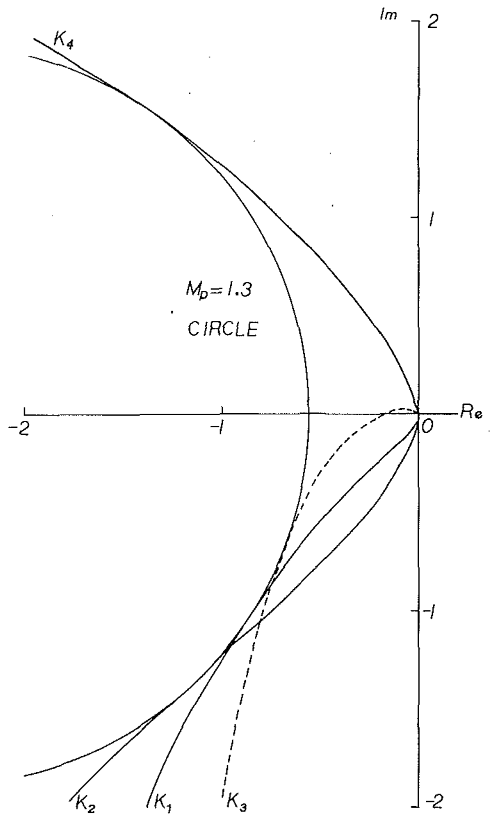

Fig. 1 Nyquist plots of the four solutions

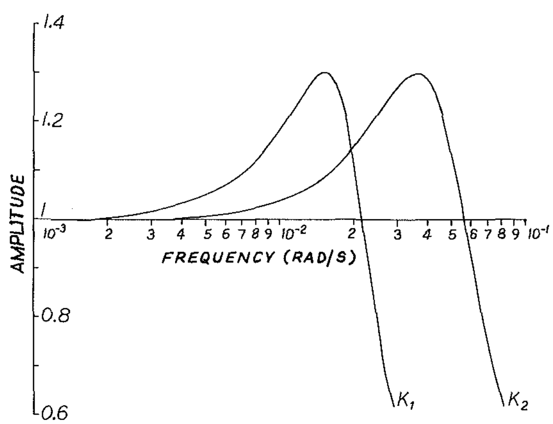

Fig. 2 Closed loop frequency response with K_1 = 8.457(10^{-2}) and K_2 = 0.46578

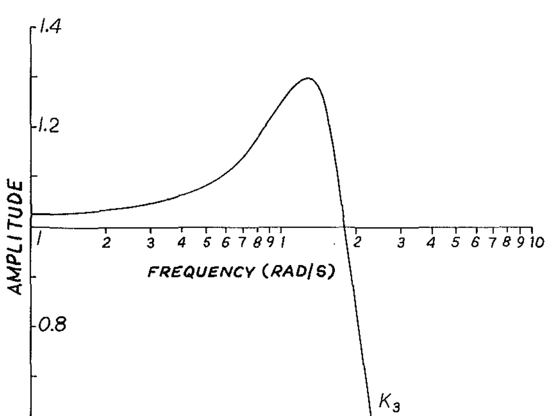

Fig. 3 Closed-loop frequency response with K_3 = 17.453

Table 1 The four possible solutions of the example problem

Solution number, i	Gain, K_i	Resonant frequency, ω_i
1	8.4575(10^{-2})	1.5(10^{-2})
2	0.46578	3.57(10^{-2})
3	17.4533	1.257
4	2543.3	10.17

$$V(\omega) = -506.89\omega^{16} - 10186.65\omega^{14} - 1283.85\omega^{12} - 2223.24\omega^{10} - 11.265\omega^{8} - 0.33678\omega^{6} \quad (19)$$

The degree of the polynomial in (16) is 18, but since this is an even polynomial and one ω^2 factor may be cancelled, it is only necessary to find the positive roots of a polynomial of degree 8 to determine the resonant frequencies. These frequencies and the corresponding gains calculated from equation (13) are displayed in Table 1.

The Nyquist plots of the four solutions and the M_p-circle are shown in Fig. 1. The fourth solution is unstable. The closed loop frequency response for the three plausible solutions are shown in Figs. 2 and 3. The largest velocity error constant and the fastest response are provided by the third solution K_3 = 17.45.

Summary

A computational scheme for setting open-loop gain to obtain a specified peak closed loop frequency response amplitude has been developed. The algorithm is easily programmed if a good polynomial solver is available. This automated alternative to the graphical technique set forth by Brown and Campbell eliminates trial and error, increases numerical accuracy, and produces all possible gains and resonant frequencies when multiple solutions exist.

References

1 Brown, G. S., and Campbell, D. P., *Principles of Servomechanisms*, Wiley, New York, 1948.

2 Chestnut, H., and Mayer, R. W., *Servomechanisms and Regulating System Design*, Vol. 1, Wiley, New York, 1966.

3 Raven, F. H., *Automatic Control Engineering*, McGraw-Hill, New York, 1978.

Trellis Representation and State Estimation for Dynamic Systems With a *K*th Order Memory and Nonlinear Interference[1]

Kerim Demirbaş[2]

A fast state estimation scheme is presented for dynamic systems with a Kth order memory and nonlinear interference. This new scheme is based upon a trellis diagram representation of dynamic models and stack sequential algorithm of Information Theory.

1 Introduction

Recursive state estimation of dynamic models with a first order memory has been extensively treated in the literature. As a result, many estimation schemes have been developed [1]–[5] and applied for practical systems [6]–[7]. The state estimation

[1] This work was carried out at Bilkent University, Ankara, Turkey.

[2] Department of Electrical Engineering and Computer Science, The University of Illinois at Chicago, Chicago, Ill. 60680.

Contributed by the Dynamic Systems and Control Division of The American Society of Mechanical Engineers. Manuscript received by the Dynamic Systems and Control Division November 15, 1988; revised manuscript received June 1989. Associate Editor: S. Jayasuriya.

of dynamic models with a higher order memory may be accomplished by first representing these dynamic models by higher dimensional dynamic models with a first order memory, and then using an estimation scheme cited above. But, the implementation of the state estimation of these higher dimensional models may become complex. In this paper, a new fast state estimation scheme is proposed for dynamic models with a memory of order K and nonlinear interference. This estimation scheme does not require these dynamic models to be represented by any higher dimensional dynamic models with a first order memory. This results in memory reduction for the implementation of the state estimation for dynamic models with a memory of order K and nonlinear interference.

2 Problem Statement

This paper deals with the state estimation of discrete dynamic models with nonlinear interference and a memory of order K. These models are defined by

$$x(k+1)=f(k,x(k),X(k),w(k)) \quad \text{the state model} \tag{1}$$

$$z(k)=g(k,x(k),X(k),I(k),v(k)) \quad \text{the observation model} \tag{2}$$

where k indicates the discrete time; $v(k)$ and $w(k)$ are an observation noise vector and a disturbance noise vector at time k with zero means and known statistics; $x(0)$ and $I(k)$ are a random initial state vector and an interference vector at time k with known statistics; $x(k)$ and $z(k)$ are a state vector and an observation vector at time k, respectively; $X(k)$ is the set of $K-1$ previous discrete values of the state $x(k)$, namely, $X(k)\triangleq\{x(k-l)\text{: } l=1, 2, 3, \ldots, K-1\}$; $f(k,x(k),X(k),w(k))$ and $g(k,x(k),X(k),I(k),v(k))$ are given (linear or nonlinear) functions; and the initial state and all samples of the disturbance noise, observation noise, and interference are independent. The next section presents a state estimation scheme which yields an estimate of the state sequence $X^L\triangleq\{x(k)\text{: } k=0, 1, \ldots, L\}$ by using the observation sequence $Z^L\triangleq\{z(k)\text{: } k=1, 2, \ldots, L\}$.

3 Estimation Scheme

First, the models of (1) and (2) are approximated by a finite state model (or machine) and an approximate observation model, respectively. This finite state model is represented by a trellis diagram, and metrics are assigned to the nodes, branches, and paths of this trellis diagram. Then, using a stack sequential algorithm [8], an estimate of the state sequence X^L is obtained. The finite state model which approximates the state model of (1) is defined by

$$x_q(k+1)=Q(f(k,x_q(k),\hat{X}(k\,|k),w_d(k))) \tag{3}$$

where $Q(.)$ is the quantizer defined in [2], which divides the n-dimensional Euclidean space into nonoverlapping generalized rectangles of equal size (called the gate size) and which then assign each rectangle (called the gate) to its center, where n is the dimension of the state vector $x(k)$; $w_d(k)$ is a discrete disturbance noise vector which approximates the disturbance noise vector $w(k)$ [2]; $x_q(0)$ is a discrete random initial state vector which approximates the initial state vector $x(0)$ and the possible values of $x_q(0)$ are said to be the initial quantization levels (or the quantization levels of the state at time zero); $x_q(k)$, $k>0$, is the quantized state at time k; $\hat{X}(k\,|k)$ is the estimate of $X(k)$, given the observation sequence Z^k, namely $\hat{X}(k\,|k)\triangleq\{\tilde{x}(k-l\,|k)\text{: } l=1, 2, 3, \ldots, K-1\}$, where $\tilde{x}(m\,|k)$ is defined by

$$\tilde{x}(m\,|k)\triangleq\begin{cases} E\{x(0)\} & \text{if } m=-1,\ -2,\ -3,\ \ldots, \\ & \text{or } (m=0 \text{ and } k=0) \\ \hat{x}(m\,|k) & \text{otherwise} \end{cases}$$

in which $\hat{x}(m\,|k)$ is the estimate of $x(m)$ given the observation sequence Z^k, and $E\{.\}$ stands for the expectation. The approximate observation model is defined by

$$z(k)=g(k,x_q(k),\hat{X}(k\,|k),I_d(k),v(k)) \tag{4}$$

where $I_d(k)$ is a discrete random interference vector which approximates the interference vector $I(k)$.

The finite state machine of (3) is represented by a diagram, called the trellis diagram of the state [2]-[5]. The following metrics are assigned to each node, branch, and path of the trellis diagram. Consider two nodes (or quantization levels) $x_{qm}(k-1)$ and $x_{qn}(k)$, where the second subscript denotes the label of the quantization level, that is $x_{qn}(k)$ is the nth quantization level of the state at time k. The metric of $x_{qn}(k)$ is defined as the natural logarithm of the occurrence probability of $x_{qn}(k)$ if $k=0$, and zero otherwise. The transition probability from the node $x_{qm}(k-1)$ to the node $x_{qn}(k)$, denoted by $\pi(x_{qm}(k-1)\rightarrow x_{qn}(k))$, is defined as the probability that the state at time k takes the quantization level $x_{qn}(k)$ when the state at time $k-1$ took the quantization level $x_{qm}(k-1)$, namely,

$$\begin{aligned}\pi(x_{qm}(k-1)\rightarrow x_{qn}(k))&\triangleq Prob\{x_q(k)=x_{qn}(k)\,|x_q(k-1)\\ &=x_{qm}(k-1)\}\\ &=\sum_r Prob\{w_d(k-1)=w_{dr}(k-1)\}\end{aligned}$$

where the summation is taken over all r such that

$$Q(g(k-1,x_{qm}(k-1),\hat{X}(k-1\,|k-1),w_{dr}(k-1)))=x_{qn}(k).$$

The metric of the branch connecting the node $x_{qm}(k-1)$ to the node $x_{qn}(k)$, denoted by $M(x_{qm}(k-1)\rightarrow x_{qn}(k))$, is defined by

$$\begin{aligned}M(x_{qm}(k-1)\rightarrow x_{qn}(k))&\triangleq ln\{\pi(x_{qm}(k-1)\rightarrow x_{qn}(k))\}\\ &+ln\{p(z(k)\,|x_{qn}(k),\hat{X}(k\,|k))\}\end{aligned}$$

where $p(z(k)\,|x_{qn}(k),\hat{X}(k\,|k))$ is the conditional probability density function of the observation at time k, given that $x_q(k)=x_{qn}(k)$ and $X(k)=\hat{X}(k\,|k)$. This density function is expressed in terms of possible values of the discrete interference vector $I_d(k)$ as

$$\begin{aligned}p(z(k)\,|x_{qn}(k),\hat{X}(k\,|k))&=\sum_{i=1}^{s_k}p(z(k)\,|x_q(k)=x_{qn}(k),X(k)\\ &=\hat{X}(k\,|k),I_d(k)=I_{di}(k))Prob\{I_d(k)=I_{di}(k)\},\end{aligned}$$

where s_k is the number of possible values of $I_d(k)$; $I_{di}(k)$ is the ith possible value of $I_d(k)$; and $p(z(k)\,|x_q(k)=x_{qn}(k),X(k)=\hat{X}(k\,|k),I_d(k)=I_{di}(k))$ is the conditional probability density function of the observation at time k, given that $x_q(k)=x_{qn}(k),X(k)=\hat{X}(k\,|k)$, and $I_d(k)=I_{di}(k)$. The metric of a path in the trellis diagram is defined as the sum of the metrics of all the nodes and branches along the path.

The trellis diagram of the state shows possible paths along which the quantization levels can be taken by the state with time. Therefore, the state estimation problem is to find a path through the trellis diagram from time zero to time L so that the quantization levels along this path become an estimate of the state sequence X^L, given the observation sequence Z^L. Finding a path, among many, is a multiple composite hypothesis testing problem. It can be shown [2] that the optimum rule which minimizes the overall error probability is to choose the path with the greatest metric (and if there exist more than one path with the same greatest metric, to choose any one of these at random). The choice of the path with the greatest metric could be accomplished by the Viterbi algorithm (VA) [2]-[3]. But, the implementation of the VA requires an

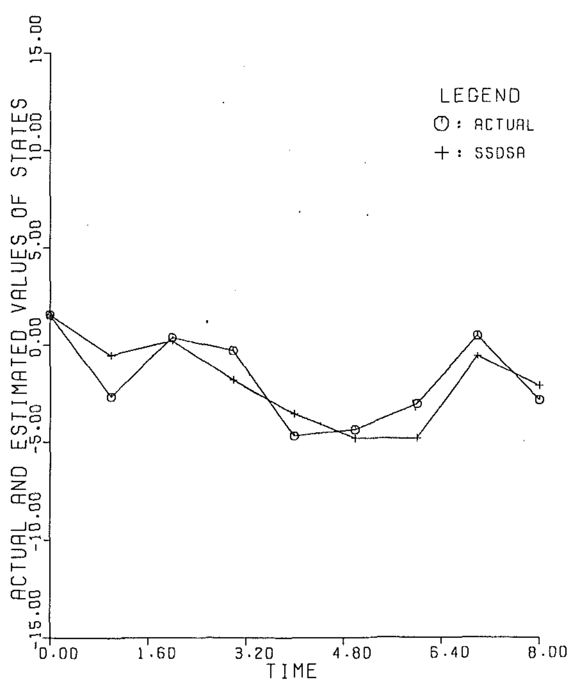

Fig. 1 Actual and estimated values of states

exponentially increasing memory with time. In order to overcome this obstacle, in this paper, a stack sequential algorithm is used to estimate the states. Stack sequential algorithms [8] guess (or estimate, but not find) the path with the greatest metric by searching only the paths which most likely contain the path with the greatest metric. The implementation of a stack sequential algorithm requires a memory increasing less than exponentially with time. Hence, a stack sequential algorithm is faster and more practical than the VA.

4 Simulations

Many examples with white Gaussian noise and interference were simulated. The stack sequential algorithm given in [8] was used to guess the path with the greatest metric. The random variables were approximated by the discrete random variables given in [2]. Fig. 1 presents the actual values (denoted by ACTUAL) and estimated values (denoted by SSDSA) of the states of a nonlinear example which contains a multiplicative interference and which has a 4th order memory. This example is given by

$$x(k+1) = 0.8x(k)cos\{0.5x(k-1)x(k-2)x(k-3)\} + w(k)$$

the state model

$$z(k) = [\ + I^2(k)]x^2(k) + 0.01[x(k-1)x(k-2)x(k-3)] + v(k)$$

the observation model

where the mean values of $x(0)$ and $I(k)$ are 1.6 and 0.4; and the variances of $x(0)$, $v(k)$, $I(k)$, and $w(k)$ are 1.1, 2, 0.3, and 4, respectively. In simulation of this example, a gate size of 0.250 was used; and $x(0)$, $w(k)$, and $I(k)$ were approximated by the discrete random variables with three possible values given in [2].

The proposed estimation scheme is faster and more practical than the estimation schemes based upon the Viterbi algorithm, even though the estimates obtained by the proposed scheme is inferior to the estimates by the estimation schemes based upon the Viterbi algorithm since a stack sequential algorithm might, once in a while, pick up a path which does not have the greatest metric and this might cause a state estimate divergence. But, the implementation of the proposed scheme requires a memory increasing less than exponentially, whereas the implementation of the estimation schemes based upon the Viterbi algorithm requires an exponentially increasing memory with time.

5 Conclusions

A fast estimation scheme is presented for dynamic models with nonlinear interference and a memory of order K. The proposed scheme can be used for any practical problems, such as target tracking under jamming or economics, where the future state is a nonlinear function of the previous values of the state, disturbance noise, observation noise, and interference.

References

1 Medich, J. S., "A Survey of Data Smoothing for Linear and Nonlinear Dynamic Systems," *Automatica*, Pergamon Press, 1973, Vol. 9, pp. 151–162.

2 Demirbaş, K., "New Smoothing Algorithms for Dynamic Systems with or without Interference," *The NATO AGARDograph Advances in the Techniques and Technology of Applications of Nonlinear Filters and Kalman Filters*, No. 256, AGARD, Mar. 1982, pp. 19–1/66.

3 Demirbaş, K., and Leondes, C. T., "Optimum Decoding based Smoothing Algorithm for Dynamic Systems with Interference," *The International Journal of Systems Science*, Vol. 17, No. 2, Feb. 1986, pp. 251–267.

4 Demirbaş, K., and Leondes, C. T., "A Stack Sequential Decoding based Smoothing Algorithm for Dynamic Systems with Interference," *The International Journal of Systems Science*. Vol. 17, No. 3, Mar. 1986, pp. 479–497.

5 Demirbaş, K., "Multidimensional State Estimation with Multiple Composite Hypothesis Testing in the Presence of Interference," ASME Journal of Dynamic Systems, Measurement and Control, Vol. 110, Sept. 1988, pp. 297–302.

6 Hutchinson, C. E., "The Kalman Filter Applied to Aerospace and Electronic Systems," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-20, No. 4, 1984, pp. 500–504.

7 Demirbaş, K., "Maneuvering Target Tracking with Hypothesis Testing," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-23, No. 6, Nov. 1987, pp. 757–766.

8 Viterbi, A. J., and Omura, J. K., *Principles of Communication and Coding*, McGraw-Hill, New York, 1979.

Design of PPD Controllers for Position Servos

C. W. de Silva[1]

In this paper the classical time domain design problem of position servoactuators having proportional plus derivative (PPD) error controllers is reconsidered. Control system stability is represented by percentage overshoot and the speed of response by peak time. The associated design equations are strongly coupled and nonlinear. Design curves are presented to facilitate the realization of fast yet accurate designs. A design algorithm that generates exact values for the controller parameters is given. A numerical example is included to illustrate the design procedure.

Introduction

Actuators with proportional plus derivative (PPD) error control are commonly used as position servos [1]. The two parameters in a PPD control element are the control gain K and the derivative time constant T. Values for these two parameters can be chosen to provide specified levels of stability and speed of response in the control system. In classical time domain design, stability is represented by percentage overshoot

[1]Professor and NSERC Chair, Department of Mechanical Engineering, The University of British Columbia, Vancouver, B.C., Canada V6T 1W5.

Contributed by the Dynamic Systems and Control Division of The American Society of Mechanical Engineers. Manuscript received by the Dynamic Systems and Control Division January 20, 1986; revised manuscript received April 14, 1988.