Enhanced Motion and Sizing of Bank in Moving-Bank MMAE

JUAN R. VASQUEZ

PETER S. MAYBECK, Fellow, IEEE
Air Force Institute of Technology

The focus of this research is to provide methods for generating precise parameter estimates in the face of potentially significant parameter variations such as system component failures. The standard multiple model adaptive estimation (MMAE) algorithm uses a bank of Kalman filters, each based on a different model of the system. Parameter discretization within the MMAE refers to selection of the parameter values assumed by the elemental Kalman filters, and dynamically redeclaring such discretization yields a moving-bank MMAE. A new online parameter discretization method is developed based on the probabilities associated with the generalized chi-squared random variables formed by residual information from the elemental Kalman filters within the MMAE. This new algorithm is validated through computer simulation of an aircraft navigation system subjected to interference/jamming while attempting a successful precision landing of the aircraft.

Manuscript received September 8, 1998; revised September 2, 1999, October 24, 2001 and May 17, 2002; released for publication May 28, 2004.

IEEE Log No. T-AES/40/3/835878.

Refereeing of this contribution was handled by P. K. Willett.

Authors' address: AFIT/ENG, 2950 Hobson Way, Building 641, WPAFB, OH 45433-7765, E-mail: (Peter.Maybeck@afit.edu).

0018-9251/04/$17.00 2004 IEEE

I. INTRODUCTION

Parameter and state estimation are critical in many of today's complex systems. A motivating example is that of providing an accurate navigation solution to an aircraft performing a precision landing. Sufficient accuracy of the state estimate is needed to provide the desired navigation solution. Adequate parameter estimation is needed so the system can adapt to a failure or other significant event, such as the onset of interference or jamming of the onboard GPS receiver, and ensure performance is maintained. The focus of the work presented here is to present methods developed for generating precise parameter estimates in the face of potentially significant parameter variations such as interference or jamming of the onboard GPS receiver.

A new algorithm is developed which works in conjunction with multiple-model adaptive estimation (MMAE). Specifically, a new online discretization method is described for use with moving-bank MMAE to enhance parameter estimation. By dynamically redeclaring which points in parameter space are used for the basis of the Kalman filters within the MMAE structure, a moving-bank MMAE is able to provide a fine enough parameter discretization to yield precise state and parameter estimation, without requiring the overwhelming number of elemental filters to operate in parallel that a conventional MMAE would. A brief overview of moving-bank MMAE is presented, followed by discussion of the new discretization method, and simulation results.

II. MMAE OVERVIEW

The basic structure of an MMAE is shown in Fig. 1. The MMAE employs multiple Kalman filters to model the dynamic nature of the system (and its sensors) to represent performance in the presence of specific hypothesized environment conditions. By running multiple filters in parallel, residual information at each update is used to identify the system parameters or failure status and to reconfigure the system rapidly to major events or failures. A logical choice for one filter is the no-fail condition, with the remainder of the filter bank comprised of filters based on hypothesized candidate failure types. The number of elemental filters will affect the granularity of the parameter estimation and ultimately the accuracy of the state estimation.

Let $\mathbf{a}$ denote the p-dimensional vector of unknown parameters in the system model and assume that, in general, the range of $\mathbf{a}$ is continuous. The MMAE will generate estimates of the parameter vector and state vector, denoted $\hat{\mathbf{a}}$ and $\hat{\mathbf{x}}$, respectively. A separate elemental filter within the MMAE is associated with discrete point values for the parameters hypothesized

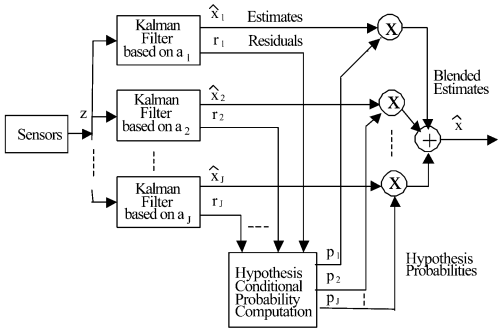


Fig. 1. MMAE algorithm.

during modeling. To make the number of filters in the bank finite and thus keep the problem tractable, the continuous range of $\mathbf{a}$ is discretized into J representative values. More explicitly, let the model for the jth filter corresponding to $\mathbf{a}_j$ (for $j = 1,2,\ldots,J$) be described by an "equivalent discrete-time model" for a continuous-time system with sampled-data measurements as fully detailed in [10].

The hypothesis conditional probability $p_j(t_i)$ is defined as the probability that $\mathbf{a}$ assumes the discrete value $\mathbf{a}_j$, conditioned on the observed measurement history to time t_i,

$$p_j(t_i) = \Pr[\mathbf{a} = \mathbf{a}_j \mid \mathbf{Z}(t_i) = \mathbf{Z}_i]. \tag{1}$$

Then it can be shown [1, 5, 8, 11] that $p_j(t_i)$ can be evaluated recursively for all j via the iteration

$$p_j(t_i) = \frac{f_{\mathbf{z}(t_i)|\mathbf{a},\mathbf{Z}(t_{i-1})}(\mathbf{z}_i \mid \mathbf{a}_j, \mathbf{Z}_{i-1}) p_j(t_{i-1})}{\sum_{k=1}^{J} f_{\mathbf{z}(t_i)|\mathbf{a},\mathbf{Z}(t_{i-1})}(\mathbf{z}_i \mid \mathbf{a}_k, \mathbf{Z}_{i-1}) p_k(t_{i-1})} \tag{2}$$

in terms of the previous values of $p_1(t_{i-1}),\ldots,p_J(t_{i-1})$ and the conditional densities for the current measurement $\mathbf{z}(t_i)$ given by

$$f_{\mathbf{z}(t_i)|\mathbf{a},\mathbf{Z}(t_{i-1})}(\mathbf{z}_i \mid \mathbf{a}_j, \mathbf{Z}_{i-1}) = \beta_j \exp\{-\tfrac{1}{2}\mathbf{r}_j^{\mathrm{T}}(t_i)\mathbf{A}_j^{-1}(t_i)\mathbf{r}_j(t_i)\} \tag{3}$$

where

$$\beta_j = \frac{1}{(2\pi)^{m/2}|\mathbf{A}_j(t_i)|^{1/2}}. \tag{4}$$

Additionally, $\mathbf{r}_j(t_i)$ is the residual vector in the jth filter, $\mathbf{A}_j(t_i)$ is the residual covariance matrix computed by the jth filter ($[\mathbf{H}_j\mathbf{P}_j^{-}\mathbf{H}_j^{\mathrm{T}} + \mathbf{R}_j]$, in terms of that filter's assumed measurement matrix $\mathbf{H}_j$, and measurement noise covariance $\mathbf{R}_j$, and that filter's computed state error covariance $\mathbf{P}_j^{-}$ before measurement incorporation), and m is the measurement vector's dimension. Furthermore, the Bayesian minimum mean square error estimate of the state is the probability-weighted average

$$\hat{\mathbf{x}}(t_i^{+}) = E\{\mathbf{x}(t_i) \mid \mathbf{Z}(t_i) = \mathbf{Z}_i\} = \sum_{j=1}^{J} \hat{\mathbf{x}}_j(t_i^{+}) p_j(t_i) \tag{5}$$

where $\hat{\mathbf{x}}_j(t_i^{+})$ is the state estimate generated by a Kalman filter based on the assumption that the parameter vector equals $\mathbf{a}_j$. The corresponding parameter estimate (the conditional mean of $\mathbf{a}$), can be generated as

$$\hat{\mathbf{a}}(t_i) = \sum_{j=1}^{J} \mathbf{a}_j p_j(t_i). \tag{6}$$

One expects the residuals of the Kalman filter based on the best model to have the mean-squared value most in consonance with its own computed $\mathbf{A}_j(t_i)$, whereas "mismatched" filters have larger residuals than anticipated through $\mathbf{A}_j(t_i)$. Therefore, the filter based on the most correct assumed parameter value receives the most probability weighting.

To handle time-varying parameter values, a lower bound is imposed on the probabilities $p_j(t_i)$ from (2) computed by an MMAE based on static parameter assumptions [11]. The alternative of a Markov model for $p_j(t_i)$ variations in time and the associated interacting multiple model (IMM) algorithm [2, 6, 7] was considered but rejected for our current applications for two reasons. First, the IMM algorithm requires specification of an entire matrix of parameter state transition probabilities for its Markov model, which often is significantly more difficult for a designer to provide accurately than a single lower bound to accomplish basically the same impact on probability flow within the MMAE. Secondly, the elemental filters within the IMM algorithm are continually reinitialized, whereas an elemental filter is only restarted in the proposed MMAE when there is an indication of divergence (the quadratic form in the exponent of (3) exceeds a specified value). The authors believe that this allows better disambiguation among different parameter value hypotheses because the effects of those different modeling assumptions are allowed to accumulate in the filter residuals over multiple sample periods.

III. MOVING-BANK MMAE

To avoid the potentially large number of elemental filters needed for an MMAE bank, the concept of a "moving bank" of fewer filters has been developed [3, 12, 14]. For instance, if there are two uncertain parameters and each can assume 10 possible values, then $J = 10^2 = 100$ separate filters must be implemented in a full-scale fixed-bank MMAE, even if the parameters are treated as unknown constants. The moving-bank MMAE is identical to the full-bank estimator discussed previously, except J corresponds to the smaller number of elemental filters in the moving bank rather than the total number of possible discrete parameter vector values. There is then an on-line dynamic redeclaration of which points in the parameter space are to be used for the basis of

the elemental filters within the MMAE, i.e., which points are to define the current "moving bank". In the example above, one might choose the three discrete values of each scalar parameter that most closely surround the estimated value, requiring $J = 3^2 = 9$ separate elemental filters, rather than 100. The issue then becomes which particular J filters are in the bank at a given time or how should the parameter space be "discretized" for accurate estimation of the parameter vector $\mathbf{a}$. A new algorithm is presented next which addresses this issue.

Throughout this discussion, it is assumed that the total number (J) of elemental filters within the MMAE algorithm is prespecified by the designer. The algorithm must determine which discrete points in parameter space $(\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_J)$ are to be used for the basis of those J elemental filters. Where those points reside in parameter space determine the physical location and size of the region of parameter space covered by the assumed parameter point values, i.e., over time this determines the "motion" and changing "size" of the bank in the moving-bank MMAE algorithm. Note that the methodology for moving and resizing the bank developed herein is entirely different from that of the variable structure IMM algorithm [6, 7], in which the number of elemental filters is allowed to vary with time and in which the parameters are assumed to obey a Markov process model inherent in the IMM structure.

IV. PROBABILITY-BASED DISCRETIZATION METHOD

The fundamental concept employed by the probability-based discretization method (PBDM), is to choose the parameter values for the elemental Kalman filters in an MMAE based on the calculation of the probability $P_\chi(\chi_j \leq T)$. Recall that each elemental filter assumes a value for the parameter vector $\mathbf{a}_j$, and the true parameter vector is identified as $\mathbf{a}_t$. If any mismodeling of the true system is present in the filter model ($\mathbf{a}_j \neq \mathbf{a}_t$), then χ_j is the generalized chi-squared random variable defined by the following quadratic form of the measurement residuals $\mathbf{r}_j$ and their associated filter-computed covariance matrix $\mathbf{A}_j$:

$$\chi_j = \mathbf{r}_j^{\mathrm{T}} \mathbf{A}_j^{-1} \mathbf{r}_j. \tag{7}$$

Note that this is the quadratic form that appears in (3) and thus is the fundamental basis of the MMAE adaptation through the $p_j(t_i)$ computation of (2). The probability calculation $P_{\chi_j} = P_\chi(\chi_j \leq T)$, is the probability that the generalized chi-squared variable (generalized because the actual residual $\mathbf{r}_j$ in (7) actually has covariance $\mathbf{A}_t$, defined explicitly later in (10), versus the jth elemental filter-computed $\mathbf{A}_j$) will lie below a threshold T. Note that χ_j provides an indication of the correctness of a filter in terms of how well the filter-assumed parameter values match the true parameter values. Specifically, a value of $\chi_j \leq m$ (the measurement dimension) indicates that the filter model is a good match to truth, whereas $\chi_j \gg m$ implies a bad match to truth. The PBDM will select parameter values for the filters such that the desired "correctness" of the filter models (in terms of how well the filter-assumed parameter values match the true parameter values) will be attained. In other words, the filter-assumed parameter values will be chosen by the PBDM such that designer-chosen P_{χ_j} values will be attained.

First, Section A develops the computation of the probability $P_\chi(\chi_j \leq T)$ as a (hyper-) volume integral under a probability density surface over an ellipsoid, or, after a transformation, over a spheroid for simpler evaluation. Section B then indicates how a designer would choose desired values of $P_\chi(\chi_j \leq T)$ based upon attaining a corresponding desired pattern of p_j values during the operation of the MMAE. Section C discusses one fruitful means of discretizing the parameter space: locating one discrete value at the value $\hat{\mathbf{a}}(t_i)$ generated by the MMAE and then setting the other $\mathbf{a}_j$ values so as to attain the desired probability values. This entails an iterative solution for the $\mathbf{a}_j$ values, and an efficient search routine for this purpose is described in Section D. Finally, Section E suggests an overall algorithm design strategy of using an off-line search for sets of $\mathbf{a}_j$ values, to enable a simple table lookup for on-line use by a moving-bank MMAE.

A. Probability Calculation and Coordinate Transformation

The probability that $\chi_j \leq T$ is given by the one-dimensional integral

$$P_{\chi_j} = P_\chi(\chi_j \leq T) = \int_0^T f_\chi(c)dc \tag{8}$$

where $f_\chi(c)$ is the generalized chi-squared density function. If the filter is a perfect match of the true system ($\mathbf{a}_j = \mathbf{a}_t$), then the density function $f_\chi(c)$ is exactly the well-known chi-squared density function, and the probability P_{χ_j} can be readily calculated. However, if any mismodeling of the true system is present in the filter model ($\mathbf{a}_j \neq \mathbf{a}_t$), then the generalized chi-squared density function $f_\chi(c)$ is not readily known. Therefore, it is necessary to evaluate an m-dimensional integral based on the density function for the residuals $f_{\mathbf{r}}(\boldsymbol{\rho})$ which can be expressed in the presence of mismodeling and is given by

$$P_\chi(\chi_j \leq T) = \int \cdots \int_{\mathcal{A}_{\mathbf{r}}} f_{\mathbf{r}}(\boldsymbol{\rho})d\boldsymbol{\rho} \tag{9}$$

where $\mathcal{A}_{\mathbf{r}}$ represents the appropriate region of integration in the parameter space.

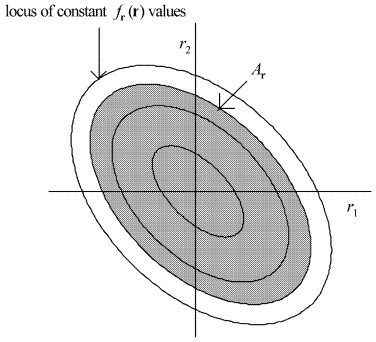


Fig. 2. Density function contour map (hyper)ellipsoids for 2-D residuals.

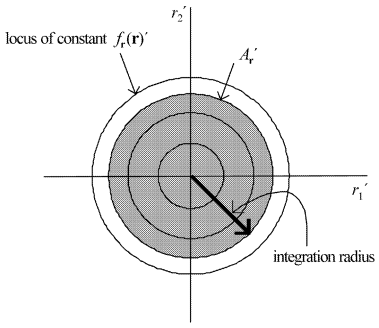


Fig. 3. Transformed density function contour map (hyper)spheroids for 2-D residuals.

The density function $f_{\mathbf{r}}(\boldsymbol{\rho})$ is fully characterized as a normally distributed random variable with mean $\boldsymbol{\mu}_j$ and covariance matrix $\mathbf{A}_t \Rightarrow \mathbf{r}_j : N[\boldsymbol{\mu}_j, \mathbf{A}_t]$. Hanlon [4] developed expressions for the residual mean vector $\boldsymbol{\mu}_j$ in the presence of mismodeling, and he showed that the actual covariance matrix for the residuals in the presence of system mismodeling is the true covariance matrix $\mathbf{A}_t$ associated with the residuals and is computed in terms of the true system matrices as

$$\mathbf{A}_t = \mathbf{H}_t \mathbf{P}_t^- \mathbf{H}_t^{\mathrm{T}} + \mathbf{R}_t \tag{10}$$

where $\mathbf{P}_t^-$ is the true error covariance matrix committed by a filter based on $\mathbf{a}_t$ against a "real-world" based on $\mathbf{a}_t$ (the superscript denotes the matrix prior to a measurement update), $\mathbf{H}_t$ is the true system output matrix, and $\mathbf{R}_t$ is the true measurement noise covariance matrix. Therefore, a fully characterized density function $f_{\mathbf{r}}(\boldsymbol{\rho})$ is available for use in the integration given by (9).

The density function $f_{\mathbf{r}}(\boldsymbol{\rho})$ of the m-dimensional residual can be visualized by means of a hyperellipsoid contour map, with a two-dimensional zero-mean example illustrated in Fig. 2. Integration over the shaded (hyper)elliptical region $\mathcal{A}_{\mathbf{r}}$ would be messy in terms of the limits of integration. However, this can be simplified if the region is transformed to the shaded (hyper)spherical region $\mathcal{A}_{\mathbf{r}'}$ associated with the density function $f_{\mathbf{r}'}$ as shown in Fig. 3. The development presented here shows that the transformed limit of integration is a radius of integration equal to the square root of the threshold $\sqrt{T}$.

Apply the following linear transformation to the residuals

$$\mathbf{r}_j' = \sqrt[C]{\mathbf{A}_j^{-1}}\,\mathbf{r}_j \tag{11}$$

where C denotes Cholesky and a Cholesky square root [10] of $\mathbf{A}_j^{-1}$ is used such that

$$\mathbf{A}_j^{-1} = \left(\sqrt[C]{\mathbf{A}_j^{-1}}\right)^{\mathrm{T}} \left(\sqrt[C]{\mathbf{A}_j^{-1}}\right). \tag{12}$$

This linear transformation on the Gaussian random variable maintains its Gaussian nature as $\mathbf{r}_j' : N[\boldsymbol{\mu}_j', \mathbf{A}_t']$ with the transformed mean vector and covariance matrix given by

$$\boldsymbol{\mu}_j' = \sqrt[C]{\mathbf{A}_j^{-1}}\,\boldsymbol{\mu}_j \tag{13}$$

$$\mathbf{A}_t' = \left(\sqrt[C]{\mathbf{A}_j^{-1}}\right) \mathbf{A}_t \left(\sqrt[C]{\mathbf{A}_j^{-1}}\right)^{\mathrm{T}} \tag{14}$$

and the generalized chi-squared variable given by

$$\chi_j = \mathbf{r}_j'^{\mathrm{T}} \mathbf{r}_j'. \tag{15}$$

This leads to the transformation of (9) into the m-dimensional integral

$$P_\chi(\chi_j \le T) = \int_{\mathcal{A}_{\mathbf{r}'}} \cdots \int f_{\mathbf{r}'}(\boldsymbol{\rho}')\, d\boldsymbol{\rho}' \tag{16}$$

which is now based on the transformed density function $f_{\mathbf{r}'}(\boldsymbol{\rho}')$. Finally, the radius of integration shown in Fig. 3 must be identified. The integration has been simplified from hyperellipsoids to hyperspheroids, and (15) can be recognized as the equation for a hypersphere; so the upper limit of integration is simply some radius of this hypersphere. Furthermore, the original limit of integration for $P_\chi(\chi_j \le T)$ in (8) is the threshold, T, and from the relationship of χ_j to $\mathbf{r}'$ given by (15), the radius of integration is recognized as the square root of the threshold $\sqrt{T}$. An m-dimensional numerical integration method was provided by Oxley [13] and Maybeck [10] and simply extends the often utilized technique of transforming the coordinate system from Cartesian to polar coordinates prior to performing the numerical integration [16].

B. Choice of P_{χ_j} Values

Consider an MMAE with five filters and the goal of accurately estimating a scalar parameter value. One approach would be to assign the filters parameter values such that the filter probabilities $p_j(t_i)$ would result in the values shown in Table I. Notice that the center filter in the bank has a relatively high probability p_j, its neighbors have less probability but still relatively substantial amounts, and the outlying

TABLE I
Example Probability Values

p_j, $j = 1 \ldots 5$	0.05	0.2	0.5	0.2	0.05
$P_\chi(\chi_j \leq T)$	0.025	0.15	0.5	0.85	0.975

filters have extremely small p_j values. Such a pattern of p_j values in the operation of an MMAE would indicate that the discretization of the parameter space is neither too coarse nor too fine: the total probability is not essentially all absorbed by a single elemental filter, nor is it impossible to distinguish between "good" and "bad" models on the characteristics of the residuals from the five filters. These values are determined in an ad hoc fashion, giving the designer significant latitude in choosing how broad or narrow to make the filter bank. However, this particular choice is motivated by human eyesight, in that the center three filters would correspond to points tightly spaced in parameter space and thus providing a "foveal view" of that space, while the two outliers would provide a coarser discretization for a "peripheral view."

In order to strive for these desired p_j values, as calculated by the MMAE, the designer would select the P_{χ_j} values shown in Table I. Note that $P_{\chi_j} = P_\chi(\chi_j \leq T)$ is a monotonic function of T, depicting the cumulative probability associated with χ_j taking on values less than or equal to T (not a conditional probability that χ_j takes on values in a small neighborhood about T, as for p_j). This discretization method provides a more systematic and more theoretically substantiated means of choosing the filter-assumed parameter values $\mathbf{a}_j$ than the ad hoc methods used to date. Specifically, the designer's primary concern is to select desired probability p_j values, leading to the associated P_{χ_j} values such as those in Table I, then apply the PBDM to determine the appropriate $\mathbf{a}_j$ values.

The following example illustrates a systematic method for relating the probabilities p_j and P_{χ_j}. Let each probability weight p_j be divided equally to the right and to the left of point j as shown in Fig. 4. This assumes proportionate distribution of the probability in the neighborhood of point j for simplicity. Recombine through summation the probability between points j and $j+1$, noting that the first and last values are unchanged. Integrate (form a cumulative sum) up to point j to generate the probability that model j is a good match to truth, i.e., its chi-squared statistic χ_j, is less than or equal to the threshold T. Therefore, the first J values represent the probability P_{χ_j} and are given by

$$P_{\chi_j} = \frac{p_j}{2} + \sum_{k=1}^{j-1} p_k. \tag{17}$$

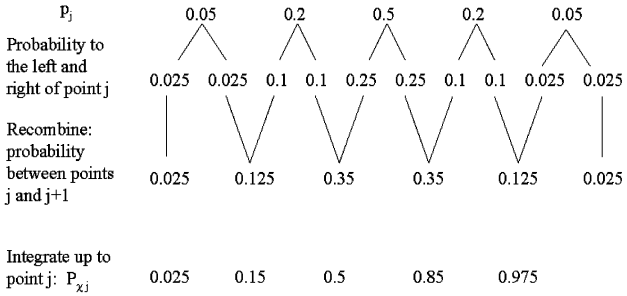


Fig. 4. Relationship of p_j to P_{χ_j}.

Note also that, if the original p_j pattern is symmetric about the center point, then the P_{χ_j} value for the center filter is always 0.5.

C. Discretization Method

The discretization method discussed in this section is one of many possible which utilizes the tools presented in the previous two sections. Other concepts including methods to move, contract, and expand the MMAE bank based on the numerical evaluation of P_{χ_j} could be explored [16]. In order to integrate these probabilities numerically, it is necessary to have the mean vector and covariance matrix for the residuals based on both a truth model and a filter model. Since the parameter value for the truth model is not exactly known (otherwise there would be no estimation needed), it is necessary to assume values for the truth model based on the best guess given by the MMAE. Specifically, assume the true parameter values are equal to the components of $\hat{\mathbf{a}}(t_i)$. This best guess is available on-line, and by design the filter in the center of the bank will assume its value.

Given this assumption, the threshold T can be determined exactly for any choice of P_{χ_t}, where the subscript t indicates that the filter-assumed parameter values are assumed to match truth. Recall that if the filter is a perfect match of the true system, then the density function $f_\chi(c)$ is exactly the ordinary chi-squared density function and the probability can be readily calculated. Inversely, the threshold can be found using the MATHCAD [9] function "$T = qchisq(P_{\chi_t}, \mathrm{m})$" which is a function of both the desired probability for the filter assumed as truth and the dimension of the residuals. Note that T is not a function of $\hat{\mathbf{a}}(t_i)$; so once P_{χ_t} and m are set, T will be known for all values of $\hat{\mathbf{a}}(t_i)$.

The remaining filters are assigned parameter values above and below $\hat{\mathbf{a}}(t_i)$ such that their P_{χ_j} values agree with those chosen by the designer for the now known threshold value T. This raises the issue of how to select parameter values for the remaining filters without simply running multiple guesses through the numerical integrator and hoping for a match with

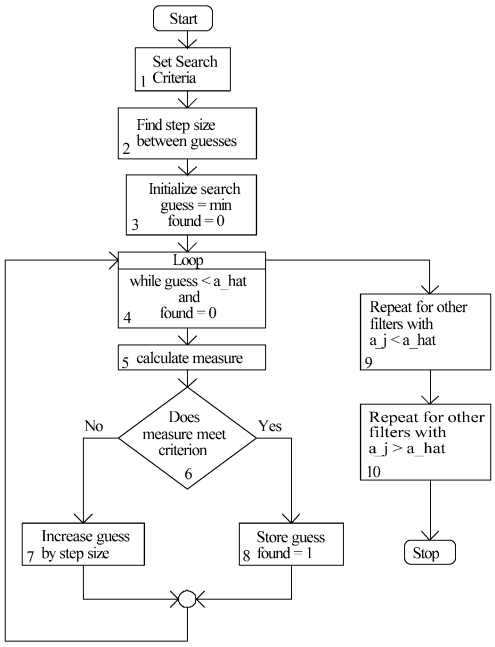


Fig. 5. Parameter value search routine.

the designer-chosen P_{χ_j} values. An automated search algorithm is presented next to overcome this problem.

D. Parameter Value Search Routine

This routine has only been developed for the special case of zero-mean residuals (thus able to address the case of uncertain parameters in the dynamics driving noise covariance $\mathbf{Q}_d$, or the measurement noise covariance $\mathbf{R}$) in the presence of mismodeling and a scalar parameter. The search routine is based on a measure of the eigenvalues of the transformed residual covariance matrix $\mathbf{A}_t'$. It is possible to use a measure of the eigenvalues of $\mathbf{A}_t'$ to predict P_{χ_j} without evaluating this probability numerically. Specifically, the product of the eigenvalues of $\mathbf{A}_t'$ was found to predict P_{χ_j} consistently:

$$M_{\text{eig}} = \prod_{i=1}^{m} \lambda_i \tag{18}$$

where

$$\lambda_i = \text{the eigenvalues of } \mathbf{A}_t'.$$

This makes sense since the product of the eigenvalues of a matrix is directly related to the volume of the hyperellipsoid defined by that matrix [15]. The flowchart of the search routine based on the measure shown in (18) is illustrated in Fig. 5.

The basic process used by the search routine is repeated for each elemental filter. First, assume a parameter value for the elemental filter, i.e., choose a "guess" value. Calculate the measure M_{eig} associated with this guess value and determine if the search criterion defined below is met. If so, the search is complete and repeat for the other elemental filters. If not, make another guess for the parameter value based on a designer-chosen guess step size and check against the criteria. Continue this process via a loop until the criterion is met or a set number of guesses have been tried.

The search criteria include: $\hat{a}$ = scalar true parameter value (current best guess), M_{eig}^d = desired eigenvalue measure, and the logic for $a_j < \hat{a}$ is given by

$$\text{IF } (M_{\text{eig}} \leq M_{\text{eig}}^d) \text{ THEN (search is successful, found = 1)}$$

and the logic for $a_j > \hat{a}$ is given by

$$\text{IF } (M_{\text{eig}} \geq M_{\text{eig}}^d) \text{ THEN (search is successful, found = 1).}$$

The value of $\hat{a}$ is given by the MMAE. The desired eigenvalue measure M_{eig}^d must be empirically determined for each filter. Note that one filter (typically the center filter) takes on the value of $\hat{a}$ and is assumed to match truth, so there is no need to search for this parameter value, and thus there is no need to find M_{eig}^d for this filter. A manual search must be done once for each of the $J-1$ values of M_{eig}^d, as explained below. The phrase manual search refers to arbitrarily selecting a value (a guess) for the parameter a_j, then performing the numerical integration to determine if the resulting $P_\chi(\chi_j \leq T)$ is equal to the designer-chosen probability value. Once a parameter value is found that produces the appropriate probability value P_{χ_j}, the eigenvalues of $\mathbf{A}_t'$ associated with this parameter value are used to calculate M_{eig}^d via (18). This process is repeated to determine each of the $J-1$ values of M_{eig}^d. However, after M_{eig}^d is determined once for any single value of $\hat{a}$, this search routine will replace the manual method for all other values of $\hat{a}$.

E. Algorithm Implementation

The search routine is performed off-line for several discrete values of $\hat{\mathbf{a}}$ which range over the span of the admissible parameter space. The filter-assumed parameter values determined for each discrete value of $\mathbf{a}$ are stored in a look-up table and referenced by the value of $\hat{\mathbf{a}}$ from the MMAE. By design, a reference value of $\mathbf{a}$ will be used as the parameter value for the filter residing in the center of the bank. This look-up table will be available in real-time and the table entry closest to the current value of $\hat{\mathbf{a}}$ will be used at each sample time. This implementation simply keeps the bank centered on the best estimate for the parameters, $\hat{\mathbf{a}}$ (as closely as possible in view of the discrete values of $\mathbf{a}$ stored in the look-up table), then relies on the new discretization method to determine the remaining parameter values. Various other implementations of the PBDM are presented with results in [16].

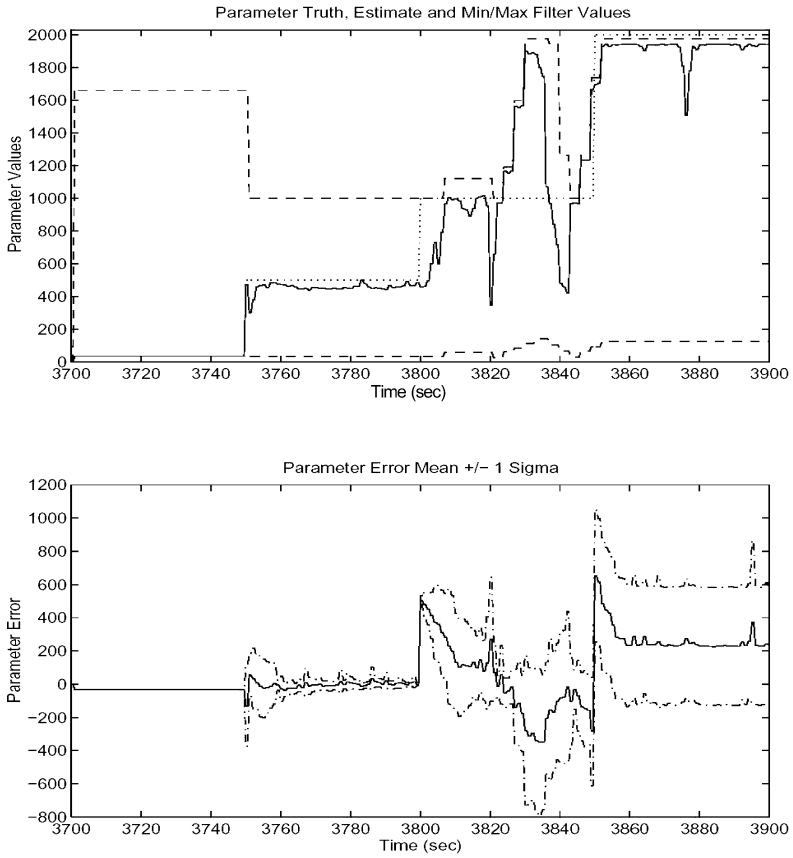


Fig. 6. Parameter estimation performance.

TABLE II
Simulated Cases (Interference/Jamming Levels)

Time (s)	3700	3750	3800	3850
R_{GPS} (ft^2)	9	4500	9000	18000

V. RESULTS

The concepts presented here are applied to a motivating example of state and parameter estimation for fault tolerant aircraft precision landing. A single failure mode representing interference/jamming of the GPS receiver is simulated for proof of concept and validation of the algorithm. The PBDM is used to discretize a moving-bank MMAE which provides state estimates and parameter estimates.

A 13-state model is used as detailed in [16] that involves an integrated GPS-INS with barometric altimeter and radar altimeter aiding. The inertial navigation system (INS) filter model is comprised of 11 states (the first nine being the standard Pinson error model states): 3 platform misalignment errors, 3 velocity errors, 3 position errors, and 2 states for barometric altimeter stabilization. The GPS filter model is comprised of 2 states: user clock bias and user clock drift. The following measurements are available: four satellite vehicle (SV) pseudo-ranges, altitude from the barometric altimeter, and height above ground level from the radar altimeter. The 10-run Monte Carlo simulations span 200 s of flight time (3700 to 3900 s) out of a 2 h flight profile. The nominal GPS measurement noise is $R_0 = 9$ ft^2, and one of the six case studies is presented here with the noise profile shown in Table II. A five-elemental-filter MMAE is implemented with a look-up table of 40 discrete parameter values for on-line use.

The designer-chosen probability values used by the PBDM are $P_{\chi_j} = [0.01, 0.24, 0.5, 0.76, 0.99]$. Note that with $P_{\chi_3} = 0.5$ and $m = 6$ for this problem, the threshold is given by MATHCAD [9] as $T = qchisq(0.5, 6) = 5.348$. The data is presented through parameter and state estimate plots. The parameter estimate plot shows the time history for the parameter being estimated in this problem for a single representative run (see Fig. 6). The first trace ($\cdots$) represents the true parameter value as dictated by the case study being simulated. The second trace (—) shows the parameter estimate provided by the MMAE. The third trace (- - -) indicates the minimum and maximum filter-assumed parameter values within the MMAE bank at each sample time. This pair of traces illustrates the breadth of the bank, how well the bank encompasses the true parameter value, and where the parameter estimate lies within the MMAE bank, i.e., close to an endpoint or somewhere in the middle. The state plots of aircraft latitude, longitude, and altitude

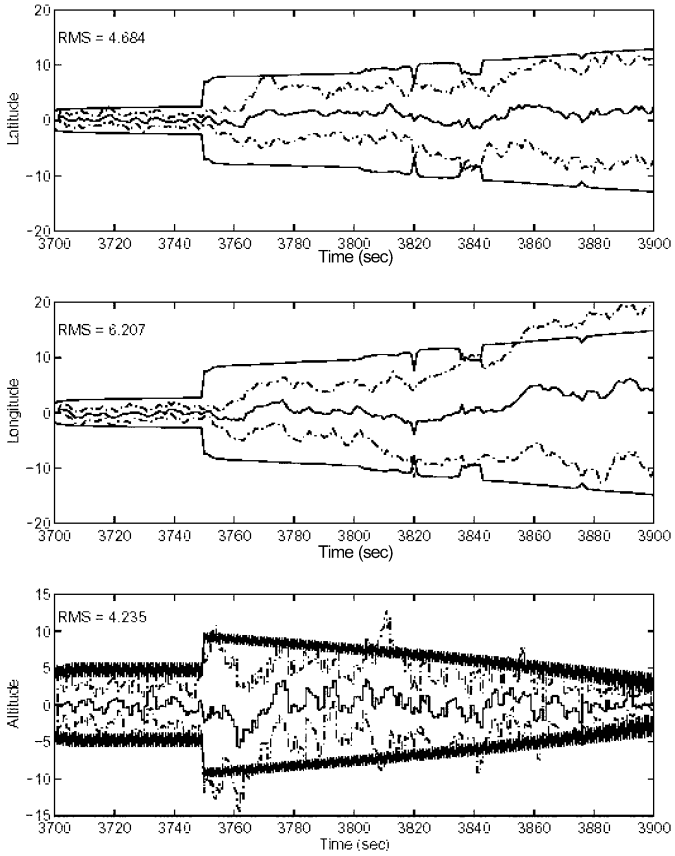


Fig. 7. MMAE state estimation errors (ft).

are plotted for the blended estimates provided by the MMAE (see Fig. 7). Each plot contains five traces. The innermost trace (—) on each data plot is the mean error time history for the applicable state. A trace pair (represented by $\cdot - \cdot -$) is plotted and identified as the state estimate error mean±sigma ($\mu_x \pm \sigma_x$). The last pair of traces (—) represent the filter-computed $\pm\sigma_{\text{filter}}$ values for the same states and are symmetrically displaced about zero because the filter "believes" that it is producing zero-mean errors [10].

The PBDM selects filter-assumed parameter values which give the MMAE the ability to react to abrupt changes in the true parameter, as seen in the plot of the parameter estimate in Fig. 6 at $t = 3750$ s. Similar performance is seen from $t = 3800$ s to 3810 s and at $t = 3850$ s. Some erratic bank movement is seen for $t = 3820$ s to 3842 s. Performance suffers when the bank size is too large, allowing severe overestimation of the parameter value ($t = 3820$ s to 3833 s). Specifically, the true parameter value is $R_{\text{GPS}} = 1000$ while the minimum and maximum filter-assumed parameter values are $R_1 = 170$ and $R_5 = 1970$, respectively. The parameter error plot in Fig. 6 shows a somewhat large bias from $t = 3850$ s to 3900 s. This is a result of restricting the bank motion at an upper bound equal to the true parameter value ($R_5 = R_{\text{GPS}} = 2000$). A better blended estimate and parameter error would result if the the upper bound on the filter bank were allowed to exceed the true parameter value. Overall, the parameter is tracked rather well and the state estimation errors are shown in Fig. 7 to be quite good. All three states show the quick response of the MMAE to the abrupt change in the true parameter at $t = 3750$ s. The state plots also indicate the adequate tuning which exists in all three channels. In particular, the filter computed standard deviation $\pm\sigma_{\text{filter}}$ encompasses the majority of the error state true mean ± one standard deviation values.

VI. CONCLUSIONS

Prior to this research, limited methods were available to determine moving-bank decisions or parameter discretization adequately within the structure of MMAE. Most of these methods rely on logic which utilize ad hoc thresholds determined through empirical analysis. The PBDM provides a new and more analytically based method for discretizing the parameter space associated with a moving-bank MMAE implementation. One application of this algorithm was demonstrated for an aircraft GPS-aided INS navigation system subjected to interference/jamming while attempting a successful precision landing of the aircraft. The results validate the ability of this algorithm to provide accurate parameter estimates in the presence of potentially

significant parameter variations such as system component failures.

REFERENCES

[1] Athans, M., and Chang, C. B. (1976)
Adaptive estimation and parameter identification using a multiple model estimation algorithm.
Technical note 1976-28, Lincoln Lab., MIT, Lexington, MA, June 1976, ESD-TR-76-184.

[2] Blom, H. A. P., and Bar-Shalom, Y. (1988)
The interacting multiple model algorithm for systems with Markovian switching coefficients.
IEEE Transactions on Automatic Control, **33**, 8 (Aug. 1988), 780–783.

[3] Gustafson, J. A., and Maybeck, P. S. (1994)
Flexible spacestructure control via moving-bank multiple model algorithms.
IEEE Transactions on Aerospace and Electronic Systems, **30**, 3 (July 1994), 750–757.

[4] Hanlon, P. D. (1996)
Practical implementation of multiple model adaptive estimation using Neyman-Pearson based hypothesis testing and spectral estimation tools.
Ph.D. dissertation, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, Sept. 1996.

[5] Lainiotis, D. G. (1976)
Partitioning: A unifying framework for adaptive systems, I: Estimation.
Proceedings of the IEEE, **64** (Aug. 1976), 1182–1197.

[6] Li, X. R., and Bar-Shalom, Y. (1996)
Multiple-model estimation with variable structure.
IEEE Transactions on Automatic Control, **41**, 4 (Apr. 1996), 479–493.

[7] Li, X. R., Zhang, Y., and Zhi, X. (1997)
Multiple-model estimation with variable structure: Model-group switching algorithm.
In *Proceedings of the 36th IEEE Conference on Decision and Control*, San Diego, CA, Dec. 1997, 3114–3119.

[8] Magill, D. T. (1965)
Optimal adaptive estimation of sampled stochastic processes.
IEEE Transactions on Automatic Control, **AC-10**, 5 (1965), 434–439.

[9] MathSoft, Inc. (1995)
MATHCAD User's Guide (version 6.0 plus edition).
101 Main Street Cambridge, MA, Sept. 1995.

[10] Maybeck, P. S. (1979)
Stochastic Models, Estimation, and Control, I.
New York: Academic Press, 1979. Republished, Arlington, VA: Navtech, 1994.

[11] Maybeck, P. S. (1982)
Stochastic Models, Estimation, and Control, II.
New York: Academic Press, 1982. Republished, Arlington, VA: Navtech, 1994.

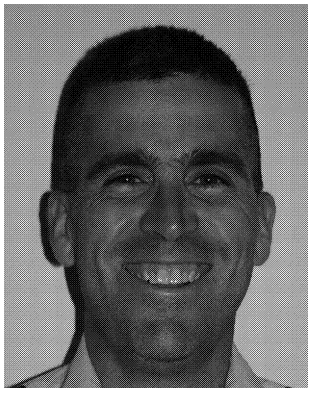
[12] Maybeck, P. S. (1989)
Moving-bank multiple model adaptive estimation and control algorithms: An evaluation.
In C. T. Leondes (Ed.), *Control and Dynamics Systems, Vol. 31*, New York: Academic Press, 1989.

[13] Oxley, M. E. (1997)
Interview with Capt. Juan R. Vasquez.
U.S. Air Force, Oct. 1997.

[14] Schiller, G. J., and Maybeck, P. S. (1994)
Space structure control using multiple model adaptive estimation and control.
In *Proceeding of the 13th IFAC Symposium on Automatic Control in Aerospace-Aerospace Control '94*, Sept. 1994.

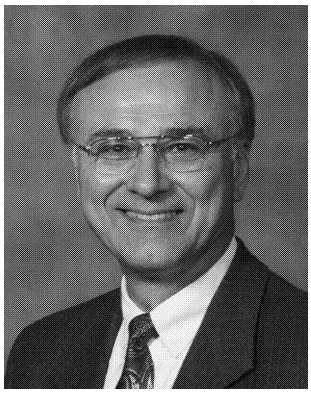
[15] Strang, G. (1976)
Linear Algebra and Its Applications.
New York: Academic Press, 1976.

[16] Vasquez, J. R. (1998)
New algorithms for moving-bank multiple model adaptive estimation.
Ph.D. dissertation, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, 1998.



Juan R. Vasquez was born on June 1, 1965. He received the B.S. degree in electrical engineering from Oklahoma State University in 1987. He received the M.S. and Ph.D. degrees in electrical engineering from the Air Force Institute of Technology in 1992 and 1998, respectively. He is a Lieutenant Colonel in the United States Air Force and is currently serving as a program manager for the Air Force Office of Scientific Research.

His military service includes assignments as an aircraft maintenance officer responsible for 330 maintenance personnel, program manager for weather satellite systems, researcher in GPS anti-jam techniques, and program manager for a multi-target multi-sensor fusion program. Dr. Vasquez's research interests include target tracking, navigation, and multiple model estimation and control.



Peter S. Maybeck (S'70—M'74—SM'84—F'87) was born in New York, NY on February 9, 1947. He received the B.S. and Ph.D. degrees in aeronautical and astronautical engineering from M.I.T., Cambridge, in 1968 and 1972, respectively.

In 1968, he was employed by the Apollo Digital Autopilot Group of The C. S. Draper Laboratory, Cambridge, MA. From 1972 to 1973, he served as a military control engineer for the Air Force Flight Dynamics Laboratory and then joined the faculty of the Air Force Institute of Technology in June of 1973. He is currently professor of electrical engineering, responsible for the graduate sequence in estimation and stochastic control and for individual advanced digital filtering and control courses. Current research interests concentrate on using optimal estimation techniques for guidance systems, tracking, adaptive systems and failure detection purposes.

Dr. Maybeck is author of over one hundred papers on applied optimal filtering as well as the book, Stochastic Models, Estimation and Control (Academic Press, vol. 1—1979, vols. 2 and 3—1982; republished by Navtech in 1994). He is a member of Tau Beta Pi, Sigma Gamma Tau, Eta Kappa Nu, and Sigma Xi. He was recipient of the DeFlorez Award (ingenuity and competence of research), the James Means Prize (excellence in systems engineering) and the Hertz Foundation Fellowship at M.I.T. in 1968. In all years from 1975 to 2003, he received commendation as outstanding Professor of Electrical Engineering at A.F.I.T. In December of 1978, he received an award from the Affiliate Societies Council of Dayton as one of the twelve outstanding scientists in the Dayton, Ohio area. In March of 1980, he was presented with the Eta Kappa Nu Association's C. Holmes MacDonald Award, designating him as the outstanding electrical engineering professor in the United States under the age of 35 (he had placed second in this national competition for 1977 as well). In 1985, he received the Frederick Emmons Terman Award, the highest national award to a Professor of Electrical Engineering given by the American Society of Engineering Education. He is a member of the AIAA, and he is the current IEEE Dayton Section Student Activities Chairman and a member of the IEEE Executive Committee of Dayton, and he previously served as chairman of the local Automatic Control Group.