

A FUZZY-TUNED ADAPTIVE KALMAN FILTER

Young Hwan Lho, and John H. Painter, Senior Member IEEE

Dept. of Electrical Engineering
Texas A&M University
College Station, TX 77843

Abstract

In this paper, fuzzy processing is applied to the Adaptive Kalman filter. The filter gain coefficients are adapted over a 50 dB range of unknown signal/noise dynamics, using fuzzy membership functions. Specific simulation results are shown for a dynamic system model which has position-velocity states, as in vehicle tracking applications such as the Global Positioning System (GPS). The filter is single-input, single-output, driven by measurements of position, corrupted by additive (Gaussian) noise. The fuzzy adaptation technique is also applicable to multiple-input, multiple-output applications for the cases where the states are higher-order moments of motion (position, velocity, acceleration, etc.).

The fuzzy processing is driven by an inaccurate on-line estimate of signal-to-noise ratio (SNR) for the signal being tracked. A robust Bayes scheme would calculate the filter gain coefficients from the signal-to-noise estimate. In our implementation, the inaccurate signal-to-noise estimate is corrected by the use of fuzzy membership functions. The resulting adaptive filter produces near optimum performance in the GPS signal-noise environment. Performance comparisons are given between optimum, Fuzzy-Tuned Adaptive, and Fixed-Gain Kalman Filters for the second-order position-velocity model.

1 INTRODUCTION

The Kalman filter [1] is a well known optimal estimation scheme, satisfying a mean square error performance criterion, which is frequently used. The optimum Kalman filter requires exact *a priori* knowledge of the driving and measurement process statistics. The main problem in application is that the covariance parameters required for Kalman gain computation are either not known precisely or change with time (the non-stationary case). Here, we may assume that such changes are not continuous, but occur at isolated times, with periods of stationarity in between changes. A solution to this case, developed by Eggers [5], is to make a stochastic estimate of the required covariances.

The Kalman gains, g_1 and g_2 , etc. are functions of the assumed known model process covariances, \mathbf{Q}_{ww} and \mathbf{R}_{vv} given by

$$E\{\mathbf{w}_i \mathbf{w}_j\} = \begin{cases} \mathbf{Q}_{ww} & \text{if } i = j \\ \mathbf{0} & \text{if } i \neq j \end{cases}$$

$$E\{\nu_i \nu_j\} = \begin{cases} \mathbf{R}_{vv} & \text{if } i = j \\ \mathbf{0} & \text{if } i \neq j \end{cases}$$

$$E\{\nu_i \mathbf{w}_j\} = \mathbf{0} \text{ for all } i \text{ and } j.$$

See Fig. 1 and equation 1.2. For the position-velocity case

the only other parameter entering the Kalman gain formulation is T , the time between samples. It has also been shown by Painter [2] that for the two-state (position, velocity) case, with scalar measurement data, the Kalman gains are actually functions of the ratio, r , of the model covariances. That is,

$$r = \left(\frac{T^2}{2} \cdot \frac{\sigma_w}{\sigma_v} \right)^2 \quad (1.1)$$

where σ_w^2 and σ_v^2 are the scalar versions of the covariances, \mathbf{Q}_{ww} and \mathbf{R}_{vv} , respectively. Then, $g_1 = g_1(r)$ and $g_2 = g_2(r)$, etc. For the robust Bayes approaches to the 2-state model, stochastic estimates of the "signal to noise ratio," r , are made as \hat{r} . Then, the Kalman gains become functions of \hat{r} , as $\hat{g}_1 = g_1(\hat{r})$, etc. The Kalman gain coefficients are then calculated from the Riccati covariance equations (cf. Brown [4], pp. 217-220). See Fig. 2, below, for a 2-state Kalman filter diagram.

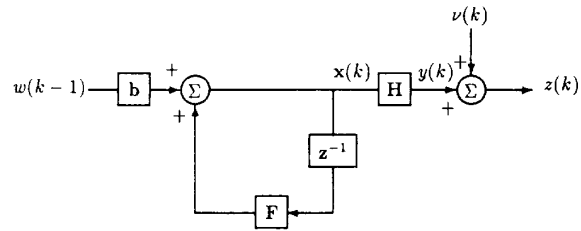


Figure 1: Block diagram of signal and data model

Our application of interest is the discrete-time Kalman filter for tracking a moving vehicle. Thus, the states of the dynamic model shown in Fig. 1 are position, velocity and acceleration, etc. The discrete-time equations in the second order case are described as:

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{F} \cdot \mathbf{x}(k) + \mathbf{b} \cdot w(k), & y(k) &= \mathbf{H} \cdot \mathbf{x}(k) \\ z(k) &= y(k) + \nu(k), & k &= 0, 1, 2, \dots \end{aligned} \quad (1.2)$$

where

$$\mathbf{F} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} T^2 \\ T \end{bmatrix}, \quad \text{and} \quad \mathbf{H} = \begin{bmatrix} 1 & 0 \end{bmatrix}.$$

In the dynamic model, the processes, $w(k)$ and $\nu(k)$, are taken to be zero mean, white, Gaussian, independent of each other. The variable $w(k)$ is the driving function for the highest moment of motion in the state, $\mathbf{x}(k)$. That is, if the highest moment modeled in $\mathbf{x}(k)$ is velocity, then $w(k)$ is acceleration. The variable $\nu(k)$ is the additive noise assumed to corrupt the measurements of the model states. The measured data, which is the input to the Kalman filter, is $z(k)$. In general, $z(k)$ may be scalar or vectorial. That is, the Kalman filter may have more than one single input. For instance, input data might be noisy measurements of the position state and of the velocity state, separately.

In our approach we will deal with the steady-state implementation of the Kalman filter, derived for the case of a dynamic model having stationary statistics. That is, we will seek the constant values of the Kalman gain coefficients, g_1 and g_2 , etc., which represent the steady-state of the Kalman gain computation, in the case where the covariances of $w(k)$ and $v(k)$ are known constants. See any Kalman filter text book, such as Brown [4], for details.

The steady-state Kalman filter (fixed-gain) for the data generating model of equation 1.2 is

$$\begin{aligned} \hat{\mathbf{x}}(k) &= \mathbf{F} \cdot \hat{\mathbf{x}}(k-1) + \mathbf{G} \cdot e(k) \\ e(k) &= z(k) - \mathbf{H} \cdot \mathbf{F} \cdot \hat{\mathbf{x}}(k), \quad \hat{y}(k) = \mathbf{H} \cdot \hat{\mathbf{x}}(k) \quad (1.3) \\ \mathbf{G} &= \begin{bmatrix} g_1 \\ g_2 \end{bmatrix}, \quad k = 1, 2, 3 \dots \end{aligned}$$

where $\hat{\mathbf{x}}(\cdot)$ and $\hat{y}(\cdot)$ are the filtered estimates of $\mathbf{x}(\cdot)$ and $y(\cdot)$, respectively.

We assume that tracking error at time k is $\mathbf{e}(k) = \mathbf{x}(k) - \hat{\mathbf{x}}(k)$. Then the error covariance matrix is given by

$$\mathbf{P}(k) = E\{\mathbf{e}(k)\mathbf{e}^T(k)\} = E\{(\mathbf{x}(k) - \hat{\mathbf{x}}(k))(\mathbf{x}(k) - \hat{\mathbf{x}}(k))^T\} \quad (1.4)$$

The block diagram for equation 1.3 is depicted in Fig. 2.

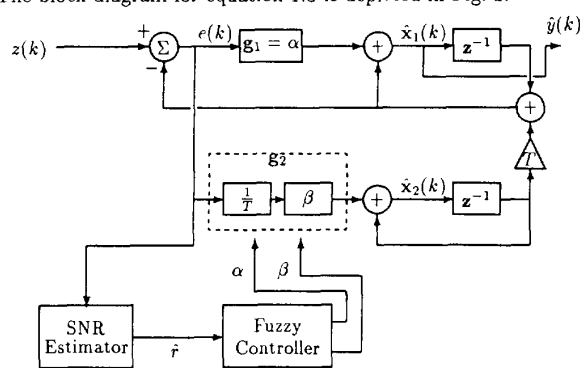


Figure 2: Fuzzy-Tuned Adaptive Kalman Filter

Eggers' covariance estimators [5] were biased. That is, the estimate, $\hat{\sigma}_w$, was biased, depending on the true value of σ_w . The effect of these biases is that for the case of \hat{g}_1 and \hat{g}_2 , etc., calculated as a function of the estimated signal to noise ratio, \hat{r} , the calculated Kalman gains are biased.

Our approach is to first estimate σ_w of the signal model and σ_v of the data model in Fig. 1 and to compute \hat{g}_1 , a "measured Kalman gain", using Eggers' estimation algorithm [5] and the Kalman gain algorithm [4]. The parameter g_1 is also known as α , for the 2-state implementation, after the " α - β filter" [2]. Fuzzy methods are then used to correct the estimated $\hat{g}_1 = \hat{\alpha}$. From the corrected α , the corresponding β is obtained in the second-order filter from equation 3.6, below.

For higher-order models, the \hat{g}_i are individually corrected.

2 ROBUST BAYES CALCULATIONS

For the discrete time Gauss-Markov generating model with state and measurement equations like 1.2, the corresponding Kalman filter system of equations [4] and Eggers' estimation algorithm [5] becomes

1. Update $\hat{\mathbf{x}}(k|k-1)$ and $\mathbf{P}^{-1}(k|k-1)$ in Kalman filter algorithm.

2. Update $\mathbf{P}^{-1}(k) = \mathbf{P}^{-1}(k|k-1) + \mathbf{H}^T(k) \cdot \mathbf{R}^{-1}(k) \cdot \mathbf{H}(k)$.
3. Compute the error covariance matrix by inverting $\mathbf{P}^{-1}(k)$ to get $\mathbf{P}(k)$.
4. Compute the Kalman gain matrix $\mathbf{G}(k) = \mathbf{P}(k) \cdot \mathbf{H}^T(k) \cdot \mathbf{R}^{-1}(k)$.
5. Update estimate $\hat{\mathbf{x}}(k) = \hat{\mathbf{x}}(k|k-1) + \mathbf{G}(k)(z(k) - \mathbf{H}(k) \cdot \hat{\mathbf{x}}(k|k-1))$.
6. Get ready for next iteration. $\hat{\mathbf{x}}(k+1|k) = \mathbf{F} \cdot \hat{\mathbf{x}}(k)$ and $\mathbf{P}(k+1|k) = \mathbf{F} \cdot \mathbf{P}(k) \cdot \mathbf{F}^T + \mathbf{b} \cdot \mathbf{b}^T \cdot \mathbf{Q}(k)$, and invert $\mathbf{P}(k+1|k)$. Go to Step 4) until iteration ends.
7. Estimate the driving signal and noise processes after computing the tracking error $e(k)$; [5]

$$\begin{aligned} e(k) &= y(k) - \hat{y}(k) \\ \hat{w}(k) &= (\mathbf{b}^T \mathbf{b})^{-1} \mathbf{b}^T \cdot \mathbf{G}(k+1) \cdot e(k+1) \\ \hat{v}(k) &= (1 - \mathbf{H}(k) \cdot \mathbf{G}(k))e(k). \end{aligned}$$

8. Estimate the mean and variance of the driving signal and noise statistics.

$$\begin{aligned} \hat{\mu}_w(k) &= a \cdot \hat{\mu}_w(k-1) + (1-a) \cdot \hat{w}(k-1) \\ \hat{\mu}_v(k) &= a \cdot \hat{\mu}_v(k-1) + (1-a) \cdot \hat{v}(k-1) \\ \hat{\sigma}_w^2(k) &= a \cdot \hat{\sigma}_w^2(k-1) + (1-a) \cdot (\hat{w}(k) - \hat{\mu}_w(k))^2 \\ \hat{\sigma}_v^2(k) &= a \cdot \hat{\sigma}_v^2(k-1) + (1-a) \cdot (\hat{v}(k) - \hat{\mu}_v(k))^2. \end{aligned}$$

In simulations, the driving signal $N(0, \sigma_w^2)$ and noise $N(0, \sigma_v^2)$ is generated for the signal and noise generating model, where σ_v^2 is considered 1024 meters² and σ_w^2 is set corresponding to the optimum value for each Kalman gain. The value of "a" is chosen to give good smoothing for the mean and variance estimators. The values $T = 1$ and $0.85 \leq a \leq 0.98$ are used here. Also, the estimated state $\hat{\mathbf{x}}$, the transformed steady-state tracking error covariance matrix \mathbf{P} , and the mean estimators and variance estimators of the signal and noise are initialized. Here all initial conditions at stage 0 are assumed to be 0.

3 FUZZY TUNING METHOD

3.1 Numerical Estimation of SNR

In a GPS-like tracking application σ_v , is usually considered fixed (although perhaps not known exactly) and the signal standard deviation, σ_w , is widely variable. The limited range of σ_v is here taken as:

$$10 \leq \sigma_v \leq 100 \quad \text{meters} \quad (3.1)$$

This range is representative of the Global Position System (GPS) of satellite radio navigation [6]. Now, the design parameter which allows us to solve for the Kalman gains, g_1 and g_2 (or α and β), is SNR :

$$SNR = \left(\frac{T^2 \sigma_w}{2 \sigma_v}\right)^2 = r \quad (3.2)$$

$$SNR_{dB} = 10 \cdot \log_{10}(SNR) \quad (3.3)$$

Thus, for simulating the $w(k)$ process, the procedure is as follows:

1. Choose T (sampling time), and σ_v -value.
2. Choose SNR -value (or SNR_{dB}).
3. Calculate σ_w -value.

After designing the parameters of σ_v and σ_w , we generate the simulated gaussian (normal) signal $N(0, \sigma_w^2)$ and the noise $N(0, \sigma_v^2)$, and then estimate both parameters, and compute a

measured SNR and the measured Kalman gain, \hat{g}_1 , using the Eggers' estimation algorithm and the Kalman filter algorithm.

The range of SNR is obtained from the Design Nomograph shown as Fig. 4 of [2]. For the range considered $-40 \text{ dB} \leq SNR_{dB} \leq 10 \text{ dB}$, SNR_{dB} is roughly proportional to α values. We partition the range of α (Kalman position gain) over which the filter must adapt. We will assign different fuzzy membership functions for each α -value in the partition. From the Nomography, we see that if we start $SNR_{dB} = -40 \text{ dB}$ and increase in 5 dB steps, α changes in roughly 0.1 steps, from about 0.1 to 0.9. A partition of the α range into 9 regions is taken to be sufficient.

3.2 Fuzzy Estimation of α and Computing β

We may apply "fuzzy" methods to the computation of α only. Then, β is computed from α . To compute α requires knowledge (an estimate) of SNR , r , given in equation 3.4.

The parameters α and β (g_1 and g_2) for the steady-state Kalman filter are computed by use of the equations, in the Alpha-Beta filter paper [2], as:

$$r = \left(\frac{T^2 \sigma_w}{2 \sigma_v}\right)^2 = SNR \quad (3.4)$$

$$\alpha = g_1 = \left(\frac{r + 4\sqrt{r}}{2}\right) \left(\sqrt{1 + \frac{4}{r + 4\sqrt{r}}} - 1\right) \quad (3.5)$$

$$\beta = Tg_2 = \frac{\alpha^2 + r(1 - \alpha)}{2 - \alpha} \quad (3.6)$$

Running with steady-state gains, the initialization for the simulated model states are determined, such that it will start in steady-state immediately.

The fuzzy method is based on making decisions [3]. The "decision-event" is "what partition of α is implied by the measured data, $\hat{\alpha}$ ". In order to set up "Decision Membership Function" (DMF) of Fig. 3 for α , we need an estimator, $\hat{\alpha}$, which reasonably approximates α .

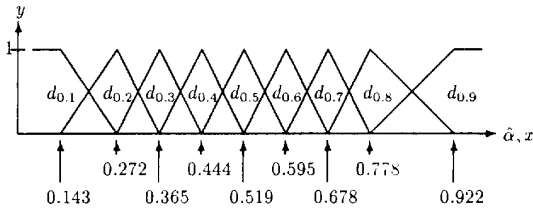


Figure 3: Decision Membership Function

There are two requirements on $\hat{\alpha}$:

1. It should have the same scale as α .
2. It should be monotonic in α .

Now, if we compute

$$\hat{r} = \left(\frac{T^2}{2} \cdot \frac{\hat{\sigma}_w}{\hat{\sigma}_v}\right)^2 \quad (3.7)$$

where $T = 1$, and $\hat{\sigma}_w$ and $\hat{\sigma}_v$ are the two available estimates, then \hat{r} is monotonic in α . However, \hat{r} appears to be logarithmic in α . Thus, we define the estimated Kalman position gain as:

$$\hat{\alpha} = \left(\frac{0.4}{5.6}\right) \cdot \ln(\hat{r}) + 0.67 \quad (3.8)$$

Once we obtain $\hat{\alpha}$, we can compute $\hat{\beta}$ from equation 3.6. We have an estimator of $\hat{\alpha}$ which is monotonic and of the same scale as α . It is also nearly linear in α , at least over the range,

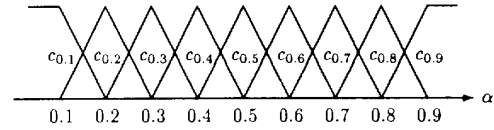


Figure 4: Control Membership Function

$0.1 \leq \alpha \leq 0.9$. The corresponding control membership functions (CMF) are shown in Fig. 4.

The estimated SNR , \hat{r} , like the true one, appears to be logarithmic in $\hat{\alpha}$. Now, if we define

$$\hat{r}_i = \left(\frac{T^2}{2} \cdot \frac{\hat{\sigma}_{w_i}}{\hat{\sigma}_v}\right)^2 \quad (3.9)$$

$$\hat{\alpha}_i = \left(\frac{0.4}{5.6}\right) \cdot \ln \hat{r}_i + 0.67 \quad (3.10)$$

where $T = 1$ and $\hat{\alpha}_i$, is estimated for $\sigma_v = 10, 32, 100$, respectively. The results are shown in Table 1 and depicted in Fig. 9. The numerically estimated α 's are clearly biased, as a function of σ_v .

$T = 1, \delta = \frac{1}{\sqrt{2}}$						
α	$\hat{\alpha}_{10}$	$\hat{\alpha}_{32}$	$\hat{\alpha}_{100}$	$\hat{\alpha}$	$\hat{\alpha}$	FTV
0.1	0.2232	0.1401	0.0653	0.143	0.139	0.100
0.2	0.3448	0.2698	0.2016	0.272	0.270	0.198
0.3	0.4324	0.3616	0.2998	0.365	0.362	0.297
0.4	0.5064	0.4400	0.3850	0.444	0.440	0.395
0.5	0.5756	0.5141	0.4664	0.519	0.514	0.493
0.6	0.6452	0.5893	0.5502	0.595	0.589	0.592
0.7	0.7205	0.6715	0.6423	0.678	0.672	0.693
0.8	0.8102	0.7705	0.7525	0.778	0.771	0.793
0.9	0.9402	0.9146	0.9099	0.922	0.915	0.895

Table 1: True & Estimated α 's and FTV of α

In order to set up Decision Membership Function, the estimated α 's are averaged in simulation to get $\bar{\alpha}$, as

$$\bar{\alpha} = \frac{\hat{\alpha}_{10} + \hat{\alpha}_{32} + \hat{\alpha}_{100}}{3} \quad (3.11)$$

The results are shown in Table 1.

What the research actually accomplishes is to find the measured values, $\hat{\alpha}$, that correspond to $\alpha = 0.1, 0.2, \dots, 0.9$, and put the partitions on these α -values to set up the DMF (Fig. 3). In this method, the membership value from the DMF is used to weight with the CMF (Fig. 4). Then the Fuzzy Tuned Values (FTV) of β are computed from equation 3.6, above.

This procedure is based on use of the so-called "soft" counterparts of the usual "hard" fuzzy logical connectives [3]. The soft connectives are directly relatable to Bayes formulation of fuzzy control.

The procedure for obtaining the FTV of α are as follows:

1. Establish the DMF based on the estimated α and the CMF with equi-triangular shape centered at the true "Kalman position gain" (α).
2. Mark the estimated α values for tuning on the x -axis, and find the corresponding values on the y -axis in the DMF.
3. Compute the FTV of α , which is obtained from the product of the two points in the DMF and the corresponding CMF values, respectively.

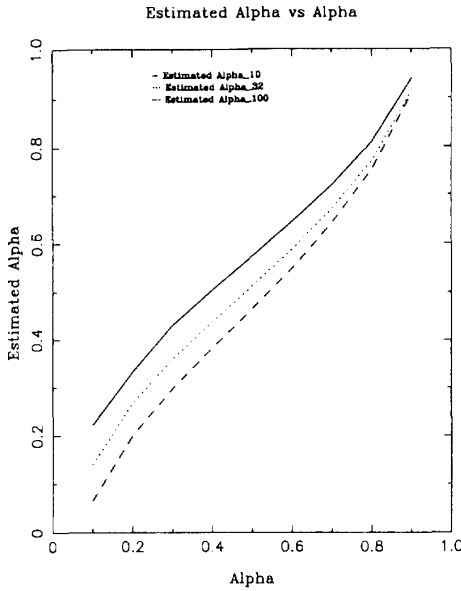


Figure 5: Est. Kalman position gains $\hat{\alpha}_i$ vs α

4 FUZZY-TUNED FILTER PERFORMANCE

The model of Fig. 1 and filter of Fig. 2 are simulated, where the standard deviation of the acceleration driving process, $w(k)$, is changed step-wise, every 100 samples. The standard deviation of the noise process, $v(k)$, is held constant at 32 meters. Thus, this is a GPS-motivated case. The individual standard deviations of $w(k)$ are chosen so as to yield optimum filter gain (g_1) values of 0.7, 0.5, 0.9, 0.8, and 0.6, in that sequence.

Fig. 6 compares the filter gains (g_1) computed, respectively, for the fuzzy-tuned filter and for an optimum filter, the latter with instantaneous, exact knowledge of the required statistics. For the fuzzy-tuned filter, the value of “ a ” in the first-order variance estimator is 0.9, which is the z-plane real pole location. This value was chosen, ad hoc, to yield a good compromise between convergence time of the gain (g_1) estimate and the noise of that estimate. Fig. 6 shows that the g_1 gain for the fuzzy-tuned filter converges rapidly to the neighborhood of the optimum gain, with a little noise evident.

Fig. 7 compares the standard deviation estimates of tracking error for the optimum filter case and fuzzy-tuned case, respectively. The tracking error in question is not that shown in Fig. 2 as $e(k)$, which is the prediction error. Rather, we use here the filtered tracking error, given by

$$e(k|k) = y(k) - \hat{y}(k|k) \quad (4.1)$$

These standard deviation functions are computed from a sequentially filtered estimate of variance, given by

$$\hat{\sigma}_w(k) = \sqrt{a \cdot \hat{\sigma}_w^2(k-1) + (1-a) \cdot (\hat{w}(k) - \hat{\mu}_w(k))^2} \quad (4.2)$$

The graphs of Fig. 7 reflect the sequential steps in acceleration and filter gain, shown previously. In the noisy graph of standard deviation of tracking error, part of the displayed noise is due to the variance estimator, itself. Sample variance, by its very definition, is a long time average. But, in order to get any reasonable display at all, over the 100-sample intervals, we have had to reduce “ a ” (the estimator memory constant) to 0.85. Thus, the variance estimator itself contributes significant noise to the graph.

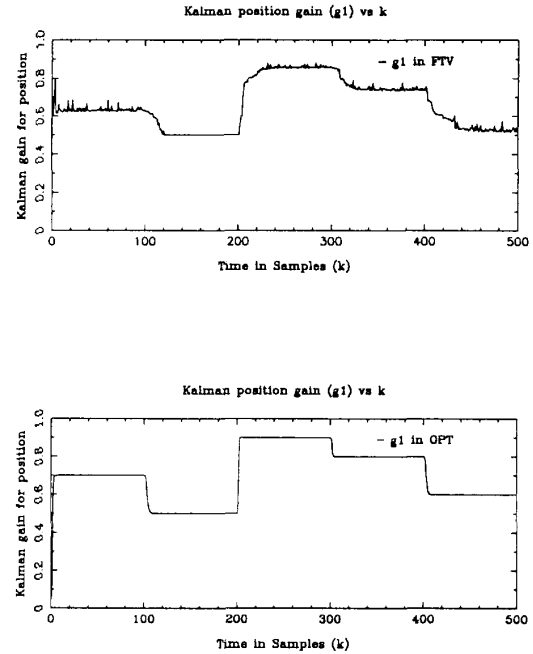


Figure 6: Five kinds of g_1 in optimum case

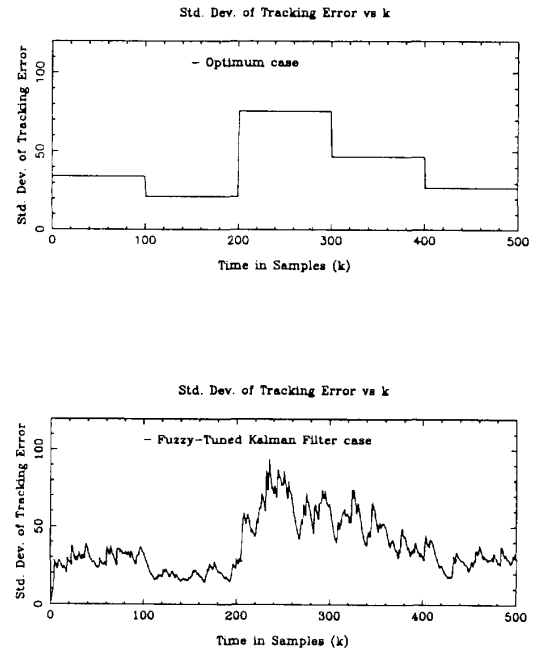


Figure 7: Five kinds of standard deviation of tracking error in optimum and fuzzy-tuned Kalman filter cases

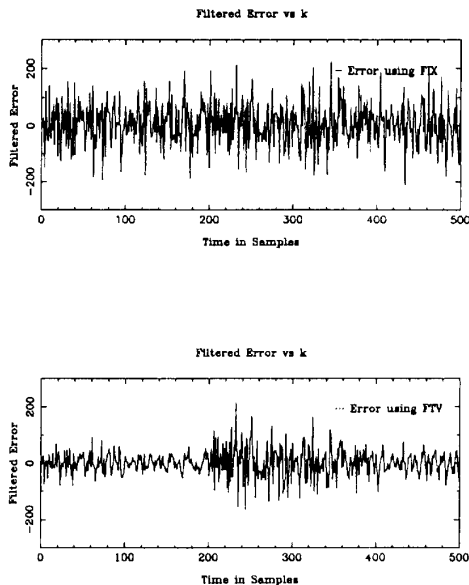


Figure 8: Filtered errors $e(k)$ using a fixed gain (for $\alpha = 0.9$) and, a Fuzzy Tuned Value (FTV) in the Kalman filter

Since the fuzzy-tuned gains of Fig. 6 are so close to optimum, one assumes that the tracking error variance would also be close to optimum. Perhaps a better view of this tracking error performance can be obtained by looking at the error waveform, itself, during the sequence of acceleration variances and their corresponding Kalman gains. This is done in Fig. 8. The first graph in Fig. 8 shows the tracking error waveform for a Kalman filter whose gains are fixed over the entire operating interval. The parameter g_1 is fixed to a value of 0.9, corresponding to operation for the highest dynamic condition of the model (greatest acceleration). For this condition, the filter is “wide-open,” so to speak, and admits the greatest amount of white measurement noise, $\nu(k)$, of the five consecutive intervals. Thus, the tracking error is large over all five intervals.

The second graph in Fig. 8 shows the fuzzy-tuned filter, wherein the gains vary from interval to interval. Here is visible the fact that the tracking error variance is indeed smaller on some intervals and larger on others. Specifically, the standard deviation is largest when g_1 is greatest (0.9) and is smallest when g_1 is least (0.5). Physically, this reduction of tracking error is because low gain corresponds to low bandwidth, when less of the (fixed-variance) white measurement noise is admitted to the filter.

Fig. 9 shows the effective straightening of the non-linear characteristics of the numerically estimated filter gains of Fig. 5. The noiseless graph of Fig. 9 implies long-term averaging, which is not available in practice.

5 CONCLUSION

What has been done in the work reported here is to use Fuzzy control to correct a biased numerical estimator of gain parameters for an adaptive Kalman filter. It might be inferred from the graphs of Figs. 5 and 9 that not much correction of the numerical-only adaptive filter is needed. And, therefore, the significance of the present results might be overlooked. What is important is that the true optimum filter, whose performance is shown in Figs 6 and 7, is not physically implementable. All approaches to physical implementation must be ad hoc in one sense or another. Our present approach works far better than

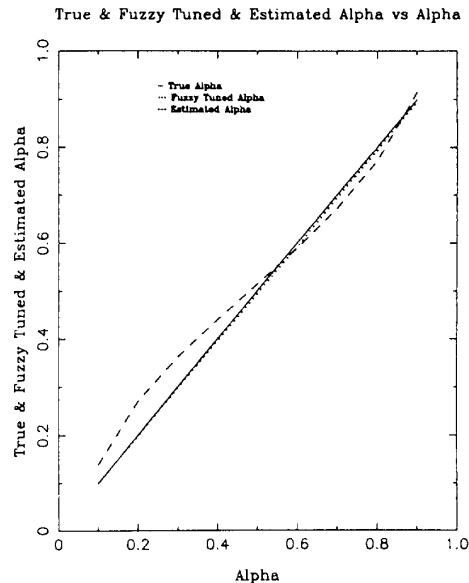


Figure 9: True, Fuzzy-tuned and Estimated α vs α

a previous attempt, using the biased numerical-only adaptive algorithms [5]. And, it is the novel Fuzzy Control approach which makes it work so well. What the present approach does is to replace the numerical-only dependence on the Ricatti gain equation in the estimation part of the procedure. It replaces Ricatti with Fuzzy plus a simple sequential averaging, whose memory constant may be varied to balance convergence time with estimator noise. Ricatti is then used, not to estimate, but only to compute the other Kalman gains from the first (g_1) estimate.

The fuzzy-tuned approach to the adaptive Kalman filter is of general interest in its own right. However, the application to the GPS scenario is of great current practical interest. Many companies are now bringing GPS receivers to market. Some use fixed-gain Kalman filters. More expensive receiver versions use various means to adapt the filter. Since receiver cost is a major market factor, the existence of a relatively simple adaptive algorithm of high performance is of great current practical interest.

References

- [1] R. E. Kalman (1960), “A new approach to linear filtering and prediction problems,” *ASME, J. Basic Eng.*, Ser. D., vol. 82, pp.35-45, March 1960.
- [2] John H. Painter, David Kerstetter and Steve Jowers (1990), “Reconciling Steady-state Kalman and Alpha-Beta Filter Design,” *IEEE Transactions on Aerospace and Electronic Systems*, **AES 26** (Nov. 1990), NO. 6.
- [3] John H. Painter (1992), “Soft Fuzzy Control,” *The Second International Workshop on Industrial Applications of Fuzzy Control and Intelligent Systems*, College Station, Texas Dec. 2-4, 1992.
- [4] Robert G. Brown, *Introduction to Random Signal Analysis and Kalman Filtering*, John Wiley and Sons, Inc.
- [5] Mitchell D. Eggers, *A Robust Empirical Bayes Approach to Adaptive Kalman Filter*, Texas A & M University (May 1984).
- [6] R. Grover Brown and Patrick Y. C. Hwang (1983-1984), “A Kalman Filter Approach to Precision GPS Geodesy,” *NAVIGATION: Journal of The Institute of Navigation*, Vol. 30, No. 4, 1983-1984.