The effect of digital compensation for computation delay in a hybrid loop

by EWART EDWARD LESLIE MITCHELL
Electronic Associates Inc.
Princeton, New Jersey

INTRODUCTION

Recent interest in hybrid computation has focussed attention on the errors incurred because of the method of solving the problem equations. In the past some attention had been given to errors incurred due to analog component accuracy or round off error in a digital solution. However, with the advent of hybrid computation and the knowledge that the sampling that had to occur at the interface would perturb the solution, a number of papers have been published to define the problem and propose solutions. Probably the reason for the stimulated interest is that the errors can be obtained analytically for a number of simple systems. The proposed solutions to the sampling delay have shown an order of magnitude increase in the accuracy of the selected problems to which they were adapted.

It is the purpose of the paper, however, to show that in general it is *not* possible to compensate for computation delay with a digital filter unless the basic damping of the system under study remains relatively constant over the duration of the problem or the sample interval can be held to 30 or more per cycle.

Miura and Iwata[1] considered the effect of digital execution time and showed that the delay would influence the poles of the system under study, obtaining expressions for the amount of the shift for a number of simple systems. To compensate for the digital execution time they suggested three compensation techniques:

(a) Assuming that the digital output is immediately integrated, the output of the integrator is modified by adding $(\tau + T/2)$ of the input. Here τ is the delay due to the digital computation and T is the hold time at the output, where normally $T = \tau$. See Figures 1 and 2.

(b) The digital computer can predict ahead by linear interpolation, i.e.,

$$y = y'_j + (\tau + T/2)\frac{(y'_j - y'_{j-1})}{T}$$

(c) The analog first order hold can be implemented in order to avoid the staircase output and provide the necessary lead. Here y'_j is the computed output of the digital computation at the j^{th} time step; y_j is the output compensated for the delay.

Figure 1—Definition of time T and τ

Figure 2—Compensation by modified integration (Miura & Iwata)

Of these techniques the first one is the only one that doesn't increase the order of the system. The other two introduce extra roots into the system equations by using past history, which can be detrimental to solution accuracy under certain conditions.

Karplus[2] has also examined the effect of digital execution time on the solution error. In addition to the compensation techniques specified above, he also suggets modifying the *input* to the digital computer—if it is the result of an integration—by $(\tau + T/2)$ times the derivative. In a sense this is equivalent to (a) above and does not affect the order of the system.

Gilbert[3] also suggests modification of the input to the digital computer when the derivatives are known in a similar manner as above. Modifying the input to the digital section avoids discontinuities in the outputs of the analog integrators. Jumps occur when the compensation of Miura and Iwata is used.

The evaluation suggested by both Karplus and Gilbert for the compensation technique is accomplished by observing the growth or decay of a simple sine wave oscillator, which is very sensitive to phase shift. Unfortunately, these compensation techniques work very well for lightly damped systems; it is the heavily damped systems that produce the big errors.

Matlock[4] extends the digital prediction technique to higher order filters to better compensate for the phase shift. These higher order filters introduce extra roots into the system which seriously affect the performance of *heavily* damped systems. This digital prediction technique has also been utilized by Deiter and Nomura[5] who obtain the prediction coefficients by Gregory-Newton extrapolation. They evaluate the technique by measuring the integral error in the representation of a sine wave oscillator. Again the choice of a lightly damped system for evaluation leads to false confidence in the technique.

Our experience with predictive filters came about a year ago when we applied the scheme to a hybrid helicopter simulation. The forces and moments are computed on the digital section and integrated on the analog. To stabilize the short period loops we included 2nd order (quadratic) prediction in the moment equations. The first evaluations—at hover—indicated that the match with the real world was extremely close. Unfortunately, when we approached the top speed, the tail started to wag, at a frequency about eight times higher than expected, and the system went unstable. When the prediction subroutine was removed, the instability was also eliminated and left us with a simulation that had frequency and damping still reasonably close to the actual vehicle.

The explanation lies in the extra roots introduced by the prediction technique and the fact that the natural air frame is heavily damped at high speed. We will show that the application of prediction to a heavily damped system can lead to gross errors in the simulation when the extra roots move into the region of interest—near the unit circle in the z-plane or the $j\omega$ axis in the s-plane.

To overcome the difficulty we had to adopt a technique similar to that proposed by Gelman[6] or Connelly.[7] In essence a simple analog model is built up to approximate the system as closely as possible, and the digital computer is used to determine errors in the analog model. Provided these errors remain small correction terms, then the major feedback loops are continuous and adequate simulation accuracy can be maintained. Adequate compensation should also have been obtained using the input compensation of Karplus or the first technique (a) suggested by Miura and Iwata, since the order of the system is not modified.

In order to demonstrate the difficulties inherent in digital prediction, consider the simulation of a simple second order system and let us examine the effects of different predictive filter coefficients and damping factors on the roots of the system equations. Naturally, actual systems will be of higher order and contain nonlinearities, but significant trends can be unearthed by treating the simplest possible configuration.

Problem statement

In order to represent a physically meaningful system let us consider the simulation of vehicle pitch plane dynamics, where the velocity vector is held constant. The pitching acceleration, M, is assumed a function of pitch rate, Q and angle of attack, α. The differential equations to simulate the motion are:

$$\dot{Q} = M$$

$$\dot{\alpha} = Q$$

Note that we have absorbed the moment of inertia into the moment M, so that M has the dimensions rad/sec^2 rather than the conventional ft-lbs.

Our digital computer is programmed to compute M, and we will assume that this is linearly related to the problem state variables. In actual practice, the computation of the forces and moments requires almost all the available digital computation time; but for the purposes of analysis we will consider perturbations about a steady state.

Expressing

$$M = M_Q Q + M_\alpha \alpha$$

the state equations become

$$\begin{pmatrix} \dot{Q} \\ \dot{\alpha} \end{pmatrix} = \begin{pmatrix} M_Q & M_\alpha \\ 1 & 0 \end{pmatrix} \begin{pmatrix} Q \\ \alpha \end{pmatrix}$$

The frequency and damping for the unforced continuous system are obtained from the eigenvalues of the state matrix or roots of:

$$\lambda^2 - M_Q\lambda - M_\alpha = 0$$

so that

$$2\zeta\,\omega_n = -M_Q$$

$$\omega_n^2 = M_\alpha$$

where ζ is the damping and ω_n the natural frequency of the undamped system.

In the s-plane the roots are at $M_Q/2$; $\pm\ [M_Q^2/4 + M_\alpha]^{1/2}$ and it will be our objective to determine how close the roots of the compensated system approach these desired values.

Now to compensate for the sampling effects at the analog-digital interface we will apply up to quadratic prediction to the computed M by taking a linear combination of the current value and past history—up to 3 time steps ago. See Figure 3.

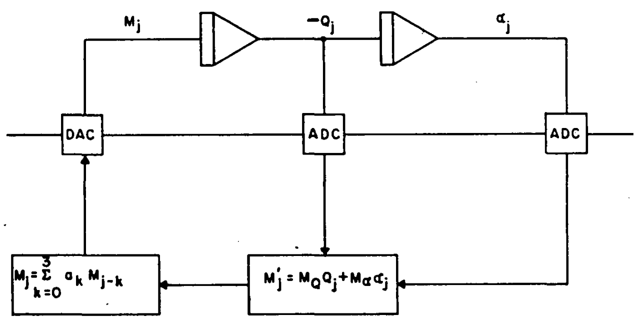

Figure 3—Flow diagram for hybrid damped sine-wave oscillator

i.e.,

$$M_j = \sum_{k=0}^{3} a_k M'_{j-k}$$

where

$$M'_j = M_Q Q_j + M_\alpha \alpha_j$$

and j is the time step count, incrementing in the sample time, T.

We want to consider the effect of various values of the predictor parameters $a_0 - a_3$ on the solution. The following interpretation can be made for the a's:

(a) $$a_0 = 1$$
$$a_1 = a_2 = a_3 = 0$$

The system corresponds to the case where the digital computation time is zero, the error being introduced by the necessity to hold the output constant over the sampling time, T.

(b) $$a_0 = 0$$
$$a_1 = 1$$
$$a_2 = a_3 = 0$$

This condition is observed in most current hybrid simulations where the computation time is equal to the sample time. Thus, the output, M, is delayed from the two inputs by one complete cycle.

Other coefficients may be obtained by solution of the set of simultaneous equations given in the appendix. The essential requirement of the prediction is the attempt to recover the lost 1.5 time steps ($=\alpha$) due to the combination of digital computation time and sample hold.

From Figure 3, the basic state equation can be expanded to develop the transition matrix that transfers the state from one time step to the next.

$$Q_{j+1} = Q_j + \int_0^T M_j dt = Q_j + M_j T$$

$$\alpha_{j+1} = \alpha_j + \int_0^T Q\, dt = \alpha_j + Q_j T + \tfrac{1}{2} M_j T^2$$

Substituting for M_j, where

$$M_j = \sum_{k=0}^{3} a_k M'_{j-k}$$

&

$$M'_j = M_Q Q_j + M_\alpha \alpha_j$$

leads to:

$$\begin{bmatrix} Q_{j+1} \\ Q_j \\ Q_{j-1} \\ Q_{j-2} \\ \alpha_{j+1} \\ \alpha_j \\ \alpha_{j-1} \\ \alpha_{j-2} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & a_{16} & a_{17} & a_{18} \\ 1 & & & & & & & \\ & 1 & & & & & & \\ & & 1 & & & & & \\ a_{41} & a_{42} & a_{43} & a_{44} & a_{45} & a_{46} & a_{47} & a_{48} \\ & & & & 1 & & & \\ & & & & & 1 & & \\ & & & & & & 1 & \end{bmatrix} \begin{bmatrix} Q_j \\ Q_{j-1} \\ Q_{j-2} \\ Q_{j-3} \\ \alpha_j \\ \alpha_{j-1} \\ \alpha_{j-2} \\ \alpha_{j-3} \end{bmatrix}$$

where

$$\begin{aligned} a_{11} &= 1+a_0M_\alpha T & a_{41} &= T+a_0M_QT^2/2 \\ a_{12} &= a_1M_QT & a_{42} &= a_1M_QT^2/2 \\ a_{13} &= a_2M_QT & a_{43} &= a_2M_QT^2/2 \\ a_{14} &= a_3M_QT & a_{44} &= a_3M_QT^2/2 \\ a_{15} &= a_0M_\alpha T & a_{45} &= 1+a_0M_\alpha T^2/2 \\ a_{16} &= a_1M_\alpha T & a_{46} &= a_1M_\alpha T^2/2 \\ a_{17} &= a_2M_\alpha T & a_{47} &= a_2M_\alpha T^2/2 \\ a_{18} &= a_3M_\alpha T & a_{48} &= a_3M_\alpha T^2/2 \end{aligned}$$

Now we can see that by adding in the prediction—or compensation—terms, we have increased the order of the system under study from 2nd to 8th, at first glance, for the quadratic predictor. However, the rank of the matrix is only 5, so just three extra roots are introduced. The eigenvalues of this transfer matrix correspond to poles in the z-plane for the sampled system. It is immediately obvious that for stability, all the eigenvalues must lie within the unit circle, otherwise successive application of the matrix will cause (some) solutions to increase without bound.

This eigenvalue equation can be reduced to:

$$[1 + a_0M_QT - a_0M_\alpha T^2/-(2 + a_0M_\alpha T^2/2)\lambda + \lambda^2]\lambda^6 + T[(1-\lambda)M_Q - (1+\lambda)M_\alpha T^2/2]\,[a_1\lambda^2+a_2\lambda + a_3]\lambda^3 = 0$$

Three of the eigenvalues are zero, but we are left with a fifth order polynomial or three extra roots. The characteristic polynomial can be rearranged to:

$$\lambda^5 + b_1\lambda^4 + b_2\lambda^3 + b_3\lambda^2 + b_4\lambda + b_5 = 0$$

where

$$\begin{aligned} b_1 &= -a_0\gamma - 2 \\ b_2 &= -a_1\gamma + a_0\zeta + 1 \\ b_3 &= -a_2\gamma + a_1\zeta \\ b_4 &= -a_3\gamma + a_2\zeta \\ b_5 &= a_3\zeta \end{aligned}$$

and

$$\gamma = T(M_Q+M_\alpha T/2)$$

$$\xi = T(M_Q-M_\alpha T/2)$$

We can show that for $T << 1$, the three extra roots move into the origin as well, leaving the two that correspond to the original system near the +1 point.

Transforming these to the s-plane by:

$$s = \frac{1}{T}\log_e z$$

results in roots at:

$$s = M_Q/2 \pm [M_\alpha + M_Q^2/4]^{1/2} + 0(T)$$

so that we have a check that the sampled system corresponds to the original continuous system for small sample times.

Our object now is to study the behavior of the root loci obtained by varying T for different prediction parameters a_0 - a_3 and for varying damping M_Q. We have kept $M_\alpha = 1$ corresponding to a natural frequency of $1/2\pi$ cycles/sec; and on all the root loci, sample time, T, is expressed as samples/cycle.

Results

Four cases of practical interest were examined and applied to systems that had inherent high and low damping. It would appear from this that at low sampling rates, the extra roots introduced by the prediction can cross the imaginary axis and become unstable, especially at high damping ($M_Q = 1.0$).

Case I: Figure 4 shows the root locus for the system,

$$a_0 = 1$$
$$a_1 = a_2 = a_3 = 0$$

for the two values of damping $M_Q = 0.2$ and 1.0. These loci correspond to the unrealized case of zero digital computation time; but since it is the only one where only two roots are maintained, can be used for comparison. The root loci are plotted on the s-plane because the variation of T causes the natural system roots to move on the z-plane. Crossing the unit circle into instability still corresponds to crossing the imaginary axis. Placing an arbitrary bound of 10% round the roots shows that we need about 160 samples/cycle when $M_Q = .2$ and 21 samples/cycle when $M_Q = 1.0$ to maintain the system simulation to this accuracy.

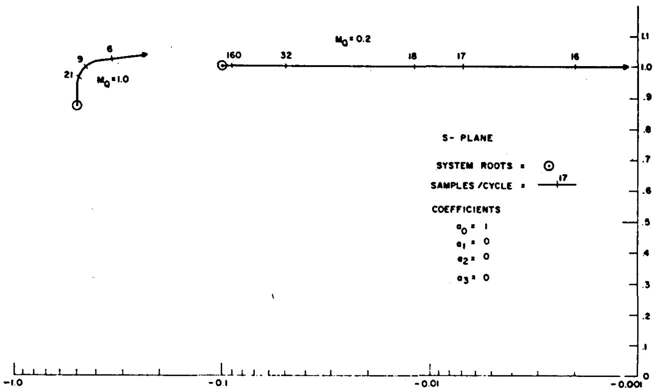

Figure 4—Root locus—no computation delay

Case II: No prediction but digital computation time corresponds to sample time, T, i.e.:

$$a_0 = 0$$
$$a_1 = 1$$
$$a_2 = a_3 = 0$$

Figure 5 shows the root loci for the same two damping values. An extra root appears, but remains well up to the left on the real axis while the system roots move toward the imaginary axis and instability. Now 520 and 64 samples/cycle are needed to keep the roots within 10% of the correct position for $M_Q = .2$ and 1.0 respectively.

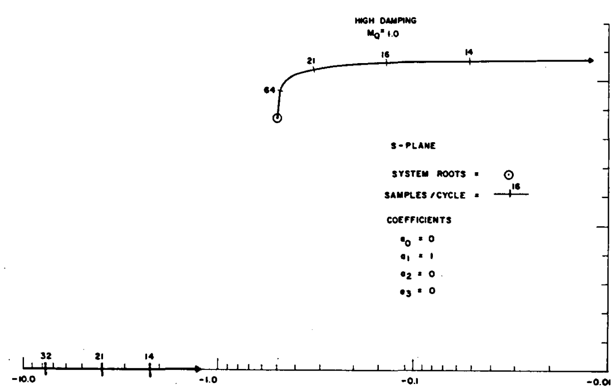

Figure 5(a)—Root locus—1 cycle computation delay

Case III: Applying the simple linear predictor to overcome the delay:

$$a_0 = 0$$
$$a_1 = 2.5$$
$$a_2 = -1.5$$
$$a_3 = 0$$

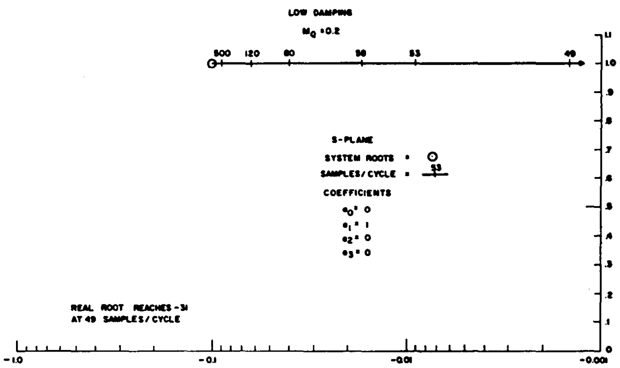

Figure 5(b)—Root locus—1 cycle computation delay

Figure 6 shows the corresponding root-loci. Of note now is the appearance of a second pair of roots since the system polynomial is now fourth order. For the low damping case, $M_Q = .2$, these extra roots stay over on the lefthand side, while the system roots move to the right, toward the stability limit. However, when we make $M_Q = 1.0$, so the basic system is heavily damped, the system roots move to the left; and the extra roots move to the right crossing the stability limit at 13 samples/cycle and producing an oscillation at 2.7 rads/sec.

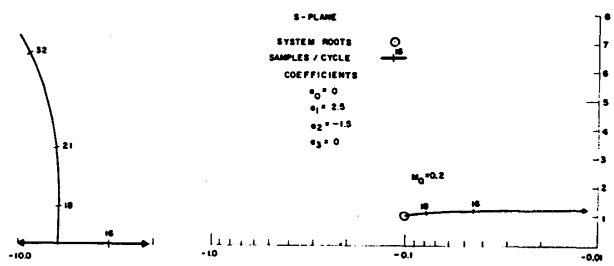

Figure 6(a)—Root locus—first order predictive compensation

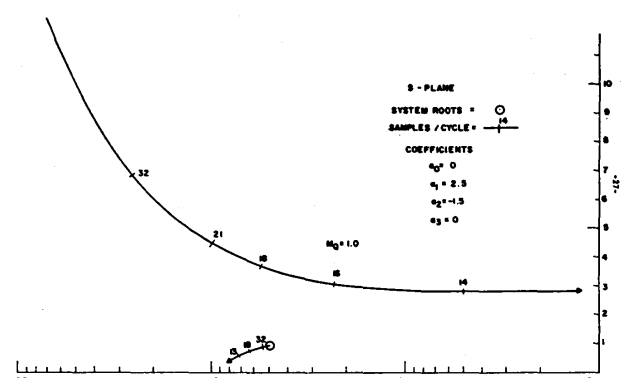

Figure 6(b)—Root locus—first order predictive compensation

Case IV: Using the quadratic filter to predict ahead for 1.5 sample times, the coefficients become:

$$a_0 = 0$$
$$a_1 = 4.375$$
$$a_2 = -5.25$$
$$a_3 = 1.875$$

Figure 7 shows the corresponding root loci. The five roots provide three extra, two imaginary and one real. In both cases the stability limit is exceeded by the extra root pair crossing the imaginary axis—at 10 samples/cycle when $M_Q = .2$ or at 28 samples/cycle when $M_Q = 1.0$.

Note that prediction controls the principal roots in the high damping case in such a way that the movement is negligible.

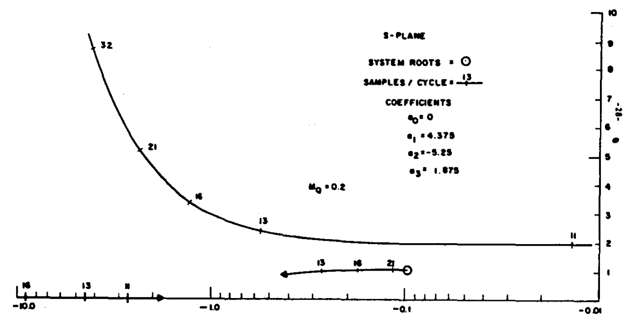

Figure 7(a)—Root locus—second order compensation

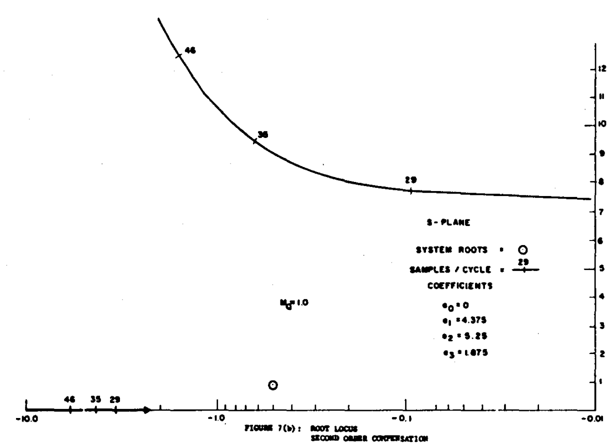

Figure 7(b)—Root locus—second order compensation

CONCLUSIONS

The application of predictive compensation to lightly damped systems, as suggested in the references, can result in accurate simulations for as low sampling rates as 10 per cycle. However, it must be pointed out that the important criteria for validity is whether we can apply this compensation to highly *stable* systems. In

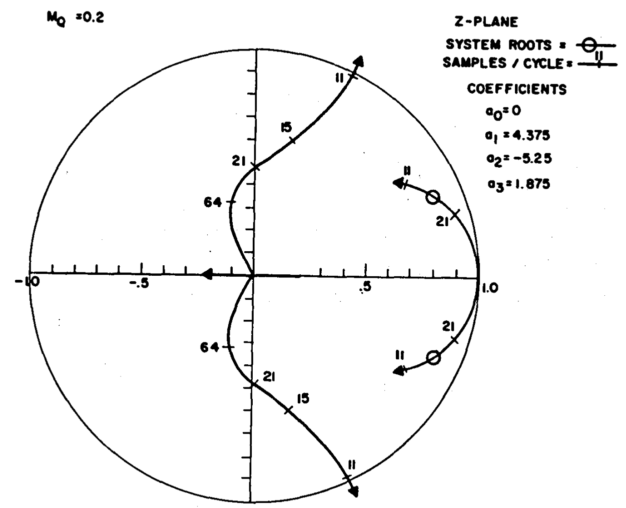

Figure 8—Root locus in Z-plane of second order compensation

fact, the majority of physical systems that come before the simulation engineer will be highly stable, since a normal design objective is to ensure suitable frequency response and adequate damping.

If digital compensation is to be applied to several systems—for instance as part of a software library—then no higher order than the first need be used and an adequate sampling rate must be maintained to keep the roots due to the compensating polynomial away from the region of interest. With unity damping ($M_Q = -2.0$) even first order prediction requires 25 samples/cycle for stability, and at least 30/cycle would be necessary for an adequate simulation. Basically, this is a restatement of the old requirement—that hybrid computation requires the fastest digital computation to maintain an adequate simulation performance.

A characteristic of these extraneous roots introduced by the prediction technique is the speed with which they move into the region of interest. That is, changing the sampling rate by 1 or 2 samples/cycle can make all the difference. Conversely, with a fixed sampling interval, no warning is given that stability limits are being approached, since a small change in natural frequency can cause the oscillation to appear.

Appendix: Filter Coefficients

Munroe[8] has shown that the coefficients $a_0 - a_n$ to pass a polynomial of degree q without distortion can be obtained from solution of the constraint equations:

$$\sum_{k=0}^{N} a_k = 1$$

$$\sum_{k=0}^{N} k\ a_k = -\alpha$$

$$\vdots$$

$$\sum_{N}^{k=0} k^q a_k = (-1)^q \alpha^q$$

where α is the number of time steps extrapolated—fraction or integer, i.e.,

$$M_{j+\alpha} = \sum_{k=0}^{N} a_k M_{j-k}$$

If α is negative, then an interpolation is obtained which is useful for data smoothing or reconstruction. If $q = N$, then the a's are completely defined, but if $q<N$, then the a_k's are chosen to minimize $\Sigma\, a_k^2$ which is the noise amplification by the filter.

REFERENCES

1 T MIURA J IWATA
Effects of digital execution time in a hybrid computer
Proc FJCC 24 251-66 1963

2 W J KARPLUS
Error analysis of hybrid computer systems
Simulation 6 121 1966

3 E G GILBERT
Dynamic error analysis of digital and combined analog-digital computer systems
Simulation 6 241 1966

4 D L MATLOCK
Pulse prediction filters applied to digital and hybrid simulation
Simulation 6 163 1966

5 R M DEITERS T NOMURA
Circle test evaluation of a method of compensating hybrid computing error by predicted integral
Simulation 8 33 1967

6 R GELMAN
Corrected inputs—a method for improved hybrid simulation
Proc FJCC 24 267-276 1963

7 M CONNELLY O FEDOROFF
A demonstration hybrid computer for real time flight simulation
AMRL-TR-65-97

8 A J MUNROE
Digital processes for sampled data systems
Wiley 1962