HYBRID SIMULATION OF KALMAN ALGORITHM IN A DIRECT DIGITAL CONTROL SYSTEM

Ming H. Kuo
Engineering Science Division
University of Wisconsin-Parkside
Kenosha, Wisconsin

ABSTRACT

A hybrid computer simulation for the interactive design of Kalman's control algorithm in a digital control system is described. Optimal control algorithm for a process system subject to both set-point changes and load disturbances is obtained using hybrid interactive search for optimization.

INTRODUCTION

The technology of applying digital computers to process control has developed rapidly in recent years. The hardware limitations and problems that plagued early installations have been mostly solved. With the advent of the minicomputer, it is becoming even more feasible to digitally implement some of the control functions such as on-line turning, adaptive control, dead-time compensation, and other similar functions in a process control system. Various control algorithms (1,2) in Z-domain have been proposed and implemented in the digital computer to accomplish the desired process performance. The formula used in the control algorithm are usually directly related to discrete transfer function of the process. However, when the process is subject to both set-point changes and load disturbance, the coefficients in the formula for the control algorithm can be obtained only if the nature of load disturbances can be formulated. Historically, the qualitative and quantitative features of the set-point changes and load disturbances were characterized in terms of simple waveforms like steps, ramps, etc., or in terms of statistical quantities such as mean, variance, power spectral density, etc. Modern treatment of the problem have continued to follow that approach, whereas worst case approaches are most common engineering solution for cases where load disturbances are unmeasureable and unpredictable. In this paper, a hybrid computer simulation is proposed for on-line, adaptive design of the control algorithm in a closed-loop, direct digital control system. The process is subjected to time-varying set-point changes and severe load disturbances. Previous studies of optimal control of process with load disturbances, using modern control theory, have relied almost exclusively on optimal control techniques to formulate and solve the problem(3). The optimal process performance has been obtained as a by-product of minimizing some contrived optimization functional. For example, linear-quadratic optimization theory has been well adapted in formulating and solving the noise-corrupted control systems. However, this type of functional minimization approach used in direct digital process control has two disadvantages: [1] it requires the solution of high-order, nonlinear matrix operation, [2] it requires a computing system with large memory capacity and high-speed operation to meet the real-time criterion. In this simulation an algebraic approach with simple parameter optimization techniques is proposed to simplify one-line design of the control algorithm.

DESIGN OF CONTROL ALGORITHM IN Z-DOMAIN

The general digital process control loop considered in this paper is illustrated in Fig. 1. The objective is to design the controller D(z) in Z-domain so that the desired loop performance is obtained. This analysis of the control loop proceeds in a manner analogous to the procedure for a continuous system. First the sampled error E(z) is given by

$$E(z) = R(z) - C(z) \tag{1}$$

The sampled output C(z) is given by

$$\begin{aligned} C(z) &= Z\{N(s)G(s)\} + D(z)\, Z\{H(s)G(s)\}E(z) \\ &= NG(z) + HG(z)D(z)E(z) \end{aligned} \tag{2}$$

Substituting the first equation for E(z) gives

$$C(z) = \frac{HG(z)D(z)E(z)}{1+HG(z)D(z)} + \frac{NG(z)}{1+HG(z)D(z)} \tag{3}$$

If the load disturbance N(s) is zero, this allows Equation (3) to be solved as the closed-loop pulse transfer function

$$G(z) = \frac{C(z)}{R(z)} = \frac{HG(z)D(z)}{1+HG(z)D(z)} \tag{4}$$

In order to design the controller D(z), an expression for HG(z) must be obtained in some manner and the desired loop-performance characteristics must be specified. Then, Equation (4) can be solved for D(z):

$$D(z) = \frac{1}{HG(z)}\,\frac{C(z)/R(z)}{1-C(z)/R(z)} \tag{5}$$

In classical process control problems, designers often ignored external disturbances altogether hoping that the final closed-loop system would resist the corruption effects of any disturbances that might occur. To those studies which did consider disturbances, it is always assumed that some prior knowledge about the characteristics of the disturbance or the disturbance functions N(s) be known. Unfortunately, in many cases algorithms designed by those assumptions perform well only when the load disturbances are insignificant as the whole control-loop variable is concerned. When the control-loop is subjected to severe load disturbance, the control algorithm must be dynamically adjusted to obtain an optimal performance. It has been suggested by R. E. Kalman (4) that instead of specifying C(z)/R(z), the control algorithm be subjected to specifications on the manipulated variable M(z) and the process performance C(z). Kalman's algorithm is more practically realistic when the process is under load changes. From Fig. 1, it can be seen that the process pulse transfer function HG(s) is the ratio of C(z) to M(z)

$$HG(z) = \frac{C(z)}{M(z)} = \frac{P(z)}{Q(z)} \tag{6}$$

Once P(z) and Q(z) are specified, the control algorithm D(z) can be derived in usual manner for the disturbance-free control loop:

$$D(z) = \frac{Q(z)}{P(z)}\,\frac{P(z)}{1-P(z)} = \frac{Q(z)}{1-P(z)} \tag{7}$$

Up to this point only the ideal situation in which one assumes the load disturbances can all be directly measured on-line or can be ignored is considered. In reality, the only quantities actually available for measurement are the current value C(z) of the process output, the current value R(z) of the set-point changes, and the output of the H(s). In this simulation a composite algebraic approach with simple optimization scheme is formulated to design the control algorithm. The manipulated variable M(z) is given by

$$M(z) = E_1(z)D_1(z) + E_2(z)D_2(z) \quad (8)$$

where $E_1(z)$, the sampled-error is given by

$$E_1(z) = R(z) - C(z) \quad (9)$$

and $D_1(z)$ is the Kalman's control algorithm given by Equation (7).

As illustrated in Fig. 2, another sampled-error $E_2(z)$ is measured which detects the difference between the disturbance-absorbing output C(z) and a disturbance-free model output $C_M(z)$.

Given mathematical model of the process G(s), $C_M(z)$ represents sampled output in disturbance-free condition:

$$E_2(z) = C(z) - C_M(z) \quad (10)$$

where $$C_M(z) = HG(z)M(z) \quad (11)$$

The design procedure can be summarized as follows. For the hypothetical ideal situation where the process is subjected to no load disturbance, the control algorithm $D_1(z)$ is implemented as Equation (7). An effective procedure for designing the overall controller is to first select an appropriate formula for $D_2(z)$. For example, a second-order control algorithm may be formulated as

$$D_2(z) = \frac{A_1(1-z^{-1}) + A_2Tz^{-1}}{(1-z^{-1})^2} \quad (12)$$

The coefficients A_1 and A_2 can then be designed by using simple optimization scheme such as relaxation method (5) for optimal process performance.

COMPUTER SIMULATION

As the proposed design procedure involve both modeling of the process G(s) and implementation of the control algorithm in Z-domain, it is evident that analog/digital hybrid computer simulation can serve as a powerful tool in the design and analysis of direct digital process control systems. In this simulation study a D.C. servo motor system is used as the process under control. The system is a Feedback servo modular unit, M550. The model of the process G(s) is simulated as a second-order dynamic system on Electronic Associates' MiniAC Analog Computer. Since any discrepancy on the accuracy of the model G(s) can be compensated by the control algorithm $D_2(z)$, only simple preimplemention steps were taken to identify the transfer function of the model. The severeness of the load disturbance is simulated by integrating a simple time function via D/A converter. Set-point changes are also generated via another D/A converter. Both $D_1(z)$ and $D_2(z)$ are implemented on a PDP-11/20 digital computer with 8K core memory, DEC tape, 16-channel A/D converter, 4 D/A converters, programmable clock, and digital interface unit. The hybrid interface units, which link two MiniAC analog computers with the PDP-11/20 digital computer consist of one homemade interface patch panel, a DEC K-series logic machine unit, and an M-series interface unit with console. The complete simulation diagram for this study is illustrated in Fig. 3.

The software available for the simulation is a newly developed interactive BASIC-like hybrid language with capability for assembly language linkages. Some of the specific features relevant to hybrid programming is summarized as follows:

1. Analog mode control statements
   ANALOG SET
   ANALOG HOLD
   ANALOG RESET

2. Hybrid interface statement
   INTERFACE A, B

   Note: when A=0 B represents contents of 8-bit digital logic output.
   when A≠0 A=represents D/A converter.
   B=represents contents for D/A conversion.

3. Interface function BIT(N)
   - i LET C=BIT(0)
     The decimal equivalent of the content on the console switch register will be read as variable C.
   - ii LET D=BIT(1)
     The decimal equivalent of the content on the digital logic input register will be read.
   - iii LET M=BIT(N)
     The decimal equivalent of the content on any memory location will be read.

4. A/D Conversion
   ADC SCAN A, B, C, ...
   ADB(N)

5. Clock interrupt
   CLOCK RATE, TICKS

6. PLOT X, Y.

Since hybrid computer installations vary considerably, the generality of hybrid programs is quite limited. Only digital computer flow chart is illustrated here (Fig. 4) to demonstrate the effectiveness of hybridization in the design and analysis of direct-digital control systems.

CONCLUSION

From the results of this study it can be seen that hybrid simulation of direct-digital control of a real process offer attractive advantages over either analog methods or pure digital simulation. Digital implementation of control algorithms, sampling, and storage provides a close-to-real-world simulation of a digital control system. Modeling of the process transfer function on the analog computer allows designers to study the whole system on a small scale basis yet leads to results which are more attractive from the practical design point of view.

REFERENCES

1. Mosler, H. A., L. B. Koppel, and D. R. Coughanowr, "Process Control by Digital Compensation," A.I.Ch.E.J., Vol. 13, No. 4, p. 768, 1967.

2. Dahlin, E. B., "Designing and Tuning Digital Controllers" Instruments and Control Systems, Vol. 41, No. 6, p. 77, 1968.

3. Kuo, B. C., Analysis and Synthesis of Sampled-Data Control Systems, Prentice-Hall, Englewood Cliffs, N.J., 1968.

4. Kalman, R. E., "Sampled-Data Processing Techniques for Feedback Control Systems," Trans. AIEE, 1959.

5. Kuo, M. H., et al, "System Modeling and Structure Optimization Using Hybrid Computer Techniques," Proc. Seventh Allerton Conference on Circuit and System Theory, Oct. 1969.

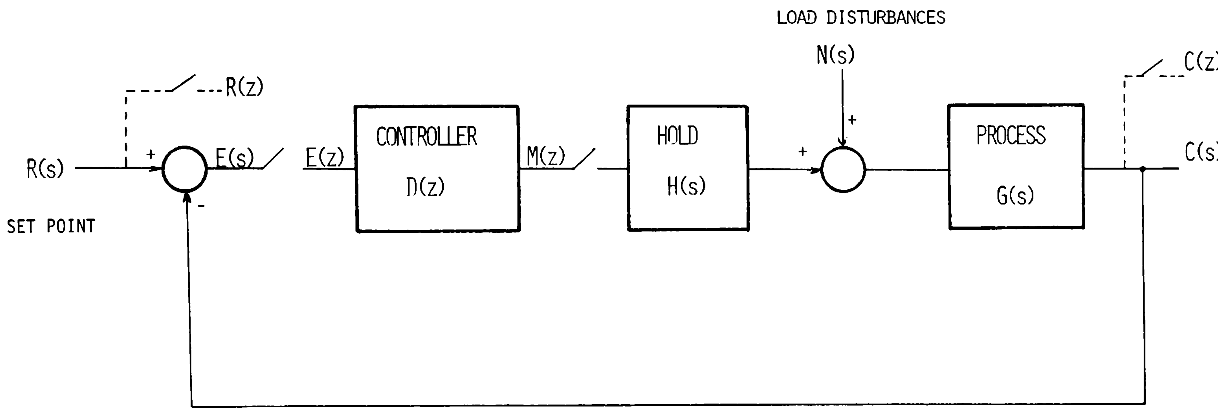

Fig.1 Block Diagram of a Direct Digital Control System.

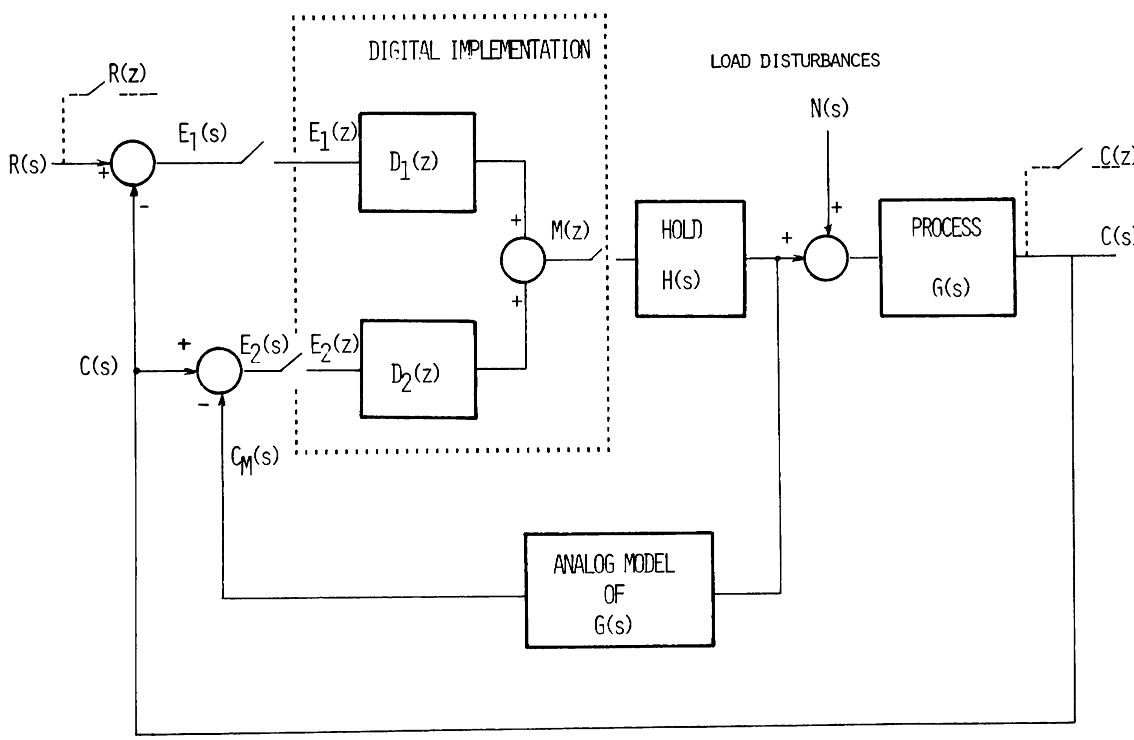

Fig. 2 Block Diagram for The Proposed Algebric Control Algorithm.

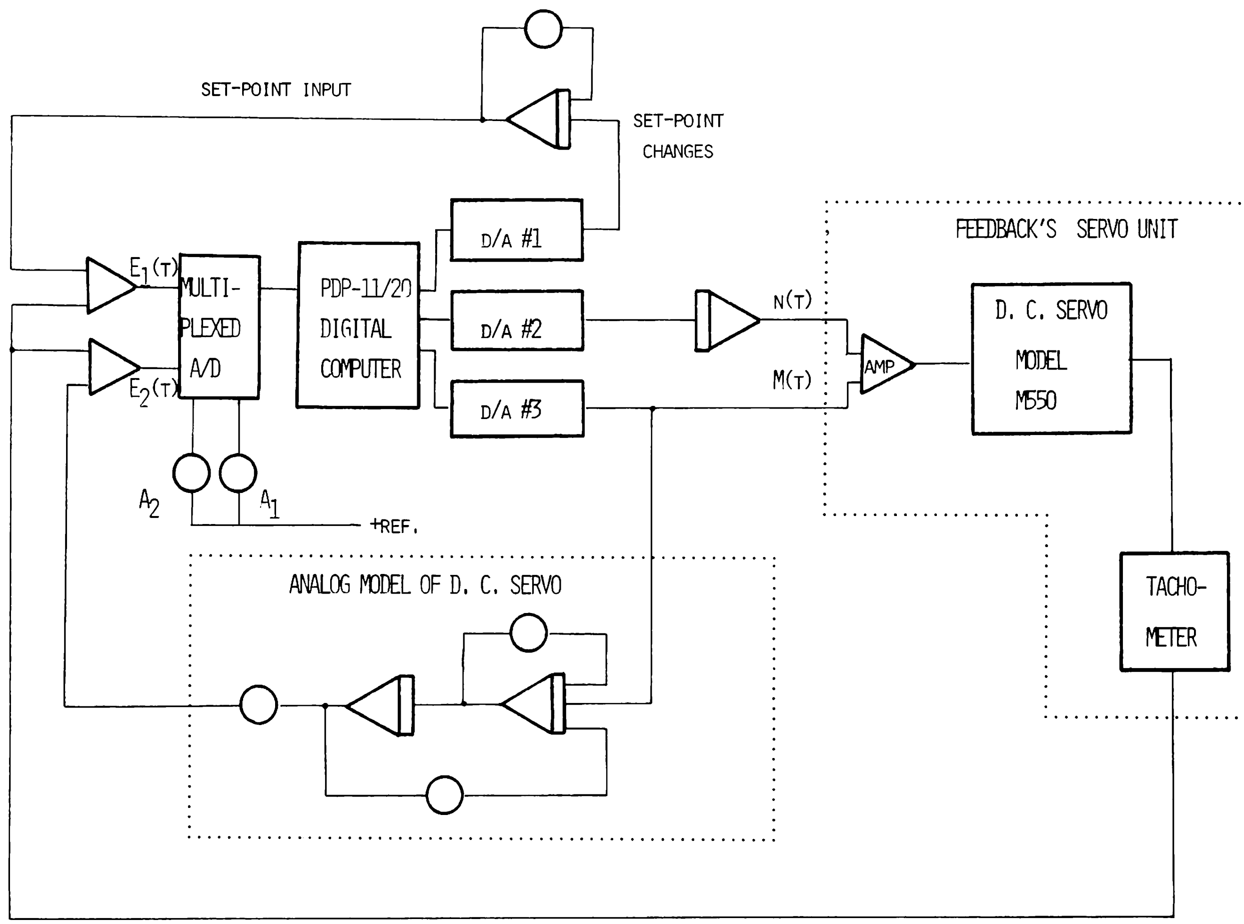

Fig. 3 Simulation Circuit Diagram for the Direct Digital Control System.

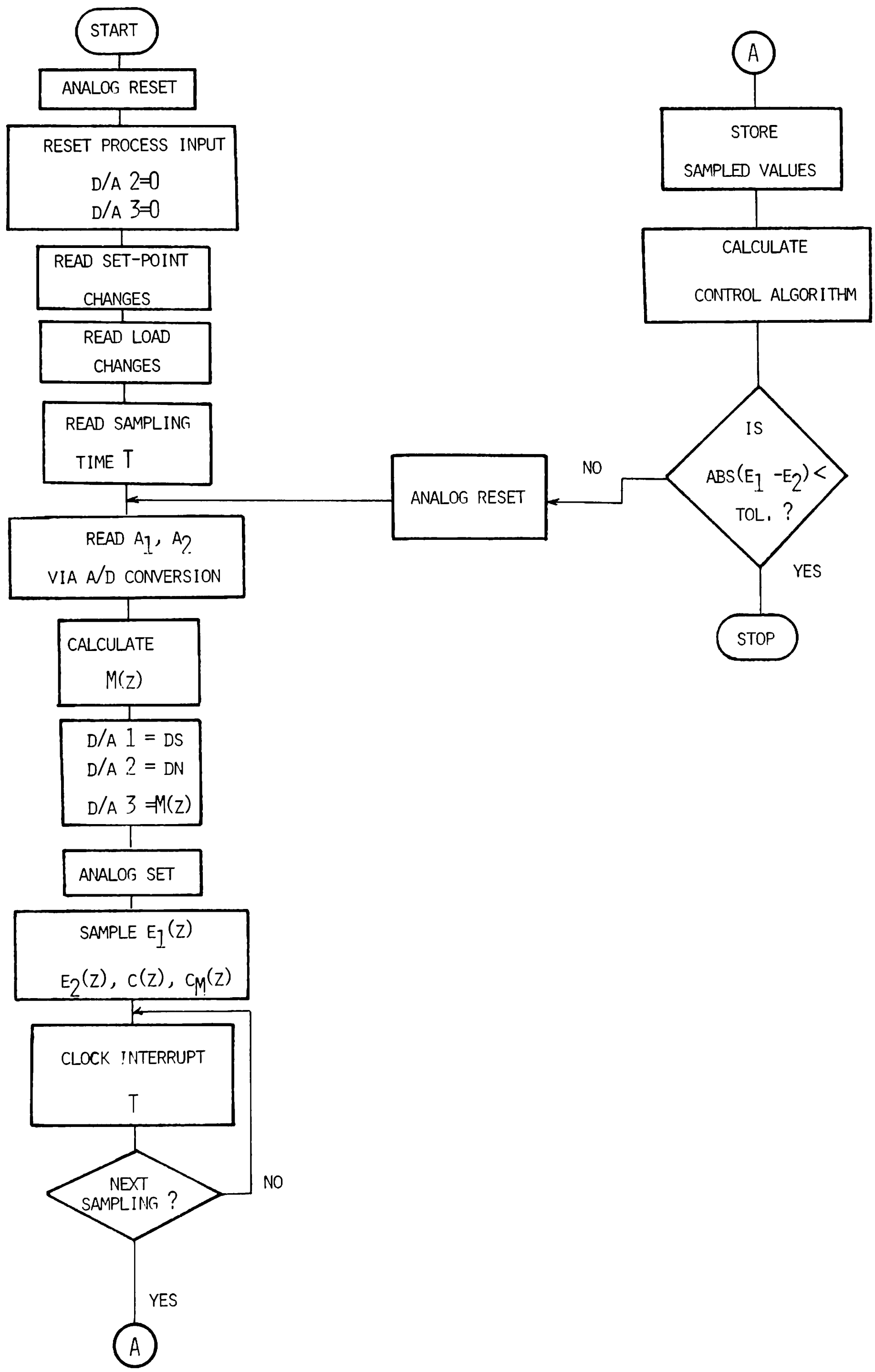

Fig. 4 Flow Chart for the Hybrid Simulation of Direct Digital Control System.