JOHN L. MARYAK and MARK S. ASHER

ISOLATING ERRORS IN STATE-SPACE MODELS OF COMPLEX SYSTEMS

One of the steps in creating a mathematical model of a system is to test the model, after it has been fully specified, to determine if it is performing adequately. Often the model does not perform acceptably (e.g., it may not give accurate predictions of the actual system's performance). This lack of fidelity can also be observed in established models that had been performing well, indicating a change in the actual system. At this point, it is necessary to diagnose where in the model the problem lies, a process called error isolation. We describe an error isolation technique for detecting the misspecified parameter (or set of parameters); this technique was designed especially for use on state-space models of large-scale systems. We report on an example of an application of the methodology to localizing errors in the model of an inertial navigation system.

INTRODUCTION

This article describes a promising error isolation (EI) procedure for models of complex systems. To set the context, consider the major steps in establishing a mathematical model of a system (for systems of any type, e.g., engineering, environmental, or economic). First, a model structure is chosen, which specifies the variables, general form, and order of the model—for example, an ARMA(2,3) process or a state-space model with a specific set of states. Next the model parameters, that is, the coefficients relating the system variables, are obtained. The parameters can be found by considering the physics of the system or by an estimation process, such as least squares, that fits the model to data derived from the system. The next critical step in system modeling is to check the performance of the model. This validation step is a standard procedure for a newly created model and is also important for established system models that are used repeatedly (as in system control). If the check shows that the model is not performing well (e.g., giving poor predictions of system behavior), the problem in the model should be located. The term error isolation is used here to refer to the process of determining which part of the model is incorrect.

Our EI procedure is an adaptation of the Bayesian error isolation methodology described by Spall.[1] Specifically, we applied Spall's general methodology to a state-space model, exploited several special features of the state-space model to enhance the efficiency of EI, and produced software to implement the procedure. The state-space model, a particularly flexible and useful model, is described in more detail in the section entitled Error Isolation for State-Space Models.

We assume a situation where a complex system is modeled using a state-space model and it has been determined (by whatever means) that the existing model is not a satisfactory representation of the system. This situation is similar to that described by Gertler[2] in his discussion of failure detection and isolation, especially using state-space models. Gertler describes methods based on residual analysis that can be used to determine that there is a failure in the system and stresses the importance of failure isolation in the system modeling process. Given that the existing model is incorrect, and under the additional assumption that the state-space form of the model is correct, we can conclude that the problem lies in the parameters of the model. Our EI methodology is specifically designed to locate the parameter (or group of parameters) most likely to be misspecified in a state-space model, given a set of observed data. As discussed in Spall,[1] EI divides the parameter set into subsets, each of which is a candidate for being identified as the misspecified subset of parameters. The goal is to isolate one of these subsets as the most likely to be misspecified; that is, given a data sample, we locate the subset with the lowest probability of being correctly specified. For a large system, a straightforward Bayesian computation of the probabilities for all the subsets would entail an impossible computational burden,[3,4] but EI avoids this burden by a novel transformation of the problem from one that requires precise numerical integration in a large-dimensional space to one involving a comparison of curves in two-space. In this way, EI combines the advantages of a Bayesian methodology with the ability to analyze large-scale systems.

This type of EI is useful not only as a step in improving the predictive performance of the model, but it also can lead to increased understanding of the system being modeled and the procedures (such as statistical estimation or physical analysis techniques) used in determining the parameters of the system model. As discussed in Spall

and Garner,[5] isolating a set of terms of the model as the most likely to be incorrectly modeled allows the engineer to concentrate on improving the estimates of these questionable terms while using the existing or nominal values for the remaining parameters. Also, for a model found to be performing poorly after a period of successful operation, EI can lead to detecting failures within the system or changes in the system parameters.

The topic of fault detection and isolation has received considerable attention in the control and statistics literature. In addition to the aforementioned article by Gertler, the survey articles[6–8] and the text by Patton et al.[9] contain many references. Recent work in this area using state-space models is described in Olin and Rizzoni[10] and Ribbens and Riggins.[11] Other reports present instances of Bayesian fault detection and isolation schemes.[7,12–14] Our EI approach, which is similar to that described by Rezayat,[14] differs from other existing fault detection schemes in several ways:

1. It can isolate modeling errors of a very general nature. For example, several methods[15,16] exist to detect a sudden jump in the the system state, and tests such as the sequential likelihood ratio test are designed to detect a nonzero bias in the differences (residuals) between predicted and observed system measurements.[2] In contrast, EI can isolate errors in any of the model parameters, such as the system transition matrix or the measurement matrix. (Litkouhi and Boustany[15] describe some of the problems involved in trying to apply a generalized likelihood ratio test to detect an increase in measurement noise.)

2. In contast to identification-based methods, which have been described in Gertler,[2] where the system parameters are repeatedly estimated, our EI does not attempt to estimate the parameters and is concerned only with detecting the parameter subset that is least likely to be correct given the available data and a fully specified (but invalid) model.

3. Our EI provides the benefit of a Bayesian methodology that properly incorporates prior information into the analysis while greatly reducing the computational demands usually associated with a Bayesian technique.

4. Our EI is designed to treat large-scale systems and has performed successfully on a thirty-three–state system in contrast to several current approaches for failure detection that treat only low-order systems (see the discussion in Kerr[6]).

The thirty-three–state example, which will be discussed in a subsequent section, also contrasts with standard Bayesian methods based on multiple integration. According to Genz,[17] current multiple integration schemes can treat a maximum of about ten dimensions. Of course, considerable research (e.g., Flournoy and Tsutakawa[18]) has been done on efficient approximate Bayesian methods for higher-dimensional problems. Our EI method differs from these methods in that accurate parameter estimates (requiring accurate approximations to Bayesian integrals) are not the end product we seek. Instead, crude, approximate integrals of posterior probabilities are sufficient for the EI process (see the next section).

In the next sections, we describe the general EI methodology, our EI methodology for state-space models, its implementation, and a numerical example illustrating its application. The boxed insert presents three theorems useful in initializing the algorithm and making it more efficient.

THE GENERAL ERROR ISOLATION METHODOLOGY

The EI methodology presented here is a specialization of a general Bayesian EI methodology developed by Spall.[1] To establish the context for our implementation of EI, and to introduce some notation, we will describe this methodology in some detail. The general methodology assumes the following:

1. A system model has been created, tested, and found to be performing poorly.

2. The modeling error lies in the parameters of the model. In particular, this assumption means that the form of the model (e.g., the discrete linear Gaussian form of the model in Equation 3) is not in question.

3. The model under consideration is parameterized by a set of m parameter vectors $\{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_m\}$, with $\{\boldsymbol{\theta}_1^*, \ldots, \boldsymbol{\theta}_m^*\}$ denoting the actual values of the parameters used in the model. This notation reflects the idea that the full set of parameters is split into m subsets, each a candidate for being identified as the misspecified subset of parameters.

4. Only one of the candidate parameter subsets ($\boldsymbol{\theta}_i$) contains misspecified parameters.

The goal of the methodology is to isolate one of these subsets as most likely to be misspecified, that is, to find the subset with the lowest probability of being correctly specified given a set of observed data. The basic quantity used in our EI is the posterior probability that the ith parameter subset is correctly specified:

$$\phi_i(\mathbf{z}) \equiv Prob\{\boldsymbol{\theta}_i^* \text{ is correct} \mid \text{data } \mathbf{z}\}, \quad i = 1, \ldots, m, \tag{1}$$

where $\mathbf{z}$ represents a data vector (this definition is explained more fully in the next paragraph). If $\mathbf{z}^*$ is the data vector actually observed in an experiment, then the $\boldsymbol{\theta}_i^*$ most likely to be misspecified (given $\mathbf{z}^*$) corresponds to the index i for which $\phi_i(\mathbf{z}^*)$ is a minimum.

To give a precise definition of the quantity $\phi_i(\mathbf{z})$, we denote $\boldsymbol{\theta}_i$-space by $\boldsymbol{\Omega}_i$ (a Euclidean subspace), and let $\mathbf{E}_i \subset \boldsymbol{\Omega}_i$ be such that $\boldsymbol{\theta}_i^* \in \mathbf{E}_i$, and it is believed that $\boldsymbol{\theta}_i \in \mathbf{E}_i$ with some large probability, say 0.9. For example, $\mathbf{E}_i$ may be a tolerance region, and $\{\boldsymbol{\Omega}_i - \mathbf{E}_i\}$ would be an unacceptable region for the parameter $\boldsymbol{\theta}_i$, for example, a region that might cause the system to become unstable. The choice of $\mathbf{E}_i$ is somewhat arbitrary, but if the same probability is used for each i, then the $\mathbf{E}_i$ regions will all be on an equal footing (*a priori*). It is usually reasonably easy for the system engineer to provide the $\mathbf{E}_i$ regions. We assume that Bayesian prior densities $p_i(\boldsymbol{\theta}_i)$ expressing the engineer's beliefs about the parameters (before collecting test data) are available (note that $\int_{\Omega_i} p_i(\boldsymbol{\theta}_i)\, d\boldsymbol{\theta}_i = 1$). Denote a stacked vector of system measurements by $\mathbf{z} = (\mathbf{z}_1^{\mathrm{T}}, \mathbf{z}_2^{\mathrm{T}}, \ldots, \mathbf{z}_n^{\mathrm{T}})^{\mathrm{T}}$ (in which T = transpose), and

Ψ-CURVE THEOREMS

In papers by Spall,[1,25] theorems and proofs of facts about ϕ-curves are presented. Implementing the results of these theorems facilitates the initialization of the error isolation (EI) algorithm and can make the stochastic approximation (SA) search more efficient. Our reformulation of the curves as $\psi_i(c)$ makes use of standard modern (innovations-based) technology for state-space models. In addition, we have proved three theorems (corresponding to Spall's Theorems 1, 2, and 3) describing the $\psi_i(c)$-curves. Because of our formulation, the proofs are simpler than those for the ϕ-curves. The theorems are presented here; the proofs are given in Asher and Maryak.[26]

The first theorem specifies conditions under which a ψ-curve approaches zero as the search variable c goes to infinity.

Theorem 1

For any i ($i = 1, 2, 3, \ldots, m$), suppose a region $\mathbf{A}_i \subset \boldsymbol{\Omega}_i - \mathbf{E}_i$ exists such that

$$\sup_{\mathbf{e}_i \in \mathbf{E}_i} \frac{1}{v_i(\mathbf{e}_i)} \geq \inf_{\mathbf{a}_i \in \mathbf{A}_i} \frac{1}{v_i(\mathbf{a}_i)} ,$$

where $v_i(\boldsymbol{\theta}_i) \equiv var(\gamma_{i1} | \boldsymbol{\theta}_i)$. Also assume $v_i(\boldsymbol{\theta})$ is continuous and $\mu_0(\boldsymbol{\theta})$ is zero for all $\boldsymbol{\theta} \in \boldsymbol{\Omega}$. Then $\psi_i(c) \to 0$ as $|c| \to \infty$. (Note: μ_0 is defined following Equation 3b.)

The second theorem specifies conditions under which the sign of the gradient of the $\psi_i(c)$ curve can be determined. This theorem can provide a monotonicity property that will eliminate some of the curves as candidates for being the minimum curve at $c = 0$ and thus simplify the SA search.

Theorem 2

Suppose a region $\mathbf{A}_i \subset \boldsymbol{\Omega}_i - \mathbf{E}_i$ exists such that

$$\sup_{\mathbf{a}_i \in \mathbf{A}_i} \frac{1}{v_i(\mathbf{a}_i)} \geq \inf_{\mathbf{b}_i \in \{\boldsymbol{\Omega}_i - \mathbf{E}_i\}} \frac{1}{v_i(\mathbf{b}_i)} .$$

Also assume that $v_i(\boldsymbol{\theta}_i)$ is continuous and that $\mu_0(\boldsymbol{\theta}_i)$ is zero for all $\boldsymbol{\theta} \in \boldsymbol{\Omega}$. Then $\text{sgn}[\psi'(c)] = -\text{sgn}(\mathbf{z}_1 + c)$. (Note: This theorem assumes that $\mathbf{z}_1$ is a scalar; see the discussion in Maryak and Asher.[21])

The third theorem specifies conditions under which a ratio of ψ-curves approaches zero as the search variable c goes to $\pm\infty$. By sequential application of the theorem, the minimum $\psi_i(c_{\text{init}})$-curve can, in principle, be identified (see the subsection on Initialization of the EI Search).

Theorem 3

Suppose a region $\mathbf{A}_l \subset \boldsymbol{\Omega}_l$ exists such that for some $i \neq l$:

$$\inf_{\mathbf{e}_l \in \mathbf{E}_l} \frac{1}{v_l(\mathbf{e}_l)} + \inf_{\omega_i \in \Omega_i} \frac{1}{v_i(\omega_i)} \geq \sup_{\mathbf{e}_i \in \mathbf{E}_i} \frac{1}{v_i(\mathbf{e}_i)} + \sup_{\mathbf{a}_l \in \mathbf{A}_l} \frac{1}{v_l(\mathbf{a}_l)} .$$

Also assume that $v_l(\boldsymbol{\theta}_l)$ and $v_i(\boldsymbol{\theta}_i)$ are continuous. Then $\psi_l(c)/\psi_i(c) \to 0$ as $|c| \to \infty$.

suppose that an actual observation $\mathbf{z}^*$ has been made. For any $\mathbf{z}$, define the ith likelihood density ($i = 1, 2, \ldots, m$) by $p_i(\mathbf{z}|\boldsymbol{\theta}_i) \equiv p(\mathbf{z}|\boldsymbol{\theta}_i, \boldsymbol{\theta}_j = \boldsymbol{\theta}_j^* \ \forall\, j \neq i)$ and denote the posterior probability of $\boldsymbol{\theta}_i \in \mathbf{E}_i$ (given $\boldsymbol{\theta}_j = \boldsymbol{\theta}_j^* \ \forall\, j \neq i$) by

$$\phi_i(\mathbf{z}) = \int_{\mathbf{E}_i} p_i(\boldsymbol{\theta}_i | \mathbf{z}) \, d\boldsymbol{\theta}_i , \tag{2}$$

where

$$p_i(\boldsymbol{\theta}_i | \mathbf{z}) = \frac{p_i(\mathbf{z} | \boldsymbol{\theta}_i) p_i(\boldsymbol{\theta}_i)}{p_i(\mathbf{z})} ,$$

and

$$p_i(\mathbf{z}) = \int_{\Omega_i} p_i(\mathbf{z} | \boldsymbol{\theta}_i) p_i(\boldsymbol{\theta}_i) \, d\boldsymbol{\theta}_i .$$

A straightforward Bayesian analysis would attempt to calculate each $\phi_i(\mathbf{z}^*)$, $i = 1, \ldots, m$, to find the minimum. However, because these calculations typically require many-dimensional numerical integrations, this attempt will fail for all but the simplest systems. The general methodology takes an indirect (although still fully Bayesian) approach to this problem, which is extremely effective. A complete description of the methodology is presented in Spall.[1]

As mentioned, the goal of EI is to find the subscript i for which $\phi_i(\mathbf{z}^*)$ is a minimum. The strategy consists of four major steps.

Step 1

Convert the problem from one involving precise numerical integration in a high-dimensional space to one involving a comparison of curves in two-space. This is done by introducing $\psi_i(c)$-curves (discussed in the subsection entitled Innovations-Based Likelihood and ψ-Curves), where c is a scalar, that have the property that $\psi_i(0) = \phi_i(\mathbf{z}^*)$. The approach is not exactly the same as in Spall,[1] which works directly with the $\phi_i(\mathbf{z}^*)$, but the idea is the same: that introducing these curves leads to an effective means of identifying the subscript i for which $\phi_i(\mathbf{z}^*)$ is a minimum by finding which $\psi_i(0)$ is smallest.

Step 2

Choose a starting point, say $c_{\text{init}} < 0$, at which the index i corresponding to the minimum $\psi_i(c_{\text{init}})$ can be identified easily (in contrast to the actually observed data where $c = 0$).

Step 3

Let c increase from c_{init} to 0, while locating intersections of the $\psi_i(c)$ curves ($i = 1, \ldots, m$) with each other, so as to keep track of which of the $\psi_i(c)$-curves is a minimum at every value of c between c_{init} and 0. A stochastic approximation (SA) technique, coupled with crude numerical integrations, is used in the search to find the intersections. The search process is depicted in Figure 1.

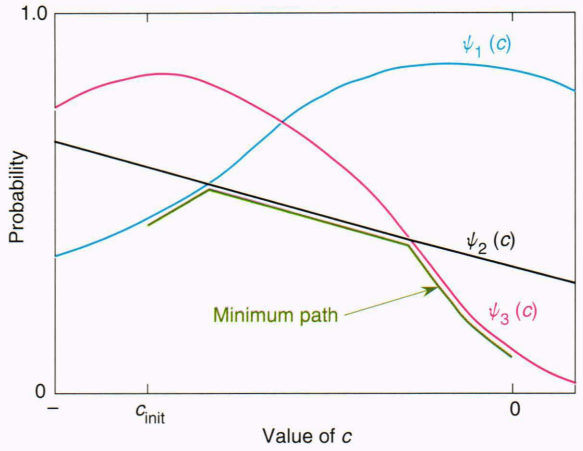


Figure 1. Example of an error isolation search (minimum path for a typical set of ψ-curves).

Given the minimum curve at c_{init} identified in step 2 (in this case, curve ψ_1), the figure illustrates how the minimum curve at $c = 0$ (curve ψ_3 in the figure) can be identified by finding intersections of the curves and switching attention from one curve to the next as the search proceeds from c_{init} toward $c = 0$ (e.g., note that curve ψ_2 becomes the new minimum curve when curves ψ_1 and ψ_2 intersect).

Step 4

Stop the search at $c = 0$. The subscript i of the minimum $\psi_i(0)$ curve identifies the minimum $\phi_i(\mathbf{z}^*)$ and hence the most likely misspecified parameter vector, $\boldsymbol{\theta}_i^*$.

These steps are described more fully in subsequent sections. The efficacy of this strategy comes from the fact that crude numerical integration is sufficient for the search to work successfully (as the SA tends to average out inaccuracies in the crude integration); precise numerical integration (which turns out to be much more computationally demanding) would be required for a straightforward Bayesian analysis.

In preliminary studies, for example by Everett,[19] this methodology has proved both effective and efficient relative to the standard Bayesian approach. Although these studies involved state-space models, none has fully exploited the special form of the state-space model.

ERROR ISOLATION FOR STATE-SPACE MODELS

Any implementation of the general EI methodology requires subroutines especially tailored to the model being used, which in our case is the standard linear-Gaussian state-space model:

$$\text{State:} \quad \mathbf{x}_k = \mathbf{T}_k \mathbf{x}_{k-1} + \mathbf{w}_k \ , \tag{3a}$$

$$\text{Measurement:} \quad \mathbf{z}_k = \mathbf{M}_k \mathbf{x}_k + \mathbf{v}_k \ , \tag{3b}$$

where the initial state and the noise terms are mutually independent and Gaussian: $\mathbf{x}_0 \sim N(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$, $\mathbf{w}_k \sim N(0, \mathbf{Q}_k)$, $\mathbf{v}_k \sim N(0, \mathbf{R}_k)$, and $k = 1, 2, \ldots, n$. The system state vector $\mathbf{x}_k$ may have large dimension (e.g., thirty-three states in the application of particular interest to us), and the observation $\mathbf{z}_k$ may also be multivariate.

In this context, the parameters of the model are the vector $\boldsymbol{\mu}_0$, the system transition matrices $\mathbf{T}_k$, the measurement matrices $\mathbf{M}_k$, and the covariance matrices $\boldsymbol{\Sigma}_0$, $\mathbf{Q}_k$, and $\mathbf{R}_k$. A natural choice of the subsets $\boldsymbol{\theta}_i$ would correspond to the state-space parameters directly; that is, $\boldsymbol{\theta}_1$ would contain all of the elements of the $\mathbf{T}_k$'s, $\boldsymbol{\theta}_2$ would contain all of the elements of the $\mathbf{M}_k$'s, and so on. The choice of the $\boldsymbol{\theta}_i$'s is flexible, however; for example, another natural choice would combine in each $\boldsymbol{\theta}_i$ all of the elements corresponding to a subsystem of the system being modeled (e.g., all of the accelerometer terms in the inertial navigation model considered in a subsequent section).

In implementing our EI methodology for state-space models, we have exploited the particular form of the state-space model (and the associated Kalman filter) to provide efficient calculations of the likelihood and to automate initialization of the algorithm. We will discuss these ideas more fully in the next subsections and present further details on our implementation of the algorithm.

Innovations-Based Likelihood and ψ-Curves

As seen in Equation 2, the EI methodology requires numerical integrations involving the likelihood $p_i(\mathbf{z}|\boldsymbol{\theta}_i)$. To do these computations, we invoke a well-known technique of using the fact that the likelihood of the Kalman filter residuals (innovations) is equal to the likelihood $p_i(\mathbf{z}|\boldsymbol{\theta}_i)$ of the observed data (see, e.g., Kailath[20]). It turns out that the likelihood of the residuals can be expressed as a product of scalar normal densities,[21] which tends to be easier to compute than the likelihood $p_i(\mathbf{z}|\boldsymbol{\theta}_i)$. Using this product form of the likelihood, we have defined the $\psi_i(c)$-curves (analogous to ϕ_i-curves) depicted in Figure 1 (see Maryak and Asher[21]).

Initialization of the EI Search

We describe here an initialization procedure for finding the minimum ψ-curve at c_{init}. This method is based on the fact that the covariance of the first residual is, for many types of parameters, a monotonically increasing function of the state-space parameters (this can be seen by inspecting the Kalman filter equations for the residuals). Examples of parameters of this type are the power spectra of the process noise and variances of the initial state and the measurement noise. This procedure is very quick, as only the first residual variances are required and then only at the extreme points of the Cartesian boxes defining the $\mathbf{E}_i$ and $\boldsymbol{\Omega}_i$ regions.

The starting point for the state-space EI search is a value c_{init} at which the minimum $\psi_i(c)$-curve ($i = 1, \ldots, m$) can easily be identified. An efficient method, based on Theorem 3 (see the boxed insert), can be used when the following conditions are satisfied: (1) $\boldsymbol{\Omega}_i$ and $\mathbf{E}_i$ are Cartesian boxes for $i = 1, \ldots, m$; and (2) $v_i(\boldsymbol{\theta}_i) \equiv$ the variance of the first residual (conditioned on $\boldsymbol{\theta}_i$ and on $\boldsymbol{\theta}_j = \boldsymbol{\theta}_j^*$ for all $j \neq i$) $i = 1, \ldots, m$, are monotonically increasing functions. The initialization proceeds as fol-

lows: For all i, compute $v_i(l_i)$, $v_i(r_i)$, $v_i(L_i)$, and $v_i(R_i)$ where l_i is the minimum point in E_i, r_i is the maximum point in $\mathbf{E}_i$, L_i is the minimum point in $\mathbf{\Omega}_i$, and L_i is the maximum point in $\mathbf{\Omega}_i$. By minimum (or maximum) point, we mean the member of the Cartesian box with the smallest (or largest) value of every component (e.g., the lower left or upper right point of a box in two-space). For any combination of i and j such that

$$\frac{1}{v_j(r_j)} + \frac{1}{v_i(R_i)} \geq \frac{1}{v_i(l_i)} + \frac{1}{v_j(L_j)} \, . \tag{4}$$

Theorem 3 (boxed insert) indicates that $\psi_j/\psi_i \rightarrow 0$ as $|c| \rightarrow \infty$, so that ψ_i can be rejected as the minimum ψ-curve at c_{init} for sufficiently large $|c_{\text{init}}|$. The minimum ψ-curve should be selected arbitrarily from the set of candidates not rejected by the above criterion (a set that might contain only one candidate). Because the state-space EI search has a self-correcting mechanism that can transfer the search to a lower curve if the initial choice is not the lowest, it is not absolutely necessary to start the algorithm with the correct minimum ψ-curve, although it is more efficient to do so, if possible.

Note that Theorem 3 applies as $c_{\text{init}} \rightarrow -\infty$. In practice, the following value of c_{init} seems to work well:

$$c_{\text{init}} = -5 \max_i \{v_i[(L_i + R_i)/2]\}^{1/2} \, , \tag{5}$$

that is, c_{init} is five nominal standard deviations to the left of zero.

Of course, the natural method of simply computing the lowest ψ-curve at an arbitrary c_{init} is prohibited by the fact that, for a large-scale system, such a computation would be infeasible (and no better than standard Bayesian in the computational sense; indeed, we might as well take $c_{\text{init}} = 0$ if standard Bayesian computations were to be used).

Two other theorems relating to the EI algorithm have been proved (both for the general and state-space EI contexts; see the discussion in the boxed insert). One specifies conditions under which a ψ-curve approaches zero as the search variable c goes to $-\infty$. The other theorem specifies conditions under which the sign of the gradient of the $\psi_i(c)$-curve can be determined. These theorems have the potential for making the EI algorithm more efficient by eliminating some of the candidate curves from the search process.

Implementation of the SA Search

As mentioned previously in Step 3, the SA search is used to pass from $c = c_{\text{init}}$ to $c = 0$, identifying the minimum $\psi_i(c)$-curve at all values of c by finding intersections of the current minimum $\psi_i(c)$-curve with all the others. The SA search uses the usual SA iteration of Robbins-Monro form (see Young[22]),

$$c_{k+1} = c_k - a_k \hat{\Delta}_{ij}(c_k) \tag{6}$$

to find a solution of $\Delta_{ij}(c) \equiv \psi_i(c) - \psi_j(c) = 0$, where c_k is the current value of c, $\hat{\Delta}_{ij}(c)$ is an estimate of $\Delta_{ij}(c)$, i is the index of the current minimum ψ-curve, j is the index of another ψ-curve being compared with ψ_i, and a_k is the SA gain sequence. The objective of the SA search is to find the next intersection of the ith ψ-curve with some other ψ-curve, say ψ_j. (By "next" we mean the first intersection between the current value of c and zero.) Then, a new SA search begins at the intersection point, with ψ_j as the new minimum ψ-curve. The search continues in this way until $c = 0$ is reached; the index of the current minimum ψ-curve (i.e., at $c = 0$) then identifies the misspecified parameter subset. This search is depicted in Figure 1.

In practice, various methods can be used to implement the search. Our approach is as follows, supposing that the ith curve has just been found to be the minimum:

1. Search for an intersection of the ith curve with curve $i + 1$ (or with curve 1, if $i = m$). After an intersection is found, its "c" value (say c') is noted; of course, if the search passes $c = 0$, it is stopped. Note that the monotonicity result of Theorem 2 in the boxed insert can sometimes be used to make this search more efficient by eliminating some of the curves as candidates for intersecting the ith curve.

2. Search for an intersection of the ith curve with the next (say curve $i + 2$). Abandon the search if the previous value of $c = c'$ is passed or if $c = 0$ is passed. Otherwise, note the c-value of the intersection point.

3. Continue to compare the ith curve with all of the remaining curves, doing the bookkeeping necessary to identify the nearest intersection to the starting point, that is, the intersection that determines the next minimum curve.

The estimate $\hat{\Delta}_{ij}(c)$ is obtained by numerically integrating to approximate $\psi_i(c)$ and $\psi_j(c)$. In the numerical integrations to calculate $\psi_i(c)$, a fairly widely spaced integration grid is used (as described in Spall[1]). In fact, for $\boldsymbol{\theta}_i$ vectors having dimension greater than about fifteen, only one or two evaluation points per axis of $\boldsymbol{\theta}_i$ are practical. At each $\boldsymbol{\theta}_i$ point of the grid, the procedure uses a Kalman filter to form the residuals and then uses the product of univariate normal densities for the likelihood calculations.

The SA gain has the standard form:

$$a_k = \frac{a_0}{k^p} \, , \tag{7}$$

where $1/2 < p \leq 1$ and $a_0 > 0$ (see Young[22]). The values $a_0 = 5$ and $p = 0.51$ worked reasonably well in our numerical studies. The SA iterations were stopped (an intersection was declared to be found) when $|c_{k+1} - c_k| < |c_{\text{init}}/200|$ for two consecutive iterations. As with almost any search algorithm, these parameters of the algorithm need to be chosen (perhaps by simulation) to be compatible with the specific application.

NUMERICAL EXAMPLE: INERTIAL NAVIGATION SYSTEM MODEL

The state-space EI methodology has been implemented in software and tested in several ways. In this section, we report on an application of EI to a large-scale system similar to one that is being analyzed at The Johns Hopkins University Applied Physics Laboratory.

Model and Parameters

This application involves a thirty-three–state model of a strapdown inertial navigation system. The state vector for this model consists of velocity, position, and orientation errors of the navigation system (three states each, for a total of nine) and twelve accelerometer and twelve gyroscope error states. The model relates the observed position error of the system to the thirty-three error states in a bench-test scenario.

The state-space model is

$$\text{State: } \mathbf{x}_k = \mathbf{T}_k \mathbf{x}_{k-1} + \mathbf{w}_k \ , \tag{8a}$$

$$\text{Measurement: } \mathbf{z}_k = \mathbf{M}\mathbf{x}_k + \mathbf{v}_k \ , \tag{8b}$$

where

$$\mathbf{w}_k \sim N(0, \mathbf{Q}) \ ,$$

$$v_k \sim N(0, \mathbf{R}) \ ,$$

$$\mathbf{x}_0 \sim N(0, \boldsymbol{\Sigma}_0) \ ,$$

$$\mathbf{T}_k = \begin{bmatrix} \mathbf{T}_{11} & \mathbf{T}_{12} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} ,$$

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} ,$$

$$\mathbf{M} = [\mathbf{I} \quad \mathbf{0}] \ .$$

Here, $\mathbf{Q}_{11}$, $\mathbf{R}$, and $\boldsymbol{\Sigma}_0$ are diagonal matrices; $\mathbf{T}_k$, $\mathbf{Q}$, and $\boldsymbol{\Sigma}_0$ are 33 × 33 matrices; $\mathbf{T}_{11}$ and $\mathbf{Q}_{11}$ are 9 × 9; $\mathbf{M}$ is 3 × 33; and $\mathbf{R}$ is 3 × 3. For further discussion of this type of model, see Eulrich et al.[23] or Upadhyay and Damoulakis.[24]

The parameter vectors are $\boldsymbol{\theta}_1$, a five-vector consisting of elements of the plant coefficient matrix (the transition matrices $\mathbf{T}_k$ are functions of this); $\boldsymbol{\theta}_2$, a nine-vector consisting of the power spectral density (diagonal) matrix of the process noise for the first nine states; θ_3, a scalar consisting of the variance of the measurement noise components; and $\boldsymbol{\theta}_4$, a thirty-three–vector consisting of the diagonal (variance) elements of $\boldsymbol{\Sigma}_0$. For convenience in setting up the input, the units of the thirty-three terms were scaled so that the $\boldsymbol{\Sigma}_0$ matrix is the identity matrix. The parameter spaces $\boldsymbol{\Omega}_i$ and subspaces $\mathbf{E}_i$ were chosen somewhat arbitrarily, although engineering judgment was used to make the model reasonably realistic given the aforementioned scaling of the components of the state.

The Software

The software was written in Fortran, using standard IMSL routines for generating random numbers and for matrix operations. We used an available in-house set of straightforward Kalman filter routines (we did not need the numerical sophistication of a square-root version of the filter, for example). The integration routine (an adaptation of the IMSL routine, TWODQ) can integrate in any number of dimensions, subject to the usual limitations imposed by hardware and time. The integration algorithm is a standard Monte Carlo method that divides the integration hyper-rectangle into a collection of subhyper-rectangles as specified by the input number, say M, of grid points per axis. For example, if the dimension of the integration space is five (as for the five-dimensional parameter $\boldsymbol{\theta}_1$), and if M is specified as equal to 2, then each of the five axes in the integration space (a subset of $\mathbb{R}^5$) is divided into two equal parts, defining a subdivision of the integration space into thirty-two (2^5) sub-hyper-rectangles. The integration then proceeds in standard Monte Carlo fashion, using points determined by selecting a uniform random variate on each of the axis partitions.

The Bayesian Priors

The priors used were truncated normal priors developed using the $\boldsymbol{\theta}_i^*$ values as means with the assumption that the $\mathbf{E}_i$ regions represented roughly plus or minus one standard deviation (±1 SD) around the mean for each component of $\boldsymbol{\theta}_i$. Further details are given in Maryak and Asher.[21]

The Tests

For the series of tests using this model, we assumed that $\boldsymbol{\theta}_2^*$ was misspecified and randomly generated 100 points of data using the true model, with $\boldsymbol{\theta}_2 \neq \boldsymbol{\theta}_2^*$. We set all of the components of $\boldsymbol{\theta}_{2,\text{true}}$ to 0.1 and all of the components of $\boldsymbol{\theta}_2^*$ to 0.5. Because the $\mathbf{E}_2$ region was defined to have all of its left end points at 0.1 and all of its right end points at 0.9, the misspecified components were all 1 SD from their true values (relative to the prior one-sigma values). But for the correlations built into the (prior) variance for this parameter, this level of misspecification might be considered extreme. In fact, we believe that the level of misspecification could be called moderate, as it is roughly comparable to obtaining three one-sigma observations in independent random tests and also because the misspecified values were actually on the boundary of the $\mathbf{E}_2$ region.

Ten tests were run, all using the same input data (the goal being to test the numerical performance of the algorithm), varying the integration grid points in $\boldsymbol{\theta}$-space from one test to the next. (The grid points were generated randomly for the Monte Carlo integration procedure.) The initialization procedure discussed in the subsection entitled Initialization of the EI Search was used and seemed to find the correct minimum curve at c_{init}, as evidenced by the fact that no automatic switching from the initial curve occurred. Of course, we are not completely sure that the minimum curve at c_{init} was found; an accurate computation of the $\psi_i(c_{\text{init}})$ values is impossible because of the large dimension of $\boldsymbol{\theta}$-space.

The integration grid used in EI to evaluate $\psi_i(c)$ was only one point per axis (i.e., only one integrand evaluation per numerical integral), resulting in an average central processing unit (CPU) time on an IBM 3090 of 4.5 min per test. The obvious alternative of trying to compute the $\psi_i(0)$ by standard Bayesian integration would be infeasible using even two function evaluation points per axis in the integration (reduced order models with twelve

and fifteen states took 14 min and 139 min of computation, respectively, for the standard Bayesian method with two points per axis; recall that computation time increases exponentially with the dimension of the state). Of course, such a standard Bayesian integration using only two grid points per axis would be too crude for most applications.

Test Results

State-space EI correctly identified θ_2 as the misspecified parameter in all ten tests. The run times and number of iterations of the EI search for each test are shown in Table 1.

Using a variety of search iteration patterns, EI reached the correct conclusion in all ten cases. The wide variation in iteration patterns and running times is to be expected because the integration grids varied randomly from one run to the next and also because the numerical integrals produced only rough approximations of the $\psi_i(c)$ values. Despite these rough approximations, the repeated comparisons of one curve with another across c-space, in conjunction with the smoothing effects of the SA procedure, succeeded in locating intersections and arriving at the correct final answer.

The search pattern was similar from one run to the next. The initialization routine described in the subsection entitled Initialization of the EI Search found that the minimum curve at c_{init} was curve 1. Then, in several runs, the next curve intersected by curve 1 as the search proceeded to the right (toward zero) in c-space was curve 4. Curve 4 was the new minimum curve until its intersection with curve 2, which remained the minimum curve until $c = 0$ was reached. So, the pattern of the search was generally as shown in Figure 1; that is, three curves and two transition points were traversed as the search proceeded from c_{init} to $c = 0$. In three of the runs, the algorithm went directly from curve 1 to curve 2 and stayed there until the end. These were runs 3, 6, and 10, which had the fastest run times. This fact relates to the discussion in the next paragraph. Also related to this discussion is the search pattern seen in run 8, which had the longest running time. In that run, the search switched among all four of the ψ_i-curves, making six transitions in all.

It seems clear that the EI algorithm will slow down when iterating in the vicinity of an intersection of two ψ-curves, since the quantity $\hat{\Delta}_{ij}(c_k)$ of Equation 6 gets smaller. If so, it would be useful to set up the search so that switching from one curve to another is minimized (see the preceding discussion in this section). In fact, the ideal search seems to be one that never encounters intersections and therefore never needs to switch attention from the starting curve to another curve. Some thoughts on accomplishing this are discussed in Maryak and Asher.[21]

Table 1. Run times and stochastic approximation iterations.

	Test no.									
	1	2	3	4	5	6	7	8	9	10
Run time (min)	5.3	5.4	2.3	3.2	4.0	1.8	3.6	13.2	4.0	2.6
No. of iterations	78	79	34	47	58	26	52	193	59	38

We ran some tests on an obvious alternative to our error isolation procedure. This alternative, which we call average Bayes, uses the same definition of $\psi_i(c)$ as EI does and then tries to compute $\psi_i(0)$ by averaging several rough approximations. That is, average Bayes computes $\psi_i(0)$ for $i = 1, 2, 3, 4$ several times with the standard Bayesian calculation using an integration grid with one point per axis (the same as state-space EI) and averages the $\psi_i(0)$ values for each i. Trials of average Bayes runs of this type were unsatisfactory, resulting in seven of ten wrong answers with run times comparable to the state-space EI times. As with standard Bayes, running average Bayes with only two points per axis is infeasible for this thirty-three–state model.

CONCLUSIONS

We have presented an efficient Bayesian algorithm for locating errors in state-space models. This algorithm is an adaptation of a general EI methodology developed by Spall.[1] The general EI methodology is designed for use with models of complex systems. Our version of EI makes use of the special form of the state-space model to produce an algorithm that is specially tailored and highly efficient for use with large-scale state-space models. We have proved three theorems useful in initializing and running the algorithm and have implemented the state-space EI methodology in software. We illustrated the application of the methodology using a state-space model describing a strapdown inertial navigation system, with promising results. This application of EI to a thirty-three–state model exceeds the current state of the art in standard Bayesian integration. Further, because these computer runs averaged less than 5 min and further enhancements of the algorithm are still possible, it seems clear that the methodology should have reasonable running times with even larger models.

REFERENCES

[1] Spall, J. C., "Bayesian Error Isolation for Models of Large-Scale Systems," *IEEE Trans. Automat. Control* **33**, 341–347 (1988).

[2] Gertler, J. J., "Survey of Model-Based Failure Detection and Isolation in Complex Plants," *IEEE Control Systems Magazine* **8**, 3–11 (1988).

[3] Geweke, J., "Generic, Algorithmic Approaches to Monte Carlo Integration in Bayesian Inference," in *Statistical Multiple Integration*, Flournoy, N., and Tsutakawa, R. K. (eds.) American Mathematical Society, Providence, pp. 117–135 (1991).

[4] Tsutakawa, R. K., "Multiple Integration in Bayesian Psychometrics," in *Statistical Multiple Integration*, Flournoy, N. and Tsutakawa, R. K. (eds.), American Mathematical Society, Providence, pp. 75–88 (1991).

[5] Spall, J. C., and Garner, J. P., "Parameter Identification for State-Space Models with Nuisance Parameters," *IEEE Trans. Aerosp. Electron. Syst.* **AES-26**, 992–998 (1990).

[6] Kerr, T., "Decentralized Filtering and Redundancy Management for Multisensor Navigation," *IEEE Trans. Aerosp. Electron. Syst.* **AES-23**, 83–119 (1987).

[7] Tsurumi, H., "A Survey of Bayesian and Non-Bayesian Testing of Model Stability in Econometrics," in *Bayesian Analysis of Time Series and Dynamic Models*, Spall, J. C. (ed.), Marcel Dekker, New York, pp. 75–99 (1988).

[8] Willsky, A. S., "A Survey of Design Methods for Failure Detection in Dynamic Systems," *Automatica* **12**, 601–611 (1976).

[9] Patton, R., Frank, P. M., and Clark, R., *Fault Diagnosis in Dynamic Systems: Theory and Application*, Prentice-Hall, New York (1989).

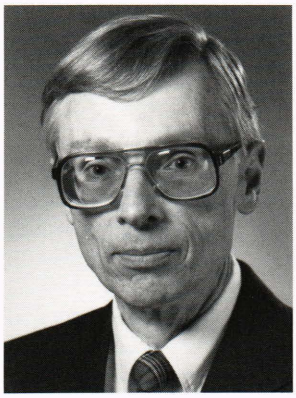
[10] Olin, P. M., and Rizzoni, G., "Design of Robust Fault Detection Filters," in *Proc. Amer. Control Conf.*, pp. 1522–1527 (1991).

[11] Ribbens, W. B., and Riggins, R. N., "Detection and Isolation of Plant Failures in Dynamic Systems," in *Proc. Amer. Control Conf.*, pp. 1514–1521 (1991).

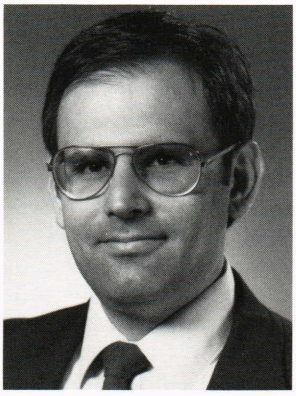
[12] Chow, E. Y., and Willsky, A. S., "Bayesian Design of Decision Rules for Failure Detection," *IEEE Trans. Aerosp. Electron. Syst.* **AES-20**, 761–773 (1984).
[13] Gordon, K., and Smith, A. F. M., "Modeling and Monitoring Discontinuous Changes in Time Series," in *Bayesian Analysis of Time Series and Dynamic Models*, Spall, J. C. (ed.), Marcel Dekker, New York, pp. 359–391 (1988).
[14] Rezayat, F., "Bayes Test for Locating the Faulty Component in a Malfunctioning Operation Process," in *Proc. Amer. Stat. Assoc., Bus. and Econ. Stat. Sect.*, pp. 140–145 (1991).
[15] Litkouhi, B., and Boustany, N. M., "On Board Sensor Failure Detection of an Active Suspension System Using the Generalized Likelihood Ratio Approach," in *Proc. 27th IEEE Conf. Decision and Control*, pp. 2358–2363 (1988).
[16] Willsky, A. S., and Jones, H. L., "A Generalized Likelihood Ratio Approach to the Detection and Estimation of Jumps in Linear Systems," *IEEE Trans. Automat. Control* **AC-21**, 108–112 (1976).
[17] Genz, A., "Subregion Adaptive Algorithms for Multiple Integrals," in *Statistical Multiple Integration*, Flournoy, N., and Tsutakawa, R.K. (eds.), American Mathematical Society, Providence, pp. 23–31 (1991).
[18] Flournoy, N., and Tsutakawa, R. K. (eds.), *Statistical Multiple Integration*, American Mathematical Society, Providence (1991).
[19] Everett, J. T., "An Evaluation of a Bayesian Approach to Isolating Sources of Errors in State-Space Models," in *Proc. Amer. Stat. Assoc., Bus. and Econ. Stat. Sect.*, pp. 280–285 (1988).
[20] Kailath, T., "An Innovations Approach to Least-Squares Estimation. Part I: Linear Filtering in Additive White Noise," *IEEE Trans. Automat. Control* **AC-13**, 646–654 (1968).
[21] Maryak, J. L., and Asher, M. S., "Isolating Errors in Models of Complex Systems," *IEEE Trans. Aerosp. Electron. Syst.*, to appear (1993).
[22] Young, P. C., *Recursive Estimation and Time Series Analysis*, Springer-Verlag, Berlin (1984).
[23] Eulrich, B. J., Andrisani, D., and Lainiotis, D. G., "Partitioning Identification Algorithms," *IEEE Trans. Automat. Control* **AC-25**, 521–528 (1980).
[24] Upadhyay, T. N., and Damoulakis, J. N., "Sequential Piecewise Recursive Filter for GPS Low-Dynamics Navigation," *IEEE Trans. Aerosp. Electron. Syst.* **AES-16**, 481–491 (1980).
[25] Spall, J. C., "Effect of the Sample on the Posterior Probability in Bayesian Analysis," *Commun. Statist. Theory Meth.* **17**, 1811–1827 (1988).
[26] Asher, M. S., and Maryak, J. L., "Isolating Modeling Errors in the Parameters of a State-Space Model," in *Proc. Amer. Stat. Assoc., Bus. and Econ. Stat. Sect.*, pp. 351–356 (1989).

ACKNOWLEDGMENT: This work was partially supported by a JHU/APL Janney Fellowship and by U.S. Navy Contract N-00039-89-C-0001.

THE AUTHORS



JOHN L. MARYAK received a Ph.D. degree in mathematics from the University of Maryland in 1972. He has worked in the APL Strategic Systems Department since 1977, where he has been involved in a variety of projects to assess and analyze the performance of Trident missile systems and submarine sonar systems. He has published many papers on Bayesian statistical methods, maximum likelihood estimation, and mathematical modeling. Dr. Maryak is a member of the American Statistical Association, the Institute of Electrical and Electronics Engineers, and Sigma Xi.



MARK S. ASHER holds B.S. and M.S. degrees in engineering from Virginia Polytechnic Institute. He has worked in the APL Strategic Systems Department since 1987, where he has been involved with modeling and estimation of inertial navigation system errors. Mr. Asher is a member of the Institute of Electrical and Electronics Engineers.