An MMAE Failure Detection System for the F-16

P. EIDE

P. MAYBECK
Air Force Institute of Technology

A multiple model adaptive estimation (MMAE) algorithm is implemented with the fully nonlinear six-degree-of-motion, Simulation Rapid-Prototyping Facility (SRF) VISTA F-16 software simulation tool. The algorithm is composed of a bank of Kalman filters modeled to match particular hypotheses of the real world. Each presumes a single failure in one of the flight-critical actuators, or sensors, and one presumes no failure. For dual failures, a hierarchical structure is used to keep the number of on-line filters to a minimum. The algorithm is demonstrated to be capable of identifying flight-critical aircraft actuator and sensor failures at a low dynamic pressure (20,000 ft, 0.4 Mach). Research includes single and dual complete failures. Tuning methods for accommodating model mismatch, including addition of discrete dynamics pseudonoise and measurement pseudonoise, are discussed and demonstrated. Scalar residuals within each filter are also examined and characterized for possible use as an additional failure declaration voter. An investigation of algorithm performance off the nominal design conditions is accomplished as a first step towards full flight envelope coverage.

Manuscript received May 7, 1995.

IEEE Log No. T-AES/32/3/05883.

Author's address: Dept. of Electrical and Computer Engineering, Air Force Institute of Technology/Engineering, 2950 P St., Bldg. 640, Wright-Patterson Air Force Base, OH 45433-7765.

0018-9251/96/$10.00 © 1996 IEEE

I. INTRODUCTION

Current state-of-the-art flight control systems (FCS) rely on costly physical redundancies to provide required aircraft reliability and survivability. What is desired is a FCS which can reduce, or even eliminate, these redundancies by exploiting the functional redundancies inherent in control surfaces and sensors. Such a FCS must be able to operate safely when its control surfaces and flight-critical sensors fail. Our goal, then, is a fault tolerant, reconfigurable FCS. The first step in that direction is a reliable failure detection algorithm.

The Air Force Institute of Technology (AFIT/ENG) has been working with Wright Laboratory (WL/FIGS) to develop multiple model, Kalman-filter-based algorithms for flight control applications [4, 6, 10, 13–17]. Past research utilized linearized "truth models" for performance simulations with good results. This effort has sought to increase confidence in these results and further demonstrate the multiple model adaptive estimator (MMAE) approach to failure detection by applying the algorithm to a more realistic, high-order, nonlinear, aircraft truth model—the most complete simulation model currently available.

The following sections include a statement of the problems which our research addresses, a short overview of the MMAE algorithm, a discussion of the models used, our results, and some concluding remarks.

II. PROBLEM STATEMENT

The task at hand is to implement an existing MMAE failure detection algorithm into as real a simulation environment of the F-16 as possible in order to facilitate a more realistic assessment of the true potential of the algorithm [5, 6, 12]. Specifically, we characterize the performance of the algorithm at a single point within the flight envelope in the following areas: single hard-over failure detection of five actuators and seven sensors; all dual hard-over combinations of these failures; and exploitation of failure information present within the filter residuals not specifically used by the MMAE algorithm. The failures included in this work are left/right stabilator (LS, RS), left/right flaperon (LF,RF), and rudder (RUD) actuators, and forward velocity (VEL), angle of attack (AOA), pitch rate (PIT), normal acceleration (Az), roll rate (ROL), yaw rate (YAW), and lateral acceleration (Ay) sensors. Additionally, we characterize operation at points in the flight envelope off the nominal design point to direct future work towards algorithm full flight envelope coverage.

III. MMAE OVERVIEW

Fig. 1 shows a functional block diagram of the MMAE algorithm. Its primary feature is a bank of

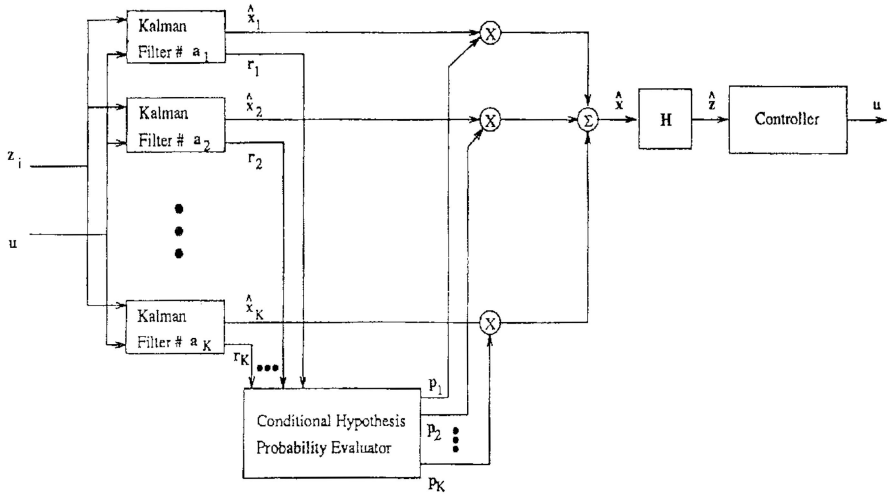

Fig. 1. MMAE block diagram.

steady-state, discrete, Kalman filters operating in parallel, with a vector of sensed measurements $\mathbf{z}_i$ and a vector of actuator control commands $\mathbf{u}$ as the inputs of the algorithm. All filter designs are based on the same reduced-order, linearized equations of motion for our nominal point in the flight envelope, but each hypothesizes a different failure condition. At every sample period, each of these K filters produce a state estimate $\hat{\mathbf{x}}_k$, and a vector of residuals $\mathbf{r}_k$, for $k = 1,2,\ldots,K$. The idea is that the filter which produces the most well-behaved residuals, contains the model which best matches the true failure status of the aircraft [4, 8, 9].

Failure identification takes place in the conditional hypothesis probability evaluator block. Each of K failure hypotheses is assigned a probability of being correct, p_k, based on the following recursive equation:

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

where

$$f_{z(t_i)|\mathbf{a},\mathcal{Z}(t_{i-1})}(\mathbf{z}_i \mid \mathbf{a}_k, \mathcal{Z}_{i-1}) = \frac{1}{(2\pi)^{m/2}|\mathbf{A}_k(t_i)|^{1/2}} \exp([-\tfrac{1}{2}\mathbf{r}_k(t_i)^T \mathbf{A}_k^{-1}(t_i)\mathbf{r}_k(t_i)]). \tag{2}$$

In these equations, $f_{z(t_i)|\mathbf{a},\mathcal{Z}(t_{i-1})}(\mathbf{z}_i \mid \mathbf{a}_k, \mathcal{Z}_{i-1})$ is the probability density function of the current measurement $\mathbf{z}(t_i)$, conditioned on the hypothesized failure status ($\mathbf{a} = \mathbf{a}_k$) and previously observed measurement history $\mathcal{Z}(t_{i-1})$, based on a filter's residuals $\mathbf{r}_k$ and internally precomputed residual covariance $\mathbf{A}_k$. When actual residuals are in consonance with filter-computed covariance $\mathbf{A}_k$, the exponential term in (2) is approximately $[-m/2]$, where m is the measurement dimension. With an incorrect hypothesis, the magnitude of that exponential is much greater, yielding a deweighted p_k for that hypothesis from (1) and (2). In practice the p_ks are artificially lower bounded ($p_{k_{\min}} = 0.001$) to prevent "lock-out" and the leading term $1/[(2\pi)^{m/2}|\mathbf{A}_k(t_i)|^{1/2}]$ is stripped to prevent a bias towards declaring sensor failures [1, 5, 6, 11–16]. The algorithm is started up with the no-failure hypothesis presumed. The output of this block is a vector of probabilites which can be used to declare a failure and also weight the state estimates as also shown in Fig. 1. The output of the algorithm is a probability weighted state estimate. In this application, the state estimate and linear combinations thereof are then passed directly as inputs to the existing Block 40 FCS for the VISTA F-16 (rather than using the raw measurements as in the existing FCS). This results in the "MMAE-based" control shown in Fig. 1.

Not shown in Fig. 1, but nonetheless important to effective operation of the MMAE algorithm, is dither. Dithering is the term given to purposeful commands sent to the actuators which "shake up" the system state. For the MMAE to detect failures, it must have some activity in the state vector from which to discern abberant behavior. Past research has investigated various forms of dithering which are subliminal to the pilot [12]. This application uses those results directly.

The architecture of Fig. 1 is used for both single- and dual-failure hypotheses. To reduce the number of filters required on line, a hierarchical approach, shown in Fig. 2, is employed. To begin with, only the K single-failure hypothesis filters are on line. Upon declaration of a failure, a new bank of filters is brought on line from memory storage. This bank contains filters designed for the declared failure, all dual failure combinations which include that failure (the doubly subscripted hypotheses in "Level 1" of Fig. 2), and the no-failure hypothesis (to "back out" of the decision tree if necessary).

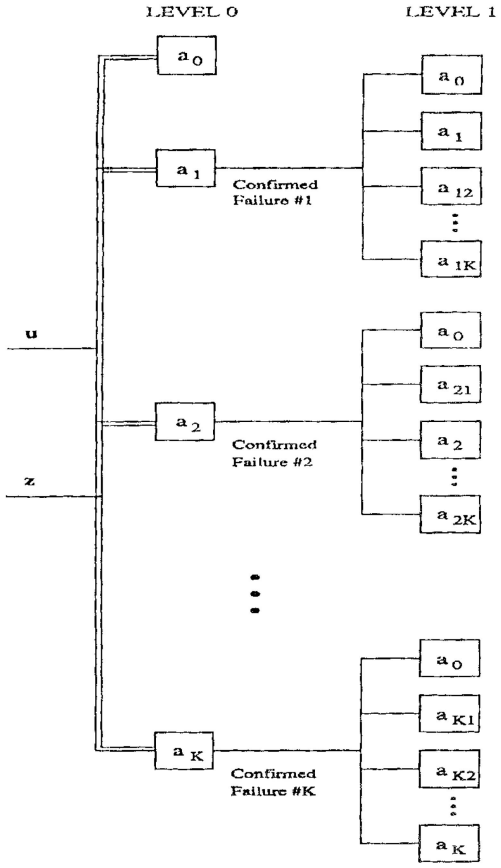

Fig. 2. Hierarchical structure.

IV. MODELS

The "truth model" used in this research is the VISTA F-16 aircraft simulation hosted in the Simulation Rapid-Prototyping Facility (SRF) at the Flight Controls Division of the Flight Dynamics Directorate, Wright Laboratory [2, 3]. This is a high-order, fully nonlinear, six-degree-of-freedom simulation which includes the Block-40 FCS. During nonrealtime simulation runs, the truth model provides required inputs $\mathbf{u}$ and $\mathbf{z_i}$ to the MMAE and the MMAE returns its estimated $\hat{\mathbf{z}}_\mathbf{i}$ to the FCS.

The design models used in the Kalman filters are based on small-perturbation-assumed, linearized, state-space equations of motion which are provided by a subroutine within the SRF VISTA F-16 software. Simple first-order lags (i.e., reduced-order models) are augmented to the state-space formulation to account for actuator dynamics. Wind buffeting is accounted for by incorporating a zero-order Dryden wind model, and sensor noise is included as well [7]. A full development of all models can be found in [4].

V. SINGLE FAILURE IDENTIFICATION RESULTS

Untuned MMAE: The MMAE design described in Sections III and IV is implemented into the SRF VISTA F-16 simulation. All twelve single failures, as well as the no-failure case, are each run a total of 10 times to give results with statistical confidence. Fig. 3 displays the results. This figure contains 13 strip plots, each of which showing the mean (over 10 runs) time history of the probability that the failure hypothesis on which it is based is correct when that failure is actually introduced at 3.0 s into the simulation. It is readily apparent from the plots that the design is unacceptable at this point. Clearly, there are higher order effects for which our filter design models cannot account. While there is good stimulation within the appropriate channels when a failure is introduced, there are just too many false alarms and missed alarms.

Tuned MMAE: In order to account for unknown higher order effects and improve on the performance of Fig. 3, we go about tuning the filters through addition of pseudonoise. In this research, we explore four methods of tuning which can exploit any available insight into the problem, three of which are incorporated into the final design.

Method 1: Direct Pseudonoise on Longitudinal Diagonal $\mathbf{Q}_d$ Entries

In the first method, we look to compensate for our reduced-order dynamics model equations by adding pseudonoise to the dynamics model. From Fig. 3 we note significant false alarming in the normal acceleration (Az) channel and some in the angle of attack (AOA) channel as well. This insight directs us to do tuning in the longitudinal channel first. To accomplish this, we insert a fictitious source of discrete white noise ($\mathbf{w}'_{d_k}$) of diagonal covariance $\mathbf{Q}'_d$ into the dynamics model equation of the filters

$$\mathbf{x}_k(t_{i+1}) = \mathbf{\Phi}_k\mathbf{x}_k(t_i) + \mathbf{B}_{d_k}\mathbf{u}(t_i) + \mathbf{G}_{d_k}\mathbf{w}_{d_k}(t_i) + \mathbf{w}'_{d_k}(t_i) \tag{3}$$

and specifically adjust the longitudinal diagonal entries ($Q_{d_\theta}, Q_{d_u}, Q_{d_\alpha}, Q_{d_q}$). The advantage here is that one can tune directly on *individual states* (pitch angle, velocity, angle of attack, pitch rate) and, with insight, correct for dynamics model deficiencies. Fig. 4 shows the probability convergence results when using this tuning method.

The improvement is quite good. With a few exceptions, the forward velocity sensor (VEL) and left stabilator actuator (L ST) most notably, unambiguous single-failure detection has been achieved. False alarms and missed alarms are nearly eliminated. It should be noted however that with increases in pseudonoises can come decreases in speed of performance. Performance improvements gained in some channels through addition of pseudonoise can cause sluggish performance in others, and in the extreme, lead to missed alarms. This tradeoff must always be considered in the context of any application.

Method 2: Direct Pseudonoise on $\mathbf{R}$ Entries

A second method which we use to improve performance in the VEL channel adds white noise

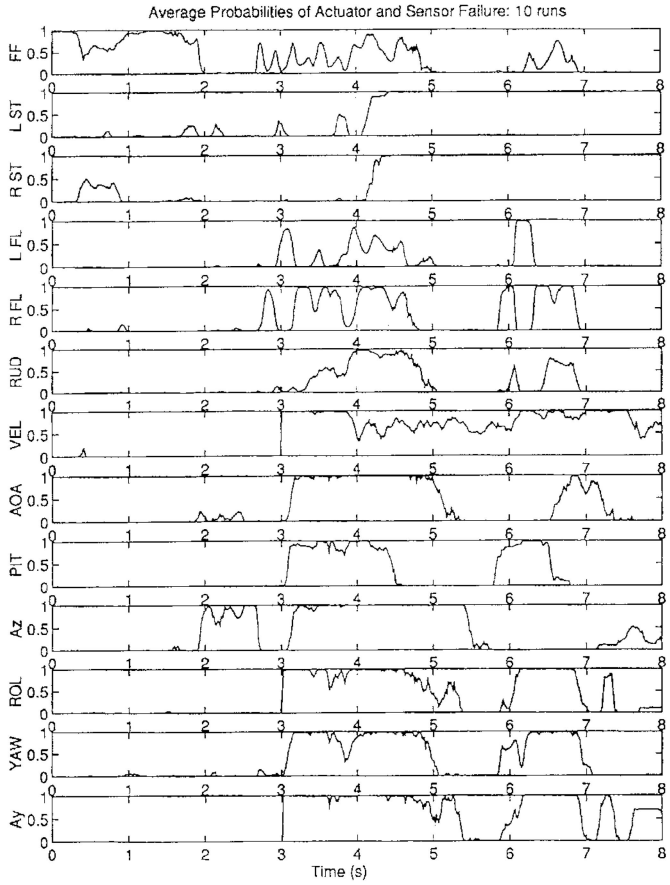

Fig. 3. Single-failure probability convergence, untuned (i.e., as tuned in [12]–[14]).

directly to the entries of $\mathbf{R}_k$, which is the covariance of the measurement corruption noise $\mathbf{v}_k(t_i)$ in the assumed measurement model:

$$\mathbf{z}_k(t_i) = \mathbf{H}_k\mathbf{x}_k(t_i) + \mathbf{v}_k(t_i). \tag{4}$$

This enters into the Kalman filter equations via the filter-computed residual covariance:

$$\mathbf{A}_k(t_i) = \mathbf{H}_k(t_i)\mathbf{P}_k(t_i^-)\mathbf{H}_k^T(t_i) + \mathbf{R}_k(t_i) \tag{5}$$

which impacts (1) and (2) directly. The advantage here is that one can tune directly in a *sensor channel* which is experiencing performance difficulty by varying the values of the diagonal entries in $\mathbf{R}_k$ $(R_u, R_\alpha, R_q, R_{Az}, R_p, R_r, R_{Ay})$.

Fig. 5 shows the performance improvement which can be achieved using this method. Comparing the VEL plot in Figs. 4 and 5, one can see that we've gained unambiguous failure detection at the cost of speed of response.

Method 3: Direct Pseudonoise on Lateral Diagonal $\mathbf{Q}_d$ Entries

For the third method of tuning, we return to the concept of Method 1, adding pseudonoise directly to *states* through $\mathbf{Q}_d$. This time however, we seek to eliminate the ambiguity still present in the left stabilator (L ST) channel. The L ST plot in Fig. 5 shows this probability "bouncing" at the onset of failure. From data not explicitly shown in Fig. 5, the ambiguity is determined to be coming from the yaw channel. We therefore seek to demonstrate the technique of Method 1 further and improve performance by tuning the lateral diagonal $\mathbf{Q}_d$ entry Q_{d_r}.

Fig. 6 displays the result of this tuning method. A comparison of the L ST and YAW plots from Figs. 5 and 6 shows that we can eliminate the ambiguity in the L ST channel at the cost of some performance degradation in the YAW channel.

Having performed the tuning in Methods 1, 2, and 3, we arrive at a design which provides unambiguous failure detection capability in all channels with no false alarms. No attempts are made to do any optimization with the tuning, since explicit performance *specifications* are not provided at the outset. Rather, Fig. 6 demonstrates the potential of the MMAE failure detection algorithm.

Method 4: Tuning for Actuator Uncertainty

This last method attempts to account for model deficiencies related to actuator effects. It is not included in our final design, but presented for possible future use. Here, a fictitious source of scalar white noise $w_B(t)$, of strength q, is added to the model dynamics equation, from which (3) is generated as an equivalent discrete-time model [8]:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{G}\mathbf{w}(t) + \mathbf{B}_{\text{act col}}w_B(t) \tag{6}$$

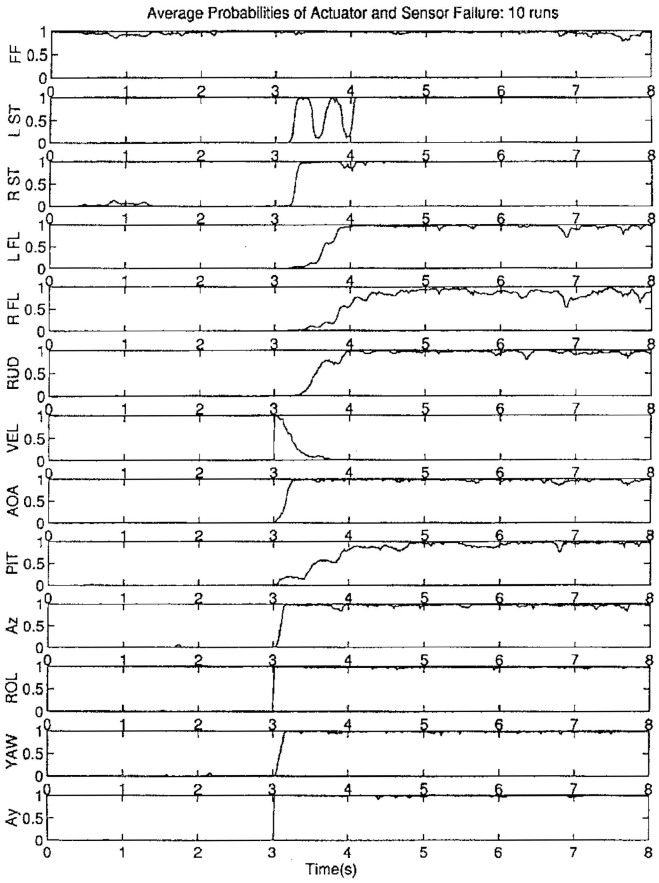

Fig. 4. Single-failure probability convergence, $Q_{d_{\text{long}}}$ tuning.

where $\mathbf{B}_{\text{act col}}$ is the column of the original $\mathbf{B}$ corresponding to the actuator of interest. Here, the continuous pseudonoise is brought through the relevant actuator dynamics stripped from the basic unaugmented state-space representation; the corresponding covariance differential equation for the Kalman filter propagation cycle would then be:

$$\dot{\mathbf{P}}(t) = \mathbf{AP}(t) + \mathbf{P}(t)\mathbf{A}^T + \mathbf{GQG}^T + \mathbf{B}_{\text{act col}} q \mathbf{B}_{\text{act col}}^T. \tag{7}$$

The advantage is that one can directly tune in order to address *poor single actuator* failure detection performance.

VI. DUAL-FAILURE IDENTIFICATION

Using a filter design which incorporates tuning Methods 1, 2, and 3 from the previous section, we proceed to investigate the ability of the MMAE algorithm to detect a second failure. The dual-failure scenarios are simulated with the first failure occurring at 3.0 s and the second two or more seconds later. This allows enough time for the first failure to be declared before the second is inserted. Other ways of inserting the second failure are possible, but this one allows us to focus on the second failure detection performance with the single-failure performance acheived in Fig. 6 as a baseline. The results shown in Fig. 7 are qualitatively placed into one of four categories. "Good" results demonstrate second failure performance commensurate with, or better, than that achieved for a lone single failure (see Fig. 6). They are often characterized by a probability lock and hold. "Fair" results demonstrate performance somewhat degraded from when a lone single failure occurs. These are often characterized by significant dropouts from probability lock. "Poor" results demonstrate performance drastically degraded from when a single lone failure occurs. They are often characterized by some spiking in the correct probability channel, but no lock whatsoever. "ND" results indicate no detection. The second failure is completely missed or falsely declared.

From Fig. 7, we see that in general, dual failure probability performance is quite good. In fact, owing greatly to the addition of pseudonoise, our dual failure performance within the more realistic, nonlinear, aircraft simulation environment produces better results than that seen initially in the linearized simulations of past research [12, 14]. There exists however, a couple of trends within the trouble spots worth mentioning.

Over eighty percent of the dual-failure difficulties lie within three identifiable problem areas: first

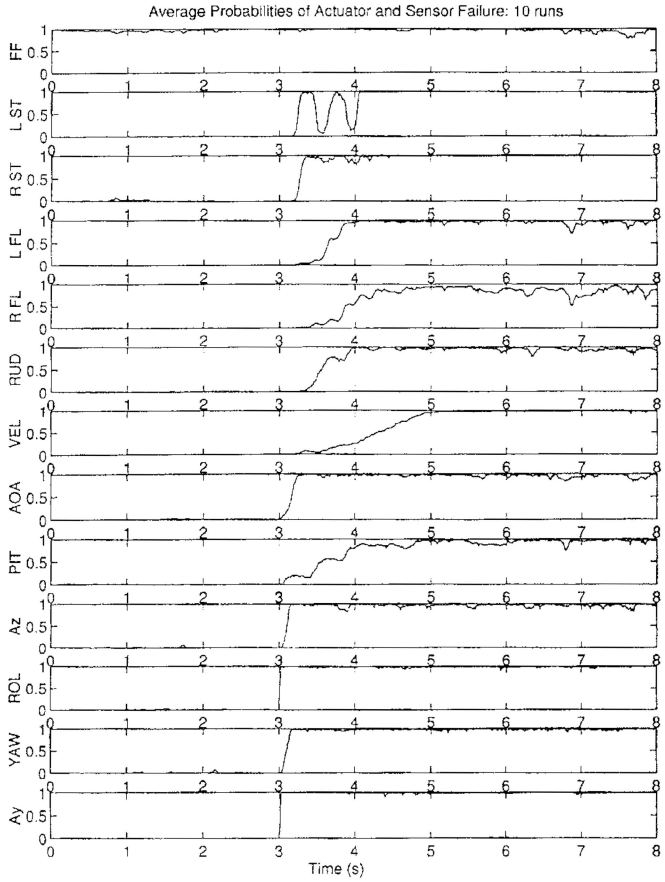

Fig. 5. Single-failure probability convergence, $Q_{d_{\text{long}}}$ and **R** tuning.

failure = stabilator, second failure = flaperon, and yaw sensor/rudder ambiguity. The two rows associated with the stabilators (LS, RS) show a high concentration of shaded squares ("Fair", "Poor", and "ND" ratings). We attribute this primarily to diminished state excitation. During benign flight conditions such as are used in this research, the system state must be artificially excited by control surfaces. Without state excitation, there can be no failure detection. Research is being conducted on optimal, subliminal, methods for "dithering" the control surfaces, but in our case, continuous, subliminal, sinusoids are used. When a stabilator actuator fails, a primary source of control authority is lost. With no reconfigurability in our dither strategy, state excitation via dithering is significantly diminished, resulting in difficulties declaring the second failure. The second problem area is reflected in the concentration of shaded squares in the LF and RF columns, i.e., when the second failure is a flaperon. This problem is also traceable to system state excitation, but for a different reason. The Block-40 FCS features an aileron-to-rudder interconnect (ARI) which provides for "feet on the floor" coordinated turns. In this application, it also provides unwanted coupling when we attempt to dither the flaperons and rudder at separate levels. The result of ARI interference is less-than-desirable deflection levels of the flaperons. This reduces its effect on state excitation and subsequently, its actuator failure identifiability. Fig. 6 shows the comparative weakness in single failure performance for flaperons, and the difficulties are exacerbated in the second failure case. When the first failure is a stabilator, overall reduced state excitation hurts performance, and when the first failure is a sensor, it could be the lack of that sensor's pertinent information which degrades detection performance. Removing the ARI for dithering could alleviate this problem area. The last problem area results from the inherent yaw sensor/rudder ambiguity. When the yaw sensor fails, there is no way to detect a rudder actuator failure, and when the rudder actuator fails, there is not enough yaw axis excitation to discover a yaw sensor failure at all. Addition of a rudder position sensor, or some third source of information, would alleviate this problem.

These dual-failure results are particularly encouraging because the tuned algorithm did not experience a proportional, or exponential, increase in performance difficulty when going from single to dual failures. The failure detection algorithm has proven itself capable within the nonlinear, high-order F-16 simulation environment which is the closest thing to real world characteristics.

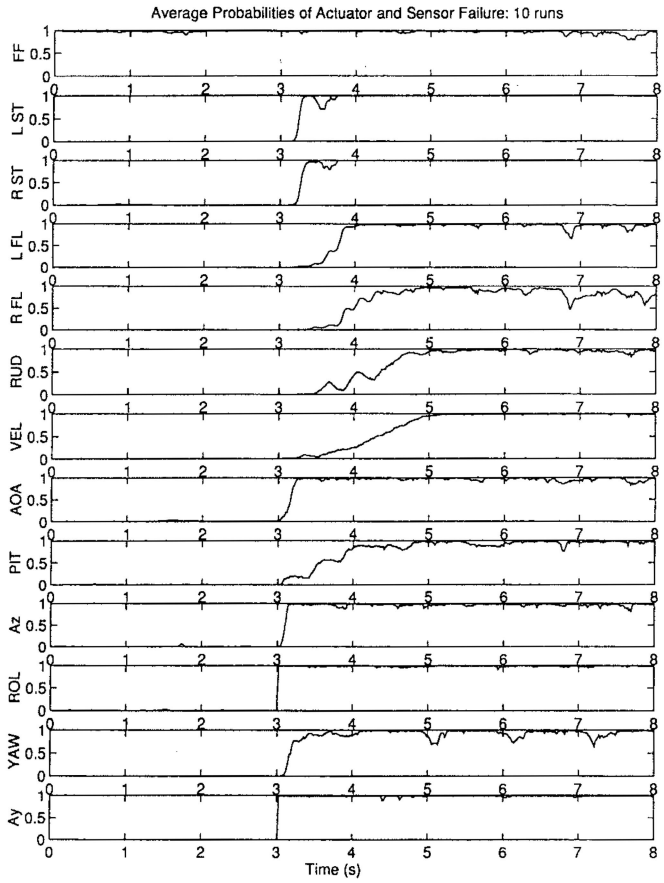

Fig. 6. Single-failure probability convergence, $Q_{d_{\text{long}}}$, $\mathbf{R}$, and $Q_{d_{\text{lat}}}$ tuning.

VII. RESIDUAL MONITORING

Research performed in the past identified an additional source of failure information within the residual terms of the algorithm, $\mathbf{r}_k$ [12]. Kalman filtering theory predicts that residuals within *well-matched* filters ought to be zero-mean, white, Gaussian, and of covariance $\mathbf{A}_k$. It has been suggested that simple tests be constructed which exploit these predictable characteristics [12]. A running average, for example, might constitute a zero-meanness test, while the number of zero crossings could test for whiteness, and magnitudes might be used in a variance test. Noting though, that our models are poorly matched initially and that substantial amounts of pseudonoise are required for good probability convergence, these predictable characteristics may no longer be valid. We wish to examine the scalar residual terms to see if such simple "additional voter" tests still exist.

To perform this investigation, the single-failure algorithm utilizing tuning methods 1, 2, and 3 is once again employed. Here, for each simulated single failure, the time histories of the seven scalar residuals (forward velocity, angle of attack, pitch rate, normal acceleration, roll rate, yaw rate, lateral acceleration) are recorded. For any given failure, the residuals in which we are interested come from the fully functional hypothesis filter and that particular failure hypothesis filter. This is because they are the best matched filters during the unfailed (0 to 3 s and failed (3 to 8 s) portions of a single-failure simulation. Fig. 8 shows data representative of sensor failures. A pitch rate sensor failure is inserted at 3.0 s) in each of ten simulations and the results averaged for statistical confidence. The first column presents the fully functional hypothesis filter residuals and the residual plots of the second column are from the failed pitch rate sensor hypothesis filter.

There are a number of trends observable from Fig. 8. First of all, over the first three seconds, when the fully functional hypothesis filter is correct, residual characteristics aren't white or Gaussian in general. This can be attributed to model mismatch, as anticipated. The second readily noticeable trend is that, with the exception of the pitch rate residual, the behavior of both sets of residuals is nearly identical. The pseudonoise added in Section V to account for model mismatch has effectively blurred the distinction between the two failure hypotheses. Only within the pitch residual do we observe evidence of any predicted behavior. This is because sensor failures show up directly and immediately within their respective scalar residual: it is here where the effects are strongest. Focusing then on the pitch rate residual pair, we see

Second Failure

First Failure	LS	RS	LF	RF	RUD	VEL	AOA	PIT	Az	ROL	YAW	Ay
LS		Good	ND	Poor	Good	Good	Good	Poor	Good	Fair	Fair	Fair
RS	Good		Poor	ND	Good	Good	Good	Poor	Good	Fair	Fair	Fair
LF	Good	Good		Good	Good	Good	Good	Good	Good	Good	Good	Good
RF	Good	Good	Good		Fair	Good	Fair	Good	Good	Good	Good	Good
RUD	Good	Good	Good	Good		Good	Good	Good	Good	Good	ND	Good
VEL	Good	Good	Good	Good	Good		Good	Good	Fair	Good	Good	Good
AOA	Good	Good	Good	Good	Good	Good		Good	Fair	Good	Good	Good
PIT	Good	Good	Fair	Fair	Fair	Good	Good		Good	Good	Good	Good
Az	Good	Good	Poor	Fair	Good	Good	Good	Good		Good	Good	Good
ROL	Good	Good	Good	Good	Good	Good	Good	Good	Good		Good	Good
YAW	Good	Good	Fair	Good	Poor	Good	Good	Good	Good	Good		Good
Ay	Good	Good	Fair	Fair	Good	Good	Good	Good	Good	Good	Good	

Fig. 7. Dual-failure performance summary matrix.

that zero-mean, white, Gaussian behavior has been maintained (if only in the affected scalar residual) when the assumed hypothesis is correct. Oscillations and bias are observable in the fully functional pitch rate residual after the failure has been inserted, while a cessation of such behavior is simultaneously noted in the failed-sensor filter pitch rate residual. Effectively, for sensor failures, the zero-meanness and whiteness tests are retained, with the variance test being lost owing to pseudonoise addition. Fortunately, the noted oscillations provide an opportunity for a replacement test. The input dither which comes through the failed scalar residual can be readily detected since we know the frequency of dither ahead of time. These results indicate that for sensor failures, scalar residual monitoring remains a viable failure detection technique.

No such definitive failure indication is seen for actuator failures. Unlike sensor failures, the effects of an actuator failure must propagate through the system and, in general, reveal itself in multiple sensors. Therefore, there is no one scalar residual to seek for an additional vote. The problem is that our tuning efforts mask the predicted effects in much the same way as we observed in the unaffected sensor scalar residuals in Fig. 8. However, there is *some* useful information which shows up in the yaw residual for rudder failures and in the acceleration residuals for stabilator and flaperon failures. It is conceivable that some kind of comparison test between the fully functional and failed hypothesis scalar residual counterparts could be generated as an additional vote. This would have to be pursued further in future investigations.

Tuning for probability convergence as done in Section V and residual monitoring are not mutually beneficial. Because probability convergence is our primary goal, we must accept the degradation in residual monitoring performance. Based upon these results however, we needn't abandon it entirely. Good indications of sensor failure still exist, and with some appropriate logic code, can be exploited in the future.

VIII. FLIGHT ENVELOPE RESEARCH

A final area of interest in our research is to investigate the robustness of the MMAE failure detection algorithm. The SRF simulation environment provides a unique opportunity to test our design throughout the flight envelope. As a first step towards

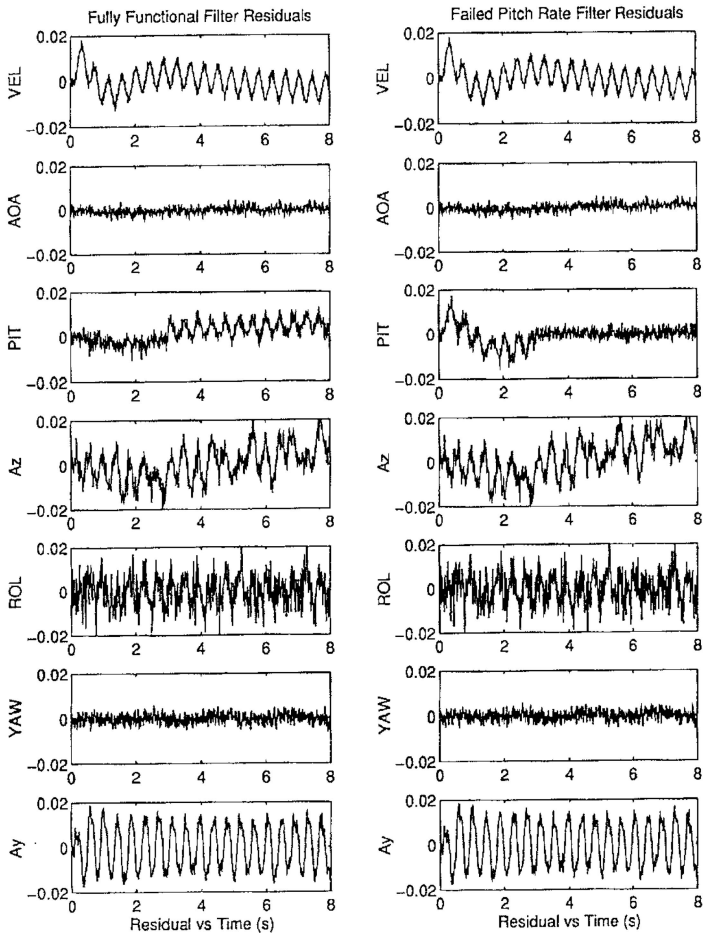

Fig. 8. Residual monitoring pair for pitch rate sensor failure.

full envelope coverage, we wish to determine whether gain scheduling or perhaps a manageable number of discrete designs will be more appropriate. To do this, we take our single-failure design, operate it at flight envelope points away from the nominal, and characterize the performance. Specifically, discrete steps are taken along the Mach and Altitude axes until performance degradation becomes unacceptable. This region defines a local envelope which we can discretize for further investigation. In Fig. 9, points 1 through 20 comprise this local envelope of interest. Four extrema, points 21 through 24, are also included for longer range trends. The exhaustive single-failure simulation of Section V is then run at each of the 24 points so that trends in performance degradation can be observed. For space considerations, probability convergence plots are not included here [4]. Instead, the important trends are discussed in some detail.

This research provides three important results which will serve to guide further work in this area: accurate trim variables are critical to acceptable performance; false alarms rather than missed alarms are the limiting factor; and lack of robustness points to a need for gain scheduling.

First, recalling that our design is perturbation based, and noting that sensor values and control variables passed to the algorithm are absolute, there is a heightened emphasis placed on trim values. Specifically, our design requires a nominal steady-state value for stabilator deflection as well as forward velocity, angle of attack, and normal acceleration. The algorithm uses these values to subtract off the appropriate inputs in order to generate perturbation inputs. We find that if these trim values are not accurate, an initial *perturbation* error is generated which leads to false alarms. Therefore, future consideration must be given to an accurate method of maintaining these trims. Possibilities include scheduling based on simulation data, or perhaps some form of on-line determination.

A second finding is that false alarms are the limiting factor in performance degradation. One might intuit that lower dynamic pressure with the same dithering strategy would result in lower state excitation and more missed alarms, while higher dynamic pressure with the same dithering strategy would violate small-perturbation hypotheses and result in false alarms. Our research doesn't bear this out. In both directions of dynamic pressure change, false alarms become the problem, and we attribute this to model mismatch. Recall that the MMAE design is based on a linearized model about a given trim. Before

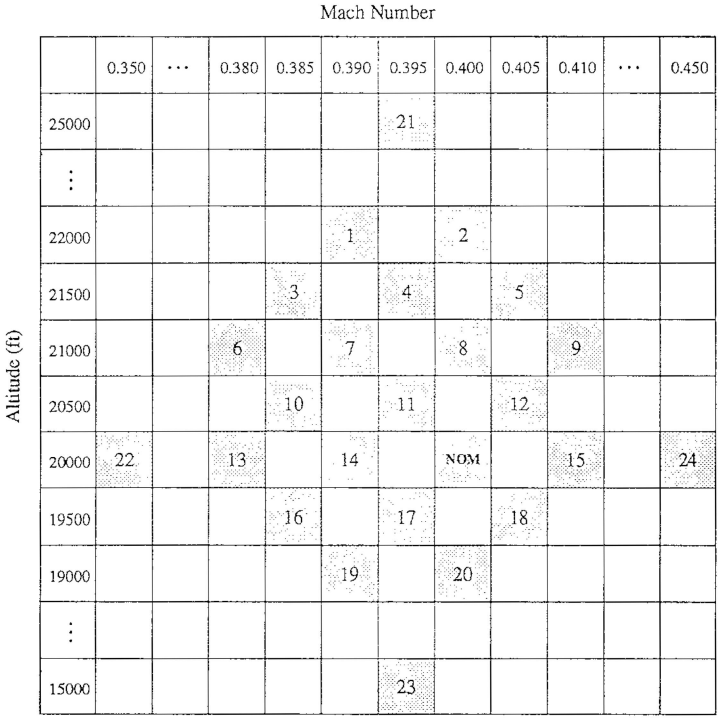

Fig. 9. Flight envelope research points matrix.

dynamic pressure can have the impact just described, model mismatch effects inherent when operating at a different point in the envelope dominate. Fig. 3 demonstrates the general effect of model mismatch, namely that false alarms present the pressing concern. Interestingly, points 9, 12, Nominal (NOM), 17, and 19 of Fig. 9 define an axis of approximately constant dynamic pressure along which reasonably acceptable performance is achieved. So while higher and lower dynamic pressure, along with model mismatch, yield false alarms, variations corresponding to constant dynamic pressure and current levels of tuning seem to yield minimal degradation.

The third and final result indicates that gain scheduling may be the best way to approach full-envelope coverage. One need only look at the small region of operability in Fig. 9 (0.03 Mach, 3000 ft) to see that our algorithm isn't very robust. This seems smaller, still, considering the addition of substantial pseudonoise which might have masked the model mismatch we induce off nominal. The small neighborhood effectively prohibits the possibility of having a small number of discrete designs cover the envelope. Gain scheduling or a similar concept will be required in future developments of this technology.

IX. CONCLUSIONS

The SRF VISTA F-16 simulation provides an opportunity to apply the MMAE failure detection algorithm in as near a real-world environment as is currently possible and further determine its potential. From this application, we learn that model mismatch due to reduced order, linearized modeling leads to false alarms and must be accounted for via addition of pseudonoise to assumed design models. Upper limits on pseudonoise addition do exist, however, which if passed, result in missed alarms. Intelligent pseudonoise addition such as that demonstrated in Methods 1 through 4, though, can result in acceptable probability convergence. Also, the algorithm did not suffer from insurmountable difficulties when the level of complexity was raised from single to dual failures. Finally, additional model mismatch which results from operating away from trim quickly yields false alarms. Thus, a lack of robustness points to a gain scheduling approach to full flight envelope implementation.

Overall, the MMAE performed well against the fully nonlinear truth model environment. With no performance specifications driving our research, there was no pressing effort to optimize the results. Implementation-specific requirements must always be considered, with the techniques just described being applied appropriately. The results of this effort argue strongly for continued research and development in this area.

REFERENCES

[1] Athans, M., et al. (1977)
The stochastic control of the F-8C aircraft using a multiple model adaptive control (MMAC) method—Part I: Equilibrium flight.
IEEE Transactions on Automatic Control, **AC-22**, 5 (Oct. 1977), 768–780.

[2] Century Computing Inc. (1992)
Simulation/Rapid-Prototyping Facility (SRF) F-16 VISTA Simulation Engineer's Guide.
Century Computing Inc., Dec. 1992.

[3] Century Computing Inc. (1992)
Simulation/Rapid-Prototyping Facility (SRF) F-16 VISTA Simulation User's Manual.
Century Computing Inc., Oct. 1992, Draft Sun/UNIX version.

[4] Eide, P. K. (1994)
Implementation and demonstration of a multiple model adaptive estimator failure detection system for the F-16.
M.S. thesis, AFIT/GE/ENG/94D-06, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, Dec. 1994.

[5] Hanlon, P. D. (1992)
Failure identification using multiple model adaptive estimation for the LAMBDA flight vehicle.
M.S. thesis, AFIT/GE/ENG/92D-19, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, Dec. 1992.

[6] Lane, D. W. (1993)
Mutiple model adaptive estimation applied to the LAMBDA URV for failure detection and identification.
M.S. thesis, AFIT/GE/ENG/93D-23, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, Dec. 1993.

[7] Martin, R. M. (1990)
LQG synthesis of elemental controllers for AFTI/F-16 adaptive flight control.
M.S. thesis, AFIT/GE/ENG/90D-36, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, Dec. 1990.

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

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

[10] Maybeck, P. S., and Hanlon, P. D. (1993)
Performance enhancement of a multiple model adaptive estimator.
IEEE Conference on Decision and Control, San Antonio, TX, Dec. 1993, 462–468.

[11] Maybeck, P. S., and Pogoda, D. L. (1989)
Multiple model adaptive controller for the STOL F-15 with sensor/actuator failures.
In *Proceedings of the 28th Conference on Decision and Control*, Dec. 1989, 1566–1572.

[12] Menke, T. E. (1992)
Multiple model adaptive estimation applied to the VISTA F-16 with actuator and sensor failures.
M.S. thesis, AFIT/GA/ENG/92J-01, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, June 1992.

[13] Menke, T. E., and Maybeck, P. S. (1992)
Multiple model adaptive estimation applied to the VISTA F-16 flight control system with actuator and sensor failures.
In *Proceedings of the IEEE National Aerospace and Electronics Conference*, Dayton, OH, May 1992, 441–448.

[14] Menke, T. E., and Maybeck, P. S. (1993)
Sensor/actuator failure detection in the VISTA F-16 by multiple model adaptive estimation.
In *Proceedings of American Control Conference*, San Francisco, June 1993, 3135–3140.

[15] Pogoda, D. L. (1988)
Multiple model adaptive controller for the STOL F-15 with sensor/actuator failures.
M.S. thesis, AFIT/GE/ENG/88D-23, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, Dec. 1988.

[16] Stevens, R. D. (1989)
Characterization of a reconfigurable multiple model adaptive controller using a STOL F-15 model.
M.S. thesis, AFIT/GE/ENG/89D-52, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, Dec. 1989.

[17] Stratton, G. L. (1991)
Actuator and sensor failure identification using a multiple model adaptive technique for the VISTA F-16.
M.S. thesis, AFIT/GE/ENG/91D-14, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, Dec. 1991.

Capt. Peter K. Eide was born on June 5, 1966, in Menomonee Falls, WI. He pursued a B.S. in electrical engineering and a commission into the U.S. Air Force through ROTC at the University of Wisconsin, Madison. He received the M.S. degree in electrical engineering from the Air Force Institute of Technology, Wright-Patterson Air Force Base, OH, in 1994.

Upon graduation from the University of Wisconsin he was assigned as Electronics Warfare Systems Engineer at the Warner Robins Air Logistics Center in Warner Robins, GA. While in this position, he brought a $69M Intermediate Test Set on line for the AN/ALQ-131 tactical ECM program. Capt. Eide then volunteered and was accepted into the graduate program at the Air Force Institute of Technology. After receiving the M.S. degree, he accepted a follow-on assignment to the Flight Dynamics Directorate of the Wright Laboratory at Wright-Patterson AFB, OH.

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, where he is currently Professor of Electrical Engineering. Current research interests concentrate on using optimal estimation techniques for guidance systems, tracking, adaptive systems and failure detection purposes.

Dr. Maybeck is the author of numerous 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 in 1994 by Navtech). 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 1994, 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, OH 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 senior member of the A.I.A.A., and is the current I.E.E.E. Dayton Section Student Activities Chairman and a member of the I.E.E.E. Executive Committee of Dayton. He previously served as Chairman of the local Automatic Control Group.