CORRECTED INPUTS—A METHOD FOR IMPROVED HYBRID SIMULATION

Robert Gelman
General Electric Company
Re-Entry Systems Department
Philadelphia 1, Pennsylvania

I. INTRODUCTION

The purpose of this paper is to describe a programming procedure which is designed to minimize some of the difficulties often encountered in hybrid simulations. The method has been worked out for the hybrid system at General Electric’s Re-Entry Systems Department, in Philadelphia. This system consists of general purpose analog and digital machines, connected by analog to digital (A-D) and digital to analog (D-A) converters, and associated logical elements. The procedure itself will be referred to as the method of corrected inputs. It is designed to minimize three major sources of error connected with data transfer. These are:

1. D-A Data
   Due to the nature of the digital computer, each variable which it sends to the analog is seen by the analog as a stepped function. This introduces errors because of the differences between this function and the actual one, and also because of the response of the analog to discontinuous inputs. Filtering this input can smooth the discontinuities somewhat, but introduces lags and distortions.

2. A-D Data
   The analog output is sampled and sent to the digital computer no more than once during each digital computation interval. If the digital time step is larger than, or even a significant portion of, the fundamental frequency of the analog output, then the A-D data could actually be a very poor representation of the analog outputs.

3. Time Lags
   The minimum time required for an output from a subsystem to affect the operation of the subsystem itself would be the sum of the A-D and D-A time lags. This could render hybrid operation very difficult, if not completely useless, in studies of the performance and stability of control systems, or other studies involving leads, lags, or phase relationships.

The method of corrected inputs provides a simple way of minimizing all these difficulties. The basic idea behind the method is to represent, on the analog, some of the functions which are also calculated on the digital computer. The outputs of the digital simulation would then be used, not as inputs to the main analog simulation, but as corrections to the comparable analog portion.

II. DESCRIPTION OF METHOD

The following is a brief summary of the system. The timing is arranged so that the digital computations lag the analog by the digital computation interval. Those inputs to
the main analog program which are sensitive to phase, delays, frequency response, or similar factors, are obtained from subsections of the analog program. These inputs are sampled and sent to the digital computer, where they are stored until the digital simulation produces the corresponding quantities, presumably with more accuracy. The differences are then returned to the analog computer which generates corrections, in the form of ramp functions, to the analog inputs. The only requirement imposed on the analog subsystems involved is that the analog functions not diverge appreciably from the comparable digital functions in less than two digital computation intervals.

The analog inputs to the digital program are not sent directly, unless they are slowly varying functions. Otherwise, each one is averaged over the time period of one computation interval, and this average is sent to the digital. The fact that the digital calculations lag the analog enables the digital computations to be made using inputs from the analog which are the average values for the period of time covered by the calculations. Thus, the method minimizes the three major drawbacks to accuracy in hybrid simulations: data transfer time lags, discontinuous D-A signals, and unrepresentative A-D signals.

The manner in which this method can be used to meet the basic difficulties described above can best be understood if it is explained in more detail. This is done with the aid of figures 1, 2, and 3.

Figure 1 is a schematic representation of a hybrid simulation. The box $F_1$ represents, let us say, the low frequency portion of the problem.
lem, solved on the digital computer, and box $f_2$ represents the high frequency portion, simulated on the analog. Output b is shown feeding back into $f_2$, in order to illustrate the possibility that the behavior of $f_2$ can depend on high frequency as well as low frequency inputs. Note that all digital functions are represented by capital F's, analog functions by small f's, digital variables by capital letters, and analog variables by small letters. When the same letter is both capitalized and lower case, it represents either a variable that is converted A-D or D-A, or a function which is simulated on both computers.

Figure 2 shows the simulation of figure 1, as modified for the method of corrected inputs. The modified circuit is not as much more complicated than the original one as these diagrams would make it appear. The analog function, $f_a$, consists of two amplifiers; $f_t$ consists of one. The digital operation $F_t$ consists of summing three numbers; $F_a$ consists of moving one number from an input location to an output. The analog function $f_t$, could be complex, but experience shows that it is usually possible to generate, by very simple means, a function which will be close enough to $F_t$ for our purposes. The function $f_a$, if used, is a sample and hold circuit.

The D-A data transfer will be explained first. The normal hybrid simulation would convert A directly to its analog equivalent, a, and feed it into $f_2$ as shown in figure 1. In the method of corrected inputs, $f_2$ receives its basic input, the quantity $m$, from $f_t$, as shown in figure 2. A correction is added to $m$ before it is fed into $f_2$, so that $f_2$ receives the quantity $d$, which is a corrected $m$. The function $f_a$ is an integration and an addition. That is,

$$d = m + f \text{edt}$$

The behavior of the variables is shown in figure 3-A. The dashed line, $A'$, shows what $A$ would look like if it were calculated continuously and in phase with the analog. At time $t_{ao}$, the quantity $d$ is sampled as $d_{ao}$, which is equal to $m_{ao}$. It is converted to the digital number $B_{ao}$ and stored in digital memory. At time $t_{ao}$, the quantity $A_{ao}$ has been computed, and the difference, $E_{ao} (A_{ao} - B_{ao})$, is converted to $e_{ao}$ and fed into $f_s$. Thus, for the period from $t_0$ to $t_1$,

$$d = m + \left( A_0 - D_0 \right) \frac{t - t_1}{t_2 - t_1}$$

Also at $t_1$, the quantity $d_1 (=m_1)$ is converted to $D_1$ and stored. Then, at $t_2$, the quantity $A_1 - D_1$ is computed. Here we have to consider one of the more subtle points of the process. The quantity $D_1$ does not reflect the correction, $e_0$, which has already been added to $m$.* For illustrative purposes, consider the special case, $m = A' - K$ is a constant. Then we would find that

$$A_0 - D_0 = K$$

$$A_1 - D_1 = K$$

$$A_2 - D_2 = 0$$

If we use, for $E$, the general formula

$$E_n = A_n - D_n$$

then we would obtain for consecutive values of $d$,

$$d_0 = A' - K$$

$$d_1 = A' - K$$

$$d_2 = A'$$

$$d_3 = A' + K$$

$$d_4 = A' + K$$

$$d_5 = A'$$

$$d_6 = A' - K$$

etc.

If we use the formula

$$E_n = A_n - D_n - E_{n-1}$$

we would obtain for consecutive values of $d$,

$$d_0 = A' - K$$

$$d_1 = A' - K$$

$$d_2 = A'$$

$$d_3 = A'$$

etc.

Thus, the subtraction of the preceding correction term removes a phase lag oscillation from the D-A input. It is apparent from the foregoing that a condition is imposed on $f_1$, requiring that the amount that $m$ diverges from $A$

* From the procedure, it follows that $d_n = m_n + \sum_{i=0}^{n-2} e_i$
in the time \( t_{n+2} - t_n \) is no greater than the acceptable error in \( d \). That is, from the time the variable \( d \) is sampled to the time the corresponding correction has been added to it, two time cycles have elapsed. Thus, as long as the inaccuracies in the generation of \( m \) do not cause it to drift more than an acceptable amount in two time cycles, it will be satisfactory.

The A-D data transfer is illustrated in figure 3-B. The variable, \( b \), is shown as having high frequency components, and large excursions, so that samples taken at each digital time interval might not be representative. The solution consists of using \( b \) to construct another function, \( h \), whose value at any sampling time, \( t_n \), is equal to the average value of \( b \) during the interval \( t_{n-1} - t_n \). The resulting value, \( h_n \), is converted to \( H_n \), and fed into \( F_1 \). The digital computer, at time \( t_n \), is just starting the computations for the problem interval between \( t_{n-1} \) and \( t_n \). All the variables will have values corresponding to time \( t_{n-1} \), except those coming from the analog, which will have values equal to what their average will be during the time interval, \( t_{n-1} \) to \( t_n \), for which the computations are to be made. It is to be noted that any high speed system responses are automatically reflected in \( b \) because of the closed analog loop through \( f_1 \), \( f_2 \), and \( f_3 \).

The manner in which \( h \) is produced from \( b \) is, in principle, very simple. In the first time interval, from \( t_0 \) to \( t_1 \), the only input to \( f_1 \) is \( b \). Assuming that \( t_n - t_{n-1} \) is a constant, then it is a simple matter to adjust \( f_1 \) such that,

\[
h_1 = \int_{t_0}^{t_1} b dt / (t_1 - t_0). \quad (19)
\]

The quantity \( h_1 \) is sampled, and fed back as soon as possible as \( j_1 \). Thus, at time \( t_2 \), we find,

\[
h_2 = h_1 + \int_{t_1}^{t_2} b dt / (t_2 - t_1) - \int_{t_1}^{t_2} j_1 dt / (t_2 - t_1). \quad (20)
\]

But, the first and third terms on the right side of this expression cancel out, leading to the general result,

\[
h_n = \int_{t_{n-1}}^{t_n} b dt / (t_n - t_{n-1}). \quad (21)
\]

The manner in which \( h \) is fed back to \( f_1 \), as \( j \) will affect the timing of the data transfer, as well as the accuracy of \( h \). The output, \( h \), can be sampled and fed directly back to \( f_1 \), by adding a sample and hold circuit, shown as \( f_6 \), to the analog simulation. Or, \( h \) can be sent to the digital computer as \( H \), sent back immediately as \( J \), and converted to \( j \). I prefer the former method, as it leads to a neater data transfer routine. Therefore, the timing sequence which follows assumes that \( f_6 \) is in the circuit, and \( F_6 \) is out. The alternate sequence, based on \( F_6 \) in the circuit and \( f_6 \) out, is included as appendix A.

Figure 4-A illustrates the timing sequence involved in the operation of the method of corrected inputs, assuming the inclusion of \( f_6 \) in the analog simulation. The actual D-A data transfer will occur in a time interval \( \tau \) seconds.
long, and the A-D transfer in an immediately subsequent interval lasting T seconds. The sequence of operations follows:

\[ t_0 \]

- Analog variables sampled. \((d_0 \text{ and } h_0)\)
  \(h_0 = 0\) fed back to \(f_1\)

\[ t_0 \rightarrow (t_0 + T) \text{ (A-D data transfer)} \]

\[ d_0 = m_0 \rightarrow D_0 \]
\[ h_0 = 0 \rightarrow H_0 \]

\(E_0 = A_0 \rightarrow D_0\)

\(t_0 \rightarrow t_1 \) (Digital computations)

\(E_0 \rightarrow e_0\)

\[ t_1 \]

- Analog variables sampled. \((d_1 \text{ and } h_1)\)
  \(h_1 = \frac{t_1}{t_0}\) fed back to \(f_1\)

\[ t_1 \rightarrow (t_1 + T) \text{ (A-D data transfer)} \]

\[ d_1 = m_1 \rightarrow D_1 \]
\[ h_1 \rightarrow H_1 \]

\(t_1 + T \rightarrow (t_2 - \tau) \) (Digital computations)

\(E_1 = A_1 \rightarrow D_1 \rightarrow E_0\)

\(t_2 - \tau \rightarrow t_2 \) (D-A data transfer)

\(E_1 \rightarrow e_1\)

\[ t_2 \]

- Analog variables sampled. \((d_2 \text{ and } h_2)\)
  \(h_2 = \frac{t_2}{t_1}\) fed back to \(f_4\)

\[ t_2 \rightarrow (t_2 + T) \text{ (A-D data transfer)} \]

\[ d_2 = m_2 + e_0 \rightarrow D_2 \]
\[ h_2 \rightarrow H_2 \]

III. RESULTS OF AN ANALOG DEMONSTRATION

A simplified three degree of freedom simulation of the flight and control of a missile was chosen to illustrate the method of corrected inputs. A description of the simulation is given in appendix B. This work was done on the analog, because the digital computer which comprises half of the hybrid facility is being replaced, a process which will not be completed until early in 1964. In order to simulate a hybrid operation on an analog computer, it was necessary to make simplifications, compromises, and omissions. Even so, it is felt that a valid and informative demonstration was obtained. The main differences between this analog demonstration and a full scale hybrid simulation are:

1. The data transfer took place at much lower rates. Most of the runs used transfer rates between one and ten cycles per second, instead of the 0.1 to 1 KC rates that are customarily used. This was not due to limitations of the equipment, but to the fact that it is much easier to demonstrate the method at the lower rates.

2. The digital equations were simulated on the same time base as the analog equations, with no lag. It is easy enough to store a value in the digital computer until the time comes to use it, but it was not felt to be worth the effort to accomplish the same thing on the analog.

3. The A-D averaging procedure was set up and run, but it was not used in the full simulation, as the fact that the digital lag was not included would have made it a hindrance, rather than a help. Figures 5 and 6 show examples of its operation at 1 and 10 cps respectively.

![Figure 5. L averaged and sampled at 1 cps.](image-url)
between a sampled and a corrected input. Note that the approximate function differs appreciably from the controlling one.

Figures 5 and 6 show the operation of the A-D correction circuit. Figure 5, obtained with a one cps sampling rate, illustrates how each sampled value of the output is equal to the average value of the input function during the preceding sample and hold cycle.

In order to get some numerical indication of the effectiveness of each procedure, a function, Q, was generated, such that

\[ Q = k \int y^2 \, dt \]

where the effectiveness is considered to be inversely proportional to Q. Table 1 shows the results of measuring Q at the termination of each run. All the runs stopped automatically, at \( x = 9000 \) ft. A comparison of the first three columns of this table is shown in figure 10. The data in these curves were taken without A-D sampling, so that they are a measure of the type of D-A transfer used, and of the sampling frequency. Curve A shows the results of D-A transfer, and represents an optimum result, obviously independent of sampling frequency. Curve B plots results with corrected D-A inputs, which can be seen to be far superior to the results obtained by sampling the D-A inputs, shown as curve C.

IV. SUMMARY

In summary, let us see how the method of corrected inputs minimizes the three sources of error listed at the beginning of this paper.
1. **D-A Data Transfer Errors**

The input to the main analog program is smooth instead of stepped. It is in phase with the simulation into which it is fed. It is constantly corrected; the correction is no more than two digital computation intervals behind the actual problem.

2. **A-D Data Transfer Errors**

The digital computer receives the analog variable, already averaged over the interval for which the digital computations are to be made, instead of sampled at its value at the start of that interval. Any errors introduced by the averaging process will tend to cancel themselves out, rather than accumulate.

3. **Time Lags**

The simulation can respond immediately to variations in any portion of the system. This ability is completely independent of either the A-D or the D-A transfer time requirements.

Thus, we have a system which ameliorates difficulties which were thought by many to be intrinsic to hybrid simulation.
APPENDIX A
ALTERNATE TIMING SEQUENCE FOR METHOD OF CORRECTED INPUTS

The timing sequence shown in this appendix would be used in the event that the correction term \(j\), which is fed to \(f_1\) (see figure 2), originates in the digital computer, rather than the analog circuitry. The comments refer to figure 4-B. All A-D data transfers occur in the time intervals \(T\), and all D-A data transfers occur in the immediately subsequent time intervals \(\tau\).

\[ t_0 \]
Analog variables sampled. (\(d_0\) and \(h_0\))
\[ t_0 \rightarrow (t_0 + T) \] (A-D data transfer)
\[ d_0 (= m_0) \rightarrow D_0 \]
\[ h_0 (= a) \rightarrow H_0 \]
\[ (t_0 + T) \rightarrow (t_0 + T + \tau) \] (D-A data transfer)
\[ J_0 (= H_0) \rightarrow j_0 \]
\[ (t_0 + T + \tau) \rightarrow t_1 \] (Digital computations)
\[ E_0 = A_0 - D_0 \]

\[ t_1 \]
Analog variables sampled. (\(d_1\) and \(h_1\))
\[ t_1 \rightarrow (t_1 + T) \] (A-D data transfer)
\[ d_1 (= m_1) \rightarrow D_1 \]
\[ h_1 = \int_{t_1}^{t_2} bdt \cdot (t_1 - t_0) \rightarrow H_1 \]
\[ t_2 \]
\[ (t_1 + T) \rightarrow (t_1 + T + \tau) \] (D-A data transfer)
\[ J_1 (= H_1) \rightarrow j_1 \]
\[ E_0 \rightarrow e_0 \]
\[ (t_1 + T + \tau) \rightarrow \] (Digital computations)
\[ E_i = A_i - D_i - E_n \]

Figure 9. Variable, \(y\), measured at various points during nominal run.

Figure 10. \(Q\) as a function of D-A sampling method and sampling rate.
Analog variables sampled. (d₂ and h₂)
\[ t_2 \rightarrow (t_2 + T) \] (A-D data transfer)
\[ d_2 = m_2 \rightarrow D_2 \]
\[ h_2 \left( \int_{t_1}^{t_2} \mathrm{d}t / (t_2 - t_1) \right) \rightarrow H_2 \]
\[ (t_2 + T) \rightarrow (t_2 + T + \tau) \] (D-A data transfer)
\[ J_2 \rightarrow j_2 \]
\[ E_1 \rightarrow e_1 \]

APPENDIX B
ANALOG SIMULATION TO DEMONSTRATE METHOD OF CORRECTED INPUTS

This all analog simulation was split into so-called “digital” and “analog” portions, to represent those parts of the problem which would normally be assigned to each computer. The equations for calculating the acceleration, velocity, and position vectors were considered digital, and the equations involving the control system, applied forces and torques, and rotational dynamics were considered analog.

Referring to figure 11, the analog equations are:
\[ \ddot{\beta} = -bF/I - dN/I - T/I \]  \hspace{1cm} (B-1)
\[ \dot{\beta} = \dot{\beta}_0 + \int \ddot{\beta} \mathrm{d}t \]  \hspace{1cm} (B-2)
\[ \beta = \beta_0 + \int \dot{\beta} \mathrm{d}t \]  \hspace{1cm} (B-3)
\[ a = \beta - \gamma^2 \]  \hspace{1cm} (B-4)
\[ N = k_1 \alpha \dot{\epsilon}^2 \]  \hspace{1cm} (B-5)
\[ F = k_2 y^4 + k_3 y^3 + k_6 \beta + k_7 \dot{\beta} \]  \hspace{1cm} (B-6)
\[ T = k_9 \dot{\beta} \dot{x} \]  \hspace{1cm} (B-7)
\[ L = N + F \]  \hspace{1cm} (B-8)

and the digital equations are:
\[ \ddot{x} = -D/M + k_1 \]  \hspace{1cm} (B-9)
\[ \dot{x} = \dot{x}_0 + \int \ddot{x} \mathrm{d}t \]  \hspace{1cm} (B-10)
\[ x = x_0 + \int \dot{x} \mathrm{d}t \]  \hspace{1cm} (B-11)
\[ D = k_4 x^2 \]  \hspace{1cm} (B-12)
\[ y = y_0 + \int \dot{y} \mathrm{d}t \]  \hspace{1cm} (B-13)
\[ y = \dot{y} \]  \hspace{1cm} (B-14)
\[ \dot{y} = L^{**}/m \]  \hspace{1cm} (B-15)
\[ y = y_0 + \int \dot{y} \mathrm{d}t \]  \hspace{1cm} (B-16)

The quantities \( y, \dot{y}, \) and \( \gamma \) are approximated by a section of the simulation which replaces equations B-13, B-14, B-15, and B-16 with the equations:
\[ y = y_0 + \int \dot{y} \mathrm{d}t \]  \hspace{1cm} (B-17)
\[ \dot{y} = \dot{y}_0 + \frac{1}{m} \int L \mathrm{d}t \]  \hspace{1cm} (B-18)
\[ \gamma = \dot{y}/\dot{x} \]  \hspace{1cm} (B-19)

For this simulation, the values of the constants were:
\[ m = 1.555 \times 10 \text{ pound-sec}^2/\text{ft} \]
\[ I = 6.22 \times 10 \text{ pound-ft-sec}^2 \]
\[ b = 5 \text{ feet} \]
\[ d = 2 \text{ feet} \]
\[ k_1 = 3 \times 10^{-4} \text{ pound-sec}^2/\text{ft}^2 \]
\[ k_2 = 1.73 \times 10^{-3} \text{ pound-sec}^2 \]
\[ k_3 = 1.06 \times 10 \text{ pounds/ft} \]
\[ k_4 = 5.8 \text{ pound-sec/ft} \]
\[ k_5 = 7.16 \times 10 \text{ pounds} \]

\( \dagger \) The quantities \( y, \dot{y}, \gamma, \) and \( \dot{x} \) are received from the digital portion of the simulation.
\( \ddagger \) The quantity \( L \) is received from the analog portion of the simulation.
The A-D correction circuit (not the sampling circuit) is shown in figure 12-B. The output, \( v_c \), is equal, during the hold period, to the average value of the input, \( v \), during the preceding sample and hold period. This circuit, while operative (see figures 5 and 6), was not used in conjunction with the entire simulation.

\[
\begin{align*}
    &k_s = 1.83 \times 10^{-1} \text{ pound-sec} \\
    &k_r = 3 \times 10^{-2} \text{ feet/sec}^2 \\
    &k_s = 1.25 \times 10^{-4} \text{ pound-sec}^2/ft^2
\end{align*}
\]

In order to simulate the operation of the corrected D-A inputs, the variables \( y, \dot{y}, \) and \( y \) were sent through the circuit of figure 12-A (the D-A correction circuit box of figure 7). In this figure,

\[
\begin{align*}
    &v = \text{the desired function} \\
    &\dot{v} = \text{the approximate function} \\
    &v_s = \text{the output of the sample and hold circuit} \\
    &v_c = \text{the correcting and feedback voltage} \\
    &v_f = \text{the final corrected output}
\end{align*}
\]

A1, A2, and A4 are summing amplifiers, and A3 is an integrator. In operation, the potentiometer, \( P \), is set so that

\[
    \dot{v}_f = v_s/T_s
\]

where \( T_s \) is the period of the sample and hold circuit. The result is that \( v_f \) is a ramp voltage whose value at the end of each hold period is equal to the difference between \( v \) and \( \dot{v} \) at the time of sampling. The problem was run using this correction circuit, both with and without sampling the variable \( L \).

**TABLE 1—VALUES OF Q**

<table>
<thead>
<tr>
<th>Type of data transfer</th>
<th>Sampling Rate, cps</th>
</tr>
</thead>
<tbody>
<tr>
<td>D-A</td>
<td>A-D</td>
</tr>
<tr>
<td>Direct</td>
<td>Direct</td>
</tr>
<tr>
<td>Sampled</td>
<td>Direct</td>
</tr>
<tr>
<td>Corrected</td>
<td>Direct</td>
</tr>
<tr>
<td>Approx.</td>
<td>Direct</td>
</tr>
<tr>
<td>Direct (2)</td>
<td>Sampled</td>
</tr>
<tr>
<td>Sampled</td>
<td>Sampled</td>
</tr>
<tr>
<td>Corrected</td>
<td>Sampled</td>
</tr>
<tr>
<td>Approx.</td>
<td>Sampled</td>
</tr>
</tbody>
</table>

* Unstable, went into overload.