Reducing Lag in Virtual Displays Using Multiple Model Adaptive Estimation

DAVID W. KYGER

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

A multiple model adaptive predictor is applied to a virtual environment flight simulator to remove the effect of computational and scene-rendering delay time. Angular orientation of the user's head is predicted a period of time into the future, so that the scene can be rendered appropriately by the time the user actually looks in that direction. Single nonadaptive predictors cannot adequately cover the dynamic range of head motion. By using three dissimilar models of head motion upon which to base the individual elemental filters within the multiple model adaptive estimator (MMAE) algorithm, an MMAE is designed which outperforms the nonadaptive Kalman predictor proposed by Liang [9].

Manuscript received May 5, 1997; revised January 20, 1998.

IEEE Log No. T-AES/34/4/07979.

Authors' current addresses: D. W. Kyger, Avionics Directorate, Wright Laboratories, Wright-Patterson AFB, OH 45433-7765; P. S. Maybeck, Dept. of Electrical and Computer Engineering, Air Force Institute of Technology/ENG, 2950 P St., Bldg. 640, Wright-Patterson AFB, OH 45433-7765.

0018-9251/98/$10.00 1998 IEEE

I. INTRODUCTION

Virtual environments have great potential as training aids. They allow training at low cost and low risk. The effectiveness of this training is, in part, dependent on the realism of the system. One of the major problems that decreases realism is lag [1]. Lag is the time delay between the human input to a virtual environment and the response of the environment [2]. When lag is present, the human operator will adapt his responses to try to compensate for the lag. If the virtual environment is used for teaching skills, the student will learn based on the presence of artificial lag. When the student moves to an actual system, with different lag characteristics, some of the techniques and strategies developed in the virtual environment may not successfully transfer, making the virtual environment training less useful [3]. There have been many efforts to understand and overcome lag [1]. Still, the problem persists.

One approach to solving the problem of lag in visual displays is to predict where the human subject will be looking (look-angle) some short time in the future, typically 100 ms to 200 ms for our research computational resources and for those envisioned for virtual environment simulators which will be fielded in the near future. This information would allow the computer that generates images to complete all necessary operations and store the appropriate graphics in a buffer before it is actually needed to render the scene at the predicted look-angle, and to finish that process by the time the subject's head actually points in that direction. To make an accurate prediction, a designer needs a model of human head motion. A single model cannot describe head motion characteristics because they cover such a broad range, from staring to rapid tracking movements to chaotic motion when a subject is attempting to acquire a new target or threat [6]. This research uses multiple models to characterize head motion. Specifically, multiple model adaptive estimation (MMAE) is used to predict look-angle, adapting to the changing characteristics of head motion (angular velocity or angular acceleration) in a real-world flight control environment. This is intended to address the shortcomings of nonadaptive predictors that have been developed in the past [7, 9].

Compared with earlier MMAE investigations, this research is unique in a number of ways. First, it concentrates on adaptive prediction rather than adaptive filtering, so that there is even more criticality associated with correct model selection and weighting to accomplish performance goals. Secondly, the elemental filters within the MMAE are based on *different model types* instead of merely *different parameter values within a single given model type*, as is more typical. Finally, the best performance is achieved by incorporating one elemental filter that provides very agile responsiveness to the onset of

a head "maneuver", but which *by itself* could not be used for this application because of instabilities it could cause—the MMAE's ability to "restart" a divergent elemental filter makes this a viable design strategy.

II. ASSUMPTIONS/RESTRICTIONS

The following is a list of assumptions made for this research.

1) The specific aim of this research is to predict pilot head orientation 100 ms to 200 ms into the future during flight simulation. The prediction time is driven by two factors. First, research [9] shows lag varies from 85 ms to 160 ms depending on factors such as sample rate and communications schemes with the head-position tracker. Second, the system developed by previous researchers [12, 13] runs at 10 Hz, making integer multiples of 100 ms intervals a natural choice. All tuning is accomplished for a 100 ms prediction.

2) Only head angular orientation is considered. Translational motion is not measured or predicted. In flight the dominant motion of a pilot's head is angular.

3) There is no reliable "truth model" [4, 6] of head motion. That is, no mathematical model exists that can accurately predict look-angles. Instead, models are tested against a cross-section of empirical data.

4) This research does not deal with eye motion, but only with head-pointing motion.

Perhaps the most obvious way to reduce lag is to increase the speed of required computations. Hardware improvements alone cannot solve the problem [6]. Sensor response time, transmission time, and computational overhead involved in image management all contribute to lag. As computer speed increases, so will the demand for more realistic graphics and other tasks. Faster computers may reduce lag, but they will not eliminate the problem.

III. MULTIPLE MODEL ADAPTIVE ESTIMATION

The ability of the Kalman filter to produce accurate estimates of the true states of a physical system is limited by the extent that its internal dynamics model adequately describes true system dynamics. This implies that the correct values of the parameters which describe the dynamic system (for example, the coefficients of the system differential equations) must be embedded in the filter model. Often, however, some of these parameters are unknown to the designer and/or changing in time. One method of dealing with this situation is through MMAE [10, 11].

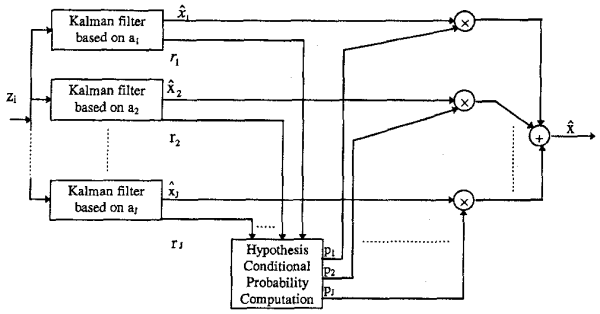


Fig. 1. MMAE algorithm.

As in the development presented in [11], consider the vector **a** containing all the uncertain and/or dynamic parameters in a given system model. Let this system model be described by the first-order, linear, stochastic differential equation of the form:

$$\dot{\mathbf{x}}(t) = \mathbf{F}(\mathbf{a})\mathbf{x}(t) + \mathbf{B}(\mathbf{a})\mathbf{u}(t) + \mathbf{G}(\mathbf{a})\mathbf{w}(t) \tag{1}$$

with physically realizable, noise corrupted, sampled-data measurements described by

$$\mathbf{z}(t_i) = \mathbf{H}(\mathbf{a})\mathbf{x}(t_i) + \mathbf{v}(t_i) \tag{2}$$

where

$\mathbf{x}(t)$ = n-dimensional system state vector,
$\mathbf{u}(t)$ = deterministic control vector,
$\mathbf{w}(t)$ = white Gaussian vector dynamics driving noise of strength $\mathbf{Q}(t)$,
$\mathbf{z}(t_i)$ = m-dimensional measurement vector,
$\mathbf{v}(t_i)$ = discrete-time white Gaussian measurement noise of covariance $\mathbf{R}(t_i)$,
$\mathbf{F}(\mathbf{a})$ = system plant matrix,
$\mathbf{B}(\mathbf{a})$ = control input matrix,
$\mathbf{G}(\mathbf{a})$ = noise distribution matrix,
$\mathbf{H}(\mathbf{a})$ = measurement matrix.

Since **a** may assume a continuous range of values over the space of allowable parameters, it is necessary, for tractability, to discretize **a** into a set of J vector values: $\mathbf{a}_1, \mathbf{a}_2, \ldots \mathbf{a}_J$. A multiple model adaptive estimator consists of J independent Kalman filters, in which the jth filter is based on, and tuned to, the specific parameter set $\mathbf{a}_j$. Fig. 1 shows an MMAE. These J filters form a bank of elemental filters which are processed in parallel. Each elemental filter produces its own estimate of the true state, denoted $\hat{\mathbf{x}}_j(t_i)$, for the jth hypothesized value of **a**. The residuals of all J elemental filters are then used to calculate the probability that **a** assumes the value $\mathbf{a}_j$ at time t_i, for $j = 1, 2, \ldots J$. This probability is called the "hypothesis conditional probability" and is denoted as $p_j(t_i)$. In essence, $p_j(t_i)$ represents the apparent validity of the jth filter's system model at time t_i. Its computation is shown subsequently.

The overall state estimation of MMAE, $\hat{\mathbf{x}}_{\text{mmae}}(t_i)$, is the probability-weighted average of the estimates of

the elemental filters; that is [11]:

$$\hat{\mathbf{x}}_{\text{mmae}}(t_i) = \sum_{j=1}^{J} p_j(t_i)\hat{\mathbf{x}}_j(t_i). \tag{3}$$

This type of multiple model filtering algorithm is referred to as Bayesian MMAE, as opposed to the maximum a posteriori form which would produce the single $\hat{\mathbf{x}}_j(t_i)$ corresponding to the highest p_j, as the algorithm output. Fig. 1 illustrates the structure of the Bayesian MMAE. The hypothesis conditional probabilities $p_j(t_i)$, $j = 1,2,\ldots,J$, are calculated at each sample time t_i, by the recursive equation [11]:

$$p_j(t_i) = \frac{f_{\mathbf{z}(t_i)|\mathbf{a},\mathbf{Z}(t_{i-1})}(\mathbf{z}_i \mid \mathbf{a}_j, \mathbf{Z}_{i-1}) * p_j(t_{i-1})}{\sum_{k=1}^{J} f_{\mathbf{z}(t_i)|\mathbf{a},\mathbf{Z}(t_{i-1})}(\mathbf{z}_i \mid \mathbf{a}_j, \mathbf{Z}_{j-1}) * p_k(t_{i-1})} \tag{4}$$

where $\mathbf{Z}(t_{i-1})$ is the measurement history from the first sample time until sample time t_{i-1}, and

$$f_{\mathbf{z}(t_i)|\mathbf{a},\mathbf{Z}(t_{i-1})}(\mathbf{z}_i \mid \mathbf{a}_k, \mathbf{Z}_{i-1}) = \frac{1}{(2\pi)^{m/2}|\mathbf{A}_k(t_i)|^{1/2}} \exp\{\cdot\}$$

$$\{\cdot\} = \{-\tfrac{1}{2}\mathbf{r}_k^T(t_i)\mathbf{A}_k^{-1}(t_i)\mathbf{r}_k(t_i)\}$$

$$\mathbf{r}_k(t_i) = \mathbf{z}(t_i) - \mathbf{H}_k(t_i)\hat{\mathbf{x}}_k(t_i^-) = k\text{th filter residual vector}$$

$\hat{\mathbf{x}}_k(t_i^-) = k$th filter's state estimation before the measurement update at time t_i is incorporated

$$\mathbf{A}_k(t_i) = \mathbf{H}_k(t_i)\mathbf{P}_k(t_i^-)\mathbf{H}_k^T(t_i) + \mathbf{R}_k(t_i) = k\text{th filter-computed residual covariance matrix}$$

$\mathbf{P}_k(t_i^-) = k$th filter's computed state error covariance before the measurement update at time t_i is incorporated.

Note that the denominator of (4) is the sum of the numerator terms for $j = 1,2,\ldots,J$. This ensures that the sum of $p_j(t_i)$ values, for $j = 1,2,\ldots,J$, equals one.

Note that the hypothesis conditional probabilities at time t_i are functions of the hypothesis conditional probabilities at time t_{i-1}. Due to the recursive nature of the calculations in (4), it is essential that an artificial lower bound be established for $p_j(t_i)$ [11]. Without this lower bound, the hypothesis conditional probability of a filter with a totally invalid parameter set for some period of time, $\mathbf{a}_q$, may go to zero and remain zero for all time: once $p_q(t_{i-1})$ reaches zero, $p_q(t_i)$ and all subsequent p_q would be zero. Thus the qth filter is essentially removed from the bank. Should the actual system change its characteristics so that the true parameter set $\mathbf{a}$ matches $\mathbf{a}_q$ at some future time, $p_q(t_i)$ would remain at zero and the MMAE would produce undesirable results. The residual of the jth filter plays a major role in determining $p_j(t_i)$. As evident from (4), the filter with the smallest ratio of $r_j^T(t_i)\mathbf{A}_j^{-1}(t_i)\mathbf{r}_j(t_i)$ assumes the largest conditional hypothesis probability. Thus the hypothesis probability algorithm is consistent with the heuristic intuition that the residuals of a well-matched filter should be smaller (relative to the filter's internally computed residual covariance, $\mathbf{A}_j$) than the residuals of a mismatched filter.

Finally, it should be noted that MMAE performance is generally degraded when too much pseudonoise is added to the elemental filters for tuning purposes. This is due to the fact that "correct" $p_j(t_i)$ values are assigned when the residuals of filters based on correct models look small relative to the filter-computed residual covariance, while the residuals of filters based on "incorrect" models look large. The addition of pseudonoise tends to blur the defining characteristics of the filter residuals, and distinguishing well-matched filters from mismatched filters becomes increasingly difficult. If substantial dynamics pseudonoise is not added during the filter tuning process, a given filter may diverge when its hypothesis is incorrect. Should such a condition be detected via real-time residual monitoring, the divergent filter can be "restarted" using the weighted average of the nondivergent filters' state estimate at the time. More is said about this in Section VIII.

IV. MODELS

Several models of head motion were investigated. This section summarizes the models.

1) *First-Order Gauss-Markov Acceleration (FOGMA) Model*: This model assumes that head motion angular acceleration is well modeled as a first-order Gauss-Markov process (the output of a first-order lag driven by white Gaussian noise), i.e., that the rate of change in acceleration is a function of the current acceleration and white noise. The FOGMA model is motivated by its successful use in describing airborne targets [5] and the fact that tracking airborne targets is a common task for pilots. This model has the form:

$$\begin{aligned} \dot{\mathbf{p}} &= \mathbf{v} \\ \dot{\mathbf{v}} &= \mathbf{a} \\ \dot{\mathbf{a}} &= -\frac{1}{\tau}\mathbf{a} + \mathbf{w} \end{aligned} \tag{5}$$

where

$\mathbf{p}$ = angular position,
$\mathbf{v}$ = angular velocity,
$\mathbf{a}$ = angular acceleration,
$\mathbf{w}$ = zero-mean white Gaussian noise of strength $\mathbf{Q}$,
τ = time constant of the first-order lag.

Notice that as τ goes to infinity, the FOGMA model becomes a constant acceleration model (plus white

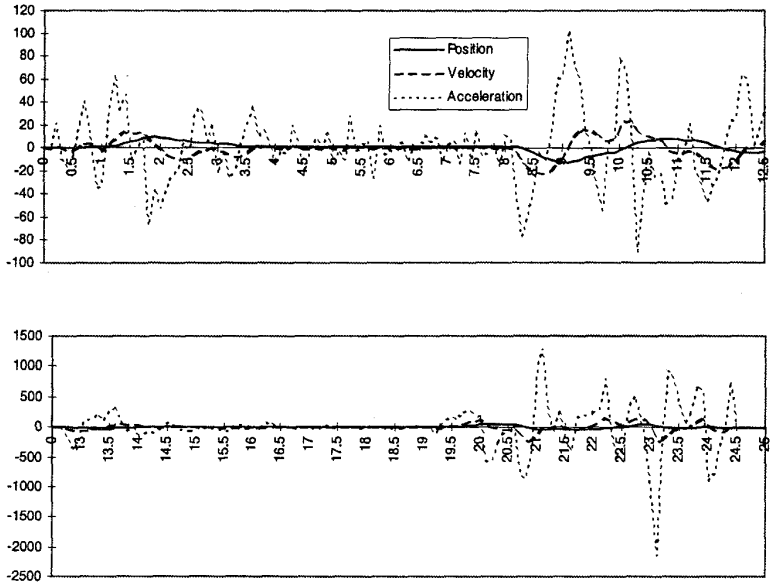


Fig. 2. Heading position (deg), velocity (deg/s), acceleration (deg/s^2).

noise for tuning and indicating that the model is imperfect). Changing the values of τ and $\mathbf{Q}$ results in very different descriptions of motion.

2) *First-Order Gauss-Markov Velocity (FOGMV) Model*: This model assumes that angular velocity is well modeled as a first-order Gauss-Markov process, i.e., that the rate of change in velocity is a function of the current velocity and white noise:

$$\begin{aligned} \dot{\mathbf{p}} &= \mathbf{v} \\ \dot{\mathbf{v}} &= -\frac{1}{\tau}\mathbf{v} + \mathbf{w}. \end{aligned} \tag{6}$$

Notice here that as τ goes to infinity, this becomes a constant velocity model (plus white noise for tuning and indicating the model is imperfect). This model describes less aggressive head motion than does the FOGMA model. A drawback to this model is that it yields a zero-mean velocity, which may not be very descriptive of head motion.

3) *Constant Position Model*: This model assumes that angular position is essentially constant (with white noise added to indicate that a constant position is not a totally adequate description):

$$\dot{\mathbf{p}} = \mathbf{w}. \tag{7}$$

This model proved useful when there was no motion for a period of time; an elemental filter based on this dynamics model effectively rejected the impact of sensor noise on estimates of essentially fixed head orientation.

V. FILTER DESIGN

The MMAE consisted of 3 Kalman filter predictors. Each elemental filter was tuned by adjusting time constants (τ) and $\mathbf{Q}$ values for a specific type of motion: benign, moderate, and reacquisition. Once the elemental filters were tuned, they were combined into an MMAE. Each MMAE was then tested against all categories of motion and concatenated segments to display onsets of changing characteristics of head motion.

The initial MMAE algorithm was first developed by O'Connor [12]. The virtual display was developed by Russell [13] on Silicon Graphics Incorporated (SGI) 4D and Reality Engine workstations using C + + and Performer [8] library graphics support. The virtual display allows researchers to see how different models and filter tunings affect estimation lag and overshoot. This simplifies the task of identifying potentially useful filters and getting approximate tunings, as well as allowing the assessment of the *visual impact* of ringing and other phenomena (that might not be obvious simply from time histories of errors, or of error statistics).

The visual display is generated in a PT-01 head-mounted display (HMD). Look-angle is measured by a Polhemus Fastrack magnetic tracker. The Polhemus tracker consists of a magnetic field transmitter and receiver. The receiver is attached to the HMD and senses relative changes in the transmitted magnetic field whenever the HMD moves. This change is translated into a unit vector, in Cartesian coordinates, which points in the direction of HMD orientation. This unit vector is the only measurement available to the MMAE.

Liang [9] proposed a single *nonadaptive* Kalman filter, using a FOGMV model, to predict motion for HMDs. Although he empirically determined the measurement noise of his system to have variance

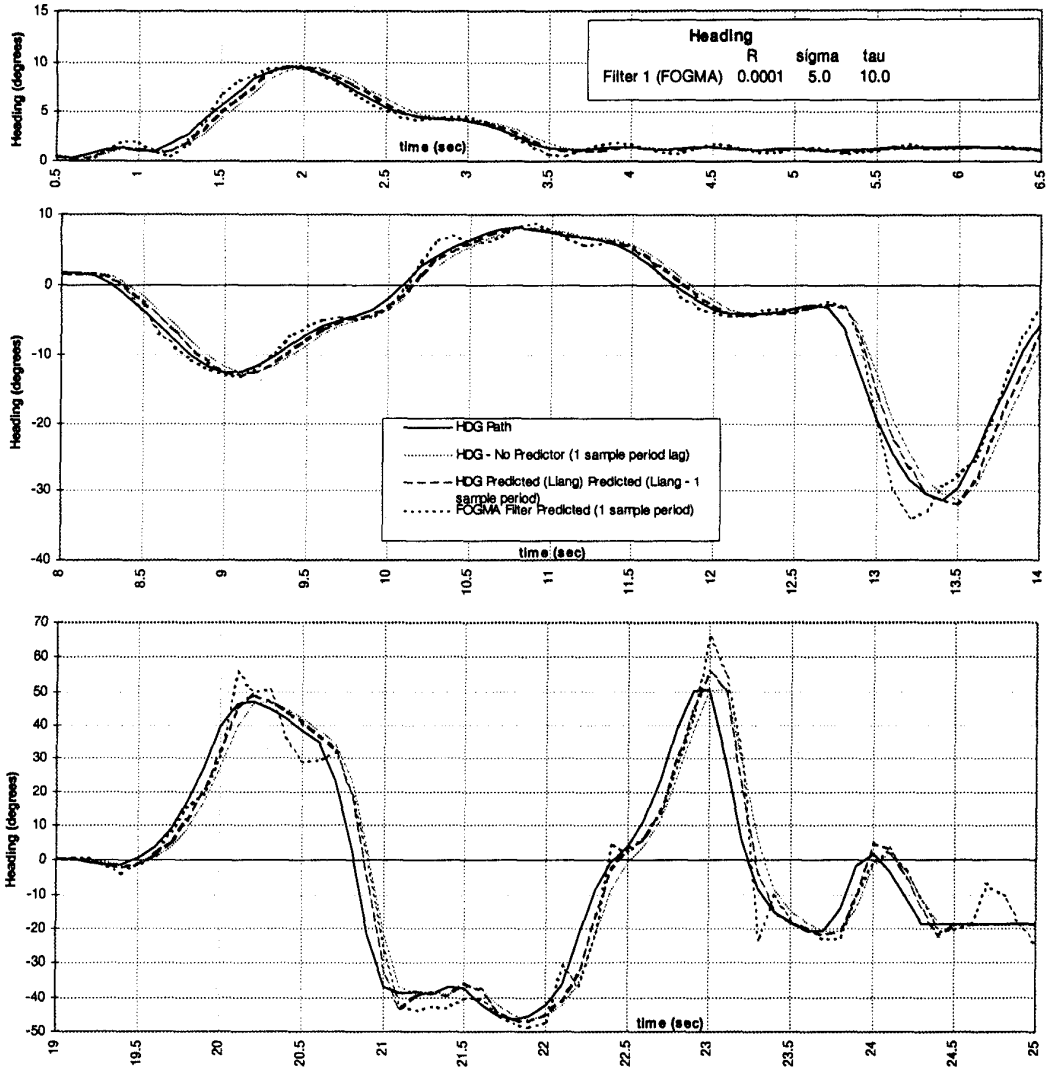


Fig. 3. Nearly-constant-acceleration FOGMA.

$R = 0.001$ deg^2, in this research predictors performed better with $R = 0.0001$ deg^2. This small value for R brought the updated look-angle estimate very close to the measured value. This is notable because the measurements were thus declared to be far more accurate than any sensor actually tested. Such a small value of R raises the risk of introducing jitter from noisy measurements. The empirically observed variance of the measurement noise varied greatly, depending on the position of the transmitter and receiver relative to each other and other electronic and metal objects in the laboratory. In the final configuration, jitter was negligible, even with $R = 0.0001$ deg^2. A different electromagnetic environment, or different look-angle measuring device, may demand a different value for R.

The initial transient on all the filters is a result of the initial conditions, $\hat{\mathbf{x}}_0$ and $\mathbf{P}_0$. The algorithm requires initial conditions, but quickly adjusts the state estimate and covariance in response to measurements. The initial conditions were arbitrarily set, resulting in a transient as the algorithm converged to the correct values. Since this initial transient was not of substantial practical interest, all of the performance plots in this paper start at 0.5 s into the simulation, after the initial transient effects have died out.

VI. PERFORMANCE EVALUATION METHODOLOGY

Since Liang's filter is the best replicatable predictor available, it serves as a benchmark. The MMAE must outperform Liang's filter to justify its greater complexity and computational demand. A second benchmark of performance is the case of no predictor in the loop at all, representative of many current implementations of virtual environment simulators. Thus, MMAE predictor performance can be bracketted by perfect (and unobtainable) prediction

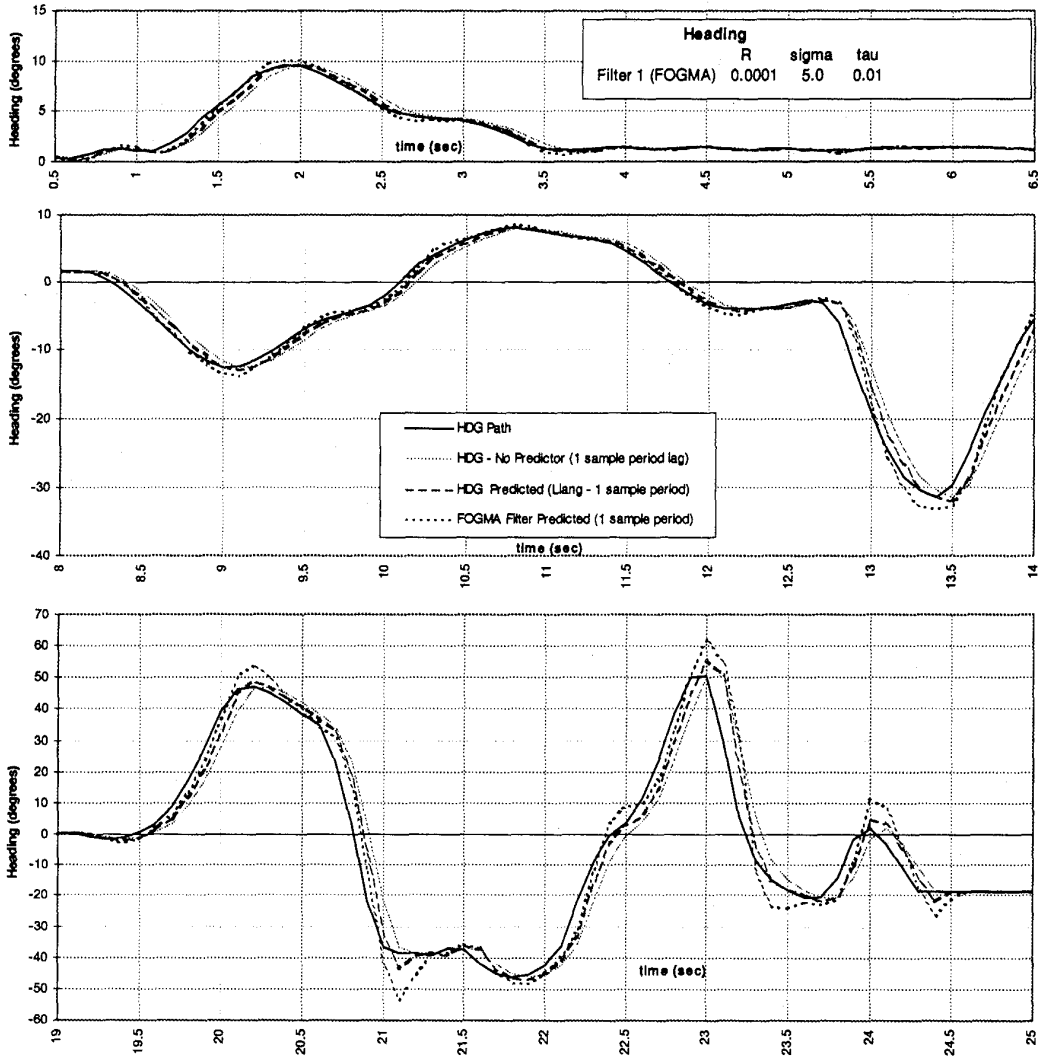


Fig. 4. Nearly-constant-velocity FOGMA.

and no prediction, with a direct comparison to the best results of a *nonadaptive* predictor (i.e., Liang's). These four plots are seen subsequently in Figs. 3–6.

Since no truth model exists for head motion, empirical data was used to test the accuracy of each design. Researchers at Armstrong Laboratories [6] collected head motion data, sampled at 60 Hz, from simulator missions flown by an experienced fighter pilot. The head movements were categorized into seven groups (circular, hourglass, chaos, vertical, horizontal, staring, and figure-eight), each group containing five sets of data. To form a single, representative data set, displaying benign, moderate, and aggressive motion, data sets from the hourglass and chaos categories, sampled at 10 Hz, were concatenated. The heading position, velocity and acceleration for this original path are shown in Fig. 2. Note that position is scaled by a factor of 10 to allow it to be put on the same graph with velocity and data. The vertical scale is in deg$*10$ for position, deg/s for velocity and deg/s^2 for acceleration. Further, note the difference in the vertical scales from the top graph to the bottom graph. The first 8 s are benign motion. Moderate motion is 8–14 s and 19 to 25 s is reacquisition (or aggressive) motion.

VII. DESIGN AND EVALUATION OF ELEMENTAL FILTERS

Individual FOGMA filters were tested over a range of time constants τ, mean-squared accelerations σ^2, and measurement noise variances R. Fig. 3 shows a single FOGMA filter with a $\tau = 10$ s (the other parameter values are listed in the figure). Recall that, for large values of τ (such as $\tau =$ 10 s), the FOGMA model approaches a constant acceleration model. Fig. 4 shows a single FOGMA filter with $\tau = 0.01$ s. This filter approximates a

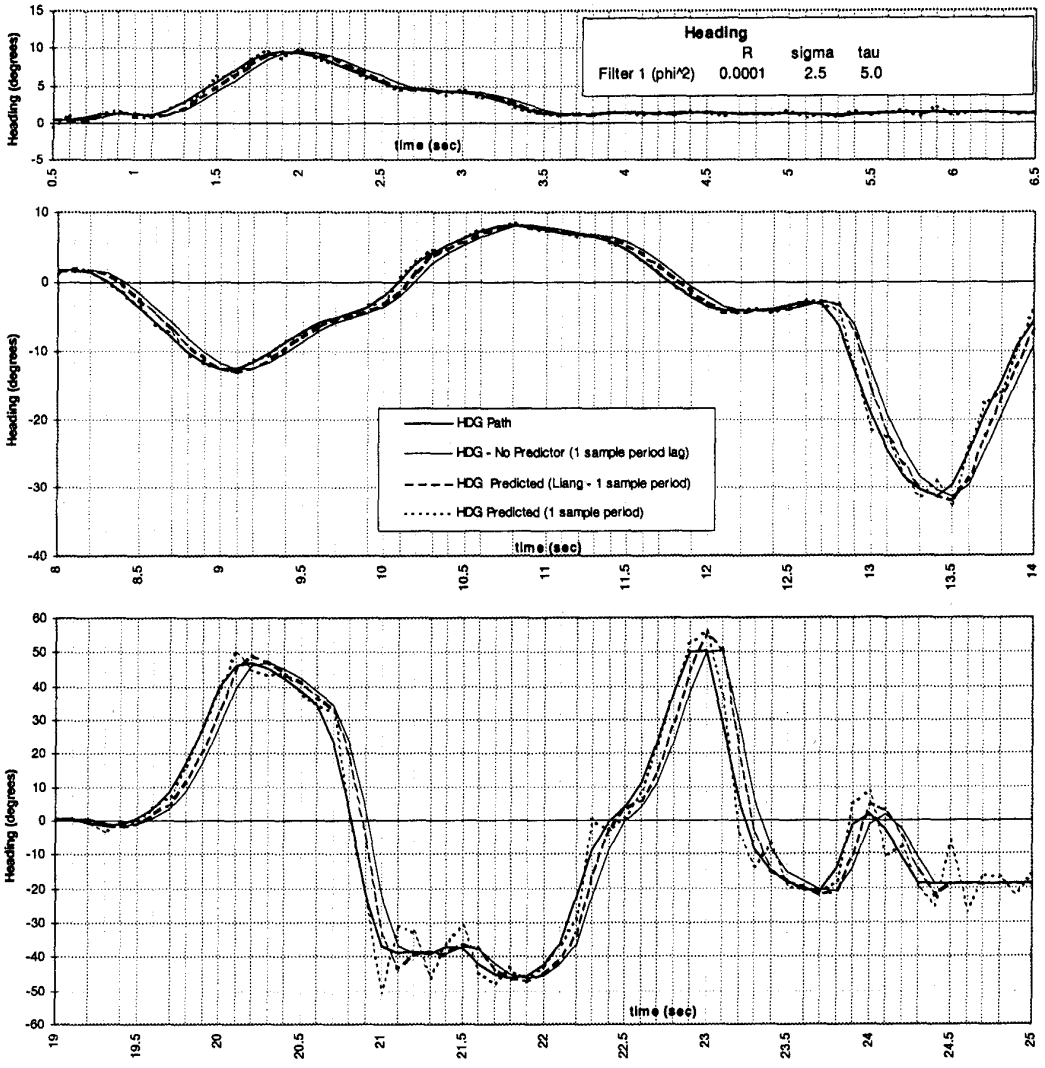


Fig. 5. Initial tuning of "Φ^2 filter."

constant velocity model, since **a** in (5) is essentially white noise. Comparing the two FOGMA filters shows that the nearly-constant-acceleration model displays less lag (see especially t = 2.0–2.5 s, t = 8.3–9.0 s) and generally larger overshoots (particularly at t = 9.1, 13.2, 24.7 s) when compared with the nearly-constant-velocity model. Since the nearly-constant-acceleration model expects any acceleration to persist, it tends to yield position predictions that oscillate (or ring) about the true measurement (t = 3.5–6.5 s, and t = 10.3–11.3 seconds) during benign and moderate motion when acceleration actually decays rapidly.

Both FOGMA and FOGMV elemental filters consistently display the same predictable behavior. As the time constant increases, predictions become more aggressive (i.e., assuming that current acceleration or velocity will persist longer into the future), resulting in less lag and more overshoot.

In an attempt to reduce the lag further, without increasing overshoot, the filter propagation equation for a single sample period was changed from the standard equation:

$$\hat{\mathbf{x}}(t_{i+1}^-) = \Phi\hat{\mathbf{x}}(t_i^+) \tag{8}$$

to

$$\hat{\mathbf{x}}(t_{i+1}^-) = \Phi^2\hat{\mathbf{x}}(t_i^+). \tag{9}$$

Simply put, what would normally be the prediction for 2 sample periods into the future becomes the prediction for a single sample period. A "Φ^2" model will predict larger changes in position between sample periods than a standard FOGMA model with the same time constant. This approach is not physically motivated and could result in filter instability if only a single filter were to be used. Squaring Φ is also somewhat arbitrary and easy to compute; a final design may use Φ appropriate for predicting over an

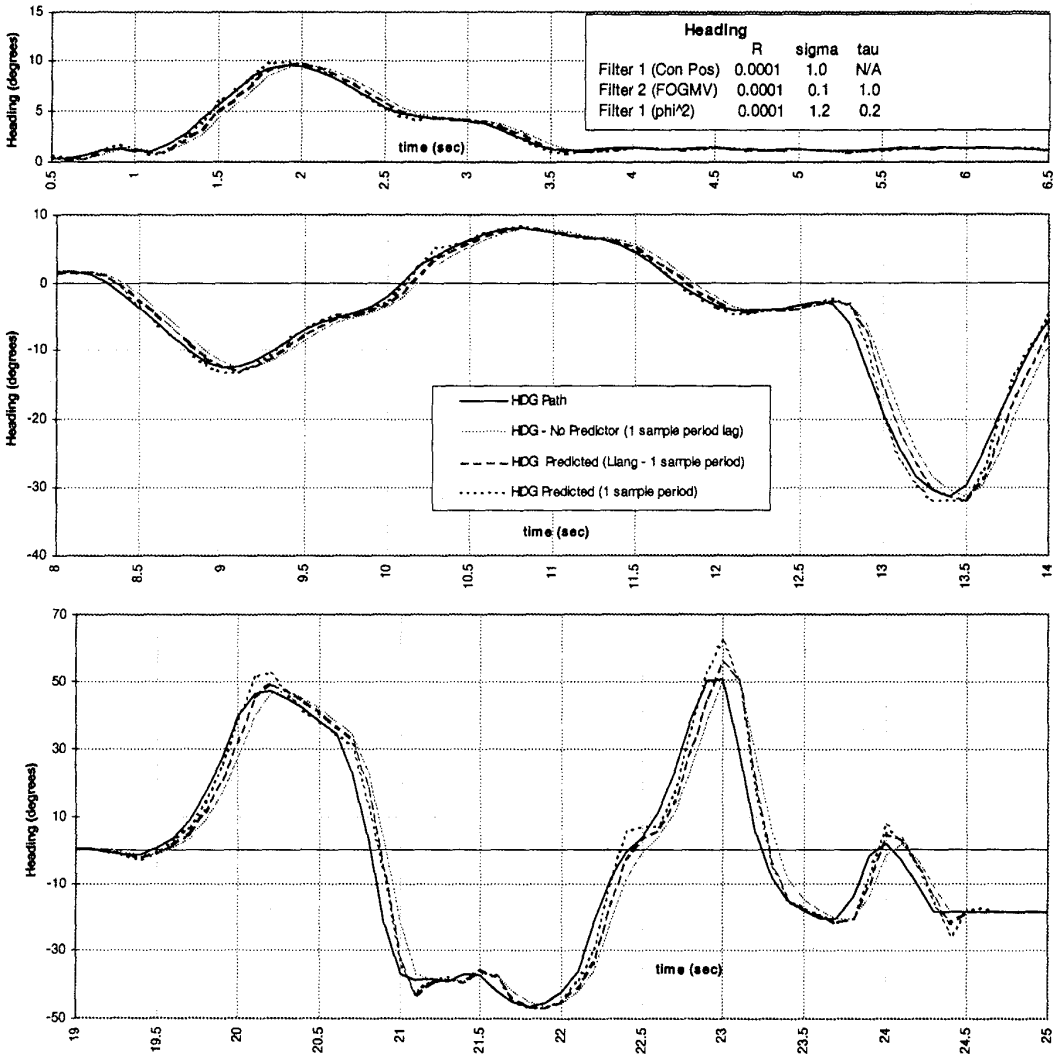


Fig. 6(a). Final MMAE design.

interval of time other than two sample periods (e.g., 1.5 or 1.6 sample periods), whatever period of time that yielded the *best* performance. Using Φ^2 is simply a convenient first attempt at further reducing lag. Fig. 5 shows the result of using a single "Φ^2 filter" with a state transition matrix Φ evaluated for a time constant of $\tau = 5$ s. The performance is especially striking during moderate levels of head motion (t = 8–14 s). Lag is almost completely eliminated, and the overshoots are no greater than in Liang's filter. However, ringing remains a significant problem (as seen during t = 13.0–13.6 s, t = 21.0–22.0 s, and t = 24.0–25.0 s).

VIII. EVALUATION OF MMAE

Initial MMAE designs were composed of three FOGMA or three FOGMV elemental filters, each tuned (through the parameters τ and either Q or σ^2) for specific types of motion (benign, moderate, and strenuous/reacquisition). Of all configurations investigated, the MMAE that yielded the best performance was composed of 1) a constant-position elemental filter for benign head motion, 2) a FOGMV elemental filter for moderate head motion, and 3) a FOGMA "Φ^2" elemental filter for strenuous/reacquisition head motion. Fig. 6 shows the final tuning of the MMAE and the best results obtained in this research. Along with the noise values (σ and R) and time constant (τ), consider the divergence thresholds. When a filter's $\mathbf{r}^T(t_i)\mathbf{A}^{-1}(t_i)\mathbf{r}(t_i)$ value exceeds its divergence threshold, the filter is declared divergent. Its prediction is given zero weighting by the MMAE and the position states of the divergent filter are replaced with the MMAE position states computed without the divergent filter

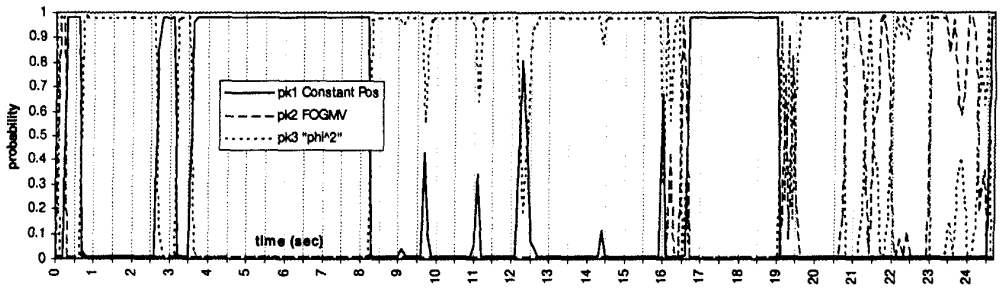


Fig. 6(b). Probability weighting of final MMAE design.

TABLE I
Comparison of Lag and Overshoot Between MMAE, Liang's Nonadaptive Filter and No Predictor

	No Predictor	Liang			MMAE		
		Benign	Moderate	Reacquisition	Benign	Moderate	Reacquisition
Lag							
Number of lag points	61	9	33	39	2	5	27
Average lag (seconds)	0.1	0.058	0.058	0.059	0.05	0.030	0.053
% improvement		42	42	41	50	70	47
Overshoot							
Number of overshoots	0	0	2	10	1	12	18
Average overshoot (degrees)	0	0	1.37	2.98	0.65	1.25	4.00
Number of predictions within 0.5 deg. (out of a possible 61)	47(benign) 15(moderate) 14(reacquisition)	52	26	12	59	44	16

(this is called an elemental filter "restart"). Since the constant-position filter has a relatively large lag (comparable to the lag with no predictor at all), it is only effective when there is very little head motion. To ensure the MMAE quickly discounts the constant-position filter's prediction when motion begins, its divergence threshold is set very low. The result is that the constant-position filter is being declared divergent and being restarted almost continuously, except when head movements are very slow.

Notice that, during benign motion, lag is almost completely eliminated (note t = 0.5–6.5 s on Fig. 6, the dotted line representing the MMAE prediction is obscured by the solid line of the true path). During moderate motion, the performance is almost as good (t = 8.0–14.0 s). The MMAE prediction oscillates about the true path with errors much smaller than those of Liang's filter, and with overshoots much less than 5 deg. In fact, the MMAE gives a smaller error than Liang's filter throughout the period of moderate motion except for individual sample periods when Liang's filter overshoots the true path later than the MMAE (t = 9.0, 13.4 s). During very rapid head motion (reacquisition: t = 19.0–24.5 s), after an initial overshoot (t = 20.2 s), the MMAE performs very much like Liang's filter. It is important to note that the MMAE could be tuned to reduce lag during the reacquisition phase further by forcing more probability to the "Φ^2 filter", but that would increase overshoots. Instead, the MMAE was tuned to demonstrate that lag could be significantly reduced during benign and moderate head motion (i.e., during tracking tasks by the pilot, when fidelity of simulation is most important, rather than during aggressive and short-term head motions representative of acquiring a new target), without a significant increase in the overshoots (compared to Liang's nonadaptive Kalman filter) or degradation in performance during reacquisition motions.

Table I compares the overshoot and lag performance of the MMAE, Liang's nonadaptive filter, and no predictor (one sample period lag). Each of the three types of motion (benign, moderate and reacquisition) lasts 6 s, yielding 61 data points each. The "% improvement" row is a measure of the improvement relative to no predictor. This table was generated by dividing each estimate into one of three categories: accurate (the estimate was within 0.5 deg of the true heading, as particularly between t = 3.7 s and t = 6.5 s), lag (the estimated heading trailed the true heading in time), and overshoot (the estimate did

TABLE II
Comparison of Temporally Averaged and Maximum Errors of Three Designs

Average magnitude of error for each type of motion (degrees)	No Predictor	Liang	MMAE
benign	0.34	0.23	0.18
moderate	1.68	0.99	0.58
reacquisition	5.90	4.05	3.37
average	2.64	1.76	1.38
% improvement (compared to no predictor)			
benign		33.5	47.1
moderate		40.9	65.0
reacquisition		31.3	42.9
average over all 3 types of motion		33.4	47.7
Maximum error (degrees)			
benign	1.47	1.00	0.65
moderate	6.48	5.06	2.77
reacquisition	24.93	21.22	20.07
% improvement (compared to no predictor)			
benign		31.6	55.8
moderate		21.9	57.3
reacquisition		14.9	19.5

not have a close or lagging connection with a specific true heading, such as at $t = 23.0$ s). Overshoot was calculated as the magnitude of the difference between the predicted and true headings. Lag was calculated as the time between the true position reaching a heading and the predicted position reaching that same heading (using linear interpolation between sample times to compute the exact time when the estimate reached the same value as the actual heading at a given earlier sample time). Notice that the MMAE outperforms Liang's filter in every category except overshoot in the reacquisition phase. Also, note that the lag in the MMAE varies much more than in Liang's design. The most dramatic improvement is in the predictions of moderate motion. The MMAE gives 65% more points within the 0.5 deg "accurate" tolerance and a 75% reduction in lag points. To achieve this, the number of overshoot points increased, but the average overshoot is very small.

Table II compares the temporally averaged and maximum error magnitudes (i.e., error absolute values) of the three designs. The MMAE gives the best improvements during benign and moderate motion (substantially better than those of Liang's filter), but even during reacquisiton motion the MMAE gives substantial error reduction. The largest errors occur during periods of lag, not during overshoot.

IX. CONCLUSIONS

Multiple model adaptive estimation has been shown to reduce lag significantly in virtual displays. It results in less lag than the single nonadaptive Kalman filter designed by Liang [9] and its overshoot characteristics show a slightly larger magnitude, but equal duration (usually one sample period). The improved performance is most dramatic during benign and moderate head motion, i.e., for the tracking most common in virtual environment simulations. During abrupt reacquisition motions, the MMAE still significantly reduces lag, but at the expense of overshoot and some ringing. Various design schemes showed different amounts of lag and overshoot, but all displayed some lag and only the constant position filter did not display overshoot. Lag cannot be *eliminated* without prior information about future head motion. Overshoot is inherent in any predictive scheme, especially when position is being predicted based only upon position measurements (velocity and acceleration are not currently measurable). Measuring angular acceleration as well as angular position could enhance prediction accuracy substantially.

Motion estimations in each axis (pitch and heading) interfered with each other. The residuals for heading and pitch contribute to decisions about the type of motion present (benign, tracking, or reacquisition). This implies that both heading and pitch are experiencing the same type of motion. In reality, heading and pitch change independently. The MMAE will form a single estimation of look-angle dynamics which will be a compromise between what is needed for pitch and heading predictions. It may be useful to consider separate filters for the two axes in future work.

MMAE can be very helpful if a single sample period overshoot is acceptable. It seems overshoot is unavoidable in a predictive algorithm based only on position measurements. A single filter

cannot model all the look-angle trajectories typical of a single simulator mission. Adaptive schemes (such as MMAE) cannot change models until after elemental filter residuals reveal the need for an altered hypothesis (i.e., a different elemental filter appropriately gaining the highest probability weighting). The MMAE in this research responded to changing dynamics in a *single* sample period (as quickly as possible) and drove the prediction error to nearly zero. Higher sampling rates have two advantages during times of changing dynamics. First, overshoot errors have smaller magnitude. Second, the overshoot error is displayed for a shorter period. If the sampling rate is high enough to make a single sample period overshoot imperceptible, or at least acceptable to the user, MMAE can greatly enhance head-mounted visual displays by reducing lag.

REFERENCES

[1] Adelstein, B. D., Johnston, E. R., and Ellis, S. (1993)
Spatial sensor lag in virtual environment systems.
Proceedings of the SPIE, **1833** (1993).

[2] Arnold, J. E. (1963)
Design and evaluation of a predictor for remote control synthesis with signal transmission.
Technical Report TND-2229, Washington, DC, NASA, 1963.

[3] Arthur, K. W., Kellogg, S. B., and Ware, C. (1993)
Evaluating 3D task performance for fish tank virtual worlds.
ACM Transactions on Information Systems, **11**, 3 (July 1993).

[4] Boff, K. R., and Lincoln, J. E. (Eds.) (1988)
Engineering Data Compendium: Human Perception and Performance, Vol. 3.
Armstrong Aeromedical Research Laboratory, Wright-Patterson AFB, OH, 1988.

[5] Chang, C.-B., and Tabaczynski, J. A. (1984)
Application of state estimation to target tracking.
IEEE Transactions on Automatic Control, **AC-29**, 2 (Feb. 1984).

[6] Eggleston, R. (1995)
Personal interview.
Director, Human Interface Technology Branch, Armstrong Laboratory, Wright-Patterson AFB, OH, May 10, 1995.

[7] Friedman, M., Starner, T., and Pentland, A. (1992)
Device synchronization using an optimal linear filter.
In *Proceedings of the 1991 ACM Symposium on Interactive 3D Graphics*, 1992.

[8] (1992)
Graphics Library Programming Guide, Vol. 1.
Document 007-1210-050, Silicon Graphics, Inc., 1992.

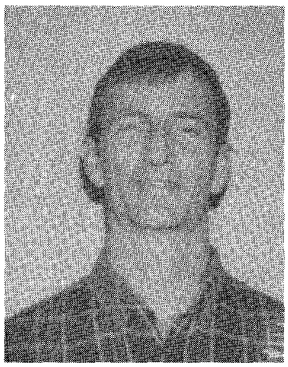
[9] Liang, J., Shaw, C., and Green, M. (1991)
On temporal-spatial realism in the virtual reality environment.
In *Proceedings of the ACM Symposium on User Interface Software and Technology*, 1991.

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

[11] Maybeck, P. S. (1994)
Stochastic Models, Estimation and Control, Vol. 2.
Arlington, VA: Navtech Book & Software Store, 1994.

[12] O'Connor, W., and Blake, D. (1992)
Unpublished notes and software for a multiple model adaptive estimator developed at the Air Force Institute of Technology, 1992.

[13] Russell, J. R. (1994)
Multiple model adaptive estimation and head motion tracking in a virtual environment: An engineering approach.
M.S. thesis, AFIT/GCS/ENG/94D-21, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, Dec. 1994.



David W. Kyger was born on June 4, 1962 in Albuquerque, NM. He accepted an appointment to the United States Air Force Academy in 1980 and in 1983 he participated in a six-month work-study program at the Air Force Weapons Laboratory, Kirtland AFB. Capt. Kyger earned a B.S. in electrical engineering from the Air Force Academy with academic distinction in May 1985.

After completing pilot training at Ft. Rucker, AL, he was assigned to Holloman AFB to fly UH-1N Huey helicopters. In September 1986, Capt. Kyger was transferred to F. E. Warren AFB. He served in Desert Shield/Desert Storm as a Search and Rescue Duty Officer. In 1991 he transitioned to the HH-60 Pave Hawk and moved to Misawa AB, Japan. Capt. Kyger entered the Air Force Institute of Technology in May 1994, and upon graduation in Dec. 1995, he was assigned to the Avionics Directorate at Wright Laboratories, Wright-Patterson AFB.

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., in Cambridge, 1968 and 1972, respectively.

In 1968, he was employed by the Apollo Digital Autopilot Group of The C. S. Draper Laboratory, Cambridge, MA. From 1972 to 1973, he served as a military control engineer for the Air Force Flight Dynamics Laboratory and then joined the faculty of the Air Force Institute of Technology in June of 1973. He is currently Professor of Electrical Engineering, responsible for the graduate sequence in estimation and stochastic control and for individual advanced digital filtering and control courses. His 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 by Navtech in 1974). 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 1997, 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 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.