The Digital Implementation of Control Compensators: The Coefficient Wordlength Issue

PAUL MORONEY, MEMBER, IEEE, ALAN S. WILLSKY, MEMBER, IEEE, AND PAUL K. HOUPT, MEMBER, IEEE

***Abstract*—There exist a number of mathematical procedures for designing discrete-time compensators. However, the digital implementation of these designs, with a microprocessor, for example, has not received nearly as thorough an investigation. The finite-precision nature of the digital hardware makes it necessary to choose a computational structure that will perform adequately with regard to the initial objectives of the design. This paper describes a procedure for estimating the required fixed-point coefficient wordlength for any given computational structure for the implementation of a single-input single-output LQG design. The results are compared to the actual number of bits necessary to achieve a specified performance index.**

I. INTRODUCTION

THE DESIGN of discrete-time compensators through the use of optimal regulators, pole-placement concepts, observer theory, optimal filtering [1], [2], and also via classical control theory [3] has received a great deal of attention in the literature. In the past such designs have usually been implemented on large, expensive, floating-point computer systems. However, the number of applications that could effectively use small-scale hardware control systems that work in real time has greatly increased, especially with the advent of the inexpensive microprocessor.

While the recent advances in digital hardware capabilities have opened many new possibilities for control system implementations, they have also raised new issues. A number of these involve the problems that arise in dealing with the fixed-point arithmetic and finite wordlengths of small-scale digital systems. As these problems are not addressed at all in the idealized mathematical design procedures that have been developed to date, a methodology must be established for treating the digital implementation of a design. The *mathematical design* procedure produces an infinite-precision ideal compensator specification. The job of the *implementation* step is to specify and order sequentially the critical computations that must take place in the compensator so that the end result, the actual finite-precision digital system, performs as close to the ideal as is consistent with the expense and speed requirements of the application. The implementation step also includes a specification of the hardware architecture and components. It is important to note that the mathematical design and the implementation phases may not be totally independent, since the implementation can be very important in determining an acceptable sampling rate and the number of operations that can be performed per sampling period. These then become restrictions on the compensator design.

Our approach draws on the field of digital signal processing [4], [5], which has generated many results concerning the realistic implementation of digital filters. Good reviews concerning the effects of finite precision in digital filters—specifically, the effects of coefficient quantization, limit cycles, and quantization noise—can be found in [6], [7], and [8]. Some work has been done in looking at similar questions for digital feedback compensators, but it has been somewhat limited. Knowles and Edwards [9] and Curry [10] have each considered a roundoff noise analysis of certain sampled-data systems. Bertram [11], Slaughter [12], Johnson [13], and Lack [14] have developed amplitude bounds on the effects of quantization in sampled-data control systems. Sripad [15] has looked in some depth at the roundoff noise and finite-precision coefficient performance of the discrete-time Kalman filter and linear-quadratic-Gaussian controller. Rink and Chong [16] have derived bounds on the effects of quantization errors in floating-point regulators. Farrar [17] has pointed out in a basic way some of the issues involved in implementing continuous-time linear-quadratic-Gaussian controllers as discrete-time fixed-point microprocessor-based systems. Willsky [18] has pointed out some of the parallels between filter and controller implementations. In this paper, we *use*, *adapt*, and *extend* the ideas of digital signal processing for digital feedback compensators; specifically we examine the issue

Manuscript received April 12, 1979; revised December 10, 1979 and February 18, 1980. Paper recommended by H. A. Spang, III, Past Chairman of the Applications, Systems Evaluation, and Components Committee. This work was supported by NASA Ames under Grant NGL-22-009-124 and was performed in part at the M.I.T. Laboratory for Information and Decision Systems and in part at the Charles Stark Draper Laboratory.

P. Moroney was with the Charles Stark Draper Laboratory, Inc., Cambridge, MA 02139, and with the Massachusetts Institute of Technology, Cambridge, MA 02139. He is now with the Linkabit Corporation, San Diego, CA 92121.

A. S. Willsky is with the Department of Electrical Engineering and Computer Sciences and the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139.

P. K. Houpt is with the Department of Mechanical Engineering and the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139.

0018-9286/80/0800-0621$00.75

of coefficient quantization in fixed-point compensator implementations. Since researchers in digital signal processing have developed a great many tools for implementing digital filters, we should try to use these concepts. However, because of the presence of a feedback loop around the digital compensator, many of these concepts do not directly apply for control, and adaptations are necessary. Finally, in our treatment, where the coefficients of the ideal compensator have been chosen to optimize some scalar criterion, we have to modify the notion of *statistical coefficient wordlength*. This modification also constitutes a possible extension for digital signal processing.

The basic idea behind the selection of a coefficient wordlength is the same for digital filters and digital compensators. Approximating the coefficients of a structure with a finite number of bits causes a degradation in the system's performance as compared to the ideal. Assuming that a given quantitative performance measure is provided, we can measure the tradeoff in the number of bits versus the degradation. Then, assuming that we specify an acceptable amount of degradation, one must determine the minimum number of coefficient bits needed to meet this goal. Clearly, a straightforward way to determine this wordlength is to simply reevaluate the measure of performance for sets of coefficients that are quantized to different wordlengths, and to choose the smallest wordlength meeting the design specification. This direct method can be quite time-consuming, even when we assume that the coefficients are to be rounded to the shorter wordlengths, and not chosen in some more complex fashion [19].

The concept of a (simpler) statistical estimate of the wordlength originated in the study of digital filters with the work of Knowles and Olcayto [20]. Avenhaus [19] applied this idea to the digital filter power transfer function (as a performance measure), and later Crochiere [21], [22] used the concept with the filter transfer function magnitude and a wordlength optimization procedure. All three of these studies chose different performance measures, none of which seem to be particularly appropriate for control problems where the compensator phase is critical. In this paper we adapt the statistical wordlength concept to the steady-state linear-quadratic-Gaussian (LQG) control problem. We have selected the LQG problem for several reasons. First, it has received a great deal of attention in the recent literature, due to its robustness, multivariate formulation, and optimal nature. Second, the LQG problem has an explicit scalar performance index J which can be used to gauge the effectiveness of an implementation. In fact, this was the performance measure used by Sripad [15].

It would also have been possible to choose a criterion such as phase margin, output noise power, or any combination of stability or noise measures. If the problem under consideration was simply a Kalman filter, then a suitable performance measure would be the trace of the error covariance matrix. We have chosen J in order to present our results in a specific context. These results extend in a straightforward manner to other measures. It should also be noted that we treat the single-input single-output case for convenience, and because it is in this setting that most digital filtering results have been developed. The following analysis can be extended easily to the multiple-input multiple-output case, once a multiple-input multiple-output structure is specified [23].

In terms of its applications, the statistical wordlength estimate that we develop is useful in the relative comparison of different structures on the basis of their required coefficient wordlengths. However, more importantly, the statistical estimate can be used as the basis for an iterative gradient-search constrained optimization procedure (see [23] and the conclusions of this paper) for generating minimum coefficient wordlength structures. This is possible because the statistical estimate is *continuous*, that is, not limited to an integral number of bits, and also because this estimate is differentiable with respect to the coefficients of the structure. These points are not true of the direct method of wordlength determination, which is essentially the method used by Sripad [15].

The organization of this paper is as follows. In Section II we describe the derivation of the LQG compensator including the real computation time constraints. The notion of a compensator *structure* and a notation adequate for expressing the computations that occur in such a structure is presented in Section III. In Sections IV, V, and VI we introduce the notion of statistical wordlength and apply it to the LQG problem. Finally, we present examples of the technique and compare the results to the direct method.

II. The LQG Controller Problem

In this section we present the single-input single-output LQG control configuration and the mathematical, or ideal, design of the compensator. Assume that we wish to design a digital discrete-time compensator for a continuous-time plant system, and that the control signal is piecewise constant. Typically, after sampling the plant output at rate $1/T$, the compensator is designed to produce an output $u(k)$ based on the compensator inputs up to and including $y(k)$. Such a design would not be implementable, since $u(k)$ and $y(k)$ refer to identical sample times, and a finite time must be allowed for the computation of $u(k)$ from $y(k)$. These two requirements are contradictory.

Kwakernaak and Sivan [1] present a design procedure where $u(k)$ depends only on compensator inputs up to and including $y(k-1)$. This allows a full sample interval for the computation of $u(k)$. If, however, the computation time is much shorter than the sample interval, this implies some inefficiency; the output $u(k)$ will be available long before it is used as a control. Thus, Kwakernaak and Sivan also include a method for skewing the sample time of the plant output with respect to the rest of the compensator. The compensator output $u(k)$ will still depend on inputs up to and including $y(k-1)$, but now $y(k-1)$ is produced only one *calculation* time before $u(k)$ is needed. This eliminates any inefficiency [1], [23].

When we discretize the continuous-time plant model at some rate $1/T$ and account for any sample skewing we obtain the following set of equations describing the plant output at the sample times:

$$x(k+1)=\Phi x(k)+\Gamma u(k)+w_1(k)$$
$$y(k)=Lx(k)+w_2(k) \tag{1}$$

where n is the system order, $\Phi(n\times n)$ is the transition matrix, $\Gamma(n\times 1)$ and $L(1\times n)$ are the input and output gains, and w_1 and w_2 are discrete white Gaussian noise sequences with covariance matrices $\Theta_1(n\times n)$ and $\Theta_2(1\times 1)$, respectively. The control law is chosen to minimize the following performance index (the discretized version of a continuous-time performance index):

$$J=E\left\{\lim_{i\to\infty}\frac{1}{2i}\sum_{k=-i}^{+i}\left(x'(k)Qx(k)+2x'(k)Mu(k)+Ru^2(k)\right)\right\} \tag{2}$$

where Q is $n\times n$, M is $n\times 1$, and R is 1×1. Assuming a piecewise-constant control signal $u(t)$ formed by applying the $u(k)$ samples to a zero-order hold, and a linear compensator, the optimal compensator design can be described as follows:

$$\hat{x}(k+1)=\Phi\hat{x}(k)+\Gamma u(k)+K(y(k)-L\hat{x}(k))$$
$$u(k+1)=G\hat{x}(k+1). \tag{3}$$

Note that the equations in (3) base the current control $u(k)$ only on *past* outputs $y(k-1), y(k-2),\cdots$,[1], as discussed above. The $(n\times 1)$ matrix K is the solution to a Kalman filter problem, and can be computed by solving the following algebraic Ricatti equation [1]:

$$\Sigma=\Phi\{I-\Sigma L'(\Theta_2+L\Sigma L')^{-1}L\}\Sigma\Phi'+\Theta_1$$

where

$$K=\Phi\Sigma L'(\Theta_2+L\Sigma L')^{-1}. \tag{4}$$

Similarly, the $(1\times n)$ matrix G results from an optimal regulator design and the following algebraic Riccati equation [1]:

$$P=(\Phi-\Gamma R^{-1}M')'P\{I-\Gamma(R+\Gamma'P\Gamma)^{-1}\Gamma'P\}\cdot(\Phi-\Gamma R^{-1}M')+Q-MR^{-1}M'$$

where

$$G=(R+\Gamma'P\Gamma)^{-1}\Gamma'P(\Phi-\Gamma R^{-1}M')+R^{-1}M'. \tag{5}$$

Fig. 1 presents a simple block diagram of the system and its (infinite-precision) compensator. This ideal compensator (3) can be described by an infinite-precision map (transfer function) in the digital frequency domain,

$$\frac{U(z)}{Y(z)}=-G(z-\Phi+KL+\Gamma G)^{-1}K. \tag{6}$$

The digital filter transfer function (6) must be implemented in finite precision and therefore will suffer some degradation in the system's measure of performance J.

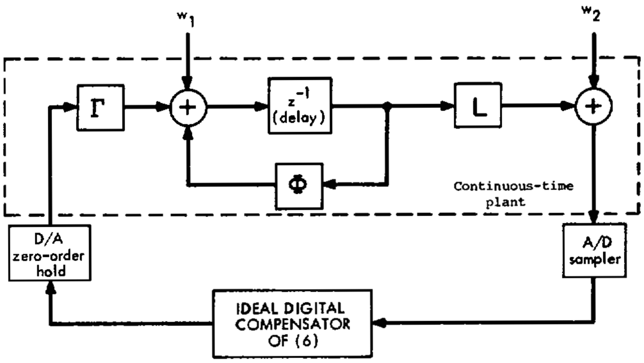


Fig. 1. Plant and compensator.

III. Algorithms and Structures

In order to discuss different implementations, one must have an accurate notation that reflects these differences. The term "structure" is employed to specify the exact finite-precision algorithm by which the compensator output samples u are generated from its input samples y. All structures for implementing a given filter or compensator would perform identically under infinite-precision arithmetic, but produce different quantization noise, coefficient quantization effects, and limit cycles when implemented in finite-precision. A good review of some of the structures used to implement single-input, single-output digital filters can be found in [22], [24], and [25].

Now let us examine the compensator equations (3) to see if they represent a possible computational structure. Consider the nodes $\hat{x}(k)$ to be the *states* of the structure, where a state corresponds to the output of a delay element in the signal flow graph [22] of the structure. Then the $\hat{x}(k+1)$ equation describes the new state values to be functions of the current states $\hat{x}(k)$, the current output $u(k)$, and the current compensator input $y(k)$. This is an accurate representation of a set of computations. However, the last equation in (3) shows the next output value to be a function of the next state values. This cannot accurately describe real computations, since some finite time must be allowed to compute $u(k+1)$ from $\hat{x}(k+1)$, and this is inconsistent with the identical $(k+1)$ indexes. There is in effect a series delay that is implied, yet such a delay is not accounted for in the compensator design.

This example points out the difference between compensator and filter structures [23]. In digital filtering, (3) would be taken to represent a structure. This is done frequently in describing the dependence of the output node on the state nodes. The series delay that *must* exist is ignored; after all, series delay in the filter output is not really important. However, in control systems, series delay is critical. Unplanned for delay adds negative phase shift and affects the performance of the closed-loop system in a negative way. Thus, we must include *all* the required computational delays in our description of a compensator structure. Simply adding the delay to the plant model and redesigning the compensator is a poor solution, since the order of the compensator will increase when we do this.

The best solution is to implement the compensator in such a way that its computations can take place in the allotted time intervals. For example, we can rewrite (3) as follows:

$$\hat{x}(k+1)=\Phi\hat{x}(k)+\Gamma u(k)+K(y(k)-L\hat{x}(k))$$
$$u(k+1)=-G\{\Phi\hat{x}(k)+\Gamma u(k)+K(y(k)-L\hat{x}(k))\}. \quad (7)$$

The equations (7) do represent a compensator structure, since the next values of state and outputs depend on only current values of states and inputs. In fact, *note that* $u(k)$, *the compensator output, is a state*. This will *always* be true for compensator structures. For any nth-order filter structure with only n unit delays (canonic in delays), a corresponding nth-order compensator structure will exist. However, the LQG compensator structure will have an extra delay at its output because of the series delay. Thus, an nth-order delay-canonic compensator structure will have $n+1$ unit delay elements.

Since the notion of a compensator structure is thus different from a filter structure, we must adapt the filter structure notation to account for the differences. Let u and y represent the compensator output and input, respectively. Let v represent the compensator states (other than u). We can adapt the notation used by Chan [24] for digital filters to produce the following *modified state-space representation* [23]:

$$\begin{bmatrix} v(k+1) \\ u(k+1) \end{bmatrix}=\Psi_q\Psi_{q-1}\cdots\Psi_1\begin{bmatrix} v(k) \\ u(k) \\ y(k) \end{bmatrix}. \quad (8)$$

Several important points make (8) useful.

1) Each (rounded) coefficient in the structure occurs once and only once as an entry in one of the Ψ_i matrices. The remainder of the matrix entries are ones and zeros.

2) The concept of a precedence to the operations (multiplies, adds, and quantizations) is maintained. The ordering of the Ψ matrices implies that the operations in computing the intermediate nodes

$$r_1=\Psi_1\begin{bmatrix} v(k) \\ u(k) \\ y(k) \end{bmatrix}$$

are completed first, then

$$r_2=\Psi_2\left[\Psi_1\begin{bmatrix} v(k) \\ u(k) \\ y(k) \end{bmatrix}\right]$$

next, and so forth. The parameter q specifies the number of such *precedence levels*. Examples of the modified state space representation appear in Section VII.

Notationally, it is also useful to define Ψ_∞ to be the infinite precision product of $\Psi_q, \Psi_{q-1}, \cdots, \Psi_1$ and to partition it as follows:

$$\Psi_\infty=[\Psi_{11} \quad \Psi_{12}] \quad (9)$$

where Ψ_{11} is $(n+1)\times(n+1)$ and Ψ_{12} is $(n+1)\times 1$.

IV. Statistical Wordlength for Digital Filters

In this section we review briefly the basic development of the statistical wordlength measure as used in digital signal processing [21]. Consider a general measure of performance f, a differentiable function of the coefficients $(c_1, c_2, \cdots, c_m)$ of the structure. The value of f associated with any particular finite-precision structure reflects a degradation in performance as compared to the ideal (unrounded coefficients) case f_∞. This degradation df can be expanded in a Taylor's series about the ideal value. To first order

$$df(c_1,c_2,\cdots,c_m)\approx\sum_{i=1}^{m}\left.\frac{\partial f}{\partial c_i}\right|_\infty dc_i \quad (10)$$

where c_i is the ith coefficient to be rounded, dc_i is the error due to quantization, and $\partial f/\partial c_i|_\infty$ is the first partial derivative of f evaluated at the unrounded coefficient values. Note that coefficients such as $3, 2, 1, 1/2, \cdots$ are not normally affected by rounding and should not be included in the sum (10).

If Δ is the quantization step size (the fraction represented by the least significant bit of the fixed-point coefficient word), then each dc_i must lie between $\pm\Delta/2$ (rounding assumed). Given the partial derivatives in (10), we could (upper) bound the error df, producing a very *pessimistic* wordlength estimate. Specifically,

$$df<\frac{\Delta}{2}\sum_{i=1}^{m}\left|\left.\frac{\partial f}{\partial c_i}\right|_\infty\right|. \quad (11)$$

The basic statistical wordlength idea is to produce a less pessimistic estimate by treating an *ensemble* of structures. Over this ensemble, the coefficient errors dc_i can be thought of as uniformly-distributed zero-mean uncorrelated random variables, each of variance $\Delta^2/12$. Using (10), we can now treat df as a random variable. With dc_i as described above, df will have a zero-mean, and a variance

$$\sigma_{df}^2=\frac{\Delta^2}{12}\sum_{i=1}^{m}\left(\left.\frac{\partial f}{\partial c_i}\right|_\infty\right)^2. \quad (12)$$

For large m, the central limit theorem can then be applied to justify a Gaussian distribution for df. Thus, with a given confidence level (probability), say 95 percent, one can predict the variance σ_{df}^2 needed for the error df to remain within some prescribed bound. In other words, 95 out of 100 of the structures in the ensemble will result in systems where df remains within this bound.

From a table of the Gaussian distribution,

$$\Pr[|df|\leqslant 2\sigma_{df}]=0.954. \quad (13)$$

If the quantity of interest f is constrained to lie within $\pm E_0$ of the ideal f_∞, then (13) implies that σ_{df} equal $E_0/2$. This result can be combined with (12) to produce an estimate of the parameter Δ,

$$\Delta=\frac{\sqrt{3}\,E_0}{\sqrt{\sum_{i=1}^{m}\left(\left.\frac{\partial f}{\partial c_i}\right|_{\infty}\right)^2}}. \tag{14}$$

Given Δ, the statistical wordlength can be defined to be

$$\mathrm{SWL}=l+\log_2\frac{1}{\Delta}. \tag{15}$$

The first term in (11) represents the number of bits necessary to represent the integer portion of the coefficients and the second term gives the number of bits necessary for the fractional portion of the coefficient word.

Crochiere [21], [22], [25] has presented a number of results comparing the statistical wordlength of structures, using the transfer function magnitude as the performance measure f. Since this choice of f is frequency-dependent, the resulting estimate is also frequency-dependent. The final wordlength can be selected as the maximum of the estimates over the frequency range of interest. In the examples treated by Crochiere, the statistical wordlength estimate was typically 1–3 bits conservative as compared to the actual minimum number of bits necessary to meet the transfer function error limit. Using the statistical wordlength idea, Crochiere [21] was also able to formulate an optimization procedure for designing shorter-coefficient-wordlength filter structures. Although this optimization method is quite different from the one we have alluded to it is a big motivation for developing the statistical approach for compensators.

V. Statistical Wordlength and the Performance Index J

As mentioned in Section II, it is convenient to use the performance index J in (2) as the measure of performance f in an LQG setting. Using the approach of the previous section, the change in J would be estimated by

$$dJ(c_1,c_2,\cdots,c_m)\simeq\sum_{i=1}^{m}\left.\frac{\partial J}{\partial c_i}\right|_{\infty}dc_i. \tag{16}$$

However, the optimal nature of the LQG control problem forces all the sensitivities $\partial J/\partial c_i$ to be zero. Therefore, a higher order approximation is necessary:

$$dJ\simeq\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\left.\frac{\partial^2 J}{\partial c_i\partial c_j}\right|_{\infty}dc_i\,dc_j. \tag{17}$$

The use of second-order terms (not used in digital filter analysis) is a unique aspect of our statistical wordlength formulation. However, the use of these terms would be necessary in any filter or compensator design analysis in which the statistical estimate is based on the degradation in a scalar performance measure that has been *optimized* with respect to the unrounded coefficients. For example, if a digital filter were designed by minimizing the integrated squared error between the desired and actual filter transfer function magnitude characteristics, then a statistical wordlength estimate based on this performance measure would have to use second-order sensitivities, since all first-order sensitivities would be zero. The statistical wordlength derivation that we have developed would have to be used in this case, and therefore our formulation also has some potential applications in digital filter design.

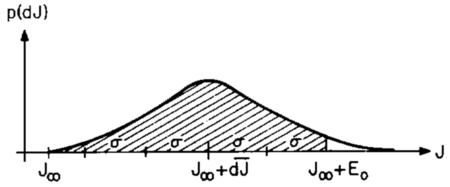


Fig. 2. Probability density of dJ.

Proceeding from (17), the mean of dJ will no longer be zero,

$$E(dJ)=\frac{1}{2}\left(\sum_{i=1}^{m}\left.\frac{\partial^2 J}{\partial c_i^2}\right|_{\infty}\right)E\left[(dc_i)^2\right]. \tag{18}$$

For convenience, define the random variable ϵ to be the square of dc_i. Its mean and variance can be shown to be $\Delta^2/12$ and $\Delta^4/180$. The variance of dJ can now be found [23],

$$\sigma_{dj}^2=\frac{\sigma_\epsilon^2}{4}\sum_{i=1}^{m}\left(\left.\frac{\partial^2 J}{\partial c_i^2}\right|_{\infty}\right)^2+(\bar{\epsilon})^2\sum_{\substack{i=1\\ i<j}}^{m}\sum_{j=1}^{m}\left(\left.\frac{\partial^2 J}{\partial c_i\partial c_j}\right|_{\infty}\right)^2. \tag{19}$$

Recall the application of the central limit theorem in Section IV. We can make the same assumption for our higher order statistical wordlength derivation. For the usual digital filtering estimate, the coefficient quantization could either decrease or increase the error in the transfer function magnitude at any specific frequency. This implied that the error was zero-mean. In the control case, the value of J can only *increase* under coefficient quantization. Thus, we need only have a specification on the maximum allowed value of J including the degradation due to coefficient quantization: $J_\infty+E_0$. Following the general approach of Section IV, we must equate this value to the two-sigma point in the distribution for dJ in order to compute our estimate of Δ (see Fig. 2):

$$J_\infty+E_0=J_\infty+\overline{dJ}+2\sigma_{dJ}. \tag{20}$$

This choice of σ_{dJ} gives a 97.5 percent confidence level in terms of remaining below the allowed deviation E_0. Combining (19) and (20) we can derive an expression for Δ^2:

$$\frac{1}{\Delta^2}=\frac{1}{6E_0}\sqrt{\sum_{\substack{i=1\\ i<j}}^{m}\sum_{j=1}^{m}\left(\left.\frac{\partial^2 J}{\partial c_i\partial c_j}\right|_{\infty}\right)^2+\frac{1}{5}\sum_{i=1}^{m}\left(\left.\frac{\partial^2 J}{\partial c_i^2}\right|_{\infty}\right)^2}+\frac{1}{24E_0}\sum_{i=1}^{m}\left(\frac{\partial^2 J}{\partial c_i^2}\right). \tag{21}$$

Using (15), the SWL can then be written

$$\mathrm{SWL}=l+\frac{1}{2}\log_2\frac{1}{\Delta^2}. \tag{22}$$

The use of second-partial derivatives in approximating dJ in (17) has given rise to a complex expression for the statistical wordlength. Efficient methods for evaluating (22) will be discussed in the next section.

VI. Computational Procedure

In order to compute the derivates of J_∞, the infinite-precision (ideal) performance index, it is convenient to use the trace form [26] of (2):

$$J_\infty = \operatorname{trace}[S \quad Z] \tag{23}$$

where the $(2n+1)\times(2n+1)$ matrices S and Z are defined by (24) and (25):

$$S = \left[\begin{array}{c|c|c} Q & 0 & M \\ \hline 0 & 0 & 0 \\ \hline M' & 0 & R \end{array}\right] \tag{24}$$

$$Z = E\left\{ \begin{bmatrix} x(k) \\ v(k) \\ u(k) \end{bmatrix} [x'(k), v'(k), u'(k)] \right\}. \tag{25}$$

Here Q, M, and R are the performance index parameters described in (2). The matrix Z, the covariance matrix for plant and compensator states, can be shown to satisfy the following Lyapunov equation,

$$Z = AZA' + \begin{bmatrix} \Theta_1 & 0 \\ 0 & \Psi_{12}\Theta_2\Psi_{12}' \end{bmatrix} \tag{26}$$

where

$$A = \left[\begin{array}{c|c|c} \Phi & 0 & \Gamma \\ \hline \Psi_{12}L & \Psi_{11} & \end{array}\right]. \tag{27}$$

Note that (23)–(26) depend on the infinite-precision (ideal) compensator and on the selection of compensator state variables v. The resulting J_∞ will be independent of structure. However, the partial derivatives of J_∞ (evaluated for ideal coefficients) will depend on the structure since each coefficient c_i resides in one of the structure's Ψ_i matrices. Taking the partial derivatives of (23) will produce

$$\frac{\partial^2 J}{\partial c_i \partial c_j} = \operatorname{trace} S\left(\frac{\partial^2 Z}{\partial c_i \partial c_j}\right) \tag{28}$$

where all the partials in (28) are evaluated at the ideal values of the coefficients.

Thus, we must compute the second partials of Z. Taking the first derivative of (22) produces

$$\frac{\partial Z}{\partial c_i} = A\frac{\partial Z}{\partial c_i}A' + \tilde{Q}_i + \tilde{Q}_i' \tag{29}$$

where

$$\tilde{Q}_i = \frac{\partial A}{\partial c_i}ZA' + \begin{bmatrix} 0 & 0 \\ 0 & \dfrac{\partial \Psi_{12}}{\partial c_i}\Theta_2\Psi_{12}' \end{bmatrix}.$$

Evaluation of the trace expression in (28) will imply solving m Lyapunov equations of the form shown in (29). Now to compute the second partials, we must take the derivative of (29) with respect to c_j [23]:

$$\frac{\partial^2 Z}{\partial c_i \partial c_j} = A\frac{\partial^2 Z}{\partial c_i \partial c_j}A' + X_{ij} + X_{ij}' \tag{30}$$

where

$$X_{ij} = \frac{\partial A}{\partial c_j}\frac{\partial Z}{\partial c_i}A' + \frac{\partial A}{\partial c_i}\frac{\partial Z}{\partial c_j}A' + \frac{\partial A}{\partial c_i}Z\frac{\partial A'}{\partial c_j} + \frac{\partial^2 A}{\partial c_i \partial c_j}ZA' + \begin{bmatrix} 0 & 0 \\ 0 & \dfrac{\partial \Psi_{12}}{\partial c_i}\Theta_2\dfrac{\partial \Psi_{12}'}{\partial c_j} + \dfrac{\partial^2 \Psi_{12}}{\partial c_i \partial c_j}\Theta_2\Psi_{12}' \end{bmatrix}.$$

Solving (30) for all i and j would require $m(m+1)/2$ more Lyapunov solutions; this would be extremely time-consuming.

Fortunately, this burden can be substantially reduced. Specifically, the concept of adjoint operators [1], [23] can be used to simplify (28) and (30). The expression in (28) can be replaced by

$$\frac{\partial^2 J}{\partial c_i \partial c_j} = 2\operatorname{trace}(UX_{ij}) \tag{31}$$

where U satisfies $U - A'UA = S$. Thus, we need to solve this one Lyapunov equation plus the m equations in (29) in order to compute the X_{ij}. This saves solving the $m(m+1)/2$ equations of (30).

There is still the problem of the m Lyapunov solutions needed for the derivatives $\partial Z/\partial c_i$ used in X_{ij}. By using the Lyapunov solution method of Barraud [27], these computations can also be simplified. Consider the general Lyapunov equation (32),

$$X = FXF' + C. \tag{32}$$

Barraud's method breaks into two distinct parts, one which transforms F into the upper Schur form, and one which back substitutes using the transformed F and C matrices. The major portion of this computation involves the initial F transformation. Thus, if there exist several Lyapunov equations with identical F matrices but different C matrices, then the F transformation need be done only once. This is exactly the situation for the Lyapunov equations (26) and (29) needed for X_{ij}. Typically, more than 75 percent of the Lyapunov computation time can be saved, depending on the particular A matrix.

Still further computational time savings are possible. A more complete description of the computational procedure is available in [23].

VII. An LQG Example

A sixth-order example was chosen to test the statistical wordlength algorithm. It was adapted from the longitudinal control system design done for the $F8$ digital fly-by-wire fighter [28]. The continuous-time plant parameters and performance index parameters are given as follows.

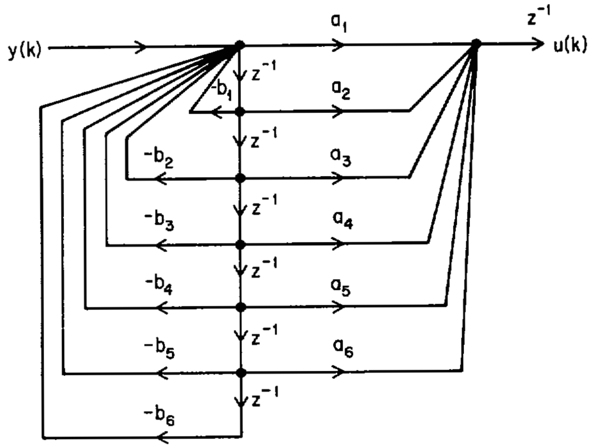


Fig. 3. Direct form II structure.

Continuous-Time System Parameters

$$A=\begin{bmatrix} -0.6696 & 5.7\times10^{-4} & -9.01 & 0 & -15.77 & 0 \\ 0 & -0.01357 & -14.11 & -32.2 & -0.433 & 0 \\ 1 & -1.2\times10^{-4} & -1.214 & 0 & -0.1394 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -12 & 12 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$B=\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$C=\begin{bmatrix} 1 & 0.003091 & 31.28 & 1 & 3.592 & 0 \end{bmatrix}.$$

Continuous-Time Performance Index Parameters

$$\hat{Q}=\begin{bmatrix} 6.637 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2.6554\times10^{-7} & 2.686\times10^{-3} & 0 & 3.085\times10^{-4} & 0 \\ 0 & 2.686\times10^{-3} & 27.174 & 0 & 3.121 & 0 \\ 0 & 0 & 0 & 27.174 & 0 & 0 \\ 0 & 3.085\times10^{-4} & 3.121 & 0 & 0.3585 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\hat{R}=5.252.$$

Continuous-Time Noise Covariances

$$\Xi_1=\operatorname{diag}\begin{bmatrix} 0 & 0 & 0 & 0 & 10^{-6} & 10^{-6} \end{bmatrix}$$

$$\Xi_2=0.0018441.$$

This continuous-time system was discretized at a sample rate of 10 Hz and the optimal regulator and Kalman filter designed. The double-precision parameters Φ, Γ, L, Q, M, R, Θ_1, Θ_2, G, and K can be found in [23].

Five structures for implementing the ideal compensator transfer function (6) were examined. These were the digital filtering-based direct form II structure, a cascade of direct form II second-order sections, a parallel structure composed of such sections, a block-optimal minimum roundoff noise structure, and the structure described in (7) which we have called the *simple* structure. In all five cases we present the initial design coefficient values. These are not typically the coefficients that are used however; if they were used the structures could exhibit overflows. Consequently, we apply a scaling procedure that we have adapted for compensators [23] from the l_2 scaling of digital filters [29]. In any case where a unity entry in the unscaled structure would become a multiplier coefficient (nonunity, nonpower of two) when scaled, we have indicated this with an asterisk.

The first structure we examine is the direct form II. Fig. 3 presents its signal flow graph. Note the presence of the delay preceding the output node: the 12 coefficients of the direct form II structure come directly from the unfactored transfer function (33), and its modified state space representation (two precedence levels) is shown in (34):

$$H(z)=\frac{a_1z^{-1}+a_2z^{-2}+a_3z^{-3}+a_4z^{-4}+a_5z^{-5}+a_6z^{-6}}{1+b_1z^{-1}+b_2z^{-2}+b_3z^{-3}+b_4z^{-4}+b_5z^{-5}+b_6z^{-6}} \tag{33}$$

$$\Psi_2\Psi_1=\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ a_6 & a_5 & a_4 & a_3 & a_2 & a_1 \end{bmatrix} \cdot \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ -b_6 & -b_5 & -b_4 & -b_3 & -b_2 & -b_1 & 0 & 1^* \end{bmatrix}. \tag{34}$$

When this structure is l_2 scaled, the unity valued marked with an asterisk becomes a 13th nonunity coefficient.

The second structure, the cascade, derives its coefficients from a multiplicative factorization (and there are several ways to group the poles and zeros [23]) of (33) and breaks into three series direct form II second-order sections. The factored transfer function is shown in (35). This structure has 12 coefficients, 4 precedence levels, and requires 3 additional scaling multipliers when l_2-scaled [see (36)] (details are available in [23]):

$$H(z)=\frac{(d_1z^{-1}+d_2z^{-2})(1+d_3z^{-1}+d_4z^{-2})(1+d_5z^{-1}+d_6z^{-2})}{(1+c_1z^{-1}+c_2z^{-2})(1+c_3z^{-1}+c_4z^{-2})(1+c_5z^{-1}+c_6z^{-2})} \tag{35}$$

$$\Psi_4=\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & d_6 & d_5 & 1^* \end{bmatrix}$$

$$\Psi_3=\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & d_4 & d_3 & -c_6 & -c_5 & 1^* \end{bmatrix} \tag{36}$$

$$\Psi_2=\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ d_2 & -c_4 & -c_3 & 0 & 0 & d_1 \end{bmatrix}$$

$$\Psi_1=\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ -c_2 & -c_1 & 0 & 0 & 0 & 0 & 0 & 1^* \end{bmatrix}.$$

The third structure, the parallel form, corresponds to a partial-fraction expansion of (33) and is divided into five parallel direct form II first- and second-order sections. The expanded transfer function (also 12 coefficients before scaling) is shown in (37), and its modified state space is given in (38):

$$H(z)=\frac{e_1z^{-1}+e_2z^{-2}}{1+c_2z^{-1}+c_2z^{-2}}+\frac{e_3z^{-1}}{1+d_3z^{-1}}+\frac{e_4z^{-1}}{1+d_4z^{-1}}+\frac{e_5z^{-1}}{1+d_5z^{-1}}+\frac{e_6z^{-1}}{1+d_6z^{-1}} \tag{37}$$

$$\Psi_2=\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ e_2 & e_1 & e_3 & e_4 & e_5 & e_6 \end{bmatrix} \tag{38}$$

$$\Psi_1=\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -c_2 & -c_1 & 0 & 0 & 0 & 0 & 0 & 1^* \\ 0 & 0 & -c_3 & 0 & 0 & 0 & 0 & 1^* \\ 0 & 0 & 0 & -c_4 & 0 & 0 & 0 & 1^* \\ 0 & 0 & 0 & 0 & -c_5 & 0 & 0 & 1^* \\ 0 & 0 & 0 & 0 & 0 & -c_6 & 0 & 1^* \end{bmatrix}.$$

To scale this structure, five additional scalers (one per section) are required [23].

The fourth structure tested was a parallel block optimal minimum roundoff noise structure analogous to the minimum roundoff noise filter structure discussed by Mullis and Roberts [29] and Hwang [30]. However, since the roundoff noise performance of an LQG control system depends on the overall closed-loop behavior, it was also necessary to adapt the techniques of Mullis and Roberts and Hwang for compensators [23]. This structure is reported to have low coefficient sensitivity when used as a filter even though it requires 25 coefficients, which is many more than the previous three structures. The modified state space is shown in (39), and has three parallel second-order sections and only one precedence level [23]:

$$\Psi_1=\begin{bmatrix} f_1 & f_2 & 0 & 0 & 0 & 0 & 0 & f_3 \\ f_4 & f_5 & 0 & 0 & 0 & 0 & 0 & f_6 \\ 0 & 0 & f_7 & f_8 & 0 & 0 & 0 & f_9 \\ 0 & 0 & f_{10} & f_{11} & 0 & 0 & 0 & f_{12} \\ 0 & 0 & 0 & 0 & f_{13} & f_{14} & 0 & f_{15} \\ 0 & 0 & 0 & 0 & f_{16} & f_{17} & 0 & f_{18} \\ f_{19} & f_{20} & f_{21} & f_{22} & f_{23} & f_{24} & 0 & f_{25} \end{bmatrix}. \tag{39}$$

This last structure, the simple structure, is taken directly from the LQG compensator equations (7). In other words, those equations exactly describe the computations that must take place. The parameters of Φ, Γ, K, L, and G are taken to be the coefficient values of the structure (before

TABLE I
SWL RESULTS FOR THE $F8$ EXAMPLE

Structure (equations)	l	SWL bits (time)	TWL bits (time)	Coefficients (incl. scaling multiplies)
Direct-II (33), (34)	16	35.99(0.81)	32(1.2)	13
Cascade (35), (36)	6	14.61(0.86)	14(1.36)	15
Parallel (37), (38)	1	6.84(0.93)	6(1.08)	17
Block optimal (39)	1	7.02(1.26)	7(1.11)	25
Simple (6), (40)	1	9.05(2.44)	9(1.71)	50

scaling). We have considered this structure because it has been used to implement LQG compensators more or less by default. The form of the transfer function containing the coefficients of this structure is that of (6), and the modified state space is shown below:

$$\Psi_3\Psi_2\Psi_1=\left[\begin{array}{c} I_6 \\ \hline -G \end{array}\right]\left[\begin{array}{c|c|c} \Phi & \Gamma & K \end{array}\right]\left[\begin{array}{c|c|c} I_6 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline -L & 0 & 1 \end{array}\right]. \tag{40}$$

Note the three precedence levels, and also the enormous number of coefficients—up to 60 for a sixth-order system.

Table I presents data comparing the statistical wordlength estimates of the five structures mentioned above to the actual required wordlength as computed by the direct, almost trial-and-error, approach mentioned in Section I. This actual value is called the "TWL" (true wordlength) in Table I, and was computed using a modified binary search algorithm [23]. The amount of computation time required for each SWL or TWL calculation is also included in parentheses.

For the system tested, a 5 percent degradation was specified as the maximum allowed deterioration in the measure of performance J. The wordlength values presented in Table I do not include a sign bit.

The effect of structure on coefficient wordlength is evident from Table I. For compensators, the most important observation to make is that although the simple structure performs fairly well in terms of its required coefficient wordlength, it is inefficient. It requires far too many multipliers compared to the parallel direct form II and parallel block optimal structures which also outperform it. Yet this structure has been commonly used.

Among the remaining structures shown, the direct form II requires the most bits by far, as is typical of digital filter applications [4]. The best structure of the five is clearly the parallel direct form II, requiring 6-bit words and 17 coefficients. In performance, the block optimal is nearly as good; however, it requires eight extra coefficients.

As an estimate, the SWL was from 0.02–0.84 bits conservative for the best four structures, which is extremely good, but 3.99 bits conservative for the direct form II. As a comparison, recall the digital filter results of Crochiere [22], in which the SWL, based on transfer function magnitude, was 1–3 bits conservative. In terms of execution time, the SWL exhibits a strong dependence on the number of coefficients in the structure. These times can be compared to the execution times for the TWL value, which should be relatively independent of the number of coefficients. Thus, the SWL is faster to compute when there are fewer than 20 coefficients, and slower to compute for more than 20. However, keep in mind that its main application is for the optimization of structures, where the TWL cannot be used in the same fashion.

VIII. CONCLUSIONS

This paper constitutes an attempt to examine the issues involved in the digital implementation of control compensators. To deal with these issues, we have sought to ally the fields of digital signal processing and control and estimation.

More specifically, this paper treats the statistical coefficient wordlength issue for the LQG compensator using fixed-point arithmetic. After reviewing the LQG design procedure and defining the notion of an implementation *structure*, the statistical wordlength concept for digital filters was described. In adapting this concept to a control and estimation problem, the index J was chosen, although the method readily extends to other measures (for example, the covariance matrix trace for Kalman filter problems). Finally, an efficient computational method was discussed and an illustrative example presented.

Our results demonstrate the feasibility of using the statistical approach in determining a sufficient LQG compensator coefficient wordlength. One application of this technique would be in the comparison of different structures for implementing a design. In addition, the statistical wordlength can also be an accurate criterion for selecting the wordlength once a specific structure is chosen.

Of more importance, the continuous nature and analytical form of the statistical wordlength estimate (it is not confined to an integral number of bits) make it possible to *synthesize* minimum coefficient wordlength structures in a straightforward manner. This would be extremely difficult and time-consuming with the nondifferentiable integer TWL. Using the statistical wordlength as described in Section IV, Chan [24] has presented a constrained optimization technique for digital *filter* design based on continuous transformations of an initial filter structure. Given a set of constrained and unconstrained coefficients in the Ψ_i matrices, the transformations are used to iteratively produce structures of lower and lower coefficient sensitivity, and thus smaller coefficient wordlengths. This idea has been adapted for the LQG compensator statistical wordlength estimate presented in this paper [23].

In addition, by computing the SWL estimate, we have available the various coefficient sensitivities $\partial^2J/\partial c_i\partial c_j$. By examining their relative values, we can determine the dominant sensitivities in the structure. This information can be exploited to direct any effort at optimizing wordlength [23] or *specializing* the hardware multiplier associated with any particular coefficient. Thus, we could optimize just one portion of a higher order structure, instead of the entire structure. This would save on the number of multiplies. Also, an examination of the different sensitivities opens up the possibility of using *different* wordlengths in different parts of the structure.

Other applications of the statistical coefficient wordlength estimate developed in this paper exist. As mentioned in Section V, this statistical procedure including second-order sensitivities can be used for digital filters that are designed through the optimization of some scalar criterion. Furthermore, in the control field, this statistical wordlength formulation would apply almost unchanged to suboptimal compensators designed via some parameter optimization approach. The need for second-order sensitivities would still exist.

As a final point, it should be mentioned that most of our development applies unchanged to multiple-input multiple-output compensators [23]. The difficulty there is in defining just how one develops structures for such compensators. However, given such a structure, we can easily compute its statistical wordlength by following the procedures described in this paper.

References

[1] H. Kwakernaak and R. Sivan, *Linear Optimal Control Systems.* New York: Wiley, 1972.

[2] M. Athans, Guest Ed., Special Issue on Linear-Quadratic-Gaussian Problem, *IEEE Trans. Automat. Contr.*, vol. AC-16, Dec. 1971.

[3] B. C. Kuo, *Analysis and Synthesis of Sampled-Data Control Systems.* Englewood Cliffs, NJ: Prentice-Hall, 1963.

[4] A.V. Oppenheim and R. W. Schafer, *Digital Signal Processing.* Englewood Cliffs, NJ: Prentice-Hall, 1975.

[5] L. R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing.* Englewood Cliffs, NJ: Prentice-Hall, 1975.

[6] T. A. C. M. Claasen, W. F. G. Mecklenbräuker, and J. B. H. Peek, "Effects of quantization and overflow in recursive digital filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-24, pp. 517–529, Dec. 1976.

[7] J. F. Kaiser, "On the limit cycle problem," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 1976, pp. 642–644.

[8] A. V. Oppenheim and C. J. Weinstein, "Effects of finite register length in digital filtering and the fast Fourier transform," *Proc. IEEE*, vol. 60, pp. 957–976, Aug. 1972.

[9] J. B. Knowles and R. Edwards, "Effect of a finite-word-length computer in sampled-data feedback systems," *Proc. Inst. Elec. Eng.*, vol. 112, pp. 1197–1207, June 1965.

[10] E. E. Curry, "The analysis of round-off and truncation errors in a hybrid control system," *IEEE Trans. Automat. Contr.*, vol. AC-13, pp. 601–604, Oct. 1967.

[11] J. E. Bertram, "The effect of quantization in sampled-feedback systems," *Trans. Amer. Inst. Elec. Eng.*, vol. 77, part 2, pp. 177–182, Sept. 1958.

[12] J. B. Slaughter, "Quantization errors in digital control systems," *IEEE Trans. Automat. Contr.*, vol. AC-9, pp. 70–74, Jan. 1964.

[13] G. W. Johnson, "Upper bound on dynamic quantization error in digital control systems via the direct method of Lyapunov," *IEEE Trans. Automat. Contr.*, vol. AC-10, pp. 439–448, Oct. 1965.

[14] G. N. T. Lack and G. W. Johnson, "Comments on 'Upper bound on dynamic quantization error in digital control systems via the direct method of Lyapunov,'" *IEEE Trans. Automat. Contr.*, vol. AC-11, pp. 331–334, Apr. 1966.

[15] A. B. Sripad, "Models for finite precision arithmetic, with application to the digital implementation of Kalman filters," Sc.D. dissertation, Washington Univ., Sever Institute, Jan. 1978.

[16] R. E. Rink and H. Y. Chong, "Performance of state regulator systems with floating-point computation," *IEEE Trans. Automat. Contr.*, vol. AC-24., pp. 411–421, June 1979.

[17] F. A. Farrar, "Microprocessor implementation of advanced control modes," in *Summer Comput. Simulation Conf. Proc.*, Chicago, IL, July 1977, pp. 339–342.

[18] A. S. Willsky, *Digital Signal Processing and Control and Estimation Theory—Points of Tangency, Areas of Intersection, and Parallel Directions.* Cambridge, MA: M.I.T. Press, 1979.

[19] E. Avenhaus, "On the design of digital filters with coefficients of limited word length," *IEEE Trans. Audio Electroacoust.*, vol. AU-20, pp. 206–212, Aug. 1972.

[20] J. B. Knowles and E. M. Olcayto, "Coefficient accuracy and digital filter response," *IEEE Trans. Circuits Syst.*, vol. CAS-15, pp. 31–41, Mar. 1968.

[21] R. E. Crochiere, "A new statistical approach to the coefficient word length problem for digital filters," *IEEE Trans. Circuits Syst.*, vol. CAS-22, pp. 190–196, Mar. 1975.

[22] R. E. Crochiere, "Digital network theory and its application to the analysis and design of digital filters," Ph.D. dissertation, Dep. Elec. Eng., M.I.T., Apr. 1974.

[23] P. Moroney, "Issues in digital implementation of control compensators," Ph.D. dissertation, Dep. Elec. Eng. Comput. Sci., M.I.T., Sept. 1979.

[24] D. S. K. Chan, "Theory and implementation of multidimensional discrete systems for signal processing," Ph.D. dissertation, Dep. Elec. Eng. Comput. Sci., M.I.T., May 1978.

[25] R. E. Crochiere and A. V. Oppenheim, "Analysis of linear digital networks," *Proc. IEEE*, vol. 63, pp. 581–595, 1975.

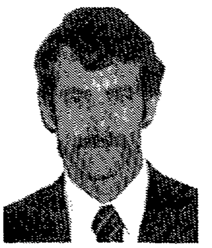
[26] G. K. Roberts, "Consideration of computer limitations in implementing on-line controls," M.I.T. Elec. Syst. Rep. ESL-R-665, Cambridge, MA, June 1976.

[27] A. Y. Barraud, "A numerical algorithm to solve $A^TXA - X = Q$," *IEEE Trans. Automat. Contr.*, vol. AC-22, pp. 883–885, Oct. 1977.

[28] A. E. Bryson, Jr., Guest Ed., Mini-Issue on the F-8 DFBW, *IEEE Trans. Automat. Contr.*, vol. AC-22, pp. 752–806, Oct. 1977.

[29] C. T. Mullis and R. A. Roberts, "Synthesis of minimum roundoff noise fixed-point digital filters," *IEEE Trans. Circuits Syst.*, vol. CAS-23, pp. 551–562, Sept. 1976.

[30] S. Y. Hwang, "Minimum uncorrelated unit noise in state-space digital filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-25, pp. 273–281, Aug. 1977.



Paul Moroney (M'79) was born in Auburn, NY, on August 27, 1952. He received the B.S., M.S., Engineers, and Ph.D. degrees from the Massachusetts Institute of Technology, Cambridge, in 1974, 1977 (2), and 1979, respectively.

From September 1974 until May 1976 he held positions as a Teaching Assistant, and then a Research Assistant at the Massachusetts Institute of Technology. From September 1977 until September 1979 he was a Draper Fellow at the Charles Stark Draper Laboratory, Cambridge, MA. He has held summer positions with Varian Associates (1973), NERComP, Inc. (1974), Watkins-Johnson (1976), and the Draper Laboratory (1977). He is currently with the Linkabit Corporation, San Diego, CA.

Dr. Moroney is a member of Eta Kappa Nu, Tau Beta Pi, and Sigma Xi.

Alan S. Willsky (S'70–M'73), for a photograph and biography, see pp. 359–360 of the June 1980 issue of this TRANSACTIONS.

Paul K. Houpt (M'67), for a photograph and biography, see p. 360 of the June 1980 issue of this TRANSACTIONS.