Fuzzy Prediction of Timeseries

PRATAP S. KHEDKAR [1]
CS Division, Department of EECS,
University of California at Berkeley,
Berkeley, CA 94720.
khedkar@diva.berkeley.edu

SRINIVASAN KESHAV [2]
CS Division, Department of EECS,
University of California at Berkeley
and
International Computer Science Institute,
Berkeley, CA 94720.
keshav@tenet.berkeley.edu

Abstract

This paper presents an approach to time series extrapolation based on fuzzy control. The standard exponential averaging scheme is inflexible in that it gives a fixed weight to past history, thus ignoring transient phases in system dynamics. We present a modification to the scheme where the control parameter of the averaging scheme is dynamically adjusted by a simple fuzzy logic controller. The design of the controller is described, and the scheme is evaluated by simulation on test workloads and by application to the real-world problem of flow control in communication networks. We also study the sensitivity of our system to its descriptive parameters.

1 Statement of the Problem

A frequently occurring problem in many areas of the physical sciences is that of extrapolating a time series into the future, given that the observed values can be corrupted by noise. Consider a scalar variable θ that assumes the sequence of values $\theta_1, \theta_2, ..., \theta_k, \theta_{k+1}$ and can be represented as

$$\theta_{k+1} = \theta_k + \omega_k$$

where ω_k is a random variable from some unknown distribution, called the *system perturbation.*

Suppose that an observer sees a sequence of values $\{\tilde{\theta}_k\}$ and wishes to use the sequence seen so far to estimate the next value of θ. In many cases, the observed sequence is corrupted by some noise (introduced by the measurement process), so that the observed value $\tilde{\theta}_k$ is not the actual value θ_k. We represent this by

$$\tilde{\theta}_k = \theta_k + \mu_k$$

where μ_k is another random variable from an unknown distribution, referred to as the *observation noise.*

Since the perturbation and noise variables can be stochastic, the exact value of θ_k cannot be determined. What we require, instead, is that $\hat{\theta}_k$, the predictor of θ_k, be optimal in some sense.

2 Classical approach

The standard solution to this problem is to use a Kalman predictor or one of its many variants [1] [3]. This is optimal in the sense that the expected squared error in $\hat{\theta}_k$ is zero. However, the system perturbation and observation noise variables must be from a zero mean, gaussian, white noise distribution and the observer must supply the variances of the system perturbation and the observation noise (though if the noise is colored, the Kalman predictor is still the optimal *linear* predictor of θ_k).

The robustness of the method has made it very popular in the control literature, but it requires the system and observation noise variances to be known in advance. One cannot obtain these values simply by looking at past observations, since the sequence $\{\tilde{\theta}_k\}$ is the result of both the system perturbation and the observation noise. A simple analysis shows that in such a case, the variance of each component cannot be extracted.

In this paper we consider a new approach to prediction based on fuzzy control. The approach is simple, yet it affords much generality, and we believe that it can be applied to a number of practical problems.

[1] P. Khedkar supported by NASA grant NCC-2-275

[2] S. Keshav supported by AT&T Bell Laboratories, MICRO and Hitachi Corporation. His current address: AT&T Bell Labs, 600 Mountain Ave., Murray Hill NJ 07974

3 Assumptions

We model the parameter θ_k as being the state variable of an *unknown* dynamical system. In many situations, there is not enough data and/or modeling power to describe the system dynamics analytically and isolate the 'system drift' component, which may be highly non–Gaussian. The variances involved may themselves change over time. Since the standard assumptions are unreasonable and restrictive in this scenario, we relax them and make some weak assumptions about the system dynamics. First, the time scale over which the system perturbations occur is assumed to be an order of magnitude larger than the corresponding time scale of the observation noise. Thus, the observations can be thought of as a high frequency signal that is amplitude modulated by the system dynamics.

Second, we assume that system can span a spectrum ranging from 'steady' to 'noisy'. When it is steady, the amplitude of the system perturbations is close to zero, and observed changes are due to the observation noise. When the system is noisy, the state can change, but with a time constant that is longer than the time constant of the observation noise. It is assumed that the sampling process does not have any systematic error.

Note that our approach is very general, since we do not make any assumptions about the distributions of ω and μ. Also, we do not require any conditions on the variances of these distributions. On the other hand, we do not guarantee optimality of the resulting predictor: we only claim that the method is found to work well in practice.

4 Exponential averaging

The basis of our approach is the variant of the Kalman predictor called the exponential average predictor. This is given by the recurrence:

$$\hat{\theta}_{k+1} = \alpha\hat{\theta}_k + (1-\alpha)\tilde{\theta}_k$$

The predictor is controlled by a parameter α, that can be related to the system perturbation and observation noise variances. However, it has another, more intuitive, interpretation: α can be thought of as the weight given to past history. The larger the value of α, the more weight past history has in relation to the last observed value of the parameter. The method is called 'exponential', since the predictor is the discrete convolution of the observed sequence with an exponential curve with a time constant 1/(1-α). Alternately, if the value of $\tilde{\theta}$ becomes fixed, the error, $\tilde{\theta} - \hat{\theta}$, decays exponentially. This averaging filters out the high-frequency components, hopefully eliminating the observation noise and letting the system perturbations through.

The exponential averaging technique is very robust, and has been used in a number of applications, ranging from computer communication protocols (for round trip time estimation), to the social sciences. However, a major problem with the exponential averaging predictor is in the choice of α. While in principle, it can be determined by knowledge of the system and observation noise variances, in practice, the variances are unknown. It would be useful to **automatically determine** a 'good' value of α, and to be able to **change** this value on-line if the system behavior changes. Our approach uses fuzzy control to effect this tuning.

5 Fuzzy exponential averaging

Our technique is based on the assumption that a system displays a spectrum of behavior ranging from 'steady' to 'noisy'. In a 'steady' system, the sequence $\{\theta_k\}$ is approximately constant, so that $\tilde{\theta}_k$ is affected mainly by observation noise. Then, α should be large, so that the past history is given more weight, and transient spikes in $\tilde{\theta}$ are ignored.

In contrast, the data could be from a 'noisy' system, so that θ_k itself varied steadily and considerably over time. In that case, $\tilde{\theta}$ reflects changes both in θ_k and the observation noise. A high α would filter out changes in the system too, and would cause a large lag in the predictor. By choosing a lower value of α, the observer quickly tracks changes in θ_k, while ignoring past history which only provides obsolete information. In the limit, α can be set to 0, so that the predictor $\hat{\theta}_k$ becomes maximally responsive to θ_k.

While the choice of α in the extremal cases is simple, the choice for intermediate values along the spectrum is hard to make. We use a fuzzy controller to determine a value of α that gracefully responds to changes in system behavior. Thus, as the system evolves along the noise spectrum, the value of α changes continuously to match the change, and this allows us to obtain a good estimate of θ_k at all times. Moreover, if the observer does not know α *a priori*, the predictor automatically determines an appropriate value.

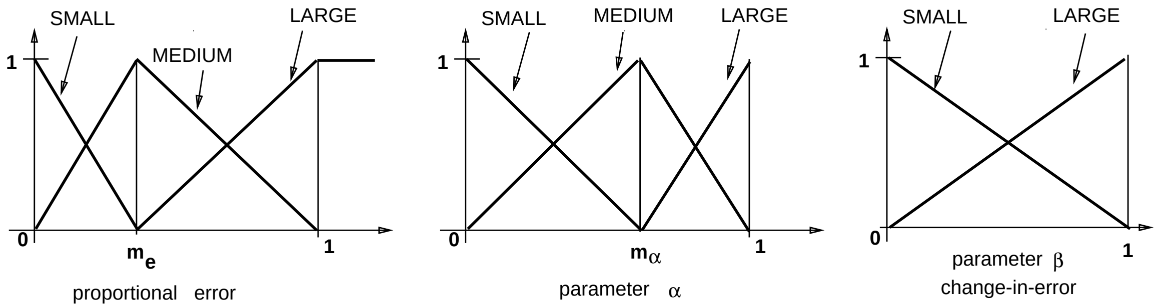

Figure 1: Membership functions for all 4 variables : there are only 2 free parameters

6 System identification

In the previous section, we linked the choice of α to the 'noise' in the system. The problem now reduces to the following: how can we determine the amount of 'noise' in the system? This is done by means of another key observation.

We will temporarily assume that the variance in the system perturbation is an order of magnitude larger than the variance in the observation noise. Given this assumption, if a system is 'steady', then the estimation error is small. That is, if the system has little perturbation, then the exponential averaging technique will produce a predictor that is usually close to the actual system state, and thus, close to the observed state. If that is so, when the prediction errors are small, the value of α should be large. In contrast, if the system is 'noisy', then the exponential averaging technique will give a predictor that usually has a large estimation error. This is because when the system noise is large, past history cannot predict the future well. So, no matter what the predictor does, it will usually have a larger error. In that case, the best thing to do is to give little weight to past history, and set α to be low, so that the observer can track the changes in the system.

To summarize, the observation is that if the predictor error is large, then α should be small, and *vice versa*. Treating **small** and **large** as fuzzy linguistic variables, we have a fuzzy controller for the estimation of α, as desired.

7 Fuzzy control

Fuzzy control requires (a) specification of the rules in terms of linguistic variables, and (b) the membership functions of the fuzzy labels used. Then a standard fuzzify–combine–defuzzify technique can be used for inference [5].

Since we do not have a good grasp of the state dynamics, we have only three gradations in the values of α and *error*. Even if additional resolution was provided, there is not enough domain knowledge to specify α values for all the finer gradations of error. In any case, a finer granularity would be justified only if the correct α value is *not* the one obtained by interpolation among the three rules below. This results in the following control rules:

- if *error* is **large** then α is **small**
- if *error* is **medium** then α is **medium**
- if *error* is **small** then α is **large**

The shapes and ranges of the fuzzy labels **small**, **medium** and **large** remain to be specified. We look at various ways in which this can be done and will argue that a reasonably simple choice works well. We make the standard assumption that these are all trapezoidal-shaped, normal fuzzy sets. This would usually require specifying four parameters for each linguistic variable (the interval of support and the interval over which membership is 1). However, we argue below that in this estimation context, the overlap should be maximal (as shown in Figure 1), and hence we have 2 free parameters (m_e and m_α), the modes for the two **medium** labels.

The reasons for this are as follows. If l_e is not 0, there exists a range of non-zero error, for which α will be set to 1. This means that the estimator will never catch up, maintaining a residual error even in the absence of all noise. On the other hand, the choice of r_e is linked to the choice of definition and scaling of the error variable, so the latter can be chosen to make r_e 1. We will show (in Section 9), that the choice of l_α and r_α do not affect the system's qualitative behavior, but a non-extremal choice here causes a larger gradient in the system response curve, which is undesirable. Hence there are two degrees of freedom left: m_e and m_α.

The tuning to compute a good value for these can be done using a neural network training algorithm (as in [2]), but here we concentrate on the effects of these two parameters as they are varied over the entire range. In any case, fine-tuning is bound to be application specific, and we are only interested in general characteristics of such a system at the moment. A further possible simplifying assumption is to force the two middle points to be identical in value. This has no justification, apart from reducing the number of parameters. We have tried this as well, but found it to be too restrictive.

Since $\alpha \in [0,1]$, and the error is unbounded, we have used proportional (relative) error, which is defined as |error/predictor|. The absolute value is good enough since there is no reason, in general, to suppose that positive and negative error should induce different behavior for α. The output of the controller is defuzzified using a standard centroid defuzzifier, and scaled up to fit the response range [0,1].

8 Smoothed proportional error

If the controller input is the absolute value of the proportional error, then the value of α can change drastically due to a single observation with a large error. This is a problem, since if α drops to a small value, the entire past history can be quickly lost. We need a way to smooth changes in the error due to observation noise.

Our solution is to use an exponential averaging filter to smooth the error as well. The value of the smoothing constant, say β, should usually be low so that the predictor is sensitive to errors, but high enough to eliminate spikes in the error. Thus, if the errors are consistent, then the result of the smoothing is the value of the error. However, if the error is transient high, probably due to observation noise, then the spike is effectively ignored, and the controller for α is unaffected by it. The result is that the predictor for θ_k responds to consistent errors in prediction, and is unaffected by single large errors. Such smoothing cannot be utilized in a system with a fixed α.

Note that with this addition, the size of the observation noise variance in relation to the system perturbation variance does not matter. Even if the observation noise variance is large, the smoothing of the observation noise removes the rapidly changing transients in the noise, and so the controller for α sees something that reflects the system perturbations alone. Hence, the earlier assumption that the variance of the observation noise be smaller than the system perturbation variance may now be removed.

The desired behavior for β is achieved by designing another fuzzy controller with the following rules:

- if *change-in-error* is **high**, β is **high**
- if *change-in-error* is **low**, β is **low**

The linguistic values **low** and **high** are defined as triangles (see Figure 1). Note that there are no additional free parameters which require specification here. The change in error was initially defined to be the difference between the current value and the previous value of the proportional error. We found that this could eliminate only a single outlier in the error. If outliers can be of longer durations, due to correlations in the observation noise, then *change-in-error* must be redefined. In practice, we found that defining it as the change over 3 successive samples, which eliminates spikes of length upto 3, works quite well. The value of β is computed before the error is smoothed; the smoothed value of error is then used to compute α.

Figure 2 is the signal flow diagram of the resultant system. It incorporates two exponential averagers, and two fuzzy controllers. In general, because of the complexity of the system, it could behave poorly, and may even be asymptotically unstable. However, both the fuzzy system and the exponential averagers are robust and insensitive to small errors. Hence the system is quite well behaved.

9 Sensitivity analysis

This section studies the sensitivity of our system to its two descriptive parameters: m_e and m_α. The first fuzzy component of the system provides a mapping from proportional error to α. This mapping is nonlinear because of the defuzzification process. However, the map is continuous and differentiable at all points of interest. Continuity is due to the use of fuzzy sets (and their overlap), and differentiability is follows from the centroid defuzzification process. A typical response curve is shown in Figure 3: it is a rational function

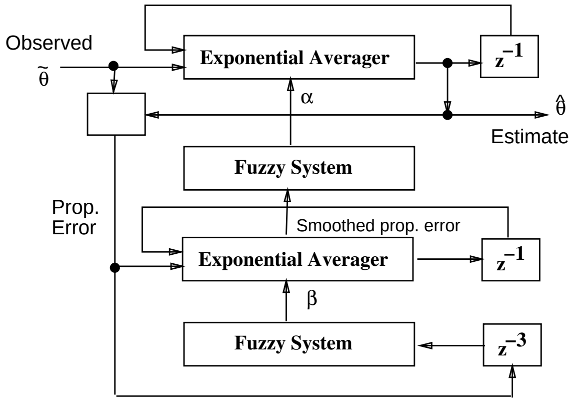

Figure 2: Schematic flow diagram of the prediction system

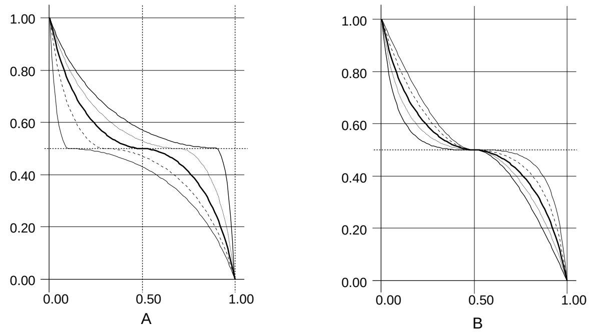

Figure 3: The Input–Output Mapping of the α-computing system, as a function of its parameters (A) $m_\alpha = 0.5$, m_e varies from 0.1 to 0.9, (B) $m_e = 0.5$, m_α varies from 0.3 to 0.7.

(the quotient of a cubic and a quadratic polynomial). It has the desirable properties of being a smooth, monotonic interpolation between the extremal cases, with small gradient in the intermediate region. The exact derivatives at 0, 0.5 and 1 are given by:

$$\frac{-m_\alpha(4-m_\alpha)}{2m_e(1-m_\alpha)}, 0, -\frac{(m_\alpha+3)(1-m_\alpha)}{2m_\alpha(1-m_e)}$$

How critical is the choice of the two parameters m_α and m_e? We need to show that the response curve as well as its derivative are not sensitive to small changes in the values of these two parameters. The former is evident from the figures (it is easy to prove this analytically as well). A similar analysis shows that the derivative $\partial f/\partial x$, at $x = 0.5, m_e = 0.5$, is 0 and independent of the value of m_α. If m_α is fixed at 0.5, then this derivative at the same point varies as $3z/(1-10z)$, where z is a small change in the value of m_e. For behavior at larger deviations, see Figure 3.

Several qualitative observations can be made about the family of curves so produced :

- The effect of varying m_e, while holding m_α fixed, is to smoothly shift the 0.5-crossing in the appropriate direction. This has the effect of stretching either the high or low error regions. If the user requires higher sensitivity in any of these, he can bring it about by tweaking the m_e value. However, the response curve (as a whole) is not sensitive to small changes in m_e around 0.5, so the exact value of m_e is not very important. Specifically, high precision in specifying this value is not needed, and fine-tuning may be dispensed with.

- The effect of varying m_α (while m_e is fixed) is complementary. The 0.5-crossing remains fixed at 0.5, whereas gradients are changed in the two regions. Again, around 0.5, variation in m_α has only limited effect and does not change the qualitative behavior.

- A suitable combination of the two effects can be obtained by varying both parameters. Extreme values for either cannot be justified (given just a gross picture of the system dynamics), and should be avoided. The actual choices are left to the user and once chosen, can remain fixed. This is an advantage in real-world applications.

We propose fuzzy control as a way to describe and manage a non-linear system. The main point to be made here is that the system response curve, though nonlinear, is well-behaved. This behavior can be analyzed and checked easily. Moreover, the nonlinearity is the result of using fuzzy logic control in a simple way. If an adaptive system were to be designed in the classical manner, the designer would have to specify this nonlinearity in all detail. Since the parametric forms as well as the parameter values would have to be chosen (quite subjectively), such an approach seems impractical.

10 Simulation Results

We tested the fuzzy predictor using two kinds of simulations. In the first set, an artificially generated input was presented to the predictor, and the behavior of the predictor was compared to a system where the value of α is fixed.

We used a few different types of system dynamics, corrupted with varying amounts of gaussian observation noise:

- system is sinusoidal, with noise
- super linear growth, with noise
- system is constant, observation noise increasing

To compare our results with the standard approach of fixing α, we have fixed α at 0.9. This is because even a moderately low, fixed α (say, below 0.5), is very sensitive to noise and is bad as a predictor. Both estimators start with a value 10.0. The results of these simulations are summarized in Figure 4. Some remarks on the observed characteristics of this predictor:

- A high, fixed α is very bad at catching up with a varying system. This may happen for a globally convex function (such as a quadratic or cubic), or may happen over a shorter timescale for a sinusoidal form (near the zero-crossing). No exponential estimator can predict a value greater than the previous value seen when the system is increasing, because of the convex combination form used. Both the fixed and adaptive system underestimate in this scenario, but the fuzzy system has a much smaller lag since it has set α to be low, because of consistently high error. This improvement can be clearly seen in Figures 4A–D. It consistently does better at the expense of some smoothness.

- A related problem is that of the long initial transient for fixed α (Figures 4C and 5A). In contrast, the fuzzy system catches up as soon as the error is reflected in the input of the fuzzy map which computes α, since it causes α to drop suddenly, allows the estimator to catch up, and then again raises α if the system is steady enough. This locks in the value. So the start-up value of $\hat{\theta}$ can be arbitrary for the fuzzy system.

- If the system is constant, the fixed-α system has a clear advantage, since fixing α at 1 would deliver perfect behavior. The fuzzy system's behavior will worsen if noise variance increases with time. Figure 5A shows this when the true value is constant at 100 while observation noise increases with time. However, the variance in the estimator (which is entirely due to observation noise) is acceptable for a significant range of noise amplitudes. Note the long transient for a fixed α.

- The essential point in favor of the fuzzy system is adaptability. It can respond to changes in system behavior by changing α, depending on feedback. We see that only two or three rules are required to cover the entire range of possible scenarios and that fuzzy logic smoothly interpolates between the rules.

We now turn to a real-world application — the problem of flow control in a high-speed wide area network. Essentially, each user determines its service rate by regularly probing the network. The sequence of probe values is used to predict the service rate in the next time interval, and the user matches its data transmission

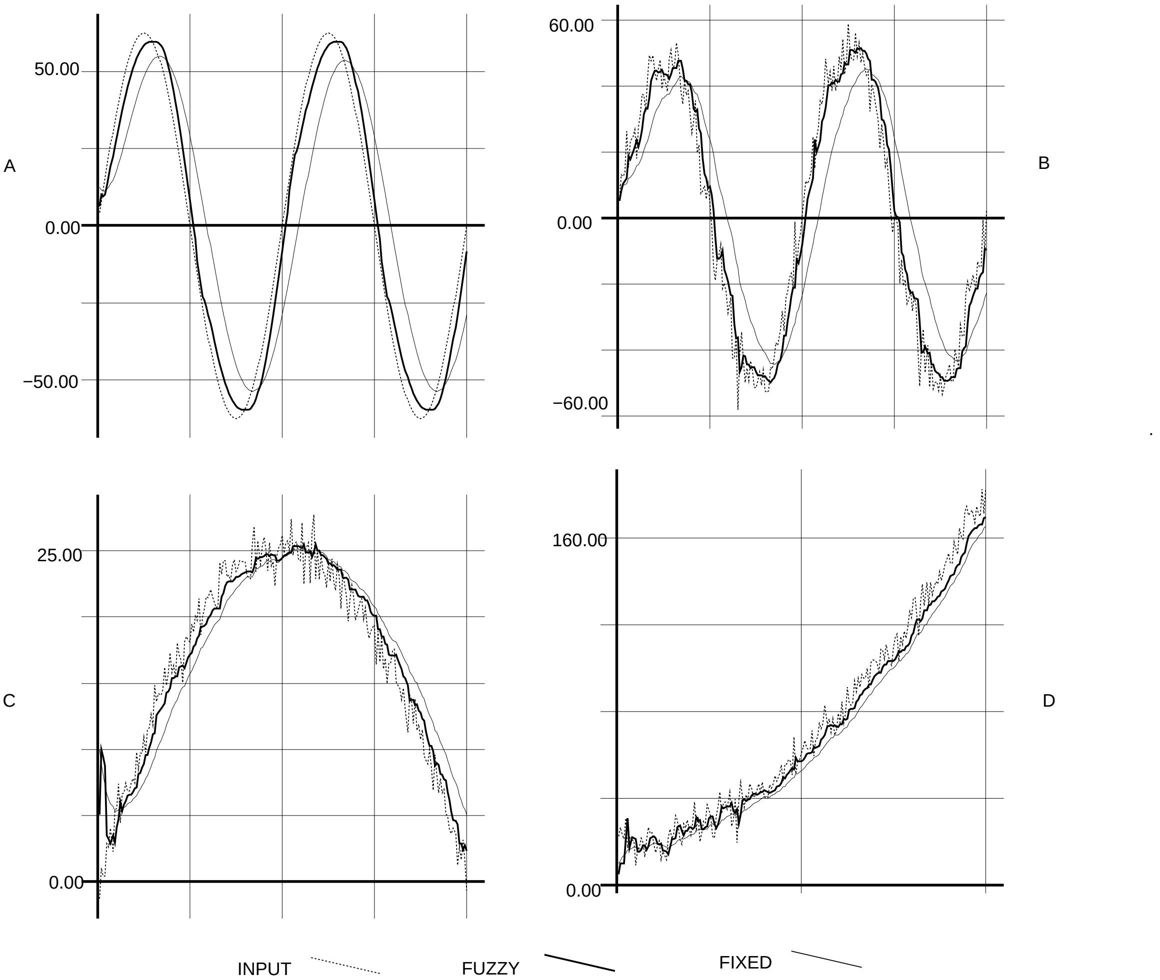

Figure 4: The predictors for different data. A, B : sinusoidal θ with variances 0 and 5. C,D : parabolic and quadratic drift with variances 1 and 5 respectively. $m_e = 0.7, m_\alpha = 0.5$.

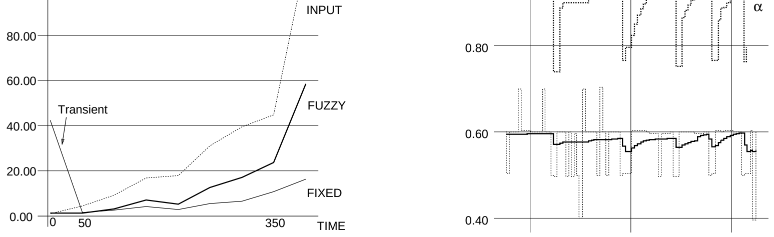

Figure 5: (A) The error of the estimators as observation noise variance increases with time (mean=200). Note the long-lasting transient for fixed α. (B) The round-trip time estimation problem. The change in α is also shown. $m_e = 0.7, m_\alpha = 0.5$.

rate to this predicted rate. Figure 5B is a sample trace showing the probe values and the corresponding choice of transmission rate over a period of 100 seconds. The averaging constant α is also shown.

The transmission rate, for the most part, tracks the probe value. Note that single spikes in the input are ignored, and that, whenever the probe values stabilize, the transmission rate exponentially catches up to the probe value. The fact that the source tracks the input rather closely shows the effectiveness of the fuzzy controller. Since the controller drops the value of α whenever the prediction error is large, it can quickly catch up with the probe value. Thus the source is able to send more data than it could otherwise. See [4] for further details.

11 Drawbacks of the approach

The most important assumption in our work is that system perturbation and observation noise are distinguished by the time-constants associated with them. We do not need specific values but their ratio must be "high enough". While dropping all distinction between the two would make the problem insoluble, it may be possible to weaken the assumption to a constraint on the power laws obeyed by the frequency spectra of the two types of variation.

Proportional error is a problem. It is better than using absolute error, but since it is measured relative to the magnitude of θ, it could get distorted significantly if the value of θ is close to zero. One simple solution to this is to artificially translate the input away from zero, then perform the inverse translation on the prediction. Presently, we tackle this problem by switching to absolute error when the proportional error gets drastically distorted. On the other hand, the cutoff for this switch in metrics is currently arbitrary, and absolute error is rarely a good metric since it is not scale-invariant.

System misidentification occurs when observation noise is very large. This happens because the β-computing system is unable to filter out all the spikes and any noise leaking through to the α-computing system will cause α to drop, and the estimator will end up reflecting the noise. Thus noise with high variance *does* affect system performance.

12 Conclusion

We have presented a very general, intuitively appealing method for time series estimation in the presence of system perturbation and observation noise. This method uses fuzzy logic control to adaptively adjust the weighting factor in the exponential averaging scheme. It uses the performance of the system in the recent past as a feedback to do this. Explicit models and variances for the two types of noise are not required. Transient behavior is adapted to automatically. The control knowledge is simply heuristic arguments, interpreted as fuzzy if-then rules. The method is shown to work well for both synthetic input data as well as a real-world application.

In addition to the problems outlined in the previous section, it would be interesting to apply this method to other applications. This would enable one to test the application-dependence of the two user-controlled parameters.

References

[1] B.D.O. Anderson, J.B. Moore, *Optimal Filtering*, Prentice Hall, 1979

[2] H. Berenji, *A Reinforcement Learning-Based Architecture for Fuzzy Control*, International Journal of Approximate Reasoning, 1992, *to appear*

[3] G.C. Goodwin, K.S. Sin, *Adaptive Filtering, Prediction and Control*, Prentice Hall, 1984

[4] S. Keshav, *Congestion Control in Computer Networks*, PhD thesis, University of California, Berkeley, August 1991

[5] C.C. Lee, *Fuzzy Logic in Control Systems: Fuzzy logic Controllers*, IEEE Trans. on Systems, Man and Cybernetics, Vol. 20, No. 2, 1990