

# Performance Enhancement of a Multiple Model Adaptive Estimator

PETER S. MAYBECK, Fellow, IEEE

PETER D. HANLON

Air Force Institute of Technology

We describe performance improvement techniques for a multiple model adaptive estimator (MMAE) used to detect and identify control surface and sensor failures on an unmanned flight vehicle. Initially, failure identification was accomplished within 4 s of onset, but by removing the “ $\beta$  dominance” effects, bounding the hypothesis conditional probabilities, retuning the Kalman filters, increasing the penalty for measurement residuals, decreasing the probability smoothing, and increasing residual propagation, the identification time was reduced to 2 s.

## I. INTRODUCTION

The requirement for high performance aircraft with instabilities that are beyond the capability of human pilots to counteract has produced the demand for sophisticated flight control systems. These flight control systems require accurate models of the aircraft to provide the necessary flight control compensation and performance. The effects of an inaccurate model can be disastrous, particularly if a sensor or a flight control surface fail and the flight control system acts on the assumption that there is no failure. One method for the control system to adapt to significant changes, like these failures, is by using a multiple model adaptive estimator (MMAE) to detect and identify the failure and then inform the flight control system of this failure.

This paper reviews the MMAE algorithm and then describes various performance enhancement techniques that were researched. In Section II we give a brief overview of the MMAE and a description of the two components that are used to build a MMAE, the Kalman filters and the hypothesis testing algorithm. The intent of this work is to present various methods, which we describe in Section III, of adjusting the MMAE to provide better performance. The MMAE correctly identified the modeled failures, so we sought to increase the MMAE performance by decreasing the time that it took to converge to this correct failure identification. The methods that we describe are removal of “ $\beta$  dominance” effects, bounding the hypothesis conditional probabilities, tuning the Kalman filters, increasing the scalar penalty for large residuals, decreasing the probability smoothing, and enlarging the residuals by propagating without updating the Kalman filters.

## II. MULTIPLE MODEL ADAPTIVE ESTIMATION ALGORITHM

### A. Overview

A MMAE consists of a bank of parallel Kalman filters, each with a different internal model, and a hypothesis conditional probability computation as shown in Fig. 1. The Kalman filters are provided a measurement vector ( $\mathbf{z}$ ) and the input vector ( $\mathbf{u}$ ), and produce a state estimate ( $\hat{\mathbf{x}}_k$ ) and a residual ( $\mathbf{r}_k$ ). Each Kalman filter has a different failure model that it uses to form the state estimate and the residual, so the sizes of the residuals from the various filters give a relative indication of how adequately each of these models represent the actual failure status of the aircraft. The residuals are used by the hypothesis testing algorithm to assign relative probabilities ( $p_k$ ) to each of the hypotheses that were used to form the Kalman filter models. The individual probabilities indicate how

Manuscript received March 24, 1993; revised July 25, 1994.

IEEE Log No. T-AES/31/4/14137.

Authors' address: Department of Electrical and Computer Engineering, 2950 P St., Air Force Institute of Technology/ENG, Wright-Patterson AFB, OH 45433.

U.S. Government work not protected by U.S. copyright.

---

0018-9251/95/\$10.00 © 1995 IEEE

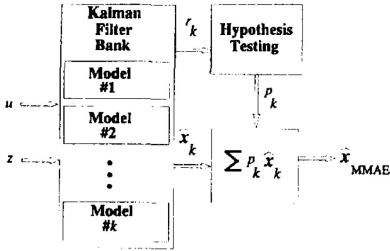


Fig. 1. MMAE.

correct each of the Kalman filter models are, and can be used to weight the individual state estimates appropriately, to form a probability-weighted average of the state estimate, ( $\hat{x}_{\text{MMAE}}$ ).

A simple example will demonstrate the operation of the MMAE. Let's assign Kalman filter 1 the fully functional aircraft model, and Kalman filter 2 the model based on a left elevator failure. Each of these filters would use the commanded input ( $\mathbf{u}$ ) and compute an estimate of what the states and measurements should be. Included in this computation is a model of the noise in these measurements, which the filters use to calculate an expected value and standard deviation of the measurement. The actual measurement is then used to form the residual, which is the difference between the actual measurement and what the filter predicts for the measurement. If the aircraft has no failures, then the residual from filter 1 would be much smaller, relative to its own internally computed standard deviation, than the residual in filter 2. The hypothesis testing algorithm would take all of the residuals from the Kalman filters and assign the highest probability to filter 1 since it has the smallest residual. Now, let us assume that a left elevator failure occurs, thus the residual in filter 2 would become quite small and the residual in filter 1 would grow. The hypothesis testing algorithm would then assign less probability to filter 1 and more to filter 2. The specific workings of these blocks are explained in the following sections.

## B. Kalman Filters

We assume that the reader is familiar with Kalman filtering and present the specific Kalman filter equations used in the MMAE. Maybeck [2] presents a development of Kalman filtering theory, if further Kalman filtering background is needed.

We require a discrete-time system model to implement this algorithm on a digital computer and, of course, we need to minimize the computational loading. Previous research [12] developed a continuous-time system model for the flight vehicle, which was converted to a discrete-time equivalent model [1]. To minimize computational loading, we assume that the system is time invariant, giving us

constant, precomputable coefficients for the system model ( $\Phi, \mathbf{B}_d, \mathbf{G}_d, \mathbf{H}$ ), and we use a linear, steady state Kalman filter, which gives us constant Kalman filter gains ( $\mathbf{K}$ ).

The discrete-time equivalent system model has the form

$$\begin{aligned}\mathbf{x}(t_i) &= \Phi \mathbf{x}(t_{i-1}) + \mathbf{B}_d \mathbf{u}(t_{i-1}) + \mathbf{G}_d \mathbf{w}_d(t_{i-1}) \\ z(t_i) &= \mathbf{H} \mathbf{x}(t_i) + \mathbf{v}(t_i)\end{aligned}\quad (1)$$

where

$\mathbf{x}$  is the system state vector,

$\Phi$  is the state transition matrix, the discrete equivalent of the system dynamics matrix,

$\mathbf{B}_d$  is the discrete equivalent of the system control input matrix,

$\mathbf{u}$  is the system input vector,

$\mathbf{G}_d$  is the discrete equivalent noise input matrix,

$\mathbf{w}_d$  is an additive white dynamics noise input with zero mean and

$$E\{\mathbf{w}_d(t_i)\mathbf{w}_d^T(t_j)\} = \begin{cases} \mathbf{Q}_d(t_i) & t_i = t_j \\ \mathbf{0} & t_i \neq t_j \end{cases} \quad (2)$$

$z$  is the measurement vector,

$\mathbf{H}$  is the system output matrix,

$\mathbf{v}$  is an additive white measurement noise input, independent of  $\mathbf{w}_d$ , with zero mean and

$$E\{\mathbf{v}(t_i)\mathbf{v}^T(t_j)\} = \begin{cases} \mathbf{R}(t_i) & t_i = t_j \\ \mathbf{0} & t_i \neq t_j \end{cases}. \quad (3)$$

Using this model we get the Kalman filter state estimate propagation equation:

$$\hat{\mathbf{x}}(t_i^-) = \Phi \hat{\mathbf{x}}(t_{i-1}^+) + \mathbf{B}_d \mathbf{u}(t_{i-1}) \quad (4)$$

where  $t_i^-$  is the time just before the  $i$ th time sample and  $t_{i-1}^+$  is the time just after the  $i-1$  time sample, and the Kalman filter update equation:

$$\hat{\mathbf{x}}(t_i^+) = \hat{\mathbf{x}}(t_i^-) + \mathbf{K}[\mathbf{z}(t_i) - \mathbf{H} \hat{\mathbf{x}}(t_i^-)]. \quad (5)$$

This model also gives us the Kalman filter residual:

$$\mathbf{r}(t_i) \stackrel{\Delta}{=} \mathbf{z}(t_i) - \mathbf{H} \hat{\mathbf{x}}(t_i^-) \quad (6)$$

which the hypothesis testing algorithm uses as a relative measure of how much the Kalman filter model differs from the true model. The residual is the difference between the true measurements ( $\mathbf{z}$ ), and the Kalman filter estimates of those measurements before they are taken ( $\mathbf{H} \hat{\mathbf{x}}(t_i^-)$ ), which are based on its model. Thus, if the Kalman filter model is correct, the residual will be small, otherwise the residual will be larger than anticipated when compared with the filter-computed residual covariance matrix, which is shown in (7) in the sequel.

### C. Hypothesis Testing

Maybeck shows [24: 228–229] that the Kalman filter residual is a white Gaussian sequence of mean zero and covariance

$$\mathbf{A}(t_i) = \mathbf{H}(t_i)\mathbf{P}(t_i^-)\mathbf{H}^T(t_i) + \mathbf{R}(t_i). \quad (7)$$

For our application, which assumes a time-invariant system, this covariance is constant and precomputable. For the  $k$ th elemental Kalman filter the covariance is

$$\mathbf{A}_k = \mathbf{H}_k \mathbf{P}_k \mathbf{H}_k^T + \mathbf{R}_k. \quad (8)$$

Since we know that the residual is a Gaussian vector with zero mean and covariance  $\mathbf{A}_k$ , we substitute these values into the known expression for a Gaussian conditional density function. Therefore we get that the conditional density function of the measurement ( $\mathbf{z}$ ) at  $t_i$ , given the Kalman filter model (based on one of  $k$  possible values of a vector of parameters that specifies an assumed failure status, i.e.,  $\mathbf{a} \in \{\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_k\}$ ) and the measurement history ( $\mathbf{Z}(t_{i-1}) = [\mathbf{z}^T(t_i) \cdots \mathbf{z}^T(t_{i-1})]^T$ ), as

$$\begin{aligned} f_{\mathbf{z}(t_i)|\mathbf{a}, \mathbf{Z}(t_{i-1})}(\mathbf{z}_i | \mathbf{a}_k, \mathbf{Z}_{i-1}) &= \beta \exp\{-\cdot\} \\ \beta &= \frac{1}{(2\pi)^{m/2} |\mathbf{A}_k|^{1/2}} \\ \{\cdot\} &= \left\{ -\frac{1}{2} \mathbf{r}_k^T(t_i) \mathbf{A}_k^{-1} \mathbf{r}_k(t_i) \right\} \end{aligned} \quad (9)$$

where  $m$  is the dimension of the measurement vector,  $\mathbf{z}(t_i)$ .

We desire the conditional probabilities of the various hypotheses,  $p_k(t_i) = \text{pr}(\mathbf{a} = \mathbf{a}_k | \mathbf{Z}(t_i) = \mathbf{Z}_i)$ , which we can compute using (9) and the previous conditional probabilities  $p_k(t_{i-1})$  via the following iteration [3]:

$$p_k(t_i) = \frac{f_{\mathbf{z}(t_i)|\mathbf{a}, \mathbf{Z}(t_{i-1})}(\mathbf{z}_i | \mathbf{a}_k, \mathbf{Z}_{i-1}) p_k(t_{i-1})}{\sum_{j=1}^K f_{\mathbf{z}(t_i)|\mathbf{a}, \mathbf{Z}(t_{i-1})}(\mathbf{z}_i | \mathbf{a}_j, \mathbf{Z}_{i-1}) p_j(t_{i-1})}. \quad (10)$$

### III. PERFORMANCE ENHANCEMENTS

We investigated various methods of improving the performance of the MMAE algorithm presented in Section II. The specific application that served as the basis for our investigation was the identification of flight control surface and sensor failures for the LAMBDA flight vehicle, an unmanned research vehicle developed by the Flight Control Division of the Flight Dynamics Directorate, Wright Laboratory. The primary performance objective was to identify all the failures correctly, with a secondary objective to identify the failures as quickly as possible. Since the correct identification was primary, we assumed that false alarms (incorrect declarations of a specific failure) were unacceptable. Therefore, we first developed an

MMAE algorithm that correctly identified the failure without false alarms, and then tried to improve the MMAE performance by decreasing the amount of time that was required to make the correct identification. Thus, the performance enhancements reported in this work are attempts to decrease the amount of time that the MMAE requires to converge to the correct hypothesis.

There are fifteen elemental filters in the MMAE algorithm for this application. One is designed under the assumption of a fully functional vehicle, with no actuator or sensor failures. Six are based upon an assumed failure in one of the six actuators (right or left elevator, right or left aileron, right or left rudder), and another eight assume a failure in one of the eight sensors (measurements of velocity, angle of attack, pitch rate, sideslip angle, roll rate, roll angle, or yaw rate).

The original performance of the MMAE is summarized in Fig. 2. Each plot in this figure presents the  $p_k$  time history results of a separate 10-run Monte Carlo simulation of a single sensor or actuator failure occurring at 2 s. The figure caption references various parameters that identify the configuration of the MMAE algorithm that produced the displayed results. The design number indicates which of three tuned Kalman filter designs were used, while the Dot parameter indicates the value of the scalar penalty for large residuals. PWINSIZ indicates the size of the data window over which the probabilities were smoothed, and NPROP indicates the number of propagations that were performed before an update was performed. These concepts are further amplified and results for various simulations are presented in subsequent sections. A representative sample of a specific failure simulation is shown in Fig. 3, where a right elevator failure occurs at 2 s. Note that the probability of no failure is the complement of the right elevator failure, while all the other filters maintain basically the minimum allowed probability (0.001). Also, we can observe the MMAE performance for this particular failure simply by looking at the filter probability for the filter that uses the correct failure model. For instance, we can completely characterize the MMAE performance for a right elevator failure simply by observing the probability plot for the right elevator filter in Fig. 3. Therefore, we can observe the MMAE performance for each of the failures that are modeled by the filter bank simply by combining these single failure simulation results into the single plot shown in Fig. 2. Clearly, the results for a right elevator failure shown in Fig. 3 are completely characterized by the right elevator plot in Fig. 2.

These results show that the MMAE does eventually declare the correct failure in every case, so attempts were made to decrease the convergence time to the correct failure declaration. First, two notable exceptions were found to the results represented

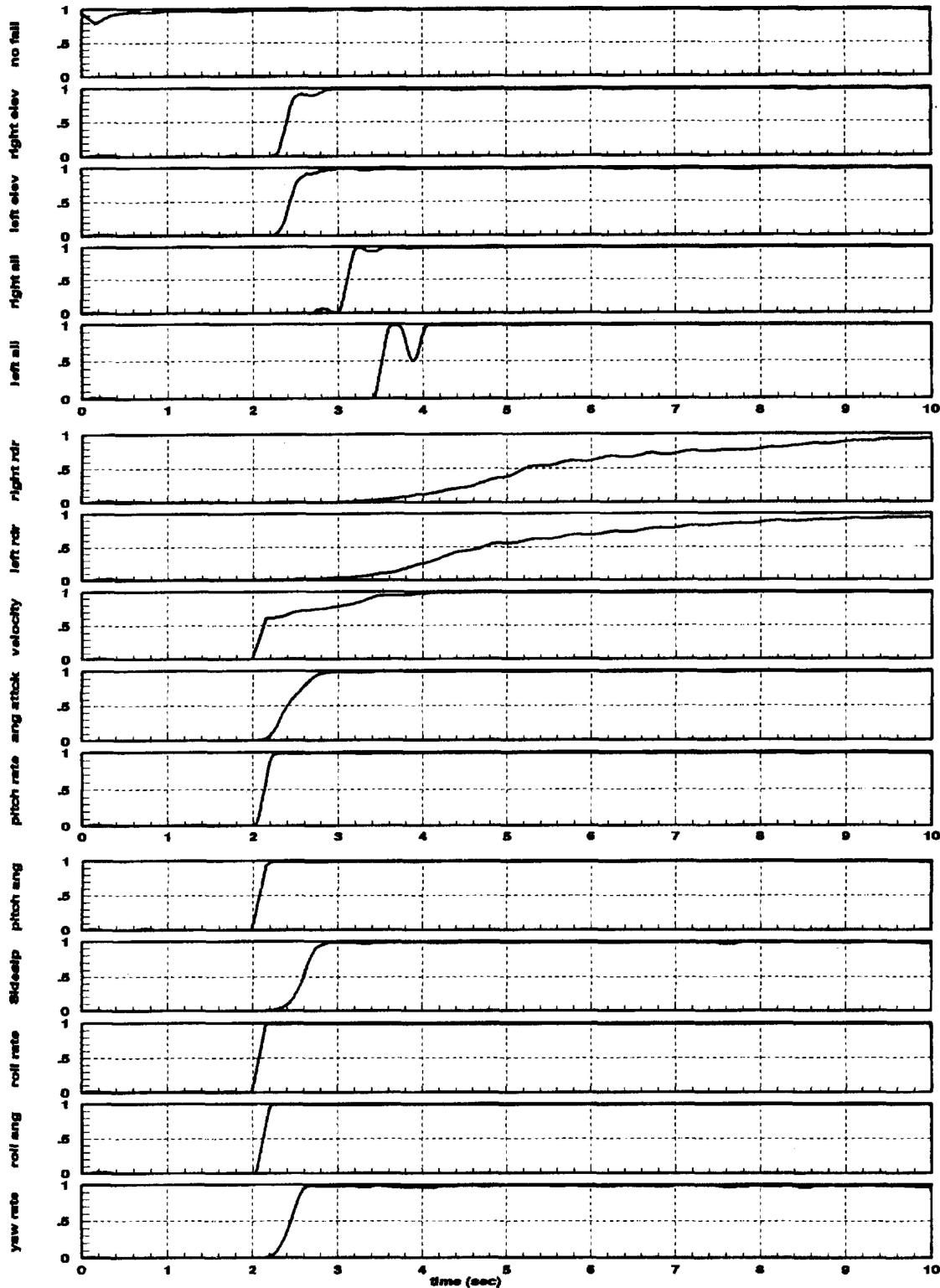


Fig. 2. Original MMAE performance. Design 1, Dot =  $-\frac{1}{2}$ , PWINSIZ = 10, NPROP = 1.

by Fig. 3. A right or left aileron failure produced momentary confusion in the sideslip and yaw rate sensor failure filters, as shown in Figs. 4 and 5. This momentary confusion caused about a 1 s delay in declaring the correct failure. Also, note that the right and left rudder failures produced much longer convergence times when compared with other failures,

as shown in Fig. 2. The performance enhancements described in this work specifically address the MMAE response to certain actuator failures, since the response to any sensor failure is generally rapid and unambiguous. In subsequent sections we describe our efforts to decrease the convergence times for the specific actuator failures.

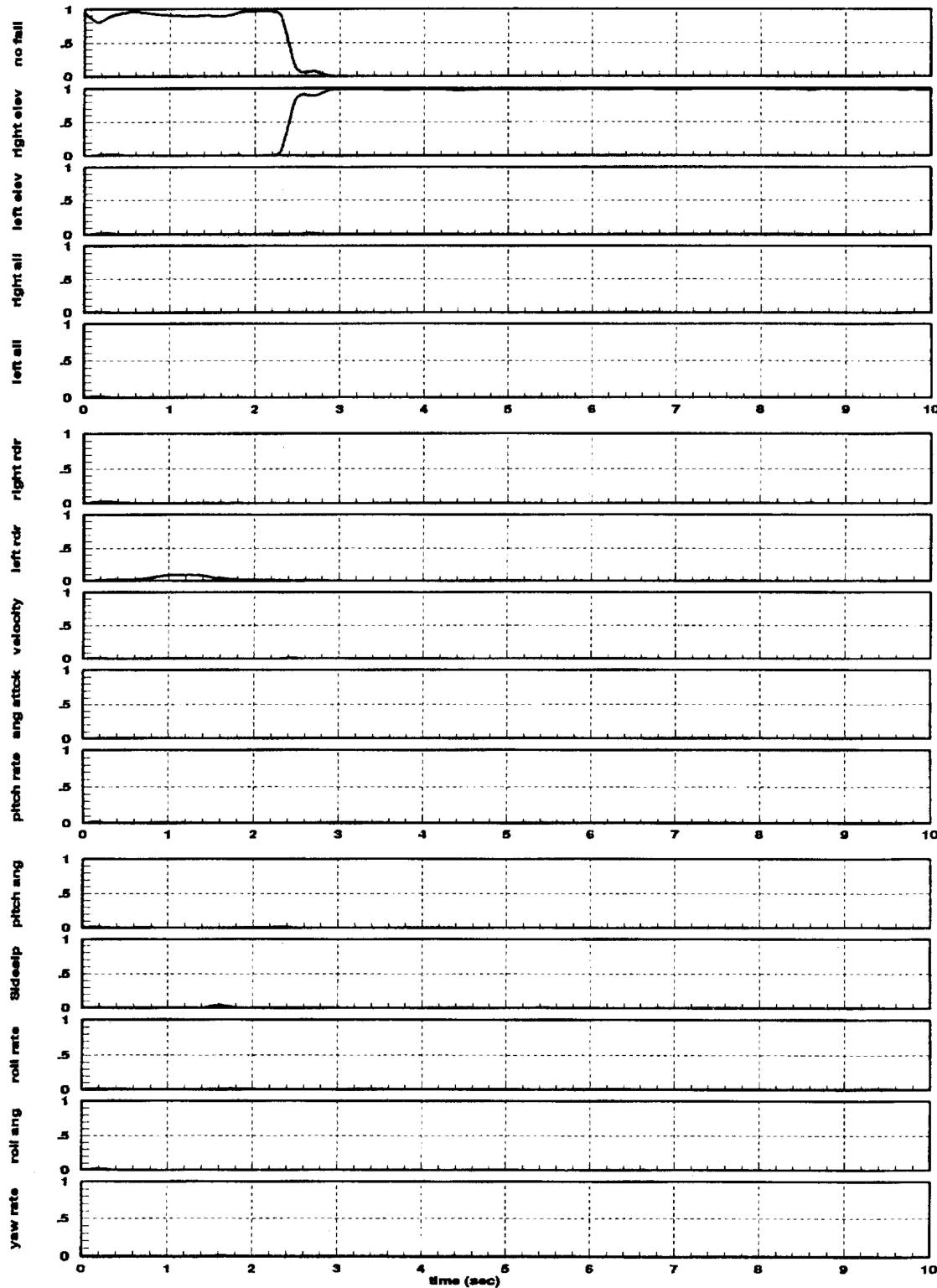


Fig. 3. Right elevator failure at 2 s. Design 1, Dot =  $-\frac{1}{2}$ , PWINSIZ = 10, NPROP = 1.

#### A. $\beta$ Dominance Effects

Maybeck and Stevens [5, 10] found that certain performance problems could be reduced by modifying (9). They altered the conditional density function in (9) by removing the  $\beta_k$  term, which was used originally to scale the area under the density function to be

equal to 1. If all the exponential terms,  $\{\cdot\}$  in (9), were approximately the same size for all elemental filters, one would desire all elemental filters to be deemed equally adequate. However, (9) and (10) would put the *highest* probability on the elemental filters with the *smallest*  $|A_k|$  value. This is an inappropriate weighting since the size of  $|A_k|$  has *nothing* to do

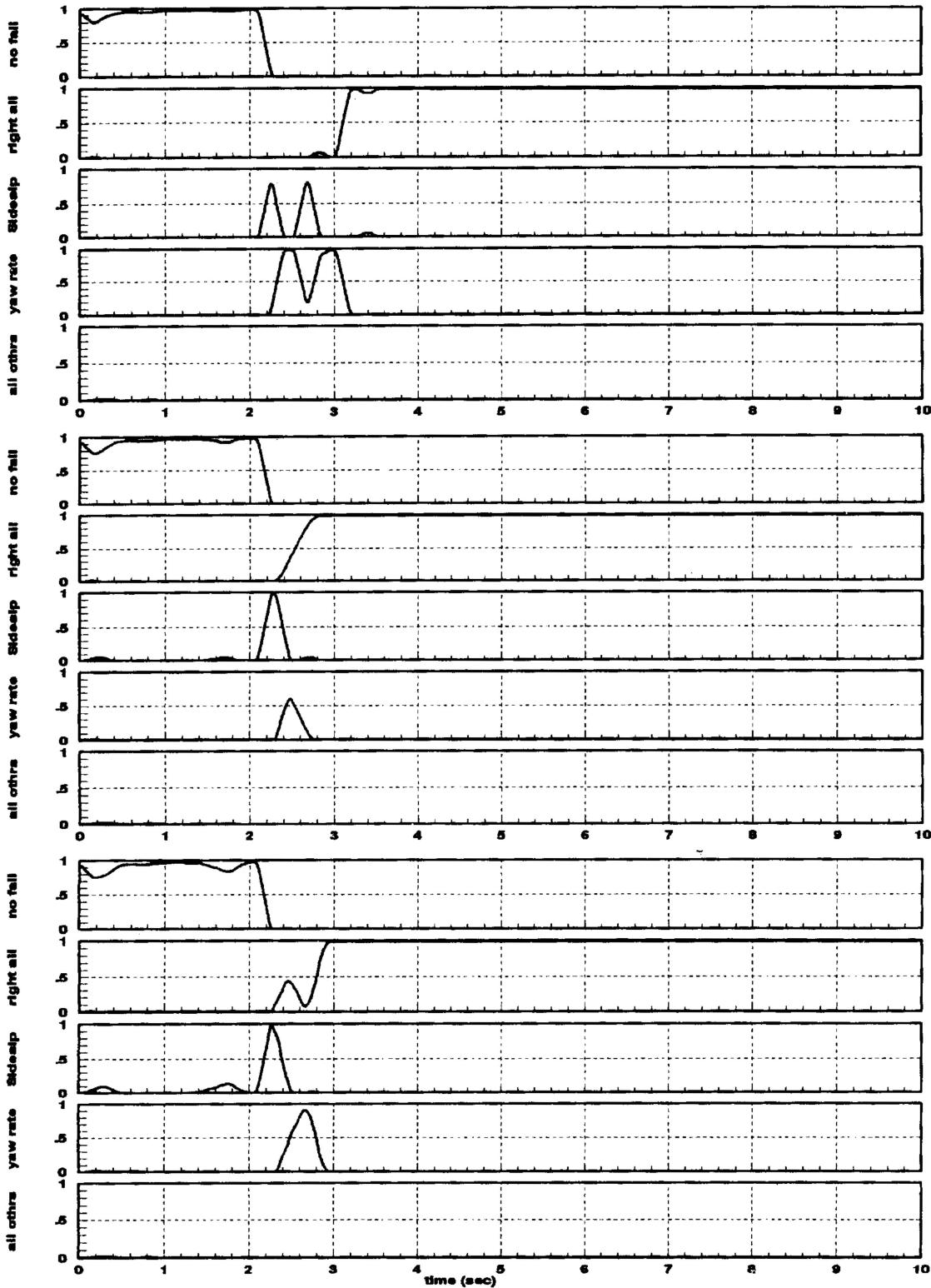


Fig. 4. Right elevator failure for various designs. Top, design 1. Middle, design 2. Bottom, design 3.

with the correctness of the hypothesis in matching the current real-world failure status. Since sensor failures exhibit themselves as a row of  $\mathbf{H}$  going to zero, filters based on the hypothesis of a failed sensor tend to have smaller  $|\mathbf{A}_k|$  values, and thus the MMAE will be prone to false alarms on sensor failures. The algorithm functions properly with the  $\beta_k$  term

removed because the denominator in (10) is the sum of all possible numerators, so the probabilities ( $p_k$ ) will still sum to 1 even if the area under each of the modified “densities,” the density functions in (9) with  $\beta_k$  removed, is no longer unity. This modification is said to remove the “ $\beta$  dominance effect” and is invoked throughout this research.

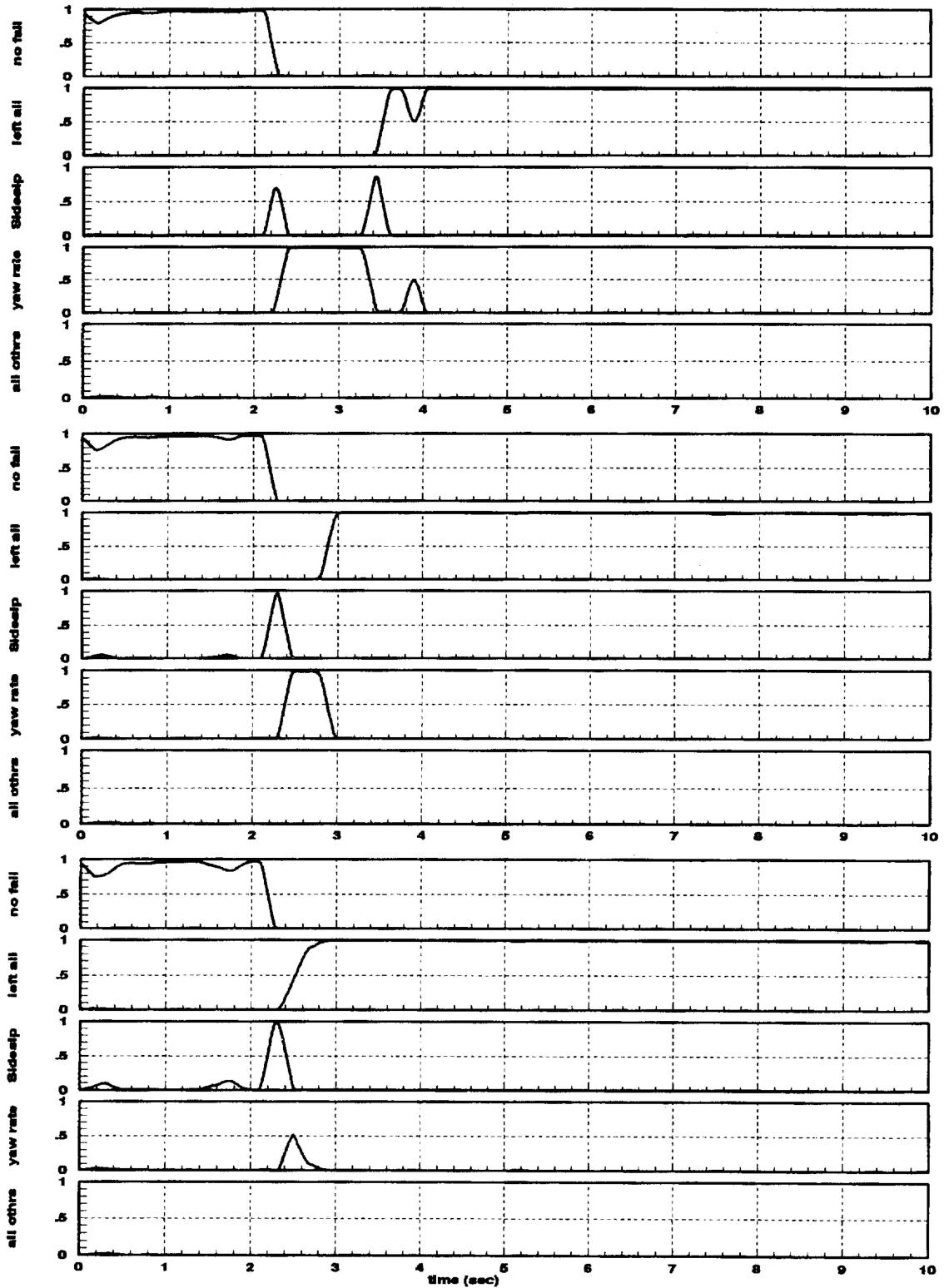


Fig. 5. Left elevator failure for various designs. Top, design 1. Middle, design 2. Bottom, design 3.

### B. Bounded Conditional Probabilities

Previous research [4, 5, 9–11] also found that a lower bound needed to be placed on the hypothesis conditional probabilities. The purpose of the MMAE is to make quick and accurate failure identifications, and it was found that if some of the probabilities were

allowed to get too small, it took a long time for the probability ( $p_k$ ) associated with the correct filter to build up when a change in the failure status occurred. This was due to the fact that the previous conditional probability,  $p_k(t_{i-1})$ , was so small for the new correct filter model and so large for the old correct model, that (10) had to be iterated several times before the

values would change significantly. Menke and Stratton [6–8, 11] found that 0.001 was a good lower bound on the conditional probabilities for a MMAE applied to a similar sensor/actuator failure detection system. This is the implementation of the hypothesis testing algorithm used for this application of the MMAE.

### C. Kalman Filter Tuning

As mentioned previously, an aileron failure produced some momentary confusion in the MMAE. We experimented with tuning the Kalman filters by increasing the assumed measurement noise variances in the models that produced this momentary confusion, in an attempt to decrease the confusion (ambiguous and/or false declarations of the modeled failure) during an aileron failure. Three filter designs were developed: design 1 used the sensor noise variances that were estimated from sensor measurement data, design 2 used nine times the estimated variances (increasing the noise standard deviation by a factor of three), and design 3 used 25 times the estimated variances (increasing the noise standard deviation by a factor of five). The overall MMAE performance using these various designs is compared in Fig. 6, with the specific performance for a right aileron failure in Fig. 4 and a left aileron failure shown in Fig. 5.

The best performance was obtained using design 2. Fig. 6 shows that the convergence time for an aileron failure using design 2 (Fig. 6, dashed line) was half of that for design 1 (Fig. 6, solid line). The specific aileron performance in Figs. 4 and 5 show that this convergence time is primarily due to a significant decrease in the amount of confusion caused by an aileron failure. Comparing the results from design 1 (top plots in Figs. 4 and 5) with those from design 2 (middle plots in Figs. 4 and 5), we see that design 2 has eliminated the second hump in the probabilities for a sideslip and yaw rate sensor failure. This caused the MMAE to identify the correct failure condition much faster.

Figs. 4, 5, and 6 show that the performance for design 3 (bottom plots in Figs. 4 and 5, dotted line in Fig. 6) is somewhat better than the performance for design 2, but the probabilities show an increase in fluctuations. We noted before that these results are an average of 10 Monte Carlo simulations, therefore the fluctuations in these plots indicate even more severe fluctuations during an actual flight, which would result in a larger number of momentary false alarms, which we deem unacceptable.

### D. Scalar Penalty Increase

It was found that by adjusting a single scalar, the most significant increase in the MMAE performance could be attained. To decrease the MMAE convergence time, it was noted that the  $-\frac{1}{2}$

in the  $\{\cdot\}$  term in the equation for the conditional density function, (9), can be viewed as a penalty for having a larger than expected residual. By increasing this term, the same residual will produce higher penalties and cause the MMAE to change its probabilities faster when a failure occurs, thus acting as a decision convergence gain because it amplifies any residuals that are larger than expected.

We experimented with replacing the  $-\frac{1}{2}$  in the Dot term (i.e., the  $\{\cdot\}$  term) of (9) with values of  $-1$  and  $-2$ . This strategy was used with the various designs that were developed during the Kalman filter tuning (described in the previous section). A rudder failure had, by far, the slowest convergence time of all the failure modes that were studied in this application. Using design 1, the convergence time (when the probability was above the 50% probability threshold) for a rudder failure decreased from 3.2 s for a Dot term of  $-\frac{1}{2}$  (Fig. 7, solid line), to 1.8 s for a Dot term of  $-1$  (Fig. 7, dashed line), and finally to 0.8 s for a Dot term of  $-2$  (Fig. 7, dotted line). It was found that even faster convergence times for aileron failures were obtained with design 2, as shown in Fig. 8.

Unfortunately, this increase in the Dot term also produced much larger fluctuations in the probabilities (similar to turning up the gain to the point where the noise becomes unacceptable), which indicates that momentary false alarms would sporadically occur during actual flight testing, which was considered to be unacceptable performance. To avoid these momentary false alarms, we chose design 1 and a Dot terms of  $-1$  (Fig. 7, dashed line). Note that by using design 1 the aileron failures take up to 1 s longer than they do for design 2, but the advantage is the lack of false alarms. Using design 1 and a Dot term of  $-1$ , the convergence to the correct failure occurs in less than 2 s for *all* the failure conditions, while minimizing momentary false alarms once convergence is attained.

### E. Probability Smoothing

Previous implementations of a MMAE for failure detection [6–8, 11] have smoothed (averaged over a number of data samples) the hypothesis conditional probabilities to minimize momentary false alarms. Good results were obtained for these implementations using a data window of 10 data samples over which the probabilities were averaged.

Decreasing the size of the data window over which the probabilities were smoothed (PWINSIZ) was explored as another possible means of decreasing the convergence time. Fig. 9 compares the results of decreasing the window size from 10 to 5. Note that the performance for a window size of 5 (Fig. 9, dashed line) does not improve significantly over the performance for a window size of 10 (Fig. 9, solid line), while the fluctuations in the probabilities increase only slightly, which indicates a very small

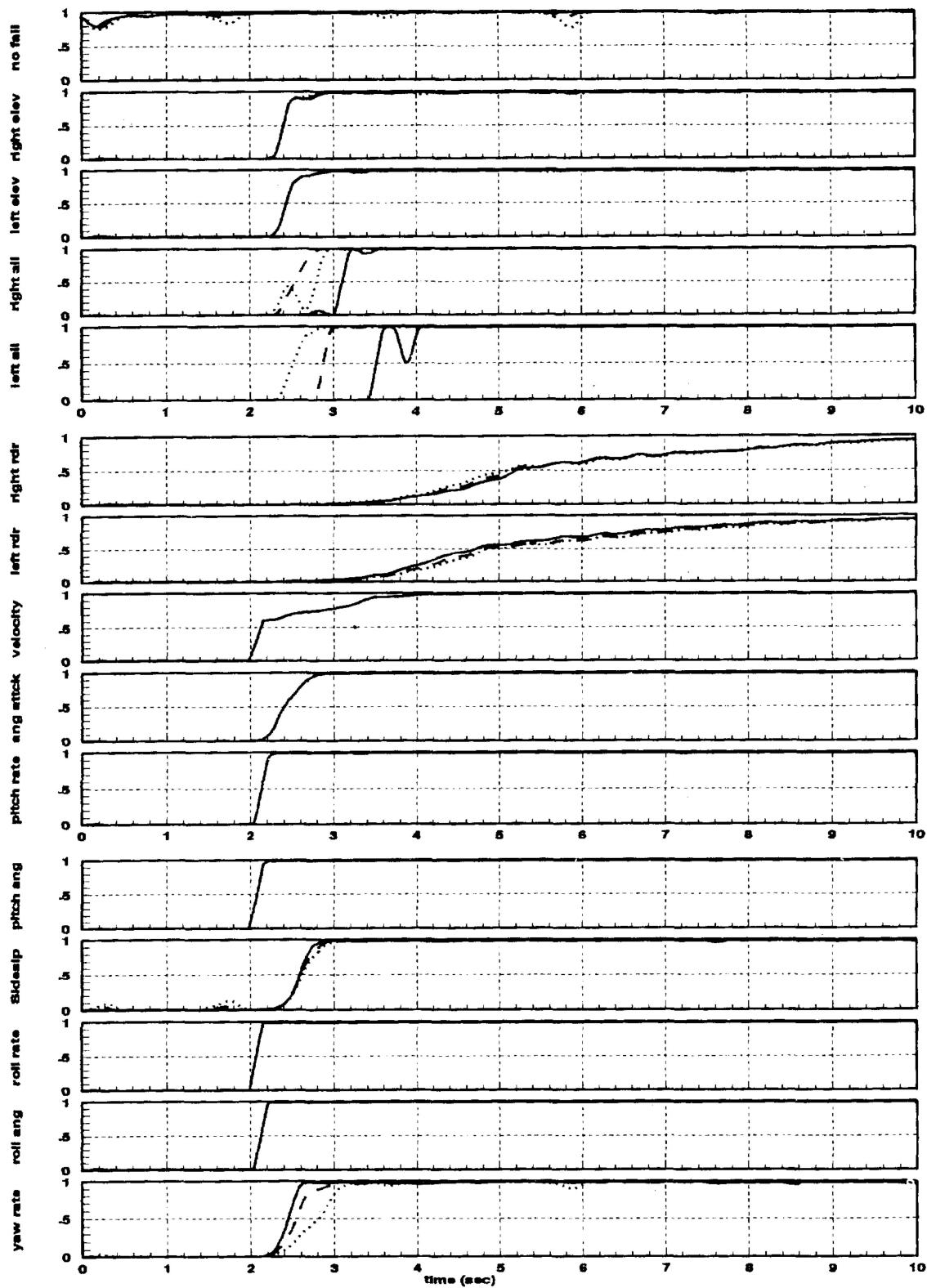


Fig. 6. MMAE performance for various designs. Solid line, design 1. Dashed line, design 2. Dotted line, design 3.

increase in momentary false alarms. Decreasing the window size increases the false alarms because high frequency fluctuations in the probabilities are no longer smoothed over several time periods. Since decreasing the smoothing does not improve the performance significantly, but it does slightly increase

the number of false alarms, we retained the original data window size of ten data samples.

#### F. Increased Residual Propagation

Another method of possibly decreasing the decision convergence time would be to propagate the Kalman

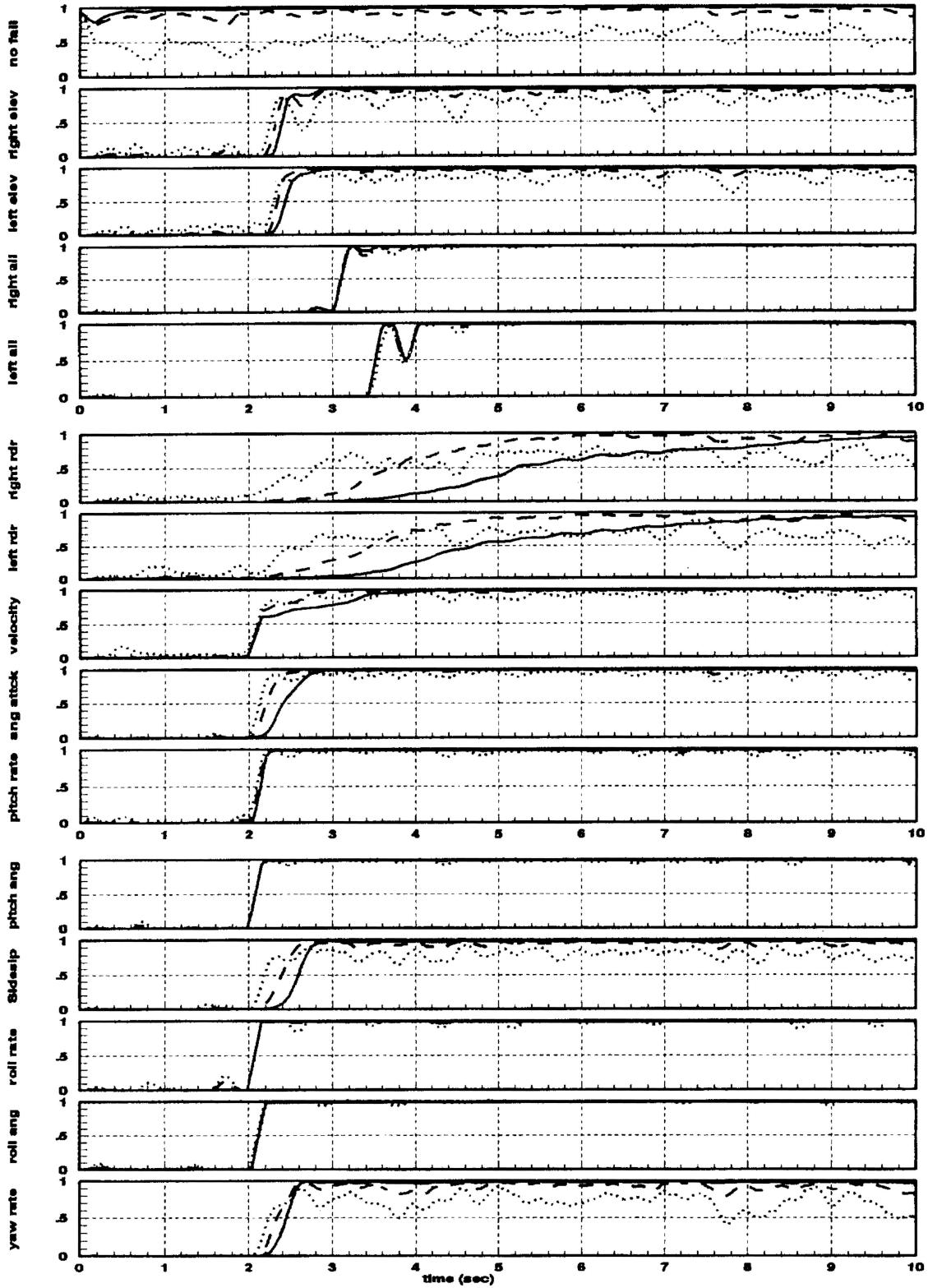


Fig. 7. Design 1 using various Dot terms. Solid line, Dot =  $-\frac{1}{2}$ . Dashed line, Dot = -1. Dotted line, Dot = -2.

filter state estimates, without updating, for a few sample periods while still monitoring the residuals. This allows the residuals in all the filters to grow much larger since the usual measurement updates are no longer correcting the state estimates toward the actual measurements; such corrections tend to

mask the impact of an incorrect hypothesis. The results for propagating various sample periods before updating are compared in Fig. 10. Unfortunately, propagating two samples periods before updating does not provide enough time for the residuals to grow very large, so the MMAE performance does

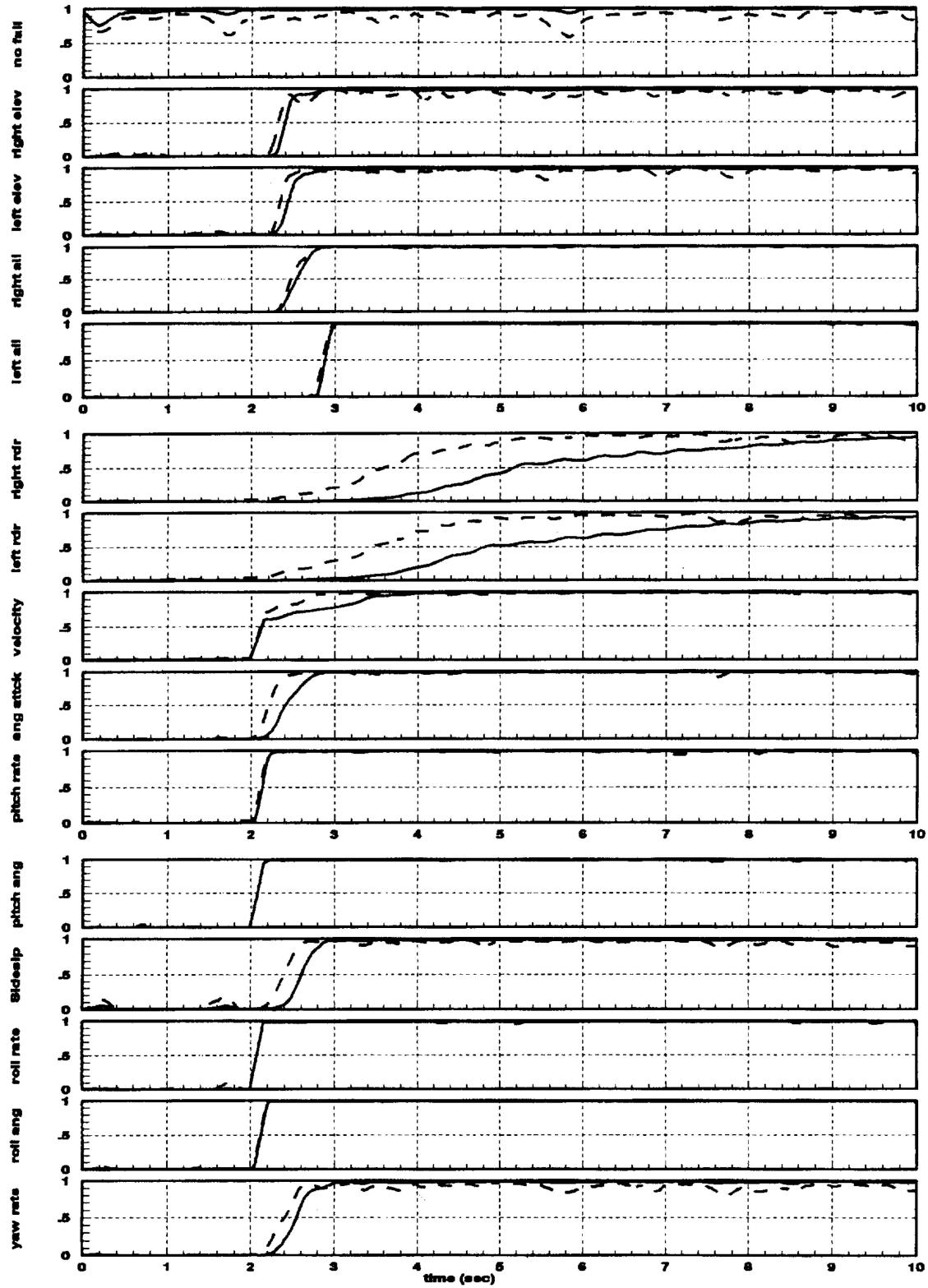


Fig. 8. Design 2 using various dot terms. Solid line, Dot =  $-\frac{1}{2}$ . Dashed line, Dot = -1.

not improve significantly. This can be seen in Fig. 10 where the solid line completely masks the dashed line, indicating that the performance for an update every other sample period is indistinguishable for an update at each sample period.

Propagating five sample periods before updating was also considered. These results are quite poor, as shown by the dotted line in Fig. 10. Note the very erratic  $p_k$  time histories, indicative of numerous momentary false alarms and misidentifications.

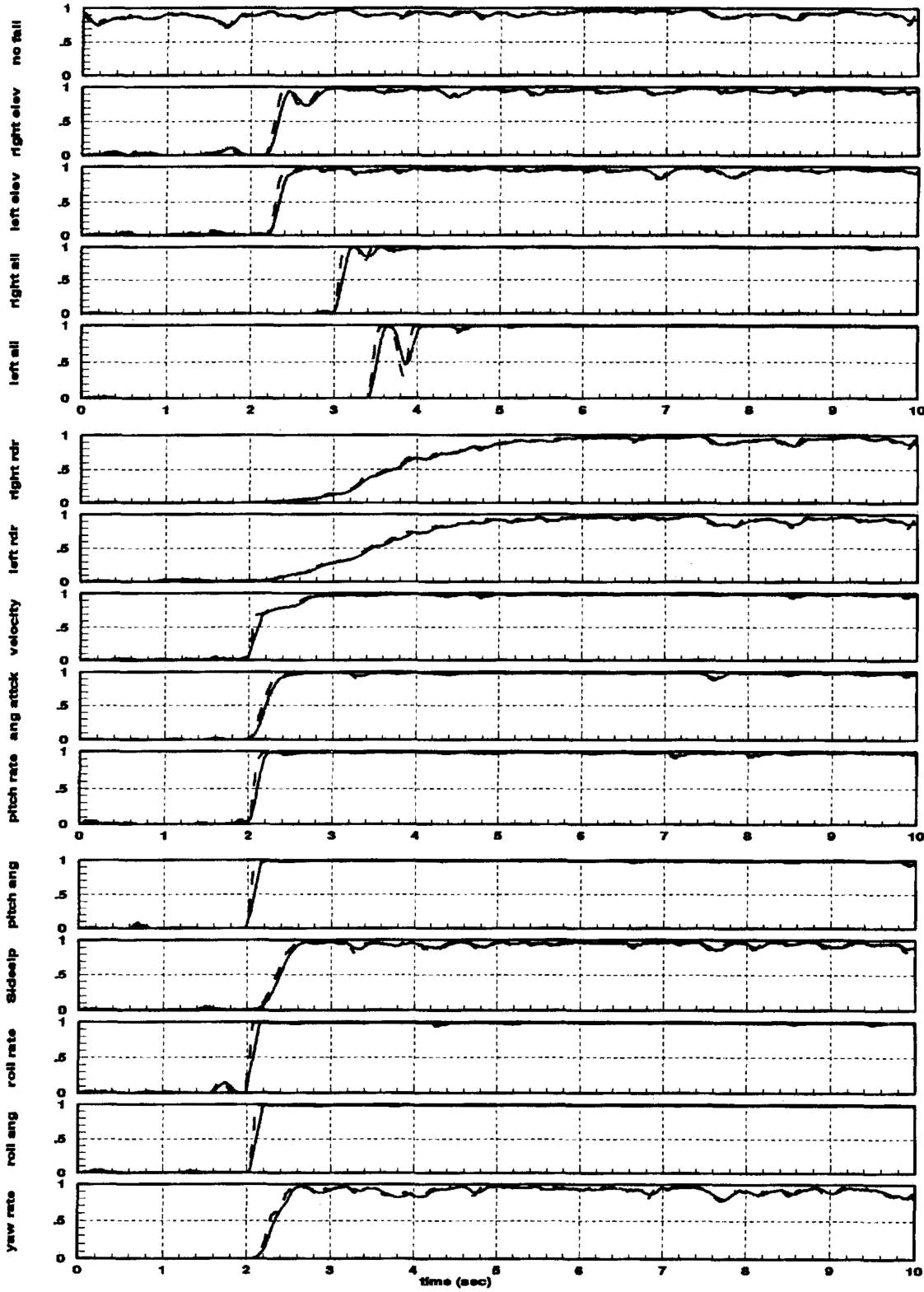


Fig. 9. MMAE performance with probabilities smoothed over different data window sizes (PWINSIZ). Solid line, PWINSIZ = 10. Dashed line, PWINSIZ = 5.

Such erratic behavior is probably caused by the fact that the same  $\mathbf{A}_k$  from (8) was used in the  $p_k$  computation of (9) and (10), even though this  $\mathbf{A}_k$  is a valid residual covariance only for the case of measurements being used *every* sample period for

updating. Using a modified residual covariance matrix that was recomputed to account for the correct number of sample periods between measurement updates might well yield significantly better performance.

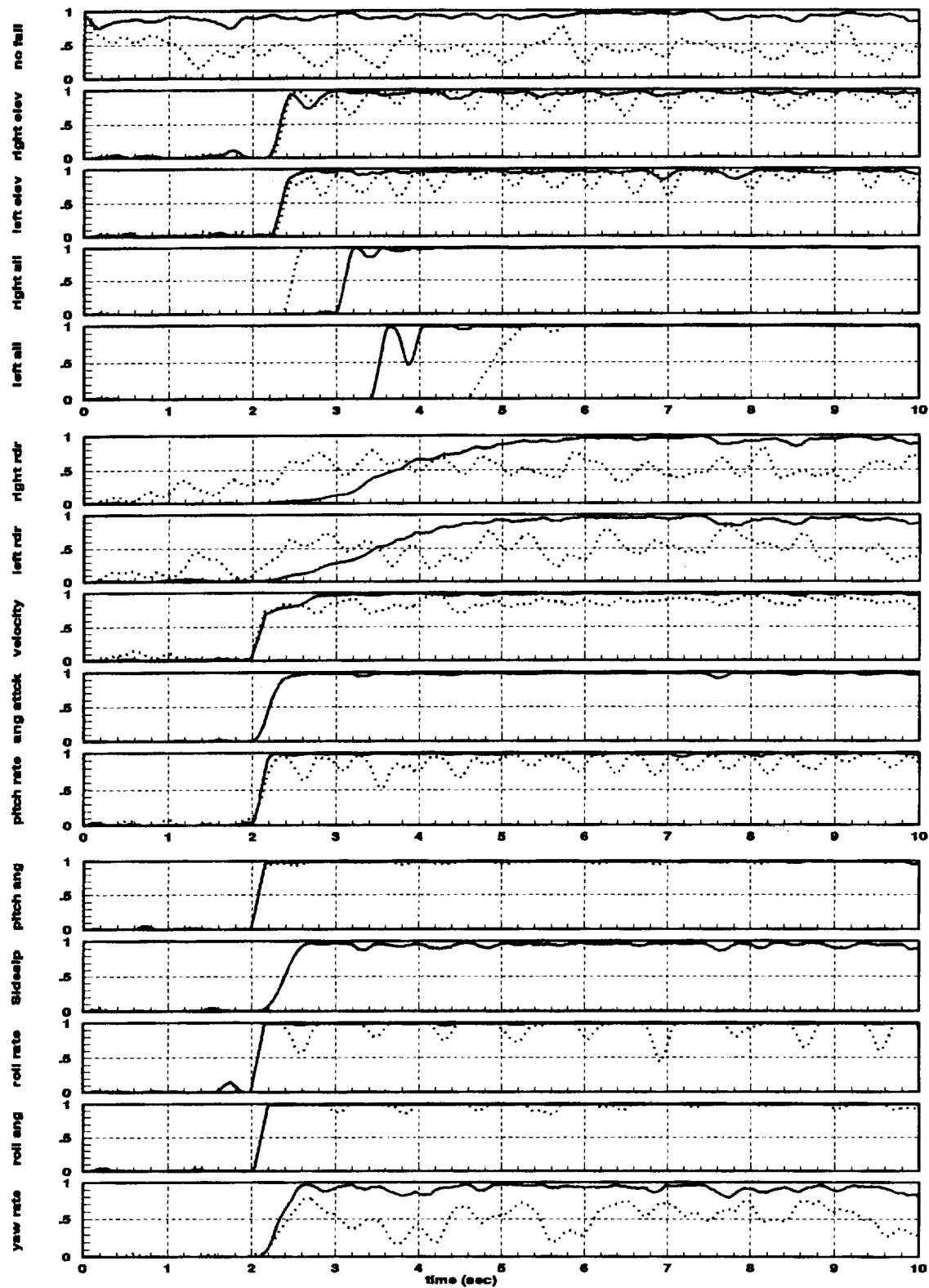


Fig. 10. MMAE performance when propagating various sample periods before updating (NPROP). Solid line, NPROP = 1. Dashed line, NPROP = 2. Dotted line, NPROP = 5.

#### IV. CONCLUSIONS

The results presented in Section III show that some of these performance enhancement techniques work well at resolving specific problems, while some proved ineffective. Removing the  $\beta$  dominance effects and

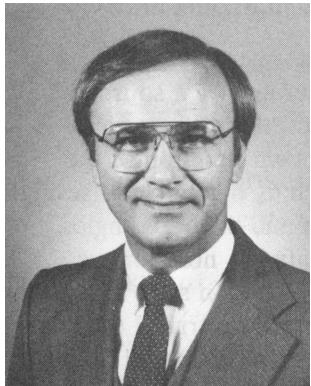
lower bounding the hypothesis conditional probabilities should be part of any MMAE used for failure identification purposes. Tuning the Kalman filters will significantly reduce the identification time for cases where the MMAE is temporarily misidentifying the failure. Increasing the scalar penalty for large

residuals works well when the hypothesis conditional probabilities converge slowly to the correct failure hypothesis. Smoothing the probability significantly reduces the momentary false alarms by removing the high frequency content from the hypothesis conditional probabilities, and does not significantly increase the identification time for this application. Propagating a few time periods without updating the Kalman filter estimates may help reduce the identification time, if the Kalman filter covariances,  $A_k$ , are recomputed using the new measurement sampling rate.

The performance enhancements that we have described allow the designer to adjust the MMAE to improve certain aspects of the MMAE performance. Note that the MMAE correctly identified the various failures that were modeled, so identification accuracy was not an issue. All the performance improvements involved a tradeoff between a decrease in the identification convergence time and an increase in the false alarm rate. In practice, an acceptable level of false alarms would need to be set, then the designer could maximize the MMAE convergence performance using the various enhancements that we have described.

## REFERENCES

- [1] Hanlon, P. D. (1992) Failure identification using multiple model adaptive estimation for the LAMBDA flight vehicle. M.S. thesis, AFIT/GA/ENG/92D-19, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, Dec. 1992.
- [2] Maybeck, P. S. (1979) *Stochastic Models, Estimation, and Control*, Vol. 1. New York: Academic Press, 1979.
- [3] Maybeck, P. S. (1979) *Stochastic Models, Estimation, and Control*, Vol. 2. New York: Academic Press, 1979.
- [4] 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 IEEE Conference on Decision and Control*, Tampa, FL, Dec. 1989, 1566–1572.
- [5] Maybeck, P. S., and Stevens, R. D. (1991) Reconfigurable flight control via multiple model adaptive control methods. *IEEE Transactions on Aerospace and Electronic Systems*, 27, 3 (May 1991), 470–480.
- [6] 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 (AU), Wright-Patterson AFB, OH, June 1992.
- [7] 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 of the National Avionics and Electronics Conference* (NAECON), Dayton, OH, May 1992, 441–448.
- [8] 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 the American Control Conference*, San Francisco, CA, June 1993, 3135–3140.
- [9] Pogoda, D. L. (1988) Multiple model adaptive controller for the STOL F-15 with sensor/actuator failures. M.S. thesis, AFIT/GE/ENG/88D-37, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, Dec. 1988.
- [10] 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 (AU), Wright-Patterson AFB, OH, Dec. 1989.
- [11] 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-53, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, Dec. 1991.
- [12] Swift, G. A. (1991) Model identification and control system design for the LAMBDA unmanned research vehicle. M.S. thesis, AFIT/GAE/ENY/91S-4, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, Sept. 1991.



**Peter S. Maybeck** (S'70—M'74—SM'84—F'87) was born in New York, NY on Feb. 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 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). 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 1993, 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. 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 Fellow of the I.E.E.E. and a member of the A.I.A.A., and he 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, and he previously served as Chairman of the local Automatic Control Group.



**Peter D. Hanlon** Capton, U.S. Air Force, received the B.S. in engineering in 1985 from the University of Central Florida, Orlando. He is currently pursuing a Ph.D. in electrical engineering at the Air Force Institute of Technology, Wright-Patterson Air Force Base, OH.

He enlisted in the Air Force in 1979, and was assigned to Athens, Greece in 1981 where he flew as a crew member on RC-135 reconnaissance aircraft. He was accepted into the Airman's Education and Commissioning Program (AECP) in 1983 and was assigned to the University of Central Florida, Orlando, until his graduation in 1985. After his commissioning, he was assigned to the Geophysics Laboratory at Hanscom Air Force Base, MA, where he served as either program manager or project engineer for several space shuttle, sounding rocket, and high altitude balloon research projects until 1989 when he was assigned to the staff of the Aeronautical Systems Division, Wright-Patterson Air Force Base, OH. In 1991 he was assigned to the School of Engineering, Air Force Institute of Technology, where he is currently pursuing the goal of a Ph.D. in electrical engineering.