COLORADO STATE
UNIVERSITY
FORT COLLINS, COLORADO
80521

department of electrical engineering

UCRL-13586

5 of 16

FINAL REPORT

by

Thomas A. Brubaker
August 15, 1973

LAWRENCE LIVERMORE LABORATORY

APPLICATION OF LINEAR TRANSFORMS AND
ESTIMATION THEORY TO THE ANALYSIS OF CHEMICAL DATA

Subcontract 8032409

between

THE REGENTS OF THE UNIVERSITY OF CALIFORNIA

and

COLORADO STATE UNIVERSITY

RECEIVED BY TIC DEC 12 1973

MASTER

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

NOTICE

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Atomic Energy Commission, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

INTRODUCTION

This report describes the results from work performed during the contract period of July 1, 1972 through June 30, 1973. Specific tasks undertaken were

1. Continued investigation of linear transforms for use with chemistry data.
2. Investigation of Kalman filtering for the analysis of chemistry data.
3. Simulation methods for evaluation of chemistry systems.
4. Teaching of three 1-week short courses on signal theory to chemistry department personnel. A fourth course was cancelled due to the cold in December.

A considerable number of the results have been written as separate reports and are included in this report as appendices.

SHORT COURSES

Three short courses were presented to chemistry and engineering personnel during the weeks of September 18, 1972, March 19, 1973 and June 25, 1973. The first week was spent giving a review of digital filters. Emphasis was given to time domain design since this fits chemistry data processing.

The second week was spent on state space descriptions for simple control systems. The purpose of state descriptions was outlined and considerable effort was spent showing how conventional control loops are described using state variables. Modeling of processes was also discussed.

During the third week an actual chemistry control system was described by Jackson Harrar. The important aspect of this system was the modeling of the chemistry process that consisted of an electrochemistry plant. After the system was discussed, the reasoning behind the use of compensation in a control loop was presented along with several examples.

Along with the short courses time was spent consulting on a number of projects in general chemistry. This activity was concerned with analysis of data and methods for estimating data parameters. With much of the work, modeling of a system is the first requirement and this aspect was emphasized on projects such as the regulatory work.

LINEAR TRANSFORMS

The linear transform activity was concerned with the development of procedures for designing digital filters and the use of Walsh functions for data representation. Important considerations in the use of digital filters are the errors introduced due to quantization of the input data and filter coefficients and the quantization of products. The evaluation of error due to product and input data quantization is done by assuming these errors are independent random variables with mean zero and variance $q^2/12$. The term q is the value of the least significant bit of the computer word. The autocovariance function of the error at the output of any discrete system is found using methods developed in Appendix 1. Here a closed form procedure for evaluating the autocovariance function is presented along with several examples.

Coefficient quantization in discrete linear transforms is under control of the programmer. However, it is important that the programmer approach the problem in the proper manner. Two strategies for coefficient quantization for digital filters are given in Appendices 2 and 3. In Appendix 2 only the filter magnitude function is considered and regions where quantization can be done to meet magnitude specifications are developed. In Appendix 3 both the magnitude and phase are considered. This is applicable to digital control work.

The work on Walsh functions was approached by assuming that the Walsh transform is taken from the sample points representing a finite length record. The Walsh coefficients then are used to represent the signal. Reconstruction of the signal is accomplished by taking the inverse Walsh transform. The results were then compared using sinc functions, the zero-order-hold and a Karhunen-Loeve expansion on the set of sample points. The initial signal that was considered consisted of a band-limited random process with a bandwidth of one Hertz. Finally, the evaluation was done using a Gaussian time signal.

Zero-Order Hold Reconstruction

Given a sampled signal of duration $(N-1)T$, where T is the sampling interval, data reconstruction using a ZOH has the form

$$y_1(t) = \sum_{k=0}^N x(kT) \left\{ \mu(t-kT) - \mu[t-(k+1)T] \right\}. \quad (1)$$

In (1) $x(kT)$ are samples from the band-limited random process. The actual signal record for $0 \leq t \leq 1$ is shown in Fig. 1. The reconstructed

signal using the ZOH is shown in Figs. 2 and 3 for four and sixteen sample points.

Since the input is a random variable, the mean square error for the ZOH reconstruction is defined as

$$\text{MSE} = E \left\{ [x(t) - y_1(t)]^2 \right\} . \quad (2)$$

Using the autocovariance of the input $x(t)$ as

$$R_x(\tau) = \int_{-1}^1 e^{j2\pi f\tau} df = \frac{2 \sin 2\pi \tau}{2\pi \tau} \quad (3)$$

the mean square error for ZOH reconstruction is given as

$$\text{MSE} = 2\sigma_x^2 - 2R_x(t-kT) \quad (4)$$

where σ_x^2 is the variance of $x(t)$. Plots of the MSE are shown in Figs. 4 and 5 for four and sixteen samples.

Sinc Reconstruction

The classical sampling theorem is based on the use of sinc functions for ideal construction of a band-limited signal over all time. For the time limited signal the summation is finite and the reconstructed signal is given by

$$\begin{aligned} y_2(t) &= \sum_{k=0}^{N-1} x(kT) \frac{\sin \left[\frac{\omega_s}{2} (t-kT) \right]}{\frac{\omega_s}{2} (t-kT)} \\ &= \sum_{k=0}^{N-1} x(kT) \operatorname{sinc} \left[\frac{\omega_s}{2} (t-kT) \right] \end{aligned} \quad (5)$$

The constructed random process for this form of reconstruction is shown in Figs. 6 and 7 for four and sixteen sample points. The mean square error is given by the expression

$$\begin{aligned} \text{MSE} &= R_x(0) - 2 \sum_{k=0}^{N-1} R_x(t-kT) \frac{\sin \frac{\omega_s}{2} (t-kT)}{\frac{\omega_s}{2} (t-kT)} \\ &= \sum_{k=0}^{N-1} \sum_{j=0}^{N-1} R_x[(k-j)T] \frac{\sin \left[\frac{\omega_s}{2} (t-kT) \right] \sin \left[\frac{\omega_s}{2} (t-jT) \right]}{\frac{\omega_s^2}{4} [t-kT] [t-jT]} \end{aligned} \quad (6)$$

Plots of the MSE are shown in Figs. 8 and 9 for four and sixteen points.

Walsh Function Reconstruction

The Walsh functions form an orthogonal complete set over any given time interval. The first sixteen Walsh functions are shown in Fig. 10. Referring to Fig. 10, to avoid discontinuity problems, the sample points for the Walsh transform are chosen exactly between the conventional sample points. Thus we have a sample at $T/2$, $3T/2$, etc. The data is represented using a Walsh transform which transforms the input set into sequency space. This implies that the Walsh coefficients used to represent the signal are given from the transform

$$A_k = \frac{1}{N} \sum_{j=0}^{N-1} x[(j+1/2)T] \text{Wal}[k, (j+1/2)T] \quad (7)$$

The reconstructed signal is given by

$$y_3(t) = \sum_{k=0}^{N-1} A_k \text{Wal}[k, (j+1/2)T] \quad (8)$$

Reconstruction signals for the Walsh function are shown in Figs. 11 and 12 for four and sixteen data points. The MSE is shown in Figs. 13 and 14.

Karhunen-Loeve Reconstruction

The Karhunen-Loeve expansion, described in many books on communication theory, was used to provide a lower bound to the error. Mathematically we have

$$y_4(t) = \sum_{i=0}^{N-1} x_i \mathcal{Q}_i(t) \quad (9)$$

where

$$x_i = \int_0^L x(t) \mathcal{Q}_i(t) dt \quad (10)$$

In (9) and (10) the eigenfunctions $\mathcal{Q}_i(t)$ are chosen to fit the class of signals being reconstructed. It can be shown that the Karhunen-Loeve expansion using prolate spheroidal wave functions provide a lower bound for band-limited signals. Actual reconstruction using the Karhunen-Loeve representation was not performed, however, the integral square error was evaluated for use in the next section.

Integral Mean Square Error Analysis

To evaluate the effect of the error over the entire interval, the integral mean square error has been evaluated for the four reconstructions. This is shown in Fig. 15. Here the Karhunen-Loeve reconstruction is obviously better, however, the computer time requirements are excessive. The sinc reconstruction is best for large sampling intervals. The Walsh and ZOH reconstructions are both better than the sinc reconstruction as

N becomes large. The Walsh representation is better than the ZOH by a factor of four over the entire range of sample points.

Reconstruction of a Gaussian Time Signal

To demonstrate the reconstruction methods for a signal that is more common in chemistry instrumentation the reconstruction of the Gaussian signal, shown in Fig. 16, from sample points was investigated. The error using the sinc reconstruction from (4) is shown in Figs. 17 and 18 for eight and thirty-two sample points. The errors using a zero-order-hold and Walsh function are shown in Figs. 19, 20, 21 and 22 for the same number of sample points.

The integral square error for the three reconstruction forms is shown in Fig. 23. A table of values is given in Table 1. The integral square errors for both the zero-order-hold and the Walsh transforms look exactly the same for the Gaussian time signal as they did for the band-limited random process. The reason for this is that these two reconstruction procedures are based on individual sample points. The sinc reconstruction on the other hand differs when the two signals are considered. This is caused by the dependence of sinc reconstruction on all sample points over the record. Note again that the Walsh function reconstruction is better than the zero-order-hold by about a factor of four. Since the reconstruction using Walsh functions requires sampling transformation and inverse transformation to recover a signal, the improved efficiency in reconstruction is questionable except when the user needs the maximum accuracy. The zero-order-hold on the other hand only utilizes sample points and is computationally very efficient.

KALMAN FILTERING

The initial defining equations for a Kalman filter are given by an expression for the signal process

$$x_{k+1} = Fx_k + G u_k$$

and an expression for the measurement process

$$y_k = Hx_k + R v_k .$$

The terms F , G , H and R are scalar constants and the u_k and v_k are samples from independent uncorrelated noise sequences. The resulting Kalman filter is given by

$$\hat{x}_{k+1} = F\hat{x}_k + K_{k+1} \left\{ y_{k+1} - HF\hat{x}_k \right\}$$

where K_{k+1} is the Kalman gain.

The use of the Kalman filter requires the designer to know the signal generating process as well as the variance of the noise sequences. At the present time these functions are not known in most chemistry instrumentation systems. As a result, the use of the Kalman filter is of little value. For these applications time invariant discrete filters still provide the best means of filtering in most chemistry applications.

In practice the modeling of chemical processes will be done by chemists over the next several years. As this progresses, use of Kalman-type filters will provide the chemist with a tool to perform optimal signal analysis. However, the implementation of these filters on digital computers will require careful evaluation of errors caused by finite

length arithmetic. Because this problem is engineering in nature preliminary results on implementation are presented in this report.

For the scalar case, the expression for the Kalman filter is a first order digital filter with a time-varying gain. Thus, the implementation is subject to exactly the same error characteristics due to quantization as are conventional digital filters. For this work the coefficient quantization errors will be assumed to be known so only input quantization errors and product quantization errors need be considered. A block diagram for the Kalman filter showing sources of error is given in Fig. 18. Since the filter is defined in discrete form, there are no input quantization errors and only product quantization errors are generated. This is a valid representation for any discrete filtering operation. Referring to Fig. 24, the error equation for the system is given by

$$e[(k+1)T] = F \left[1 - H K[(k+1)T] \right] e[kT] + K[(k+1)T] \left[\delta_1[(k+1)T] + \delta_2[(k+1)T] \right] + \delta_1[(k+1)T] + \epsilon_0[(k+1)T]$$

This is a time varying equation and the variance of the response will depend on the value of $K[(k+1)T]$ at each value of k . For many Kalman filters the time-varying gain will approach a constant for large time and the variance of error will converge to a steady state value. This can be easily found using the results from Appendix 1.

Higher order Kalman filters will exhibit the same type of characteristics and the effect of product errors can be evaluated in the same way. From these evaluations a word-length requirement can be determined using

methods described in Appendix 4. Note that this form of analysis is essential in the implementation of any filter on a digital computer.

The error analysis in this report provides the designer with the necessary tools for analyzing the implementation of any digital signal processing algorithm. However, the reader should be aware that considerable work is still being done in this area. It is anticipated that new implementation procedures will be developed in the next few years that will be useful in chemistry data processing. For this reason the progress of algorithm implementation should be followed within general chemistry.

SIMULATION

With chemistry systems becoming more and more complex it is important to consider simulation in the investigation of many chemistry problems. One important aspect of simulation is in the control of processes. Currently, digital control is handled in most applications by utilizing large machines to implement the control algorithms. Little attention is given to the word length requirements needed to implement a given control algorithm. Furthermore, sampling errors and scaling considerations often are completely neglected.

Analytical procedures can be used to determine the effect of errors and to provide some insight into the scaling problems. However, for systems of any complexity these analytical procedures often are not satisfactory. To obtain realistic results a combination of analysis and simulation is required. These ideas are demonstrated in the report contained in Appendix 4. Here the variance of error due to input data quantization

and product rounding in the control algorithm is evaluated at the system output. Then word length requirements to satisfy the error specifications are obtained. The scaling to avoid overflow is handled using simulation and the response of the control algorithm input and output are simulated for two design examples to illustrate that no overflow can occur. For the second example a gain factor was taken out of the control algorithm to avoid initial overflow.

It is important that chemistry personnel start to utilize such simulation procedures in their system designs. Although the chemical processes often are not well understood, the use of interactive graphics allows the chemists to quickly evaluate system performance using different models and thus the system can be made to meet specifications in a short time.

$\text{LOG}_2 N$	SINC	ZOH	WALSH
2	6.89×10^{-3}	1.76×10^{-1}	5.39×10^{-2}
3	1.14×10^{-6}	4.32×10^{-2}	1.20×10^{-2}
4	7.08×10^{-12}	1.13×10^{-2}	3.06×10^{-3}
5	2.94×10^{-11}	2.78×10^{-3}	8.05×10^{-4}
6	3.65×10^{-11}	6.75×10^{-4}	2.21×10^{-4}
7	2.76×10^{-11}	1.55×10^{-4}	4.14×10^{-5}
8	1.68×10^{-11}	3.94×10^{-5}	8.11×10^{-6}

Table 1

INTEGRAL SQUARE ERROR FOR VALUES
OF SAMPLE POINTS WITH A GAUSSIAN TIME INPUT SIGNAL

Fig. 1 Finite Length Record of a Zero Mean Stochastic Process

Fig. 2 ZOH Reconstruction of the Signal Using 4 Sample Points

Fig. 3 ZOH Reconstruction of the Signal Using 16 Sample Points

Fig. 4 Mean Square Error for the ZOH Reconstruction
Using 4 Sample Points

Fig. 5 Mean Square Error for the ZOH Reconstruction Using 16 Sample Points

Fig. 6 Sinc Reconstruction of the Signal Using 4 Sample Points

Fig. 7 Sinc Reconstruction of the Signal Using 16 Sample Points

Fig. 8 Mean Square Error for the Sinc Reconstruction
Using 4 Sample Points

Fig. 9 Mean Square Error for the Sinc Reconstruction
Using 32 Sample Points

Fig. 10 A Plot of the First 16 Walsh Functions

Fig. 11 Walsh Function Reconstruction of the Signal Using 4 Sample Points

Fig. 12 Walsh Function Reconstruction of the Signal Using 16 Sample Points

Fig. 13 Mean Square Error for the Walsh Function Reconstruction
Using 4 Sample Points

Fig. 14 Mean Square Error for the Walsh Function Reconstruction Using 16 Sample Points

Fig. 15 Integral Mean Square Error vs. Number of Sample Points

Fig. 16 Gaussian Time Signal

Fig. 17 Error in Sinc Reconstruction
For Eight Sample Points

Fig. 18 Error in Sinc Reconstruction
For Thirty-Two Sample Points

Fig. 19 Error Using ZOH Reconstruction
With Eight Sample Points

Fig. 20 Error in ZOH Reconstruction
For Thirty-Two Sample Points

Fig. 21 Error Using Walsh Function
Reconstruction With Eight
Sample Points

Fig. 22 Error Using Walsh Function
Reconstruction With Thirty-
Two Sample Points

Fig. 23 Integral Square Error Curves

Fig. 24 Scalar Kalman Filter Showing
Multiply Errors

COMPUTATION OF THE DISCRETE AUTOCOVARIANCE

by

Thomas A. Brubaker
Department of Electrical Engineering
Colorado State University
Fort Collins, Colorado 80521

and

John N. Gowdy
Department of Electrical and Computer
Engineering
Clemson University
Clemson, South Carolina 29631

May 1, 1973

ABSTRACT

Given an N th order discrete system the autocovariance sequence is formulated as a set of recursive and nonrecursive difference equations. Application of the final value theorem for a z transform then permits the first N terms of the steady state autocovariance sequence to be computed via one matrix inversion. The remaining terms are calculated using a simple recursive relationship.

INTRODUCTION

In the design of discrete systems it is often necessary to compute the variance or the autocovariance due to input and/or quantization noise. In many instances these noise sources can be represented as independent zero mean sequences [1] which allows the steady state autocovariance sequence $\{R[kT]\}$ to be represented by the discrete Wiener-Khinchine relationship [1,2]

$$R[kT] = \frac{\sigma^2}{2\pi j} \oint H[z] H[z^{-1}] z^{k-1} dz \quad (1)$$

In (1) σ^2 is the variance of the noise and $H[z]$ is the discrete transfer function from the noise source to the system output. For a noise sequence starting at $t=0$, the transient autocovariance sequence $\{R_n[kT]\}$ time $t=nT$ is given by

$$R_n[kT] = \sigma^2 \sum_{i=0}^{n-k} h[(i+k)T] h[iT] \quad (2)$$

where $h[nT]$ is the inverse z transform of $H[z]$. As n becomes large, each term $R_n[kT]$ in the autocovariance sequence converges to $R[kT]$ given by (1).

From a computational point of view, (1) can be evaluated using the residue theorem from complex variables. If only the steady state variance is of interest, Jury [2] has derived an expression for $R[0T]$ that is the ratio of two determinants whose terms are simple functions of the transfer function coefficients. The use of (2) requires the evaluation of the weighting sequence and a series summation for all desired values of k .

In this paper the autocovariance sequence is formulated as a set of difference equations. If only the steady state autocovariance is of interest, the use of z transforms leads to a set of simultaneous equations that can be easily evaluated via one matrix inversion. Remaining terms in the autocovariance sequence are found using a simple nonrecursive relationship.

DIFFERENCE EQUATIONS FOR THE AUTOCOVARIANCE

First consider an N th order discrete transfer function of the form

$$H[z] = \frac{1}{1 + \sum_{j=1}^N b_j z^{-j}} \quad . \quad (3)$$

For an independent zero mean input noise sequence $\{w[nT]\}$ with variance σ_w^2 that starts at $t=0$, the difference equation describing the system is

$$y[nT] = - \sum_{j=1}^N b_j y[(n-j)T] + w[nT] \quad . \quad (4)$$

The mean value of $y[nT]$ is zero for all n and terms in the autocovariance sequence at time $t=nT$ are now defined as

$$R_n[kT] = E\{y[nT] y[(n-k)T]\} \quad k=0,1,2,\dots,n \quad . \quad (5)$$

From (4) and (5) the variance of $y[nT]$ is given by the quadratic form

$$R_n[0T] = \bar{b}^T Q_n \bar{b} + \sigma_w^2 \quad (6)$$

where Q_n is a time varying symmetric autocovariance matrix with terms

$$\begin{aligned} q_{ij} &= R_{n-\min(i,j)} [|i-j|T] & n-\min(i,j) \geq 0 \\ &= 0 & n-\min(i,j) < 0 \end{aligned} \quad . \quad (7)$$

The vector $\bar{b}^t$ is defined as the row matrix $[-b_1 -b_2 \dots -b_n]$.

Since $w[nT]$ is taken from an independent noise sequence, $y[(n-k)T]$ and $w[nT]$ are uncorrelated for $k \geq 1$ and the next N terms in the autocovariance sequence are found by multiplying both sides of (4) by $y[(n-k)T]$, $k=1,2,\dots,N$ and taking the expectation. This yields a set of equations described in vector form as

$$\bar{R}_n = Q_n \bar{b} \quad (8)$$

where $\bar{R}_n^t$ is the row matrix $[R_n[T] R_n[2T] \dots R_n[NT]]$.

The $N+1$ equations described by (6) and (8) consist of a set of $N/2$ coupled recursive difference equations for N even and $(N+1)/2$ coupled recursive equations for N odd. The remaining equations are nonrecursive difference equations. For $k > N$ the $R_n[kT]$ terms are given by

$$R_n[kT] = - \sum_{i=1}^N b_i R_{n-i}[(k-i)T] \quad (9)$$

A closed form steady state solution is derived by first defining the z transform of the autocovariance term $R_{n-i}[kT]$ as

$$Z\{R_{n-i}[kT]\} = z^{-i} R_z[kT] \quad (10)$$

where the subscript $(n-i)$ represents the discrete time $(n-i)T$.

Taking the z transform of (6) and (8) now yields the expression for the z transform of the variance

$$R_z[0T] = \bar{b}^t Q[z] \bar{b} + \frac{z}{z-1} \sigma_w^2 \quad (11)$$

and the next N terms of the autocovariance sequence

$$\bar{R}_z = Q[z] \bar{b} \quad (12)$$

In (11) the matrix $Q[z]$ has terms

$$q_{ij}[z] = z^{-\min(i,j)} R_z[|i-j|T] \quad (13)$$

where $|i-j|$ takes on values $0, 1, 2, \dots, N-1$. Expanding the quadratic form given by (11) and applying the final value theorem to this and the first $N-1$ equations given by (12) now yields a set of N simultaneous equations whose solution forms the first N terms of the steady state autocovariance sequence. In matrix form we have

$$\begin{bmatrix} R[0T] \\ R[T] \\ . \\ . \\ R[(N-1)T] \end{bmatrix} = M^{-1} \begin{bmatrix} \sigma_w^2 \\ 0 \\ . \\ . \\ 0 \end{bmatrix} \quad (14)$$

where M is the matrix

$$M = \begin{bmatrix} 1 - \sum_{i=1}^N b_i^2 & -2 \sum_{i=1}^{N-1} b_i b_{i+1} & -2 \sum_{i=1}^{N-2} b_i b_{i+2} & . & . & -2b_1 b_N \\ (b_1) & (1+b_2) & . & . & . & b_N \\ (b_2) & . & . & . & . & . \\ . & . & . & . & . & . \\ (b_N) & (b_{N-2} + b_N) & . & . & . & 1 \end{bmatrix} \quad (15)$$

The remaining terms in the autocovariance sequence are given by

$$R[kT] = - \sum_{i=1}^N b_i R[(k-i)T] \quad k \geq N. \quad (16)$$

For a system transfer function of the form

$$H[z] = \frac{\sum_{i=0}^{V-1} a_i z^{-i}}{1 + \sum_{j=0}^N b_j z^{-j}} = \frac{N[z^{-1}]}{D[z^{-1}]} \quad (17)$$

the transfer function is separated by defining an intermediate operation

$$x[z^{-1}] = \frac{W[z^{-1}]}{D[z^{-1}]} \quad (18)$$

so that the response is given by

$$y[z^{-1}] = N[z^{-1}] x[z^{-1}] \quad (19)$$

The autocovariance sequence for x_n is found using the preceeding results.

The terms in the output autocovariance sequence are given by

$$E\{y[nT] y[(n-k)T]\} = \bar{a}^t C_n \bar{a} \quad (20)$$

where $\bar{a}^t$ is the row matrix $[a_0 \ a_1 \ \dots \ a_{V-1}]$ and C_n is the appropriate V by V time varying matrix. In steady state C_n is a constant matrix with terms

$$C_{ij} = R[(k-j+i)T] = \lim_{n \rightarrow \infty} E\{x[nT] x[(n-(k-j+i))T]\} \quad (21)$$

When computing the steady state autocovariance for $K < V$ certain terms in the C matrix are found from the relationship $R[kT] = R[-kT]$. Thus,

a fixed nonrecursive expression exists only for $K \geq V$.

EXAMPLE

To illustrate the procedure, we consider a low pass fourth-order Butterworth filter with a -3db point at $\omega_0 = 10$. The filter was designed using the bilinear z transform with $T=0.01$ and has the general transfer function

$$H[z] = \frac{K[z+1]^4}{z^4 + b_1 z^3 + b_2 z^2 + b_3 z + b_4} = \frac{N[z]}{D[z]} \quad (22)$$

where

$$k = 4.69832343 \times 10^{-3},$$

$$b_1 = -2.53346973,$$

$$b_2 = 2.65559567,$$

$$b_3 = -1.28757608,$$

and

$$b_4 = 0.24062331.$$

For a direct form of implementation [1] with nine product rounding errors the difference equation for the output error is

$$e[nT] = p[nT] - \sum_{j=1}^4 b_j e[(n-j)T] \quad (23)$$

The variance of $p[nT]$ is $9q^2/12$. The variance of $e[nT]$ was computed as a function of n using (6) and (8) and is plotted in Fig. 1. This was checked using the form given by (2) for $k=0$ and $n=\infty$. The steady state autocovariance for the filter is shown in Fig. 2. Here the M^{-1} matrix

from (14) is given by

$$M^{-1} = \begin{bmatrix} 64.818991 & -213.413104 & 129.627082 & -27.676674 \\ 57.510375 & -188.201279 & 115.638784 & -24.832401 \\ 38.381329 & -122.830831 & 79.495504 & -17.239467 \\ 14.134665 & -40.901684 & 33.388940 & -6.391421 \end{bmatrix} . \quad (24)$$

The multiplication shown in (14) gives the first four terms of the sequence with the remaining terms given by (16).

For two second order sections in series given by

$$H_1[z] = \frac{(5.78776100)10^{-2} (z+1)^2}{z^2 - 1.07350061 z + 0.30805006} = \frac{N_1[z]}{D_1[z]} \quad (25)$$

and

$$H_2[z] = \frac{(7.99359506)10^{-2} (z+1)^2}{z^2 - 1.45996913 z + 0.77971293} = \frac{N_2[z]}{D_2[z]} \quad (26)$$

the implementation of each section in direct form requires the following two calculations to compute the autocovariance sequence at the output of $H_2[z]$.

First the autocovariance $R_{12}[kT]$ due to the noise in section one is found using the coefficients in the transfer function

$$H_{12}[z] = \frac{N_2[z]}{D_1[z] D_2[z]} . \quad (27)$$

Thus the autocovariance sequence shown in Fig. 2 is modified by the zeros of $H_2[z]$. From (20) this is given by

$$R_{12}[kT] = \bar{a}^t C \bar{a} . \quad (28)$$

The autocovariance $R_{22}[kT]$ due to the noise in section two using the coefficients from

$$H_{22}[z] = \frac{1}{D_2[z]} \quad . \quad (29)$$

The steady state autocovariance of the filter is now given by

$$R_y[kT] = R_{12}[kT] + R_{22}[k] \quad (30)$$

since the product rounding errors are assumed to be uncorrelated [1].

A plot of $R_y[kT]$ is shown in Fig. 3 as a function of k . This illustrates the well-known result that implementation of high order filters as a series or parallel combination of first and second order sections can reduce the effect of product rounding errors.

CONCLUSIONS

A method for computing the autocovariance at the output of a digital filter has been presented. For the steady state case, the inversion of an N by N matrix is required where N is the order of the system from the noise source to the filter output. The remaining terms are computed using simple nonrecursive expressions involving the filter coefficients. The procedure is suitable for programming on a computer or on one of the programmable scientific calculators that now exist in many design facilities.

REFERENCES

1. B. Gold and C. M. Rader, Digital Processing of Signals, McGraw-Hill, 1969.
2. E. Jury, Theory and Application of the z-Transform Method, John Wiley and Sons, 1964.
3. J. L. Melsa and A. P. Sage, An Introduction to Probability and Stochastic Processes, Prentice Hall, 1972.

FIG. 1 Variance of y_n plotted as a function of n for a fourth order Butterworth filter implemented in direct form.

FIG. 2 Autocovariance sequence for a fourth order Butterworth filter implemented in direct form

FIG. 3 Steady state autocovariance for a fourth order Butterworth filter implemented as two second order sections in cascade

A STRATEGY FOR COEFFICIENT
QUANTIZATION IN DIGITAL FILTERS

by

Thomas A. Brubaker

and

Phillip R. Risch

ABSTRACT

A strategy for quantizing digital filter coefficients in a general second order section is presented. By specifying a maximum allowable error in the magnitude function over a given frequency range and an allowable error in the resonant frequency, quantization regions are established for the filter coefficients. A detailed example is included to illustrate the procedure.

T. A. Brubaker and P. R. Risch are with the Department of Electrical Engineering, Colorado State University, Fort Collins, Colorado 80521.

INTRODUCTION

A digital filter described by the difference equation

$$y_n = \sum_{k=0}^{M-1} a_k x_{n-k} - \sum_{j=1}^N b_j y_{n-j} \quad (1)$$

transforms an input sequence $\{x_n\}$ into an output sequence $\{y_n\}$.

Digital filters can be programmed on general purpose digital computers or can be implemented using digital hardware.

The accuracy of digital filters is dependent on the finite word length used to represent the input data, the filter coefficients and the intermediate products. A recent review paper on this topic along with a number of references is by Liu [1].

Error due to deterministic coefficient quantization is described by Kaiser [2], Otnes and McNamee [3], and Mantey [4]. Gold and Rader [5,6] have investigated the effect of coefficient error on the filter pole locations and have developed filter forms that are less sensitive to error. A statistical approach to the problem is described by Knowles and Olcayto [7].

A general result from the references is that higher order digital filters should be implemented as cascade or parallel connections of first and second order filter sections. When the filter sections have been determined, the filter coefficients must then be quantized in such a way that the filter specifications are satisfied.

The implementation of a filter with current digital hardware also places constraints on the coefficient quantization. Using medium or large scale integrated circuits, it is often necessary to minimize the input and output pin connections. Computations done within the integrated circuit on the other hand can often utilize more bits. This means, for example,

that sums of products can often be truncated only after all multiply-add operations have been completed. Since the coefficients represent one input to a filter, it is desirable to establish a strategy for coefficient quantization that allows a minimum number of bits to represent each coefficient.

In this paper a strategy for quantizing the coefficients in a second order digital filter section is developed. First, the error in the magnitude function is expressed in terms of the filter coefficients. Combining this expression with a result obtained by Gold and Rader [5] for error in the resonant frequency, closed regions can be established for coefficient quantization. The procedure is illustrated via a fourth order elliptic type of filter implemented as two second order sections in cascade.

ERROR ANALYSIS

Filter Magnitude Function

Given the general second order digital filter with discrete transfer function

$$H(z) = \frac{a_0 z^2 + a_1 z + a_2}{z^2 + b_1 z + b_2} \quad (2)$$

the filter frequency function is given by letting $z = e^{j\omega T}$. The resulting magnitude function is written as

$$|H(\omega)| = \frac{A(\omega)}{B(\omega)} \quad (3)$$

where

$$A(\omega) = (a_0^2 + a_1^2 + a_2^2 + (2a_0a_1 + 2a_1a_2) \cos\omega T + 2a_0a_2 \cos 2\omega T)^{\frac{1}{2}} \quad (4)$$

and

$$B(\omega) = (1 + b_1^2 + b_2^2 + 2b_1(1 + b_2) \cos\omega T + 2b_2 \cos 2\omega T)^{\frac{1}{2}}. \quad (5)$$

When the coefficients are now allowed to vary (3) can be considered as a function of five variables. This leads to the differential approximation

$$\begin{aligned} \Delta |H(\omega)| \approx & \frac{\partial |H(\omega)| \Delta b_1}{\partial b_1} + \frac{\partial |H(\omega)| \Delta b_2}{\partial b_2} \\ & + \frac{\partial |H(\omega)| \Delta a_0}{\partial a_0} + \frac{\partial |H(\omega)| \Delta a_1}{\partial a_1} + \frac{\partial |H(\omega)| \Delta a_2}{\partial a_2} \end{aligned} \quad (6)$$

where each partial derivative is a function of frequency given by

$$\frac{\partial |H(\omega)|}{\partial b_1} = \frac{-A(\omega)\{b_1 + (1 + b_2) \cos\omega T\}}{[B(\omega)]^3}, \quad (7)$$

$$\frac{\partial |H(\omega)|}{\partial b_2} = \frac{-A(\omega)\{b_2 + b_1 \cos\omega T + \cos 2\omega T\}}{[B(\omega)]^3}, \quad (8)$$

$$\frac{\partial |H(\omega)|}{\partial a_0} = \frac{(a_0 + a_1 \cos\omega T + a_2 \cos 2\omega T)}{A(\omega) B(\omega)}, \quad (9)$$

$$\frac{\partial |H(\omega)|}{\partial a_1} = \frac{a_1 + (a_0 + a_2) \cos\omega T}{A(\omega) B(\omega)} \quad (10)$$

and

$$\frac{\partial |H(\omega)|}{\partial a_2} = \frac{\{a_2 + a_1 \cos\omega T + a_0 \cos 2\omega T\}}{A(\omega) B(\omega)}. \quad (11)$$

In the above expressions $A(\omega)$ and $B(\omega)$ are given by (4) and (5).

For an error $\Delta|H(\omega)|$ the expression described by (6) is a linear algebraic equation in five variables with coefficients that are frequency dependent. To simplify the quantization (6) is now broken into two error components, one due to the denominator coefficients and one due to the numerator coefficients. This gives

$$\Delta|H(\omega)|_D = \frac{\partial|H(\omega)|}{\partial b_1} \Delta b_1 + \frac{\partial|H(\omega)|}{\partial b_2} \Delta b_2 \quad (12)$$

and

$$\Delta|H(\omega)|_N = \frac{\partial|H(\omega)|}{\partial a_0} \Delta a_0 + \frac{\partial|H(\omega)|}{\partial a_1} \Delta a_1 + \frac{\partial|H(\omega)|}{\partial a_2} \Delta a_2. \quad (13)$$

For a specified error $\Delta|H(\omega)|_D$, (12) is a linear algebraic equation in the variables Δb_1 and Δb_2 . To establish the region of quantization, the frequency dependent coefficients are varied with frequency to obtain the smallest region where quantization can take place. The choice of Δb_1 and Δb_2 within this region then forces the error $\Delta|H(\omega)|_D$ to be satisfied over the frequency range of interest.

Equation (13) is a three variable equation; however, for many designs the coefficients a_0 and a_2 are equal and (13) simplifies to

$$\Delta|H(\omega)|_N = \frac{\partial|H(\omega)|}{\partial a_1} \Delta a_1 + 2 \frac{\partial|H(\omega)|}{\partial a_2} \Delta a_2. \quad (14)$$

The expression is used in the same way as (12) when a value $\Delta|H(\omega)|_N$ is specified. By utilizing the two quantization regions the specification on the total allowable error $\Delta|H(\omega)|$ can be satisfied over the frequency range of interest.

Resonant Frequency

Another parameter which is useful in determining the region of quantization is the shift in resonant frequency. For the denominator of (2) this shift is given by Rader and Gold [5] as

$$\Delta\omega_{OD} = \frac{1}{T} \frac{-\Delta b_1}{(4b_2 - b_1^2)^{1/2}} + \frac{b_1}{2b_2} \frac{\Delta b_2}{(4b_2 - b_1^2)^{1/2}} \quad (15)$$

For complex conjugate zeros on the unit circle, the shift in the frequency where the filter gain is zero is given by

$$\Delta\omega_{ON} = \frac{1}{T} \frac{\Delta a_1}{\sqrt{4a_0a_2 - a_1^2}} - \frac{a_1 \Delta a_2}{a_0 \sqrt{4a_0a_2 - a_1^2}} \quad (16)$$

For a specified change in the resonant frequency, (15) and (16) are expressions for a straight line. These expressions can be used along with those given by (12) and (14) to form quantization boundaries in the $(\Delta b_1 \Delta b_2)$ and $(\Delta a_1 \Delta a_2)$ planes, respectively. The procedure for coefficient quantization is now illustrated via an example.

EXAMPLE

The filter coefficients for a fourth order elliptic filter implemented as two second order sections in cascade are given in Table 1. The magnitude function is shown in Fig. 1. Curves of the partial derivatives of $|H(\omega)|$ with respect to the filter coefficients are shown in Figs. 2 through 5. For section one, the expressions for the resonate and zero gain frequency shifts are found from (15) and (16) to be

$$\Delta\omega_{OD1} = - (5.3438 \times 10^3) \Delta b_1 - (5.3404 \times 10^3) \Delta b_2 \quad (17)$$

and

$$\Delta\omega_{ON1} = (1.7960 \times 10^4) \Delta a_1 + (3.5803 \times 10^4) \Delta a_2 \quad (18)$$

For section two the expressions are

$$\Delta\omega_{OD2} = - (4.8398 \times 10^3) \Delta b_1 - (4.8353) \Delta b_2 \quad (19)$$

and

$$\Delta\omega_{ON2} = (1.1937 \times 10^4) \Delta a_1 + (2.3697 \times 10^4) \Delta a_2 \quad (20)$$

To illustrate the procedure, the error in the magnitude function for each section is specified to be ± 0.2 . The allowable values for $\Delta\omega_{OD1}$, $\Delta\omega_{ON1}$, $\Delta\omega_{OD2}$ and $\Delta\omega_{ON2}$ are given as ± 1 radian.

To establish the quantization regions for the first section, the magnitude error is separated into two components of ± 0.1 . This assignment of error due to numerator and denominator coefficients is arbitrary, however, the amplitude values of the partial derivatives shown in Figs. 2 through 5 indicated this is a reasonable choice. Combining (17) and (12) the smallest quantization region for the denominator coefficients is shown in Fig. 6. Note that the region extends well into the second and fourth quadrants which indicates conventional rounding or chopping may not be as effective.

For the numerator coefficients, the effect of (14) can be neglected and only (18) is required. The resulting region is shown in Fig. 7. Similar regions exist for section two and are shown in Figs. 8 and 9.

Working within the quantization regions, the filter coefficients shown in Table 2 were established. The point in each quantization region is

shown by a dot. Note that conventional rounding to four decimal digits can result in quantization errors that can lie outside of the quantization regions. As an example, if the b_1 coefficient in section one is rounded to four digits the resulting error is $\Delta b_1 = 3.865 \times 10^{-4}$. Rounding off the b_2 coefficient results in an error $\Delta b_2 = 0.351 \times 10^{-4}$. The resulting point is outside of the quantization region and a simulation run shows that the specification on the magnitude function is not satisfied.

The error in each section along with the total filter error is shown in Fig. 10. The magnitude function for the total filter is shown in Fig. 11. If the error in the filter from 90 to 100 radians is not acceptable, a new quantization could be initiated; however, the given filter is accurate to about 13 percent in the magnitude function which is better than the accuracy of section one and less than that of section two.

	<u>SECTION 1</u>	<u>SECTION 2</u>
$a_0 = a_2$	.3442564	.3446042
a_1	- .6862578	- .6840982
b_1	- 1.9836135	- 1.9803224
b_2	.9924351	.9914921

TABLE 1
 ELLIPTIC FILTER COEFFICIENTS FOR A
 SAMPLING TIME $T = .001$

	<u>SECTION 1</u>	<u>ERROR</u>	<u>SECTION 2</u>	<u>ERROR</u>
$a_0 = a_2$	.3444	$- 1.436 \times 10^{-4}$	.3448	$- 1.958 \times 10^{-4}$
a_1	- .6865	2.422×10^{-4}	- .6845	4.018×10^{-4}
b_1	- 1.983	$- 6.135 \times 10^{-4}$	- 1.981	6.776×10^{-4}
b_2	.9918	6.351×10^{-4}	.9922	$- 7.079 \times 10^{-4}$

TABLE 2
QUANTIZED FILTER COEFFICIENTS

REFERENCES

- [1] B. Liu, "Effects of finite word length on the accuracy of digital filters - a review," IEEE Trans. on Circuit Theory, Vol. CT-18, No. 6, November 1971.
- [2] J. F. Kaiser, "Some practical considerations in the realization of linear digital filters," Proc. Third Annual Allerton Conference on Circuit and System Theory, pp. 621-633, 1965.
- [3] R. K. Otnes and L. P. McNamee, "Instability thresholds in digital filters due to coefficient rounding," IEEE Trans. on Audio and Electroacoustics, Vol. AV-18, Dec. 1970.
- [4] P. E. Mantey, "Eigenvalue sensitivity and state-variable selection," IEEE Trans. on Automatic Control, Vol. AC-13, June 1968.
- [5] C. M. Rader and B. Gold, "Effects of parameter quantization on the poles of a digital filter," Proc. IEEE 55(5), pp. 688-689, May 1967.
- [6] B. Gold and C. M. Rader, Digital Processing of Signals, McGraw-Hill Book Company, New York, 1969.
- [7] J. B. Knowles and E. M. Olcayto, "Coefficient accuracy and digital filter response," IEEE Trans. on Circuit Theory, Vol. CT-15, March 1968.

FIG. 1 FILTER MAGNITUDE FUNCTION

FIG. 2 PARTIAL DERIVATIVES OF $|H(\omega)|$ WITH RESPECT TO a_0

FIG. 3 PARTIAL DERIVATIVES OF $|H(\omega)|$ WITH RESPECT TO a_1

FIG. 4 PARTIAL DERIVATIVES OF $|H(\omega)|$ WITH RESPECT TO b_1

FIG. 5 PARTIAL DERIVATIVES OF $|H(\omega)|$ WITH RESPECT TO b_2

FIG. 6 QUANTIZATION REGION FOR THE DENOMINATOR COEFFICIENTS OF SECTION 1

FIG. 7 QUANTIZATION REGION FOR THE NUMERATOR
COEFFICIENTS OF SECTION 1

FIG. 8 QUANTIZATION REGION FOR THE DENOMINATOR COEFFICIENTS OF SECTION 2

FIG. 9 QUANTIZATION REGION FOR THE NUMERATOR
COEFFICIENTS OF SECTION 2

FIG. 10 ERROR IN THE MAGNITUDE FUNCTION FOR EACH SECTION AND FOR THE CASCADE

FIG. 11 FILTER MAGNITUDE FUNCTION FOR QUANTIZED COEFFICIENTS

A STRATEGY FOR COEFFICIENT
QUANTIZATION IN DIGITAL CONTROL ALGORITHMS

by

Thomas A. Brubaker
Electrical Engineering Department
Colorado State University
Fort Collins, Colorado 80521

ABSTRACT

A strategy for quantizing the coefficients in a general second order digital control algorithm is presented. First, the errors in the magnitude and phase functions for the algorithm are derived in terms of the filter coefficients. By specifying a maximum allowable error for each function over a given frequency range quantization regions can then be established. An example consisting of a lag-lead digital control algorithm is included to illustrate the procedure.

INTRODUCTION

The use of discrete data processing algorithms as an integral part of a control system is well known. Books by Tou [1], Monroe [2] and Kuo [3] describe many of the design concepts for these systems called sampled data or digital control systems. In a somewhat independent effort, design and implementation procedures have been developed for digital filters. A collection of papers in this area has recently been edited by Rabiner and Rader [4].

The accuracy of any digital control algorithm is dependent on the finite word length used to represent the input data, the filter coefficients and the intermediate products. A recent review paper on this topic that is concerned only with open loop digital filters is by Liu [5].

Error due to deterministic coefficients in digital filters is described by Kaiser [6], Otnes and McNamee [7] and Mantey [8]. Gold and Rader [9,10] have investigated the effect of coefficient error on the filter pole locations and have developed filter forms that are less sensitive to error. A statistical approach to the problem is described by Knowles and Olcayto [11].

A general result from the references is that higher order digital filters and/or control algorithms should be implemented as a cascade or parallel connection of first and second order filter sections. When the filter sections have been determined, the filter coefficients must then be quantized in such a way that the specifications are satisfied.

The implementation of digital control algorithms with current digital hardware can also place constraints on the coefficient quantization. Using medium or large scale integrated circuits it is often necessary to minimize

the input and output pin connections. Computations done within the integrated circuit on the other hand can often utilize more bits. This means, for example, that sums of products can often be truncated only after all multiply-add operations have been completed. Since the coefficients represent a set of inputs to the control algorithm it is desirable to establish a strategy for coefficient quantization that allows a minimum number of bits to represent each coefficient.

In this paper a strategy for quantizing the coefficients in a second order control algorithm is developed. First the errors in the magnitude and phase functions are expressed in terms of the filter coefficients. By comparing the derivatives of the magnitude and phase functions with respect to each coefficient closed regions can be established for the quantization of the coefficients over a frequency range of interest. The procedure is illustrated using a second order discrete lag-lead compensator designed using the bilinear z transform.

ERROR ANALYSIS

Magnitude Function

Given a second order discrete transfer function

$$H(z) = \frac{a_0 z^2 + a_1 z + a_2}{z^2 + b_1 z + b_2} \quad (1)$$

the frequency function is given by letting $z = e^{j\omega T}$. The resulting magnitude function is written as

$$|H(\omega)| = \frac{A(\omega)}{B(\omega)} \quad (2)$$

where

$$A(\omega) = \left\{ a_0^2 + a_1^2 + a_2^2 + (2a_0a_1 + 2a_1a_2) \cos\omega T + 2a_0a_2 \cos 2\omega T \right\}^{\frac{1}{2}} \quad (3)$$

and

$$B(\omega) = \left\{ 1 + b_1^2 + b_2^2 + 2b_1(1 + b_2)\cos\omega T + 2b_2\cos 2\omega T \right\}^{\frac{1}{2}} \quad (4)$$

When the coefficients are allowed to vary (2) can be considered as a function of five variables for any value of frequency. This leads to the differential approximation

$$\begin{aligned} \Delta |H(\omega)| &= \frac{\partial |H(\omega)|}{\partial b_1} \Delta b_1 + \frac{\partial |H(\omega)|}{\partial b_2} \Delta b_2 \\ &+ \frac{\partial |H(\omega)|}{\partial a_0} \Delta a_0 + \frac{\partial |H(\omega)|}{\partial a_1} \Delta a_1 + \frac{\partial |H(\omega)|}{\partial a_2} \Delta a_2 \end{aligned} \quad (5)$$

where each partial derivative is a frequency function given by

$$\frac{\partial |H(\omega)|}{\partial b_1} = \frac{-A(\omega)\{b_1 + (1 + b_2)\cos\omega T\}}{[B(\omega)]^3}, \quad (6)$$

$$\frac{\partial |H(\omega)|}{\partial b_2} = \frac{-A(\omega)\{b_2 + b_1 \cos\omega T + \cos 2\omega T\}}{[B(\omega)]^3}, \quad (7)$$

$$\frac{\partial |H(\omega)|}{\partial a_0} = \frac{a_0 + a_1 \cos\omega T + a_2 \cos 2\omega T}{A(\omega) B(\omega)}, \quad (8)$$

$$\frac{\partial |H(\omega)|}{\partial a_1} = \frac{a_1 + (a_0 + a_2) \cos\omega T}{A(\omega) B(\omega)} \quad (9)$$

and

$$\frac{\partial |H(\omega)|}{\partial a_2} = \frac{a_2 + a_1 \cos \omega T + a_0 \cos 2\omega T}{A(\omega) B(\omega)} \quad (10)$$

In the above expressions $A(\omega)$ and $B(\omega)$ are given by (3) and (4).

Phase Function

Given the transfer function described by (1) the resulting phase function is

$$\begin{aligned} \theta(\omega) = & \tan^{-1} \left\{ \frac{a_0 \sin 2\omega T + a_1 \sin \omega T}{a_0 \cos 2\omega T + a_1 \cos \omega T + a_2} \right\} \\ & - \tan^{-1} \left\{ \frac{\sin 2\omega T + b_1 \sin \omega T}{\cos 2\omega T + b_1 \cos \omega T + b_2} \right\} \end{aligned} \quad (11)$$

For simplicity let

$$N_1(\omega) = a_0 \cos 2\omega T + a_1 \sin \omega T + a_2 \quad (12)$$

$$N_2(\omega) = a_0 \sin 2\omega T + a_1 \sin \omega T \quad (13)$$

$$D_1(\omega) = \cos 2\omega T + b_1 \cos \omega T + b_2 \quad (14)$$

and

$$D_2(\omega) = \sin 2\omega T + b_1 \sin \omega T \quad (15)$$

The differential phase approximation is now described by

$$\begin{aligned} \Delta \theta(\omega) = & \frac{\partial \theta(\omega)}{\partial b_1} \Delta b_1 + \frac{\partial \theta(\omega)}{\partial b_2} \Delta b_2 + \frac{\partial \theta(\omega)}{\partial a_0} \Delta a_0 \\ & + \frac{\partial \theta}{\partial a_1} \Delta a_1 + \frac{\partial \theta}{\partial a_2} \Delta a_2 \end{aligned} \quad (16)$$

where

$$\frac{\partial \theta(\omega)}{\partial b_1} = \frac{-\{D_1(\omega) \sin \omega T - D_2(\omega) \cos \omega T\}}{[D_1(\omega)]^2 + [D_2(\omega)]^2} \quad (17)$$

$$\frac{\partial \theta(\omega)}{\partial b_2} = \frac{D_2(\omega)}{[D_1(\omega)]^2 + [D_2(\omega)]^2}, \quad (18)$$

$$\frac{\partial \theta(\omega)}{\partial a_0} = \frac{N_1(\omega) \sin 2\omega T - N_2(\omega) \cos 2\omega T}{[N_1(\omega)]^2 + [N_2(\omega)]^2}, \quad (19)$$

$$\frac{\partial \theta(\omega)}{\partial a_1} = \frac{N_1(\omega) \sin \omega T - N_2(\omega) \sin \omega T}{[N_1(\omega)]^2 + [N_2(\omega)]^2} \quad (20)$$

and

$$\frac{\partial \theta(\omega)}{\partial a_2} = \frac{-N_2(\omega)}{[N_1(\omega)]^2 + [N_2(\omega)]^2} \quad (21)$$

Determination of Quantization Regions

The expressions described by (5) and (16) are linear algebraic equations in five variables with coefficients that are frequency dependent. In theory, given a specification on $\Delta|H(\omega)|$ and $\Delta\theta(\omega)$ the smallest five dimensional regions that allow the specifications to be satisfied can be established by substituting the proper coefficients, that are functions of frequency, into each algebraic equation. The values of Δb_1 , Δb_2 , Δa_0 , Δa_1 and Δa_2 can then be chosen to lie inside of the regions thus giving values for the quantized coefficients.

In practice, however, it is difficult to work in a five dimensional region and it is desirable to assign a portion of each error term to one

or two coefficients. This results in one and two dimensional quantization regions that can easily be graphed. For good results the designer must carefully evaluate the effect of each error in (5) and (16) to establish the quantization regions that fit the filter being implemented. For example, in a band-pass Butterworth filter designed using the bilinear z transform only the b_1 and b_2 coefficients need be considered. The final quantization gives quantized coefficients represented in binary form for programming on a computer or for implementation with digital hardware.

EXAMPLE

In many unity feedback control systems the compensation consists of a lag-lead network. For an analog lag-lead compensator described by

$$C(s) = \frac{(s + .1)(s + 1)}{(s + .01)(s + 10)} \quad (22)$$

the equivalent discrete compensator designed using the extended bilinear z transform is given by

$$c(z) = \frac{0.8356618816z^2 - 1.626383584z + 0.7909250553}{z^2 - 1.592691562z + 0.592894916} \quad (23)$$

In the design, the sampling time was chosen to be $T = 0.05$.

Graphs of the partial derivatives described by (6) through (10) and (17) through (21) are shown in Figs. 1 through 5 as functions of frequency. Values for each partial derivative were computed over a range of zero to twenty radians, however, to illustrate the important features the frequency

range for each derivative was reduced. For higher frequencies each partial derivative asymptotically approaches zero.

Referring to the graphs the following assumptions can be made. First, equations (6) and (7) are approximately the same as are (17) and (18). Secondly, equations (8), (9) and (10) are approximately the same as are (19), (20) and (21). From computer printout, the approximations are good to three decimal places. This implies that (5) and (16) can be rewritten as

$$\begin{aligned} \Delta |H(\omega)| \approx & \frac{\partial |H(\omega)|}{\partial a_0} \{\Delta a_0 + \Delta a_1 + \Delta a_2\} \\ & + \frac{\partial |H(\omega)|}{\partial b_1} \{\Delta b_1 + \Delta b_2\} \end{aligned} \quad (24)$$

and

$$\begin{aligned} \Delta \theta(\omega) \approx & \frac{\partial \theta(\omega)}{\partial a_0} \{\Delta a_0 + \Delta a_1 + \Delta a_2\} \\ & + \frac{\partial \theta(\omega)}{\partial b_1} \{\Delta b_1 + \Delta b_2\} \end{aligned} \quad (25)$$

Quantization regions are now established by letting $\Delta x_1 = \{\Delta a_0 + \Delta a_1 + \Delta a_2\}$ and $\Delta x_2 = \{\Delta b_1 + \Delta b_2\}$ and specifying values of $\Delta |H(\omega)|$ and $\Delta \theta(\omega)$. For $\Delta |H(\omega)| = \pm 0.1$ the minimum quantization region using (24) is shown in Fig. 6. This region is established by substituting various values of $\partial |H(\omega)| / \partial a_0$ and $\partial |H(\omega)| / \partial b_1$ into (24) to find the set of lines that enclose a minimum area. For $\theta(\omega) = \pm 1$ degree the minimum region is shown in Fig. 7. The actual coefficient quantization is done so that Δx_1 and Δx_2 lie inside of each quantization region. The quantized coefficients and quantization errors are shown in Table 1. Note that the quantization

was done to yield coefficients that can be represented by a minimum number of binary bits. The binary two's complement representation for each coefficient is also given in Table 1. Graphs of the error plotted as a function of frequency are shown in Figs. 8 and 9. These graphs show that the specifications are met. For higher frequencies the errors converge toward zero.

The reader should be aware that quantization leading to a binary representation for each coefficient is more difficult than quantizing a decimal number. To achieve a minimum word length it is necessary to carefully quantize each group of coefficients so that error cancellation takes place. For example, in Table 1 the sum of Δb_1 and Δb_2 gives a value of Δx_2 that is valid even though both Δb_1 and Δb_2 are too large in magnitude to fit into the quantization region.

To give a shorter word length for some coefficients the control algorithm can be separated into two first order sections in cascade. The resulting discrete transfer functions are given by

$$G_1(z) = \frac{.1002249443z - .09972506821}{z - .999500125} \quad (26)$$

and

$$G_2(z) = \frac{8.337871639 - 7.931063313z}{z - .5931914378} \quad (27)$$

Quantization regions for each section were established using the procedures described previously and the quantized coefficients are shown in Table 2. Comparison of the results in Tables 1 and 2 shows that fewer bits are required for some coefficients when the discrete lag-lead compensator is implemented as two first order sections in cascade. However, one additional coefficient is required for this realization.

COEFFICIENT	QUANTIZED COEFFICIENT	ERROR	BINARY EQUIVALENT
$b_1 = -1.592691562$	-1.5927734370	8.1875×10^{-5}	10.0110100001
$b_2 = .592894916$	$.5930175782$	-12.26622×10^{-5}	0.100101111101
$a_0 = .835661882$	$.8356933593$	-3.14773×10^{-5}	0.110101011111
$a_1 = -1.626383584$	-1.6264648440	8.126×10^{-5}	10.01011111101
$a_2 = .790925055$	$.7910156250$	-9.057×10^{-5}	0.110010101

$$\Delta x_1 = -4.07876 \times 10^{-5}$$

$$\Delta x_2 = -4.07872 \times 10^{-5}$$

Table 1

Table of Quantized Coefficients
and Binary Equivalents

Each Binary Variable is Given in Two's
Complement Form With a Leading Sign Bit

COEFFICIENT	QUANTIZED COEFFICIENT	ERROR	BINARY EQUIVALENT
$b_2 = - .999500125$	.9995117187	$- 1.15937 \times 10^{-5}$	1.00000000001
$a_1 = .1002249443$	.1000976562	1.272881×10^{-4}	0.00011001101
$a_2 = - .09972506821$	- .099609375	$- 1.15949 \times 10^{-4}$	1.111001101

(a)

Section 1

$b_2 = - .5931914378$	.59375	5.585622×10^{-4}	1.01101
$a_1 = 8.337871639$	8.3359375	1.934139×10^{-3}	01000.0101011
$a_2 = - 7.931063313$	- 7.9296875	$- 1.37581 \times 10^{-3}$	1000.0001001

(b)

Section 2

Table 2
Table of Quantized Coefficients and Binary
Equivalents for Two First Order Sections in Cascade

REFERENCES

1. J. T. Tou, Digital and Sampled Data Control Systems, McGraw-Hill, 1959.
2. A. J. Monroe, Digital Processes for Sampled Data Systems, John Wiley and Sons, 1962.
3. B. C. Kuo, Discrete Data Control Systems, Prentice-Hall, 1970.
4. L. R. Rabiner and C. M. Rader, Digital Signal Processing, IEEE Press, 1972.
5. B. Liu, "Effects of Finite Word Length on the Accuracy of Digital Filters--A Review," IEEE Trans. on Circuit Theory, Vol. CT-18, November 1971.
6. J. F. Kaiser, "Some Practical Considerations in the Realization of Linear Digital Filters," Proc. Third Annual Allerton Conference on Circuit and System Theory, pp. 621-633, 1965.
7. R. K. Otnes and L. P. McNamee, "Instability Thresholds in Digital Filters Due to Coefficient Rounding," IEEE Trans. on Audio and Electroacoustics, Vol. AV-18, December 1970.
8. P. E. Mantey, "Eigenvalue Sensitivity and State-Variable Selection," IEEE Trans. on Automatic Control, Vol. AC-13, June 1968.
9. C. M. Rader and B. Gold, "Effects of Parameter Quantization on the Poles of a Digital Filter," Proc. IEEE, pp. 688-689, May 1967.
10. B. Gold and C. M. Rader, Digital Processing of Signals, McGraw-Hill, 1969.
11. J. B. Knowles and E. M. Olcayto, "Coefficient Accuracy and Digital Filter Response," IEEE Trans. on Circuit Theory, Vol. CT-15, March 1968.

Fig. 1 Partial Derivatives of the Magnitude and Phase Function with Respect to the b_1 Coefficient

Fig. 2 Partial Derivatives of the Magnitude and Phase Function with Respect to the b_2 Coefficient

Fig. 3 Partial Derivative of the Magnitude and Phase Functions with Respect to the a_0 Coefficient

Fig. 4 Partial Derivatives of the Magnitude and Phase Function with Respect to the a_1 Coefficient

Fig. 5 Partial Derivatives of the Magnitude and Phase Function with Respect to the a_2 Coefficient

Fig. 6 Quantization Region for $\Delta|H(\omega)| = \pm 1$

Fig. 7 Quantization Region for $\Delta\theta(\omega) = \pm 1$ Degree

Fig. 8 Magnitude Function Error

Fig. 9 Phase Function Error

IMPLEMENTATION OF
DIGITAL CONTROLLERS

by

Thomas A. Brubaker
Electrical Engineering Department
Colorado State University
Fort Collins, Colorado 80521

and

William Loendorf
Electrical Engineering Department
Colorado State University
Fort Collins, Colorado 80521

ABSTRACT

The variance of error at the output of a digital control system due to input data quantization and product rounding in a digital controller is derived. Two forms of controller implementations are considered. Then an expression for the word length that yields a specified variance of error is determined. Scaling is then discussed and an example consisting of a discrete integral plus proportional controller is presented.

INTRODUCTION

The use of discrete data processing algorithms as an integral part of a control system is well known. Books by Tou [1], Monroe [2] and Kuo [3] describe many of the design concepts for these systems called sampled data or digital control systems. In a somewhat independent effort design and implementation procedures have been developed for digital filters. A collection of papers in this area has recently been edited by Rabiner and Rader [4]. A tutorial paper that provides an overview of errors in the implementation of digital filters has been written by Liu [5].

The difference between a digital filter and a digital control system is in the system structure. A digital filter is inherently an open loop signal processing component that is described entirely by a difference equation. In a digital control system, the discrete control algorithm, which is a digital filter, is a component in a feedback loop that also usually contains analog elements. Analysis of the mixed system then requires the use of artificial sampling elements within the system.

Currently, discrete control algorithms are often implemented using a digital computer that is programmed to handle one or more control paths. However, for some applications, the real-time speed requirements can tax the capacity of a computer. Furthermore, the real-time programming of many parallel control algorithms is costly and difficult to manage.

As an alternative, the recent development of small microprogrammable processors now permits a different design philosophy for digital control systems. The use of these processors to implement discrete control algorithms

as independent units or under control of a central computer leads to a distributed computing structure that is well suited for use in a variety of control systems. In a typical system, each microprogrammable processor is initially programmed from a central computer to perform a system function. If operating conditions change the processors can be reprogrammed to meet new requirements.

In the implementation of any discrete control algorithm, the designer should consider the effect of errors due to finite length arithmetic. The errors that are introduced are caused by the quantization of the input data, quantization of the algorithm coefficients and the quantization of products. When floating point hardware is used, sums and differences also introduce errors. For satisfactory system operation, the algorithm should be implemented in a form that minimizes the effect of these errors and the word length needed for the implementation should be determined.

Quantization has been studied by a number of researchers. Bennett [6] and Widrow [7] have shown that for a rounding form of quantization, the errors can be modeled as independent random variables with mean zero and variance $q^2/12$. Here q represents the value of the least significant bit in the computer word. This model has been experimentally verified for input signals that transverse several quantization levels from sample to sample. The truncation form of quantization is similar to rounding except that the mean value for each error is $q/2$.

A method for finding the maximum upper bound on the error in a digital control system due to quantization was introduced by Bertram [8]. Slaughter [9] determined an upper bound on the steady-state system error

by replacing each error source by a step function equal to the maximum value of the error source. The steady state error is then evaluated by applying the final value theorem from z-transforms. Knowles and Edwards [10, 11] assumed random quantization errors and developed statistical upper bounds for the error at the output of the system.

In this paper, the variance of the error at the system output due to rounding of the input data and the intermediate products in a digital control algorithm is evaluated for two common algorithm implementations. A unity feedback is assumed, however, the procedure can be easily extended. Fixed point arithmetic is also assumed since this form of arithmetic is common in small microprocessors. Because scaling is also a consideration, scaling procedures are described and their effect is discussed. Finally, an example consisting of a discrete form of an integral plus proportional controller is investigated and the word length needed for a given variance is compared to that obtained using methods described in reference [9].

EVALUATION OF THE ERROR VARIANCE IN DIGITAL CONTROL SYSTEMS

A block diagram for a unity feedback digital control system is shown in Fig. 1. Here the input signal $x(t)$ and the return signal from the plant are assumed to be analog. These signals are converted into digital form and subtracted to form the discrete error signal $e[nT]$. This signal serves as the input to the digital controller with transfer function

$$C(z) = \frac{\sum_{k=0}^{M-1} a_k z^{-k}}{1 + \sum_{j=1}^R b_j z^{-j}} = \frac{C_N[z]}{C_D(z)} \quad (1)$$

Errors in the implementation of (1) are due to rounding of the analog data by the analog-to-digital converter and the rounding of products. Coefficient quantization errors are deterministic and under control of the designer. These errors will not be considered here. For simplicity, the least significant quantization level will be taken as $\pm q/2$ for all errors. This is not a serious restriction and different quantization levels can be used. Each rounding error is also assumed to be an independent random variable with mean zero and variance $q^2/12$.

Quantization of the Input Data

For the system shown in Fig. 1, two input data quantization errors occur. These errors are represented by noise sources shown in the block diagram of Fig. 2. The error at the system output due to the two errors at discrete times $t=nT$ is given by

$$e_{y1}[nT] = \sum_{k=0}^n h[kT] \{e_1[(n-k)T] - e_2[(n-k)T]\} \quad (2)$$

where $h[kT]$ is the system weighting sequence given by the inverse z transform of the system transfer function

$$H[z] = \frac{C[z] G[z]}{1 + C[z] G[z]} \quad (3)$$

In (3), $G[z]$ is the z transform equivalent of the data reconstruction element and the fixed plant. If $e_1[kT]$ and $e_2[kT]$ are independent

random variables, the variance at the system output at time $t=nT$ is

$$\text{Var } \{e_{y1}[nT]\} = \frac{2q^2}{12} \sum_{k=0}^n h^2[kT] \quad . \quad (4)$$

The expression given by (4) is monotonically increasing as n increases and approaches steady state as n approaches infinity. The steady state variance is also given by the discrete Wiener-Khinchine relationship

$$\text{Var } \{e_{y1}[\infty]\} = \left(\frac{1}{2\pi j} \oint H[z] H[z^{-1}] z^{-1} dz \right) \frac{2q^2}{12} \quad . \quad (5)$$

The evaluation of the steady state variance and/or the steady-state auto-variance sequence is easily accomplished using the procedure developed in Appendix 1.

Product Rounding Errors

Product rounding errors occur within the digital control algorithm and their effect on the system depends on the form of implementation. Typical forms are the direct form one and direct form two. To minimize product rounding errors, high order algorithms are usually implemented as a cascade or parallel connection of first and second order sections [5]. The error analysis for these cases follows the general results presented in this paper and will not be considered further.

The block diagram for a third order algorithm implemented in direct form one is shown in Fig. 3 where each multiply operation consists of a rounded product plus a noise component. For this form of implementation, the transfer function from each error source to the controller output is

given by

$$C_1[z] = \frac{1}{C_D[z]} \quad . \quad (6)$$

For an Nth order algorithm described by (1) the resulting error at the output of the control system for a direct form one implementation is given by

$$e_{y2}[nT] = \sum_{k=0}^n h_1[kT] p[(n-k)T] \quad (7)$$

where

$$p[(n-k)T] = \sum_{k=0}^{M-1} \epsilon_k[(n-k)T] + \sum_{j=1}^R \delta_j[(n-k)T] \quad (8)$$

and $h_1[kT]$ is the inverse z transform of

$$H_1[z] = \frac{C_1[z] G[z]}{1 + C[z] G[z]} \quad . \quad (9)$$

The variance of the error for independent product rounding errors is

$$\text{Var}\{e_{y2}[nT]\} = (M+R) \frac{q^2}{12} \sum_{k=0}^n h_1^2[kT] \quad . \quad (10)$$

The variance of the total error due to input data quantization and product rounding for the digital control algorithm implemented in direct form one is now given by

$$\text{Var}\{e_{yD1}[nT]\} = \frac{q^2}{12} \left\{ 2 \sum_{k=0}^n h^2[kT] + (M+R) \sum_{k=0}^N h_1^2[kT] \right\} \quad . \quad (11)$$

The other common form of implementation for a digital controller is the direct form two shown in Fig. 4 for a third order system. Here, the errors generated by multiplying with the denominator coefficients can be

considered as input errors to the digital controller. The errors due to the a_1 coefficients are simply additive at the controller output. The resulting variance of error at the system output due to input data quantization and product rounding for a direct form two implementation is

$$\text{Var}\{e_{yD2}[nT]\} = \frac{q^2}{12} \left\{ 2 \sum_{k=0}^n h^2[kT] + R \sum_{k=0}^n h^2[kT] + (M) \sum_{k=0}^n h_2^2[kT] \right\} \quad (12)$$

where $h_2[kT]$ is the inverse z transform of

$$H_2[z] = \frac{G[z]}{1 + C[z] G[z]} \quad (13)$$

WORD LENGTH REQUIREMENTS USING THE VARIANCE OF ERROR AT THE SYSTEM OUTPUT

For a given analog signal range of $\pm A$, the level of least quantization referred to the algorithm input is given by the inequality

$$q \geq \frac{A}{2^{N-1}} \quad (14)$$

In (14), N represents the number of magnitude bits that are combined with a sign bit within the computer. The inequality is used because the least level of quantization is often chosen so the range of the scaled binary variables slightly exceeds the value of A . For example, with a q of 10 mv and a ten bit and sign word length, the maximum value of the binary variables in terms of signal units is 10.23 volts. Carefully note that q is given in terms of the analog range at the system input.

Given a specification on the variance of error at the system output, the word length that allows the specification to be met is found by substituting (14) into (11) or (12) and solving the resulting inequality for N . For a specified steady state variance σ_s^2 , the word length requirement to meet or exceed the specification is found by substituting (14) into (11) or (12) with $\text{Var}\{e_{yD1}[\infty]\} = \sigma_s^2$ or $\text{Var}\{e_{yD2}[\infty]\} = \sigma_s^2$. This gives the word length for a direct form one implementation of the digital control as

$$N_{D1} \geq \log_2 \left\{ \left(\frac{A^2}{12 \sigma_s^2} \left[2 \sum_{k=0}^{\infty} h^2[kT] + (M+R) \sum_{k=0}^{\infty} h_1^2[kT] \right] \right)^{\frac{1}{2}+1} \right\}. \quad (15)$$

For the direct form two we have

$$N_{D2} \geq \log_2 \left\{ \left(\frac{A^2}{12 \sigma_s^2} \left[2 \sum_{k=0}^{\infty} h^2[kT] + R \sum_{k=0}^{\infty} h^2[kT] + (M) \sum_{k=0}^{\infty} h_2^2[kT] \right] \right)^{\frac{1}{2}+1} \right\}. \quad (16)$$

In (15) and (16) N_{D1} and N_{D2} are the smallest integers that satisfy the inequalities. In addition a sign bit is normally required.

SCALING

Along with word length considerations for input data quantization and product rounding, the designer of a digital control algorithm must consider scaling. For the algorithm to operate properly, the output $r(nT)$ as shown in Fig. 1, cannot overflow for any given input signal $x(t)$. If this happens, the control loop is effectively open and the system will not function in a desired manner.

For the common case of fixed point two's complement arithmetic, Jackson, Kaiser and McDonald [11] have shown that intermediate overflow at the output of any multiplier or adder will not alter the algorithm performance as long as the algorithm response does not overflow. This means that for the filter form shown in Fig. 3 the response $r(nT)$ is the only variable that must be scaled. For the direct form two, both $r(nT)$ and $r_1(nT)$ must be scaled. Scaling for these implementations has been investigated by Jackson [12] who used Holders inequality to establish steady state bounds for various digital filter forcing functions. However, these bounds do not include the effect of transients.

In practice, analytical methods for determining bounds on $r(nT)$ often gives pessimistic results and the use of simulation for a given class of input signals can be useful. Once the maximum value or a bound for $r(nT)$ and/or $r_1(nT)$ is determined the actual scaling is not difficult. For the direct form one implementation shown in Fig. 3, each a_1 coefficient is reduced in magnitude by a common factor. This effectively reduces the gain of the algorithm. For a direct form two implementation, the input signal $e(nT)$ is multiplied by a scale factor so that the values of $r_1(nT)$ do not overflow. The a_1 coefficients are reduced in magnitude to prevent $r(nT)$ from overflowing. Thus, again the scaling is accomplished by effectively reducing the filter gain. This gain may be reintroduced at other points in the system if desirable. The error analysis for the scaled filters or control algorithms follows from previous results.

Another important alternative is the use of an adaptive scale factor that is utilized only when an overflow is detected in the system. If this occurs and if sufficient computational time is available the filter can be automatically scaled until overflow ceases to exist.

EXAMPLE

To illustrate the procedure consider the system shown in Fig. 1 when $C(z)$ is a discrete integral plus proportional controller

$$D(z) = \frac{(k_i T/2 + k_p)z + k_i T/2 - k_p}{z - 1} \quad (17)$$

and the plant is given by $G(s) = 2/s+5$. The sampling interval was chosen as .01 and the constants were first assigned values $K_i = .75$ and $K_p = .25$. This results in a sluggish system with no overshoot for a step input. For a step input the maximum value of $e(nT)$ is one and the maximum value of the controller output $r(nT)$ is $5/2$. To obtain a controller with a gain of one, a multiplying factor of $5/2$ is placed in the data reconstruction element. The resulting z transforms are

$$G'(z) = \frac{5}{2} \frac{1-e^{-j s T}}{s} \frac{2}{s+5} = \frac{.04877}{z-.95123} \quad (18)$$

and

$$D'(z) = \frac{.1015 z - .0985}{z - 1} \quad (19)$$

Plots of $e(nT)$, $r(nT)$ and $y(t)$ are shown for the scaled system with $x(t) = \mu(t)$ in Fig. 5.

For a direct form one implementation shown in Fig. 6 there are two input rounding errors. The two product rounding errors are introduced between the numerator and denominator portions of $D'(z)$. If the errors are each assumed to be step functions with magnitude $q/2$, the method developed by Slaughter [9]] results in a steady state error at the system output

$$E_{ss} = \lim_{n \rightarrow \infty} e[nT] = \left\{ \left[\frac{q z G'(z) D'(z)}{1 + G'(z) D'(z)} \right] + \left[\frac{q z \frac{G'(z)}{z-1}}{1 + G'(z) D'(z)} \right] \right\}_{z=1}, \quad (20)$$

which gives upon substitution

$$E_{ss} = (334.33)q. \quad (21)$$

For a direct form two implementation there are two input rounding errors. The two product roundings are introduced at the output of $D'(z)$. The steady state error is given by

$$E_{ss} = \lim_{n \rightarrow \infty} e[nT] = \left\{ \frac{q z G'(z) D'(z)}{1 + G'(z) D'(z)} + \frac{q z \frac{G'(z)}{z-1}}{1 + G'(z) D'(z)} \right\}_{z=1} \quad (22)$$

which yields

$$E_{ss} = q. \quad (23)$$

Substituting (14) into (21) and (23) now gives an expression for the word length in terms of E_{ss}/A as

$$N_{df1} \geq \log_2 \left\{ \frac{(334.33)}{(E_{ss}/A)} + 1 \right\} \quad (24)$$

for the direct form one implementation and

$$N_{df2} \geq \log_2 \left\{ \frac{1}{(E_{ss}/A)} + 1 \right\} \quad (25)$$

for the direct form two implementation. Note that the variable A refers to the maximum value of the variable $e(nT)$ for a controller with a gain of one and no overflow. Values of N are tabulated in Table 1.

If the rounding errors are modeled as independent random variables with mean zero and variance $q^2/12$, the steady state variance of the error at the system output for a direct form one implementation is given by

$$\text{Var}(e) = \frac{2q^2}{12} \left\{ \sum_{k=0}^{\infty} h^2(kT) + \sum_{k=0}^{\infty} h_1^2(kT) \right\} \quad (26)$$

where $h(kT)$ is the inverse z transform of

$$H'(z) = \frac{D'(z) G'(z)}{1 + D'(z) G'(z)} \quad (27)$$

and $h_1(kT)$ is the inverse z transfer

$$H_1'(z) = \frac{G'(z)/(z-1)}{1 + G'(z) D'(z)} \quad (28)$$

Substituting gives

$$\text{Var}(e) = (25.3)q^2 \quad (29)$$

For the direct form two implementation

$$\text{Var}(e) = (4.07 \times 10^{-3})q^2 \quad (30)$$

To establish word length requirements the variable e/A is now assumed to be a Gaussian random variable. The probability that e/A is between $\pm E_{ss}$, where E_{ss} is the previously given steady state error, is now defined as

$$\text{Prob} \left\{ -\frac{E_{ss}}{A} \leq \frac{e}{A} \leq \frac{E_{ss}}{A} \right\} = \text{confidence level.} \quad (31)$$

For a given confidence level, normalized distribution tables can be used by writing

$$\text{Prob} \left\{ -\frac{E_{ss}}{A\sigma} \leq \frac{e}{A\sigma} \leq \frac{E_{ss}}{A\sigma} \right\} = \text{confidence level}$$

where

$$\sigma = \left\{ \text{Var}\left(\frac{e}{A}\right) \right\} \geq \begin{cases} \left(\frac{25.3}{2^N - 1} \right)^{1/2} & \text{direct form one} \\ \left(\frac{4.07 \times 10^{-3}}{2^N - 1} \right)^{1/2} & \text{direct form two} \end{cases} \quad (32)$$

For a 95% confidence level

$$\frac{E_{ss}}{A\sigma} = 2 \quad (33)$$

and substitution of (32) into (33) now gives

$$N_{df1} \geq \log_2 \left\{ \frac{10}{E_{ss}/A} + 1 \right\} \quad (34)$$

for the direct form one. For the direct form two

$$N_{df2} \geq \log_2 \left\{ \frac{.128}{E_{ss}/A} + 1 \right\} \quad (35)$$

The value of N_{df1} and N_{df2} are also tabulated in Table 1.

Referring to Table 1, the form that is used to implement the controller is obviously a critical factor. The use of the variance of error results in a somewhat shorter word length for a given value of E_{ss}/A . However, in practice a very short word length will give rise to other undesirable effects due to large deadbands caused by the rounding. Therefore, for the

example, a word length of ten bits plus sign might be utilized. This gives an error E_{ss}/A less than 10^{-3} for the controller implemented in direct form two.

To improve the response a new control algorithm was designed using $K_p = 3$ and $K_i = 15$. The resulting discrete controller transfer function is

$$D'[z] = \frac{1.23 z - 1.18}{z - 1} \quad (36)$$

Simulation, however, shows the controller output will overflow so a value of 1.23 was factored from (36) giving

$$D''[z] = \frac{z - .9593495935}{z - 1} \quad (37)$$

and

$$G''[z] = \frac{.0599878078}{z - .9512294245} \quad (38)$$

Response curves for the resulting system are shown in Fig. 6. Note that the steady state output for $r(nT)$ is about .8 units which means the full range of the digital controller is not being utilized fully. Word length requirements for the system are given in Table 2. For the direct form one the improved system requires fewer bits. While this result may not be expected careful examination of the system structure shows the new system to be less critical to noise generated within the system when the controller is implemented in direct form one.

E_{ss}/A	Word Length Using Slaughters Method		Word Length Using The Variance of Error and a 95% Confidence Level	
	N_{df1}	N_{df2}	N_{df1}	N_{df2}
10	6	1	2	1
1	9	1	4	1
10^{-1}	12	4	7	2
10^{-2}	15	7	10	4
10^{-3}	19	10	14	7
10^{-4}	22	14	17	11
10^{-5}	26	17	20	14
10^{-6}	29	20	24	17

Table 1

Table of Word Length Requirements

$K_i = .75$ $K_p = .25$

E_{ss}/A	Word Length Using Slaughters Method		Word Length Using The Variance of Error and a 95% Confidence Level	
	N_{df1}	N_{df2}	N_{df1}	N_{df2}
10	2	1	1	1
1	4	1	2	1
10^{-1}	8	4	4	2
10^{-2}	11	7	8	5
10^{-3}	14	10	10	8
10^{-4}	18	14	14	8
10^{-5}	21	17	18	14
10^{-6}	24	20	21	18

Table 2

Table of Word Length Requirements

$K_i = 15$ $K_p = 3$

APPENDIX A

In this section a closed form method for computing the discrete autocovariance function is derived. Consider a discrete transfer function

$$H[z] = \frac{\sum_{k=0}^M a_k z^{-k}}{1 - \sum_{j=1}^N b_j z^{-j}} = H_1[z] H_2[z] \quad (1a)$$

where

$$H_1[z] = \frac{1}{1 - \sum_{j=1}^N b_j z^{-j}} \quad (2a)$$

and

$$H_2[z] = \sum_{k=0}^M a_k z^{-k} . \quad (3a)$$

If an independent noise sequence $\{e[jT]\}$ with mean zero and variance σ^2 is applied to $H_1[z]$ the difference equation representation is given by

$$x[nT] = \sum_{j=1}^N b_j x[(n-j)T] + e[nT] . \quad (4a)$$

The discrete autovariance at time $t = nT$ is now defined as

$$R_n[kT] = E\{x[nT] x[(n-k)T]\} \quad k = 0, 1, 2, \dots, n . \quad (5a)$$

Since $e[nT]$ is taken from an independent noise sequence $x[(n-k)T]$ and $e[nT]$ are uncorrelated. Multiplying (4a) by $x[(n-k)T]$, $k = 0, 1, 2, \dots, N-1$ and taking the expectation now results in a set of

coupled difference equations in the variables $\{R_n[0T], R_n[T], \dots, R_n[(N-1)T]\}$.

Taking the z transform and applying the final value theorem now permits the first N terms of the steady state autocovariance sequence to be written in matrix form as

$$\begin{bmatrix} R[0T] \\ R[T] \\ \vdots \\ R[(n-1)T] \end{bmatrix} = M^{-1} \begin{bmatrix} \sigma^2 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (6a)$$

where M is the matrix

$$M = \begin{bmatrix} 1 - \sum_{j=1}^N b_j^2 & -2 \sum_{k=1}^{N-1} b_j b_{j+1} & -2 \sum_{j=1}^{N-2} b_j b_{j+2} & \dots & -2b_1 b_N \\ (b_1) & (1 + b_2) & \cdot & \cdot & b_N \\ (b_2) & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ (b_N) & (b_{N-2} b_N) & \cdot & \cdot & 1 \end{bmatrix} \quad (7a)$$

The remaining terms in the autocovariance sequence are found from

$$R[kT] = - \sum_{j=1}^N b_j R[(k-j)T] \quad k \geq N \quad (8a)$$

The steady state autocovariance at the output of $H[z]$ is given by

$$\lim_{n \rightarrow \infty} E\{y[nT] y[(n-k)T]\} = R_y[kT] = \bar{a}^{-t} c \bar{a} \quad (9a)$$

where $\bar{a}^t = [a_0, a_1 \dots a_M]$ and c is a constant matrix with terms

$$c_{ij} = R[(k-j+1)T] = \lim_{n \rightarrow \infty} E[x[nT] x[n-(k-j+1)T]] \quad (10a)$$

When computing the steady state autocovariance for $k < M-1$ certain terms in the C matrix are found using the relationship $R[kT] = R[-kT]$.

Thus, a fixed nonrecursive expression only exists for $k \geq M-1$.

The variance at the output of the system is found using the results of (6a). The expression at the output is

$$\lim E[y_n^2] = \bar{a}^t C_1 \bar{a} \quad (11a)$$

where C_1 is the autocovariance matrix

$$\begin{bmatrix} R[0T] & R[T] & \cdot & R[MT] \\ R[T] & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ R[MT] & \cdot & \cdot & R[0T] \end{bmatrix} \quad .$$

REFERENCES

1. J. T. Tou, Digital and Sampled Data Control Systems, McGraw-Hill, 1959.
2. A. J. Monroe, Digital Processes for Sampled Data Systems, John Wiley and Sons, 1962.
3. B. C. Kuo, Discrete Data Control Systems, Prentice-Hall, 1970.
4. L. R. Rabinier and C. M. Rader, Digital Signal Processing, IEEE Press, 1972.
5. B. Liu, "Effects of Finite Word Length on the Accuracy of Digital Filters--A Review," IEEE Trans. on Circuit Theory, Vol. CT-18, November 1971.
6. W. R. Bennett, "Spectra of Quantized Signals," Bell System Technical Journal, Vol. 27, No. 3, 1948.
7. B. Widrow, "Statistical Analysis of Amplitude Quantized Sampled-Data Systems," AIEE Trans. Part 2, pp. 555-562, 1961.
8. J. E. Bertram, "The Effect of Quantization in Sampled-Data Feedback System," AIEE Transactions, Vol. 77, pp. 177-182, 1958.
9. J. B. Slaughter, "Quantization in Digital Control Systems," IEEE Trans. on Automatic Control, Vol. AC-9, pp. 70-74, 1964.
10. J. B. Knowles and R. Edwards, "Effect of a Finite Word Length Computer in a Sampled-Data Feedback System," Proceedings of the IEE, Vol. 112, No. 6, pp. 1197-1207, 1965.
11. J. B. Knowles and R. Edwards, "Computational Error Effects in a Direct Digital Control System," Automatica, Vol. 4, pp. 7-29, 1966.
12. L. B. Jackson, J. F. Kaiser and H. S. McDonald, "An Approach to the Implementation of Digital Filters," IEEE Trans. on Audio and Electroacoustics, AU-18, September 1968.
13. L. B. Jackson, "On the Interaction of Roundoff Noise and Dynamic Range in Digital Filters," Bell System Technical Journal, Vol. 49, February 1970.

Figure 1 Block Diagram for a Unity Feedback Digital Control System

Figure 2 Block Diagram Showing Data Quantization Errors

Figure 3 Block Diagram for a Third Order Digital Filter or Controller Implemented in Direct Form I.

Figure 4 Block Diagram for a Third Order Digital Filter or Controller Implemented in Direct Form 2

Figure 5 Response Curves for $K_i=0.75$ and $K_p=0.25$

Figure 6 Response Curves for $K_i=15$ and $K_p=3$