EVALUATION OF THE WEIGHTING FUNCTION OF A LINEAR SYSTEM BY THE METHOD OF DECONVOLUTION

Thesis for the Degree of Ph. D.
MICHIGAN STATE UNIVERSITY

Arvydas Joseph Kliore

1962

This is to certify that the

thesis entitled

EVALUATION OF THE WEIGHTING FUNCTION OF A LINEAR SYSTEM BY THE METHOD OF DECONVOLUTION

presented by

ARVYDAS JOSEPH KLIORE

has been accepted towards fulfillment of the requirements for

PH. D degree in Electrical Engineering

Herman E. Koenig
Major professor

Date February 12, 1962

ABSTRACT

EVALUATION OF THE WEIGHTING FUNCTION OF A LINEAR SYSTEM BY THE METHOD OF DECONVOLUTION

by Arvydas Joseph Kliore

In an adaptive control-system, one of the basic problems is that of identifying the dynamic characteristics or mathematical model of the section of the system which varies with time in an unpredictable manner.

For slowly time-varying linear systems it is proposed to continuously monitor the weighting function of the system by employing the method of deconvolution to effect a step-by-step solution of the convolution summation, the convolution summation being defined as a finite approximation of the convolution integral.

Two types of error are inherent in this method of deconvolution. One is caused by imperfect knowledge of the weighting (response) function of the system, which in general must be estimated prior to the application of the deconvolution method. The second type of error is caused by the truncation of the convolution summation. The propogation of both types of error through iterations of the deconvolution procedure is investigated, and a set of sufficient conditions for the convergence of these errors is derived in terms of certain matrices which are functions of the variations of the input function. These conditions can be applied directly if the input function is known in advance. If the input function is not known, as is the case usually, the investigation of the convergence of errors in advance of the deconvolution computation is difficult. However, these conditions may be applied while the computation is in process to check the behavior of errors during the actual computation.

Finally, it is shown that if the input function meets certain conditions under which the system may be considered initially quiescent, the deconvolution computation may be carried out without the effect of these errors.

EVALUATION OF THE WEIGHTING FUNCTION OF A LINEAR SYSTEM BY THE METHOD OF DECONVOLUTION

By

Arvydas Joseph Kliore

A THESIS

Submitted to the
School of Advanced Graduate Studies of
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Department of Electrical Engineering

1962

HE

ACKNOWLEDGEMENT

The author wishes to express his appreciation to Professor Herman E. Koenig, his major professor, for his encouragement and guidance in the development of this thesis, and for his patience in its editing.

****

TABLE OF CONTENTS

LIST OF FIGURES

TH

LIST OF APPENDICES

THE

I. INTRODUCTION

In recent years the field of automatic control has witnessed a rapid growth of interest in the concept of adaptive control. In most general terms, an adaptive control system is one in which the input-output characteristics of the process to be controlled are measured continually and these measurements are used for continuous self-optimization of the entire system regardless of the criteria of optimization. As is the case with ordinary feedback control systems, a concise mathematical definition of an adaptive system is difficult to realize, and for this reason the usual definition is conceptual rather than mathematical.

Historically, the shift of interest to adaptive control was prompted by the increasing complexity of control problems, which made standard feedback control techniques inadequate, and by the simultaneous advancement of computer technology, which made possible the inclusion of complex digital or analog computers as real time elements of a control system.[(1-2)*] For example, a system whose dynamic characteristics vary with time in an unpredictable manner can only be handled from the adaptive viewpoint.

A conceptual diagram of a general adaptive system appears in Figure 1. The identification computer continuously monitors the dynamic characteristics of the process and the actuating-signal computer

---

* The superscript indicates the number of the reference in the Bibliography, and the page number of a specific reference. Thus, (1-2) refers to reference No. 1, page 2.

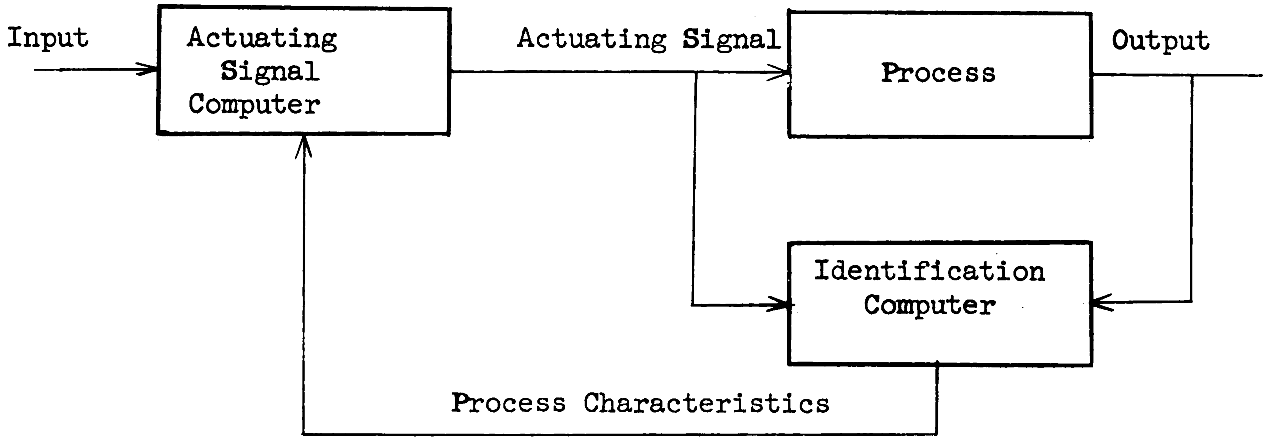

Figure 1. A general adaptive system

generates an actuating signal, based on the input and the process characteristics to realize a desired output. The two basic problems in such an adaptive system are identification and actuation. The problem of identification may be considered the primary requisite of any adaptive control system, since adaptivity implies an automatic and frequent determination of the dynamic characteristics of the process to be controlled. These characteristics may be expressed as the weighting function, transfer function, differential equation, or some other mathematical model.[1-9]

Various methods of solving the identification problem have appeared in the literature.[10,11] One class of identification schemes requires test input signals in addition to the normal operating signals of the system. An example of such a scheme is one in which a multichannel correlator is used to measure the process weighting function as the cross-correlation function between the process output and a binary

THE

white-noise which is added to the system input.[12,13]

A second example of an identification scheme employing a test signal is found in the Sperry adaptive autopilot, in which a train of pulses is added to the ordinary command signals, and the response of the system to this input is used to evaluate a performance criterion.[14]

Other identification procedures have been devised based entirely upon the information contained in the normal input and output signals of a process. Among these is a technique due to Kalman[15] which is considered to be the first significant contribution in the field of adaptive control. This technique employs the solution of a difference-equation representation of the input-output relationship of a process to obtain a Z-transform representation of the process transfer function.

An identification scheme employing orthonormal expansion was proposed by Braun.[16] The input of the system is expressed as an expansion in Laguerre functions or exponentials, and the coefficients in the corresponding expansion of the unit-step response are computed. An orthogonal-spectrum analyzer is applied to the implementation of this procedure.

In a method proposed by Mishkin and Haddad,[17] the unit step response of a process is obtained by an approximate solution of the convolution integral when the input is assumed to be composed of a combination of impulse, step, ramp or parabolic functions. This method requires repeated differentiation of the output function to derive the necessary Taylor-series coefficients.

The identification methods mentioned above are only several of the many that have been proposed. They were outlined here because they are typical of the various methods that have been published.

This thesis presents a method of process identification based on a finite summation representation of the convolution integral which may be implemented through relatively simple computation. The approximations inherent in this method lead to errors which are comprehensively analyzed in Section IV.

TH

II. MATHEMATICAL DESCRIPTION OF PHYSICAL SYSTEMS

In the broadest sense of the term, a physical system is a collection of components that are connected in some rational manner to perform a specific function. If the exact characteristics of all components of the system are known, along with their interconnection, then the characteristics of the system can be completely determined analytically. However, when such information is not available, it is necessary to establish the characteristics of the system from external measurements. This is the well known "black-box" approach, in which a system, no matter how complex, is assumed to have one input and one output of interest. In the work that follows the output variable of the system is assumed to be dependent only upon the one input variable, and the characteristics of the system itself.

The methods of mathematically describing such unilaterally dependent systems constitute the major part of this section. Only those topics which are relevant to later development are discussed. This discussion is given here only as background for later work, and is not a rigorous development in itself.

2.1 Linear Systems

Let a system be represented schematically as in Figure 1. Let $r(t)$ be the input variable and $c(t)$ the output, or response variable of the system. In general, the output variable of the system depends on the input variable and the system characteristics. The system characteristics may in turn be dependent on the input variable or any of its

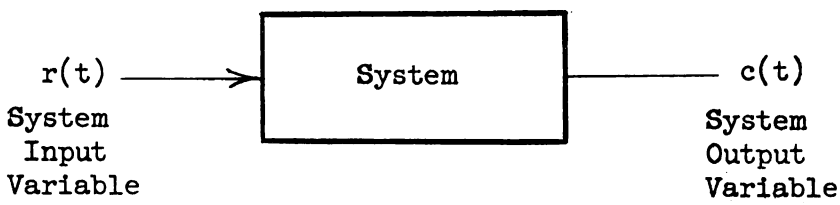

Figure 2. Representation of a system

derivatives, and also upon one or more independent variable such as time, temperature, etc. Systems that have characteristics which are dependent on the input variable are classified as nonlinear. The following discussion will consider only systems that can be assumed to be linear over some range of the input variable.

A linear system is defined as one for which the derivatives of the input and output variables are related by a linear equation, as shown below:

$$\sum_{k=0}^{q} a_k(t) \frac{d^k}{dt^k} c(t) = \sum_{j=0}^{p} b_j(t) \frac{d^j}{dt^j} r(t) \qquad (2\text{-}1)$$

Time-Invariant Linear Systems

In Eq. (2-1) the coefficients $a_k(t)$ and $b_j(t)$ are functions of time, and this characterizes the system as being time-varying. However, the simplest mathematical analysis results when the system characteristics can be assumed to be time-invariant, and hence it is advantageous if the mathematical characterization of the system can be expressed in the form

of a differential equation with constant coefficients

$$\sum_{k=0}^{q} a_k \frac{d^k}{dt^k} c(t) = \sum_{j=0}^{p} b_j \frac{d^j}{dt^j} r(t) \qquad (2\text{-}2)$$

In this discussion, a system will be assumed to be time-invariant if the coefficients in the differential equation do not change over the interval of time required for measurement.

2.2 The Convolution Integral[2]

Suppose that an input r(t) is applied to a linear, time-invariant system. Furthermore, let the input be subdivided into a series of rectangular pulses of width T, as shown in Figure 3.

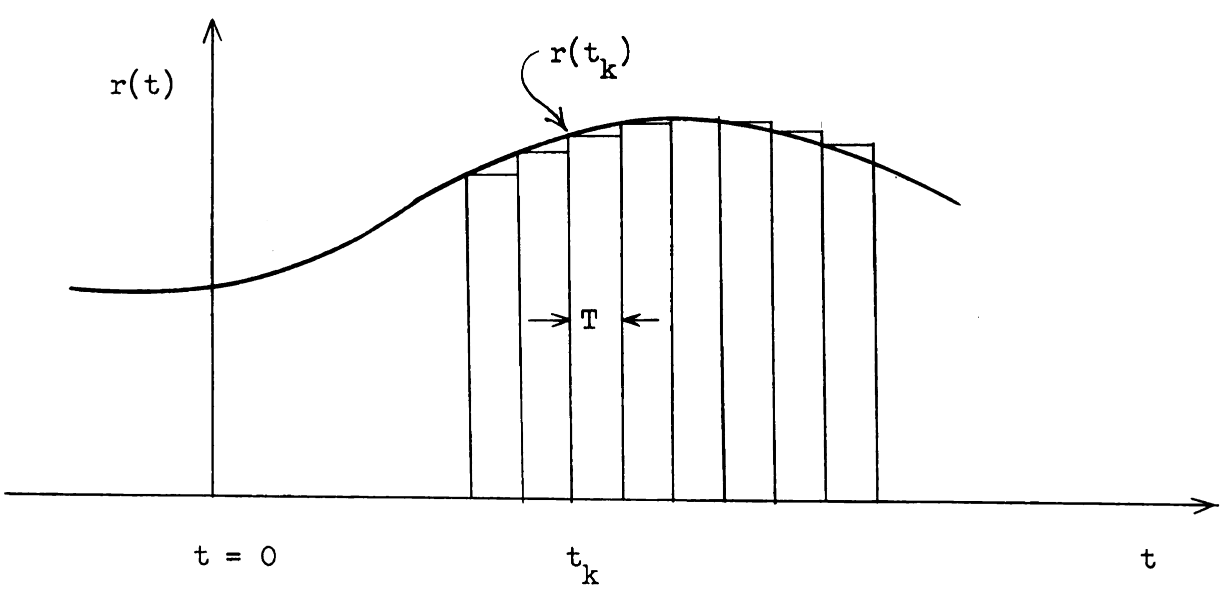

Figure 3. Decomposition of a time-function into rectangular pulses.

When a unit-pulse function is defined as below, then the input function may be approximately represented by

$$r(t) \simeq \sum_{k=-\infty}^{\infty} r(t_k)\ p(t_k,T)T$$

where

$$p(t_k,T) = \frac{1}{T} \quad \text{for } t_k \leq t \leq t_k + T \tag{2-3}$$

$$= 0 \quad \text{otherwise.}$$

and as $T \to 0$

$$r(t) = \lim_{T\to 0} \sum_{k=-\infty}^{\infty} r(t_k)\ p(t_k,T)T \tag{2-4}$$

Let the response of the system to a unit pulse, $p(t_k,T)$, at $t = t_k$ be denoted by $v(t-t_k,T)$. Then, the response to a pulse of height $r(t_k)$ and width T at $t = t_k$ is

$$c_k(t) = v(t-t_k, T)\ r(t_k)\ T \tag{2-5}$$

where

$$v(t-t_k, T) = 0 \quad \text{for } t_k > t$$

Now, using the superposition property, the total response can be approximated as the sum of the various pulse responses of the form of Eq. (2-5) and is given by

$$c(t) \simeq \sum_{k=-\infty}^{\infty} c_k(t) = \sum_{k=-\infty}^{\infty} v(t-t_k, T)\ r(t_k)T \tag{2-6}$$

Let $g(t-t_k)$ be defined as the limit of the unit-pulse response $v(t-t_k,T)$ when $T \rightarrow 0$

$$g(t-t_k) = \lim_{T \rightarrow 0} v(t-t_k, T) \tag{2-7}$$

From the definition of an integral it follows that

$$c(t) = \lim_{T \rightarrow 0} \sum_{k=-\infty}^{\infty} v(t-t_k, T)\, r(t_k)\, T$$

$$= \int_{-\infty}^{\infty} g(t - t')\, r(t')\, dt' \tag{2-8}$$

and since

$$g(t - t') = 0 \qquad \text{for } t < t'$$

the response of the system is

$$c(t) = \int_{-\infty}^{t} g(t - t')\, r(t')\, dt' \tag{2-9}$$

Letting $\tau = t-t'$ the expression may also be written as

$$c(t) = \int_{0}^{\infty} g(\tau)\, r(t-\tau) d\tau \tag{2-10}$$

This is **commonly** known as the convolution, or superposition integral, used extensively in the analysis of linear, time-invariant systems.

The function $g(\tau)$ appearing in the convolution integral is commonly called the weighting function. It is a characteristic of the

system, and by means of the convolution integral the knowledge of this function is sufficient to determine the response of a system to any input function r(t).

2.3 The Transfer Function

Many of the properties of the weighting function $g(\tau)$ can be considered in terms of its Laplace transform.

In Eq. (2-9), let the function r(t) be such that $r(t) = 0$ for $t < 0$, then

$$c(t) = \int_0^t g(t - \tau)\, r(\tau) d\tau \tag{2-11}$$

If c(t), r(t) and g(t) are Laplace transformable,[3-332] then taking the Laplace transform of both sides of Eq. (2-11) gives

$$C(s) = \mathcal{L}\left[\int_0^t g(t - \tau)\, r(\tau) d\tau\right] \tag{2-12}$$

The application of the complex multiplication theorem[3-228] of Laplace-transform theory to the right-hand side of Eq. (2-12) yields

$$C(s) = G(s)\, R(s)$$

where

$$\begin{aligned} C(s) &= \mathcal{L}[c(t)] \\ R(s) &= \mathcal{L}[r(t)] \\ G(s) &= \mathcal{L}[g(t)] \end{aligned} \tag{2-13}$$

THE

The s-domain function $G(s) = \frac{C(s)}{R(s)}$ is commonly called the transfer function of the system having a weighting function $g(t) = \mathcal{L}^{-1}[G(s)]$.

2.4 Properties of the Weighting Function of a Linear System (3-159)

It can be shown, that for a general linear system, G(s) is a rational function of s

$$G(s) = \frac{B(s)}{A(s)} = \frac{\sum_{j=0}^{p} b_j s^j}{\sum_{k=0}^{q} a_k s^k}$$

$$= \frac{b_p s^p + b_{p-1} s^{p-1} + \dots + b_1 s + b_o}{a_q s^q + a_{q-1} s^{q-1} + \dots + a_1 s + a_o} \tag{2-14}$$

where for convenience a_q is taken as unity.

In the development that follows it is assumed that $p < q$. This restriction applies to a very large class of physical systems, particularly to those that are described as having "inertia" effects. Thus, when $p < q$, $B(s)/A(s)$ is a proper fraction, and if the polynomial equations $A(s) = 0$ and $B(s) = 0$ have no common roots, the q-th order polynomial equation (commonly designated the characteristic equation)

$$A(s) = s^q + a_{q-1} s^{q-1} + \dots + a_1 s + a_o = 0 \tag{2-15}$$

in general has n distinct roots $s_1, s_2, \ldots, s_n$, with each root s_i appearing with some multiplicity m_i.

Thus, G(s) may be written as

$$G(s) = \frac{B(s)}{(s-s_1)^{m_1}(s-s_2)^{m_2}\ldots(s-s_n)^{m_n}} \tag{2-16}$$

The fraction $\frac{B(s)}{A(s)}$ can now be expressed as a sum of partial fractions. For each pole s_k of multiplicity m_k, there are m_k partial fractions of the form

$$\frac{M_{k1}}{(s-s_k)^{m_k}} ; \quad \frac{M_{k2}}{(s-s_k)^{m_k-1}} ; \quad \ldots ; \quad \frac{M_{km_k}}{s-s_k}$$

and G(s) may be expressed as a sum of fractions

$$G(s) = \sum_{k=1}^{n} \sum_{j=1}^{m_k} \frac{M_{kj}}{(s-s_k)^{m_k-j+1}} \tag{2-17}$$

where
$$M_{kj} = \lim_{s \to s_k} \frac{1}{(j-1)!} \left[\frac{d^{j-1}}{ds^{j-1}} \frac{(s-s_k)^{m_k} B(s)}{A(s)} \right]$$

The inverse Laplace transform of G(s) may now be evaluated, to yield g(t)

$$g(t) = \mathcal{L}^{-1}[G(s)] = \sum_{k=1}^{n} \sum_{j=1}^{m_k} \frac{M_{kj}}{(m_k - j)!} t^{m_k - j} e^{s_k t} \tag{2-18}$$

where M_{kj} is defined as in Eq. (2-17).

Stability[3-197]

For the purpose of this discussion a system is defined to be absolutely stable if all the roots s_i of the characteristic equation $A(s) = 0$ have negative real parts. If this condition is satisfied, then from Eq. (2-18)

$$\lim_{t \to \infty} g(t) = \lim_{t \to \infty} \sum_{k=1}^{n} \sum_{j=1}^{m_k} \frac{M_{kj}}{(m_k - j)!} t^{m_k - j} e^{\sigma_k t} e^{j\omega_k t}$$

and, if $\sigma_k < 0$ for all k

then

$$\lim_{t \to \infty} g(t) = 0 \tag{2-19}$$

and for an absolutely stable system, g(t) vanishes with increasing t. If s_j is that root of A(s) which has a negative real part such that

$$|\sigma_j| < |\sigma_k| \qquad \text{for all } k \neq j$$

then, for t sufficiently large, the effect of all other roots is negligible and g(t) may be approximated by

$$g(t) \simeq \left[\sum_{j=1}^{m_i} \frac{M_{ij}}{(m_i - j)!} t^{m_i - j} \right] e^{-|\sigma_i| t} e^{j\omega_i t}$$

$$\simeq K t^n \exp(-a t) \exp(j\omega_i t) \qquad (2\text{-}20)$$

where

$$K t^n \simeq \sum_{j=1}^{m_i} \frac{M_{ij}}{(m_i - j)!} t^{m_i - j}$$

and

$$a = |\sigma_i|$$

The Weighting Function at $t = 0$

In later work, it is necessary to specify the behavior of $g(t)$ as t approaches zero. This information is most conveniently obtained from the transfer function, $G(s)$. If

$$\mathcal{L}[g(t)] = G(s)$$

then, from the initial value theorem[3-267]

$$\lim_{t \to 0} g(t) = \lim_{s \to \infty} s\,G(s) \qquad (2\text{-}21)$$

If

$$G(s) = \frac{B(s)}{A(s)} = \frac{b_p s^p + b_{p-1} s^{p-1} + \ldots + b_1 s + b_o}{s^q + a_{q-1} s^{q-1} + \ldots + a_1 s + a_o}$$

then

$$\lim_{t \to 0} g(t) \Rightarrow \lim_{s \to \infty} \frac{s\,B(s)}{A(s)}$$

Thus, if it is required that $\lim_{t \to 0} g(t) = 0$, it is necessary that $\lim_{s \to \infty} \frac{s\,B(s)}{A(s)} = 0$. This is true only if the order of A(s) is greater than the order of s B(s), i.e.,

$$q > p+1 \quad \text{or} \quad q \geq p+2 \tag{2-22}$$

This property is important in the development of forthcoming sections.

2.5 Time-Series Representation of Continuous Systems[4,5]

Let a continuous input function be considered as a series of pulses at t = 0, T, 2T, .. , kT, ... , of width T and height r(kT). If T is chosen to be such that r(t) and g(t) do not change appreciably in the period T, and if the width T of the pulses is very small, then from the superposition property, the response of the system may be approximated by the sum of the responses to each pulse, which may be assumed to be of the form of Eq. (2-7). Thus for $kT < t < (k + 1)T$, the response of the system may be approximated by

$$c(t) \simeq \sum_{n=-\infty}^{k} g(t - nT)\, r(nt)T \tag{2-23}$$

The approximation becomes better as T becomes smaller. For T sufficiently small, it follows from Eq. (2-23) that at t = kT

$$c(kT) = T \sum_{n=-\infty}^{k} r(nT)\, g[(k-n)T] \qquad (2\text{-}24)$$

or, changing the order of summation,

$$c(kT) = T \sum_{j=0}^{\infty} g(jT)\, r[(k-j)T] \qquad (2\text{-}25)$$

In the special case where $r(t) = 0$ for $t < 0$, Eq. (2-24) shows the sequence of values of the output function $c(t)$, at intervals of time T, as an explicit function of the corresponding values of the input function $r(t)$ and the values of the weighting function $g(t)$ evaluated at $t = nT$, and can be written as

$$\begin{bmatrix} c(0) \\ c(T) \\ c(2T) \\ \vdots \end{bmatrix} = T \begin{bmatrix} g(0) & 0 & 0 & \cdots \\ g(T) & g(0) & 0 & \cdots \\ g(2T) & g(T) & g(0) & \cdots \\ \vdots & \vdots & \vdots & \end{bmatrix} \begin{bmatrix} r(0) \\ r(T) \\ r(2T) \\ \vdots \end{bmatrix}$$

(2-26)

This time series representation of the characteristics of linear systems is useful in various numerical techniques of system analysis and synthesis, and forms the basis for the computational deconvolution technique described in Section III.

III. THE METHOD OF DECONVOLUTION

For the purposes of this discussion, the term deconvolution is defined as the step-by-step solution of the convolution summation to obtain the weighting function $g(t)$, or more precisely, the values of the continuous weighting function at uniformly spaced points in time. The convolution summation is defined as the finite approximation to the convolution integral arising from the time-series representation of the response characteristics of a linear, time-invariant system, as defined in the preceding section.

3.1 The Finite Approximation of the Convolution Summation

Let a linear system having a weighting function $g(t)$ be subjected to an input $r(t)$, and let the resulting output be represented by $c(t)$. Furthermore, let $r(kT)$ and $c(kT)$ be the values of $r(t)$ and $c(t)$ at $t = t_o + kT$, where t_o is some arbitrary time origin, as shown in Figure 4.

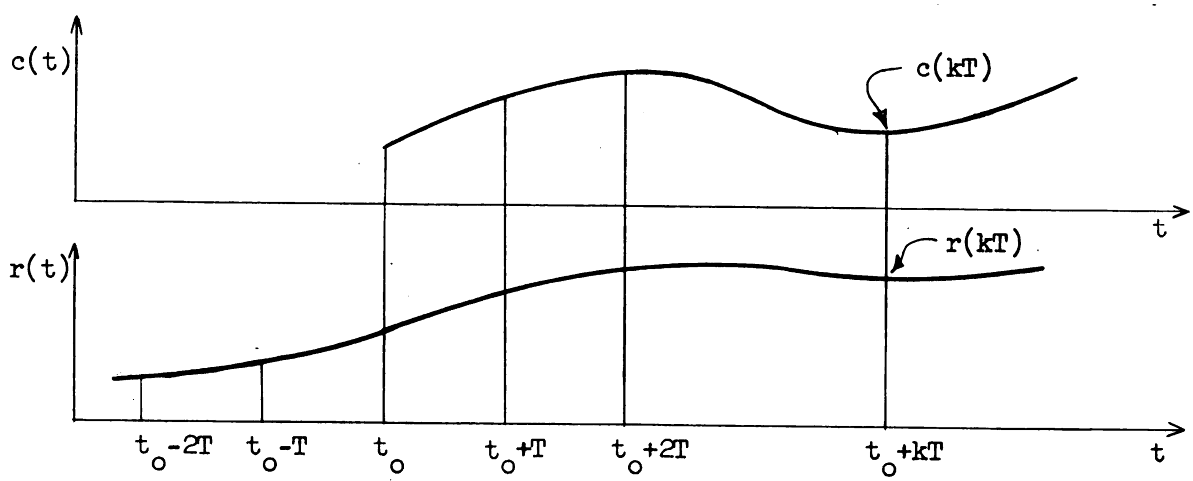

Figure 4. Input-output conventions for the convolution summation

Let the weighting function of the system, g(t), be such that

$$1)\quad \lim_{t \to 0} g(t) = 0$$

$$2)\quad \lim_{t \to \infty} g(t) = 0 \qquad (3\text{-}1)$$

It was shown in Section II, that the conditions in Eq. (3-1) are satisfied by an absolutely stable system with "inertia" effects. The form of such weighting function is shown in Figure 5.

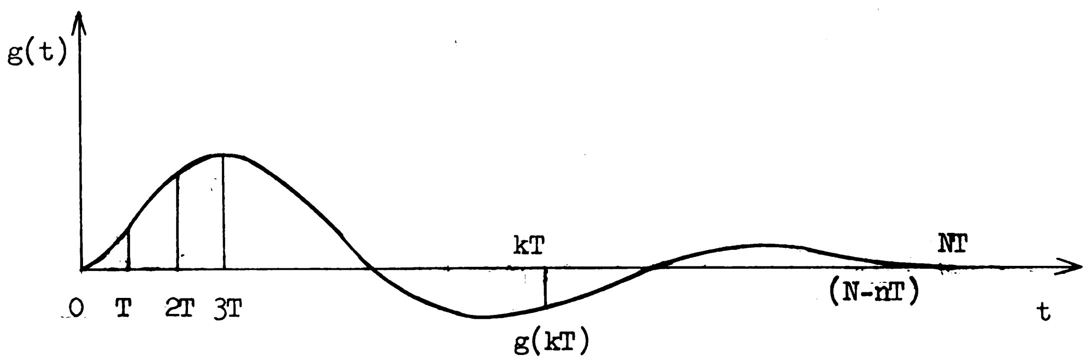

Figure 5. Weighting function of a stable system with inertia effects

As shown in Section 2.5, if T is chosen such that the variation in r(t) and g(t) over any interval of time T is small, the response of the system may be expressed in the form of a convolution summation

$$c(nT) = T \sum_{k=0}^{\infty} g(kT)\, r[(n-k)\, T] \qquad (3\text{-}2)$$

where in order to simplify the notation $c(nT) = c(t_o + nT)$

$$r[(n-k)T] = r(t_o + nT - kT)$$

$$g(kT) = g(0 + kT)$$

and T is the sampling interval.

Since $\lim_{t \to \infty} g(t) = 0$, for any $\epsilon > 0$, there exists a positive integer N, such that for $k > N$, $g(kT) < \epsilon$, and if ϵ is chosen sufficiently small with respect to the precision of observation, the approximation can be made that

$$g(kT) = 0 \qquad \text{for } k \geq N \tag{3-3}$$

and the convolution summation (3-1) becomes

$$c(nT) = T \sum_{k=0}^{N-1} g(kT)\, r[(n-k)\, T] \tag{3-4}$$

Thus, under the finite assumption, the value of the output function c(t) at any sampling instant is dependent only on the values of the input function r(t) at the preceding N-1 sampling instants.

3.2 The Procedure of Deconvolution

For purposes of clarification, let it be assumed that the system is at rest prior to the time t_o, at which time it is desired to begin the deconvolution computation, i.e., $r(t) = 0$ for $t < t_o$.

Furthermore, let the following notation be adopted:

$$r_k = r(t_o + kT)$$

$$c_k = c(t_o + kT)$$

$$g_k = g(kT) \tag{3-5}$$

Using this notation, the expression for the convolution summation is

$$c_n = T \sum_{k=0}^{N-1} g_k \, r_{n-k} \tag{3-6}$$

where $r_{-k} = 0 \quad \text{for } k = 1, 2, \ldots.$

At $t = t_o$, the value of the output c_o is given by

$$c_o = T \, r_o \, g_o$$

or

$$g_o = \frac{c_o}{Tr_o} \tag{3-7}$$

As $g(t)$ was assumed to be such that $g(0) = 0$, the above computation is theoretically not necessary, because the result is known beforehand. However, this initial computation will be useful in the error analysis of Section IV.

At $t = t_o + T$, the convolution summation gives

TH

$$c_1 = T(r_1 g_o + r_o g_1)$$

but, since $g_o = 0$,

$$c_1 = T r_o g_1, \quad \text{or} \quad g_1 = \frac{c_1}{Tr_o} \tag{3-8}$$

Similarly, at $t = t_o + 2T$

$$c_2 = T(r_1 g_1 + r_o g_2)$$

and since g_1 is available from the preceding computation,

$$g_2 = \frac{c_2}{Tr_o} - \frac{r_1}{r_o} g_1 \tag{3-9}$$

Continuing the procedure, at $t = t_o + 3T$, the convolution summation gives

$$c_3 = T\ (r_2 g_1 + r_1 g_2 + r_o g_3)$$

and since g_1 and g_2 have been previously computed,

$$g_3 = \frac{c_3}{Tr_o} - \frac{r_2}{r_o} g_1 - \frac{r_1}{r_o} g_2 \tag{3-10}$$

Continuing in this manner, at $t = t_o + iT$, the output is given by the convolution summation

$$c_i = T \sum_{k=1}^{i} r_{i-k}\, g_k = T\ (r_o g_i + \sum_{k=1}^{i-1} r_{i-k}\, g_k)$$

and

$$g_i = \frac{c_i}{Tr_o} - \frac{1}{r_o} \sum_{k=1}^{i-1} r_{i-k}\, g_k \tag{3-11}$$

where g_k has been previously computed for $k = 1, 2, \dots, i-1$.

TH

Letting $s_k = \frac{r_k}{r_o}$, Eq. (3-11) may be written as

$$g_i = \frac{c_i}{Tr_o} - \sum_{k=1}^{i-1} s_{i-k}\, g_k \tag{3-12}$$

Thus, at any time $t = t_o + iT$, g_i can be computed using the results of previous computations and the appropriate values of the input function, together with the value of the output at $t = t_o + iT$. If the computation is continued through $t = t_o + (N-1)T$, all values of $g(kT)$ from $k = 1$ to $(N-1)$ will have been computed, and the entire procedure may be started again at $t = t_o + NT$ or at any arbitrary time thereafter.

Again, let t_o' be defined as the time at which the new iteration is begun, and let

$$r_k = r(t_o' + kT)$$

and

$$c_k = c(t_o' + kT) \tag{3-13}$$

Furthermore, let $\bar{g}_k$ be the values of g_k computed during the first iteration. If the system has been operating for a time greater than NT before the beginning of the computation, the values $\bar{g}_k$ will be taken as estimates of the actual values. Then, at $t = t_o'$ the convolution summation gives

$$c_o = T \sum_{k=0}^{N-1} r_{-k} \, g_k$$

or, solving for g_o,

$$g_o = \frac{c_o}{Tr_o} - \sum_{k=1}^{N-1} s_{-k} \, g_k \qquad (3\text{-}14)$$

In general, the value of g_o as calculated from Eq. (3-14) is non-zero only because of one or both of two sources of error:

1) Errors in the initial estimates of g_k.
2) Truncation errors resulting from the finite summation.

Since the actual value of g_o is known to be zero the result of the computation in Eq. (3-14) represents the error, which will be designated as E_o. In order to partially compensate for the effect of these systematic errors let E_o be subtracted from the calculated values of each g_i.

Thus, although at $t = t_o' + T$, assuming $g_o = 0$, the convolution summation states that

$$c_1 = T \left(r_o \, g_1 + \sum_{k=2}^{N-1} r_{-k+1} \, g_k \right)$$

the computation for g_1 will be defined as

$$g_1 = \frac{c_1}{Tr_o} - \sum_{k=2}^{N-1} s_{-k+1} \, \bar{g}_k - E_o \qquad (3\text{-}15)$$

where $\bar{g}_k$ are the previously computed or estimated values of g(kT). Similarly, at $t = t_o' + 2T$

$$c_2 = T\ (r_1\ g_1 + r_o\ g_2 + \sum_{k=3}^{N-1} r_{-k+2}\ g_k)$$

or, using the value of g_1 obtained from the preceding computation

$$g_2 = \frac{c_2}{r_o T} - s_1\ g_1 - \sum_{k=3}^{N-1} s_{-k+2}\ \bar{g}_k - E_o \qquad (3\text{-}16)$$

At $t = t_o' + 3T$ the convolution summation gives

$$c_3 = T\ (r_2 g_1 + r_1 g_2 + r_o g_3 + \sum_{k=4}^{N-1} r_{-k+3}\ g_k)$$

and using the values of g_1 and g_2 computed during the two previous intervals

$$g_3 = \frac{c_3}{Tr_o} - s_2 g_1 - s_1 g_2 - \sum_{k=4}^{N-1} s_{-k+3}\ \bar{g}_k - E_o \qquad (3\text{-}17)$$

Finally, at $t = t_o' + iT$, for any i = 1, 2, ... , (N - 1),

$$c_i = T \sum_{k=1}^{N-1} r_{-k+i}\ g_k$$

TH

or

$$\frac{c_i}{T} = \sum_{k=1}^{i-1} r_{-k+i} g_k + r_o g_i + \sum_{k=i+1}^{N-1} r_{-k+i} g_k$$

All g_k $(k = 1, 2, \ldots, i-1)$ have been computed previously in this iteration and g_k $(k = i+1, \ldots, N-1)$ are known either from previous iterations, or as estimates. Therefore, g_i can be computed as follows

$$g_i = \frac{c_i}{Tr_o} - \sum_{k=1}^{i-1} s_{-k+i} g_k - \sum_{k=i+1}^{N-1} s_{-k+i} \bar{g}_k - E_o \tag{3-18}$$

When all N-1 values have been computed, the iteration is complete and the next iteration may be started at $t = t_o' + NT$ or at any time thereafter. The iteration cycle starts with a computation of E_o, and continues with the computations of each g_i $(i = 1$ to $i = N-1)$ during successive sampling intervals, using the most recently computed values of g_i in each computation. Thus, this computational procedure may be carried on indefinitely, completely regenerating all g_i during each iteration. If the system weighting function $g(t)$ varies slowly with time in such a manner that the variation is small over a period of time NT, the deconvolution computation provides a revised representation of $g(t)$ at intervals of time $t = NT$.

IV. ERROR ANALYSIS

The success of any computational procedure, such as that which was discussed in the preceding section, depends to a great extent on the errors that arise as a result of various inaccuracies which distinguish an actual computation from an idealization.

From a strictly computational viewpoint, these sources of error may be divided into two groups; (1) the errors that are caused by arithmetical operations with finite precision and, (2) those which arise as a result of approximations, estimates, and other sources of inaccuracy present in a particular computation.

The first of these is common to all computational procedures and various methods of evaluating its effect may be found in mathematical literature,[6] and for this reason it is not discussed.

The other sources of error are characteristic of the particular computational procedure. These errors are discussed here in detail.

As the deconvolution procedure is applied in an iterative manner, the behavior of errors propogated from iteration to iteration is of great importance, and the determination of conditions under which these propogated errors converge to zero is the primary objective of this error analysis.

4.1 Analysis of the Error Caused by Inaccurate Initial Estimates

It is reasonable to assume that the weighting function, $g(t)$, of a time-varying system is initially known to some degree of accuracy.

The purpose of this section is to analyze the effect of initial error in the values of each $g(kT)$ on the results of successive computations of the weighting function.

The Propogated Error

Let the deconvolution computation begin at some arbitrary time t_o. Furthermore, let the values of the input function $r(t)$ be known for $t = (t_o - T), (t_o - 2T), \ldots\ldots, [t_o - (N-1)T]$, and let these values be designated by

$$r_{-k} = r(t_o - kT)$$

Also, let it be assumed that the estimated values of the weighting function $g(t)$ at $t = T, 2T, \ldots, kT, \ldots, (N-1)T$ are available, and are represented by

$$\bar{g}_k = g_k + e_k \tag{4-1}$$

where $g_k = g(kT)$ and e_k is the error associated with the estimate of g_k. Furthermore, let

$$g_k = 0 \qquad \text{for } k \geq N$$

As shown in Section III, the value of the output function $c(t)$ at $t = t_o+$ is given by the convolution summation

$$c_o = T \sum_{k=0}^{N-1} r_{-k} \, g_k \, ,$$

or

$$\frac{c_o}{T} = r_o \, g_o + \sum_{k=1}^{N-1} r_{-k} \, g_k$$

where

$$c_o = c(t_o)$$

Setting $s_{-k} = \dfrac{r_{-k}}{r_o}$, it follows that

$$g_o = \frac{c_o}{Tr_o} - \sum_{k=1}^{N-1} s_{-k} \, g_k$$

However, since the actual computation is performed with values of g_k that are not exact, the result is:

$$\bar{g}_o = \frac{c_o}{Tr_o} - \sum_{k=1}^{N-1} s_{-k} \, \bar{g}_k$$

$$= \frac{c_o}{Tr_o} - \sum_{k=1}^{N-1} s_{-k} \, g_k - \sum_{k=1}^{N-1} s_{-k} \, e_k$$

or

$$\bar{g}_0 = g_0 + {}_1E_0 ,$$

where

$${}_1E_0 = - \sum_{k=1}^{N-1} s_{-k} e_k$$

However, as shown in Section II, for all systems under consideration, $g_0 \equiv 0$, and thus the result of the first computation gives the error

$${}_1E_0 = \bar{g}_0 = - \sum_{k=1}^{N-1} s_{-k} e_k \qquad (4-2)$$

Now, for $t = t_0 + T$, the convolution summation with $g_0 = 0$ is

$$c_1 = T(r_0 g_1 + \sum_{k=2}^{N-1} r_{-k+1} g_k)$$

or, solving for g_1

$$g_1 = \frac{c_1}{Tr_0} - \sum_{k=2}^{N-1} s_{-k+1} g_k$$

Since ${}_1E_0$ is available from the previous computation let the computation of $\bar{g}_1$ be defined as follows

$$\bar{g}_1 = \frac{c_1}{Tr_o} - \sum_{k=2}^{N-1} s_{-k+1}\, \bar{g}_k - {}_1E_0$$

$$= \frac{c_1}{Tr_o} - \sum_{k=2}^{N-1} s_{-k+1}\, g_k - \sum_{k=2}^{N-1} s_{-k+1}\, e_k - {}_1E_0$$

Setting $\bar{g}_1 = g_1 + {}_1E_1$

it follows that the error associated with the second computation is

$${}_1E_1 = - {}_1E_0 - \sum_{k=2}^{N-1} s_{-k+1}\, e_k \qquad (4\text{-}3)$$

Similarly, for $t = t_o + 2T$, the convolution summation is

$$\frac{c_2}{T} = r_1\, g_1 + r_o\, g_2 + \sum_{k=3}^{N-1} r_{-k+2}\, g_k$$

and

$$g_2 = \frac{c_2}{Tr_o} - s_1\, g_1 - \sum_{k=3}^{N-1} s_{-k+2}\, g_k$$

Again, the actual computation for $\bar{g}_2$ using the previously computed value of $\bar{g}_1$ gives

$$\bar{g}_2 = \frac{c_2}{Tr_o} - s_1\, \bar{g}_1 - \sum_{k=3}^{N-1} s_{-k+2}\, \bar{g}_k - {}_1E_0$$

$$\bar{g}_2 = \frac{c_2}{Tr_o} - s_1 g_1 - \sum_{k=3}^{N-1} s_{-k+2}\, g_k - s_1\, {}_1E_1 - \sum_{k=3}^{N-1} s_{-k+2} e_k - {}_1E_0$$

and letting

$$\bar{g}_2 = g_2 + {}_1E_2,$$

The expression for the error associated with the evaluation of g_2 is

$$ {}_1E_2 = - s_1\, {}_1E_1 - \sum_{k=3}^{N-1} s_{-k+2}\, e_k - {}_1E_0 \qquad (4\text{-}4)$$

Continuing the error analysis, at $t = t_o + 3T$, the convolution summation is

$$c_3 = T\left(r_2 g_1 + r_1 g_2 + r_o g_3 + \sum_{k=4}^{N-1} r_{-k+3}\, g_k\right)$$

or

$$g_3 = \frac{c_3}{Tr_o} - s_2 g_1 - s_1 g_2 - \sum_{k=4}^{N-1} s_{-k+3}\, g_k$$

However, the actual computation using the previously computed values $\bar{g}_1$ and $\bar{g}_2$ gives

$$\bar{g}_3 = \frac{c_3}{Tr_o} - s_2\, \bar{g}_1 - s_1\, \bar{g}_2 - \sum_{k=4}^{N-1} s_{-k+3}\, \bar{g}_k - {}_1E_0$$

$$\bar{g}_3 = \frac{c_3}{Tr_o} - s_2 g_1 - s_1 g_2 - \sum_{k=4}^{N-1} s_{-k+3} g_k - s_2\, {}_1E_1 - s_1\, {}_1E_2 - \sum_{k=4}^{N-1} s_{-k+3} e_k - {}_1E_0$$

and the error associated with g_3 is

$$ {}_1E_3 = - s_2\, {}_1E_1 - s_1\, {}_1E_2 - \sum_{k=4}^{N-1} s_{-k+3} e_k - {}_1E_0 \qquad (4\text{-}5)$$

where

$$\bar{g}_3 = g_3 + {}_1E_3.$$

Proceeding with the computation of each successive g_i, each time subtracting ${}_1E_0$, it will be found that the expression relating all E_i, $(i = 1, 2, \ldots, N-1)$ is

$$\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 0 \\ s_1 & 1 & 0 & \cdots & 0 & 0 \\ s_2 & s_1 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ s_{N-3} & s_{N-4} & s_{N-5} & \cdots & 1 & 0 \\ s_{N-2} & s_{N-3} & s_{N-4} & \cdots & s_1 & 1 \end{bmatrix} \begin{bmatrix} {}_1E_1 \\ {}_1E_2 \\ {}_1E_3 \\ \vdots \\ {}_1E_{N-2} \\ {}_1E_{N-1} \end{bmatrix} = \begin{bmatrix} -{}_1E_0 - \Omega_2 \\ -{}_1E_0 - \Omega_3 \\ -{}_1E_0 - \Omega_4 \\ \vdots \\ -{}_1E_0 - s_1 e_{N-1} \\ -{}_1E_0 \end{bmatrix}$$

(4-6)

where

$$\Omega_i = \sum_{k=i}^{N-1} s_{-k+i-1}\, e_k$$

To transform Eq. (4-6) into a form convenient for analysis, premultiply both sides by the nonsingular matrix

$$[M] = \begin{bmatrix} 1 & 0 & 0 & --- & 0 & 0 \\ -1 & 1 & 0 & --- & 0 & 0 \\ 0 & -1 & 1 & --- & 0 & 0 \\ | & | & | & & | & | \\ | & | & | & & | & | \\ | & | & | & & | & | \\ 0 & 0 & 0 & --- & 1 & 0 \\ 0 & 0 & 0 & --- & -1 & 1 \end{bmatrix} \qquad (4\text{-}7)$$

to obtain

$$\begin{bmatrix} 1 & 0 & 0 & --- & 0 & 0 \\ (s_1-1) & 1 & 0 & --- & 0 & 0 \\ (s_2-s_1) & (s_1-1) & 1 & --- & 0 & 0 \\ | & | & | & & | & | \\ | & | & | & & | & | \\ | & | & | & & | & | \\ (s_{N-3}-s_{N-4}) & (s_{N-4}-s_{N-5}) & (s_{N-5}-s_{N-6}) & --- & 1 & 0 \\ (s_{N-2}-s_{N-3}) & (s_{N-3}-s_{N-4}) & (s_{N-4}-s_{N-5}) & --- & (s_1-1) & 1 \end{bmatrix} \begin{bmatrix} {}_1E_1 \\ {}_1E_2 \\ {}_1E_3 \\ | \\ | \\ | \\ {}_1E_{N-2} \\ {}_1E_{N-1} \end{bmatrix}$$

$$= \begin{bmatrix} -{}_1E_0 - \Omega_2 \\ \Omega_2 - \Omega_3 \\ \Omega_3 - \Omega_4 \\ | \\ s_{-1}e_{N-2} - (s_{-1}-s_{-2})e_{N-1} \\ s_{-1}\, e_{N-1} \end{bmatrix} \qquad (4\text{-}8)$$

Now, let the differences between successive values of the normalized input function be defined as

$$s_i - s_{i-1} = \Delta_i \qquad (4\text{-}9)$$

Then, since in Eq. (4-8)

$$_1E_0 = -\sum_{k=1}^{N-1} s_{-k}e_k = -s_{-1}e_1 - s_{-2}e_2 - s_{-3}e_3 - \cdots - s_{-(N-1)}e_{N-1}$$

and

$$\Omega_2 = \sum_{k=2}^{N-1} s_{-k+1}e_k = s_{-1}e_2 + s_{-2}e_3 + \cdots + s_{-(N-2)}e_{N-1}$$

it follows that the first entry on the right-hand side of Eq. (4-8) is

$$-_1E_0 - \Omega_2 = s_{-1}e_1 - (s_{-1}-s_{-2})e_2 - (s_{-2}-s_{-3})e_3 - \cdots - [s_{-(N-2)}-s_{-(N-1)}]e_{N-1}$$

$$= (1 - \Delta_o)e_1 - \sum_{k=2}^{N-1} \Delta_{-k+1}\, e_k \qquad (4\text{-}10)$$

Similarly, for all i = 1, 2, ..., N-1, since

$$\Omega_i = \sum_{k=i}^{N-1} s_{-k+i-1}\, e_k = s_{-1}e_i + s_{-2}e_{i+1} + \cdots + s_{-(N-i)}e_{N-1}$$

and

$$\Omega_{i+1} = \sum_{k=i+1}^{N-1} s_{-k+i}\, e_k = s_{-1}\, e_{i+1} + \dots + s_{-(N-i-1)}\, e_{N-1}$$

it follows that the i-th entry on the right-hand side of Eq. (4-8) is

$$\Omega_i - \Omega_{i+1} = s_{-1}\, e_i - (s_{-1}-s_{-2})\, e_{i+1} - \dots - [s_{-(N-i-1)} - s_{-(N-i)}]e_{N-1}$$

$$= (1 - \Delta_o)e_i - \sum_{k=i+1}^{N-1} \Delta_{-k+1}\, e_k \qquad (4\text{-}11)$$

Substituting Eq. (4-9), Eq. (4-10) and Eq. (4-11) into the matrix equation (4-8) a set of equations relating the errors in the first iteration to the errors in the original estimates and the normalized input differences Δ_i is obtained.

$$\begin{bmatrix} 1 & 0 & 0 & --- & 0 & 0 \\ \Delta_1 & 1 & 0 & --- & 0 & 0 \\ \Delta_2 & \Delta_1 & 1 & --- & 0 & 0 \\ | & | & | & & | & | \\ | & | & | & & | & | \\ | & | & | & & | & | \\ \Delta_{N-3} & \Delta_{N-4} & \Delta_{N-5} & --- & 1 & 0 \\ \Delta_{N-2} & \Delta_{N-3} & \Delta_{N-4} & --- & \Delta_1 & 1 \end{bmatrix} \begin{bmatrix} {}_1E_1 \\ {}_1E_2 \\ {}_1E_3 \\ | \\ | \\ | \\ {}_1E_{N-2} \\ {}_1E_{N-1} \end{bmatrix} =$$

$$\begin{bmatrix} (1-\Delta_o) & -\Delta_{-1} & -\Delta_{-2} & --- & -\Delta_{-(N-3)} & -\Delta_{-(N-2)} \\ 0 & (1-\Delta_o) & -\Delta_{-1} & --- & -\Delta_{-(N-4)} & -\Delta_{-(N-3)} \\ 0 & 0 & (1-\Delta_o) & --- & -\Delta_{-(N-5)} & -\Delta_{-(N-4)} \\ | & | & | & & | & | \\ 0 & 0 & 0 & --- & (1-\Delta_o) & -\Delta_{-1} \\ 0 & 0 & 0 & --- & 0 & (1-\Delta_o) \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ | \\ e_{N-2} \\ e_{N-1} \end{bmatrix} \quad (4\text{-}12)$$

Eq. (4-12) may be rewritten in symbolic matrix notation

$$[\,A\,]_1 \quad [\,E\,]_1 = [\,B\,]_1\,[\,e\,] \qquad (4\text{-}13)$$

where

$$[\,A\,]_1 = \begin{bmatrix} 1 & 0 & --- & 0 \\ \Delta_1 & 1 & --- & 0 \\ | & | & & | \\ \Delta_{N-2} & \Delta_{N-3} & --- & 1 \end{bmatrix}$$

$$[B\,]_1 = \begin{bmatrix} (1-\Delta_o) & -\Delta_{-1} & --- & -\Delta_{-(N-2)} \\ 0 & (1-\Delta_o) & --- & -\Delta_{-(N-3)} \\ | & | & & | \\ 0 & 0 & --- & (1-\Delta_o) \end{bmatrix}$$

$$[E]_1 = \begin{bmatrix} {}_1E_1 \\ {}_1E_2 \\ | \\ {}_1E_{N-1} \end{bmatrix} \quad \text{, and } [e] = \begin{bmatrix} e_1 \\ e_2 \\ | \\ e_{N-1} \end{bmatrix}$$

Since [A] is triangular, the inverse $[A]_1^{-1}$ exists[6], and Eq. (4-13) can be solved to obtain the errors in the first iteration as explicit functions of the errors in the initial estimates and differences between the successive values of the normalized input function

$$[E]_1 = [A]_1^{-1} \; [B]_1 \; [e] \tag{4-14}$$

$$= [C]_1 \; [e]$$

where

$$[C]_1 = [A]_1^{-1} \; [B]_1$$

During the second iteration, begun at some time $t \geq t_o + NT$, the computational procedure will be exactly the same, except that the error associated with each value of the weighting function g_i used in the computation is designated by ${}_1E_i$, $(i = 1,2,\ldots, N-1)$, i.e., the error resulting from the first iteration.

Thus, the expression for the error propogated through the second iteration of the deconvolution procedure is of the same form as Eq.(4-14), namely

$$[E]_2 = [A]_2^{-1} \; [B]_2 \; [E]_1$$

$$= [C]_2 \; [E]_1 \tag{4-15}$$

where

$$[E]_2 = \begin{bmatrix} {}_2E_1 \\ {}_2E_2 \\ \vdots \\ {}_2E_{N-1} \end{bmatrix}$$

and the $[A]_2$ and $[B]_2$ matrices are identical in form to $[A]_1$ and $[B]_1$, differing only in the values of the entries Δ_i. Substituting Eq. (4-14) into Eq. (4-15) the errors in the second iteration are given by

$$[E]_2 = [C]_2 \; [C]_1 \; [e] \tag{4-16}$$

For each successive iteration the computational procedure is repeated, using the $\bar{g}_i$ computed in the preceding iteration. Thus, the expression for the error ${}_jE_i$ associated with each g_i computed during the j-th iteration is

$$[E]_j = [A]_j^{-1} \; [B]_j \; [E]_{j-1} = [C]_j \; [E]_{j-1}$$

where

$$[E]_j = \begin{bmatrix} {}_jE_i \\ {}_jE_2 \\ \vdots \\ {}_jE_{N-1} \end{bmatrix}$$

and $[A]_j$ and $[B]_j$ are identical in form to $[A]_1$ and $[B]_1$. But since

$$[E]_{j-1} = [C]_{j-1} \; [E]_{j-2}$$

$$[E]_{j-2} = [C]_{j-2} \; [E]_{j-3}$$

$$\vdots$$

$$[E]_1 = [C]_1 \; [e]$$

it follows that the error associated with the j-th iteration, as an explicit function of the error associated with the original estimates, is

$$[E]_j = [C]_j \ [C]_{j-1} \ [C]_{j-2} \ \cdots \ [C]_2 \ [C]_1 \ [e]$$

$$= \left\{ \prod_{\ell=1}^{j} [C]_\ell \right\} [e] \qquad (4\text{-}17)$$

where

$$[C]_\ell = [A]_\ell^{-1} \ [B]_\ell$$

and $[A]_\ell$ and $[B]_\ell$ are matrices identical in form to $[A]_1$ and $[B]_1$ shown after Eq. (4-13).

Conditions for Convergence of the Propogated Error

In previous discussion the nature of the initial errors e_i was not defined. However, before any statement concerning the convergence of the propogated error can be made, the properties of the initial errors e_i must be known.

Since these errors arise from the imperfect knowledge of the values of the weighting function g(t) at t = T, 2T, ... (N-1)T, it is reasonable to assume that each e_i is an independent random variable. Furthermore, let it be assumed that each e_i is normally distributed, with a mean $m_i = 0$, and a variance $\sigma_i^2 = \sigma_e^2$ for all i = 1, 2, ..., (N-1).

The error propogated through the j-th iteration of the deconvolution procedure is given by Eq. (4-17)

Letting

$$\prod_{\ell=1}^{j} [C]_\ell = [D]_j \qquad (4\text{-}18)$$

Eq. (4-17) can be written as

$$[E]_j = [D]_j \; [e]$$

or in detail

$$\begin{bmatrix} {}_jE_1 \\ {}_jE_2 \\ \vdots \\ {}_jE_{N-1} \end{bmatrix} = \begin{bmatrix} {}_jd_{11} & {}_jd_{12} & \cdots & {}_jd_{1,N-1} \\ {}_jd_{21} & {}_jd_{22} & \cdots & {}_jd_{2,N-1} \\ \vdots & \vdots & & \vdots \\ {}_jd_{N-1,1} & {}_jd_{N-1,2} & \cdots & {}_jd_{N-1,N-1} \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_{N-1} \end{bmatrix} \qquad (4\text{-}19)$$

Thus, $${}_jE_i = \sum_{k=1}^{N-1} {}_jd_{ik}\, e_k \,, \qquad \text{for } i = 1, 2, \ldots, (N-1).$$

Now , since all e_k are independent and have the same variance σ_e^2, it can be shown that the variance of each ${}_jE_i$ is(9-99)

$$\text{Var}\,({}_jE_i) = \left(\sum_{k=1}^{N-1} {}_jd_{ik}^2 \right) \sigma_e^2 \qquad (4\text{-}20)$$

$$i = 1, 2, \ldots, (N-1)$$

and the mean $M({}_jE_i)$ is

$$M\,({}_jE_i) = \sum_{k=1}^{N-1} {}_jd_{ik} \;\; m_k$$

However, since $m_k = 0$ for all k, it follows that:

$$M({}_jE_i) = 0 \tag{4-21}$$

The propogated errors will be considered to be convergent if the variance of each ${}_jE_i$ obtained in the j-th iteration is less than the variance of each ${}_{(j-1)}E_i$ obtained in the (j-1) iteration, i.e.,

$$\text{Var}\ ({}_jE_i)\ < \text{Var}\ ({}_{j-1}E_i) < \ldots\ldots < \text{Var}\ ({}_1E_i) < \sigma_e^2$$

$$\text{for all } i = 1,\ 2,\ \ldots\ ,\ (N-1) \tag{4-22}$$

If this condition is met, then

$$\lim_{j \to \infty} \quad \text{Var}\ ({}_jE_i) = 0$$

and thus, after many iterations of the deconvolution procedure, the variance of the error approaches zero. Since the mean of each ${}_jE_i$ is zero, the value of each ${}_jE_i$ vanishes.

From the expression for the variance of each ${}_jE_i$ given in Eq (4-20) the condition in Eq. (4-22) implies first of all that

$$\text{Var}\ ({}_1E_i) < \sigma_e^2$$

Since for the first iteration $[E]_1 = [C]_1\ [e] = [D]_1\ [e]$

$$\text{Var}\ ({}_1E_i) = \left(\sum_{k=1}^{N-1} {}_1c_{ik}^2 \right) \sigma_e^2 \quad = \quad \left(\sum_{k=1}^{N-1} {}_1d_{ik}^2 \right) \sigma_e^2$$

To satisfy Eq. (4-22) it is necessary that

$$\sum_{k=1}^{N-1} {}_1c_{ik}^2 = \sum_{k=1}^{N-1} {}_1d_{ik}^2 < 1$$

for all i = 1, 2, ... , (N-1)

For the second iteration

$$[E]_2 = [C]_2 \quad [C]_1 \quad [e] \quad = \quad [D]_2 \quad [e]$$

$$\text{and, Var } ({}_2E_i) = \left(\sum_{k=1}^{N-1} {}_2d_{ik}^2 \right) \sigma_e^2 \quad , i = 1, 2, \ldots , N-1$$

To satisfy the condition of Eq. (4-22) it is necessary that

$$\sum_{k=1}^{N-1} {}_2d_{ik}^2 < \sum_{k=1}^{N-1} {}_1d_{ik}^2 < 1$$

for all i = 1, 2, ... , (N-1)

Finally, to satisfy the condition of Eq. (4-22) for j iterations, the entries in the rows of the respective [D] matrices must be such that

$$\sum_{k=1}^{N-1} {}_jd_{ik}^2 < \sum_{k=1}^{N-1} {}_{(j-1)}d_{ik}^2 < \ldots < \sum_{k=1}^{N-1} {}_1d_{ik}^2 < 1$$

for all rows i = 1, 2, ... , (N-1) (4-23)

$$\text{where } [D]_j = \prod_{\ell=1}^{j} [C]_\ell$$

$$\text{and } [C]_\ell = [A]_\ell^{-1} \; [B]_\ell$$

Thus, if the conditions of Eq. (4-23) are satisfied, the propogated errors resulting from inaccurate initial estimates have decreasing variances, and are considered to converge in a statistical sense.

4.2 Analysis of the Truncation Error

In the previous discussion, the assumption was made that the weighting function g(t) vanishes for $t \geq NT$, and thus the effect of all g_k for $k \geq N$ was neglected. However, as shown in Section II, for any absolutely stable system, g(t) approaches zero asymptotically for large t, but it does not vanish identically for any finite value of t.

In this section the error introduced by the truncation of g(t) at t = NT is analyzed to determine its effect on successive iterations of the process of deconvolution.

The Propogated Error

As in previous sections, let the following notation be used:

$$g_k = g(kT) \quad \text{for } k = 0, 1, 2, \ldots,$$

$$\left.\begin{aligned} r_k &= r(t_o + kT) \\ c_k &= c(t_o + kT) \end{aligned}\right\} \text{for } k = 0, \pm 1, \pm 2, \ldots, \pm \infty$$

where t_o is the time at which the deconvolution computation is started. Furthermore, let the discussion be restricted to systems for which $g_o = 0$.

The finite approximation of the convolution summation for the value of the output, c(t), at $t = t_o$ is

$$c_o = T\left(r_o g_o + \sum_{k=1}^{N-1} r_{-k}\, g_k\right)$$

and thus, letting $s_k = \dfrac{r_k}{r_o}$, the computed value of g_o is

$$\bar{g}_o = \frac{c_o}{Tr_o} - \sum_{k=1}^{N-1} s_{-k}\, g_k$$

However, the exact expression for c_o is

$$c_o = T \sum_{k=0}^{\infty} r_{-k}\, g_k$$

or

$$c_o = T\left(r_o g_o + \sum_{k=1}^{N-1} r_{-k}\, g_k + \sum_{k=N}^{\infty} r_{-k}\, g_k\right)$$

and hence, the true value of g_o is:

$$g_o = \frac{c_o}{Tr_o} - \sum_{k=1}^{N-1} s_{-k}\, g_k - \sum_{k=N}^{\infty} s_{-k}\, g_k$$

The difference ${}_1E_o$ between the computed value, $\bar{g}_o$, and the true value, g_o, is

$$ {}_1E_o = \bar{g}_o - g_o = \sum_{k=N}^{\infty} s_{-k}\, g_k \tag{4-24} $$

Since $g_o = 0$, it is apparent that the result obtained in the computation of $\bar{g}_o$ represents only the error, ${}_1E_0$.

At $t = t_o + T$, the actual value of the output c_1 is

$$ c_1 = T\left(r_1 g_o + r_o g_1 + \sum_{k=2}^{N-1} r_{-k+1}\, g_k + \sum_{k=N}^{\infty} r_{-k+1}\, g_k\right) $$

and therefore

$$ g_1 = \frac{c_1}{Tr_o} - s_1 g_o - \sum_{k=2}^{N-1} s_{-k+1}\, g_k - \sum_{k=N}^{\infty} s_{-k+1}\, g_k $$

However, in the actual computation for $\bar{g}_1$, the convolution summation is truncated. Furthermore, the value of g_o is taken to be zero, and the quantity ${}_1E_o$ which was computed previously is subtracted in order to partially compensate for the truncation error.

$$ \bar{g}_1 = \frac{c_1}{Tr_o} - \sum_{k=2}^{N-1} s_{-k+1}\, g_k - {}_1E_0 $$

$$ = g_1 + {}_1E_1 $$

where
$$ {}_1E_1 = \sum_{k=N}^{\infty} s_{-k+1}\, g_k - {}_1E_0 \tag{4-25} $$

Continuing in the same manner, at $t = t_o + 2T$, the actual value of c_2 is

$$ c_2 = T\left(r_1 g_1 + r_o g_2 + \sum_{k=3}^{N-1} r_{-k+2}\, g_k + \sum_{k=N}^{\infty} r_{-k+2}\, g_k\right) $$

and therefore,

$$ g_2 = \frac{c_2}{Tr_o} - s_1 g_1 - \sum_{k=3}^{N-1} s_{-k+2}\, g_k - \sum_{k=N}^{\infty} s_{-k+2}\, g_k $$

However, the computed value of $\bar{g}_2$ is

$$ \bar{g}_2 = \frac{c_2}{Tr_o} - s_1\, \bar{g}_1 - \sum_{k=3}^{N-1} s_{-k+2}\, g_k - {}_1E_0 $$

$$ = \frac{c_2}{Tr_o} - s_1(g_1 + {}_1E_1) - \sum_{k=3}^{N-1} s_{-k+2}\, g_k - {}_1E_0 $$

$$ = g_2 + {}_1E_2 $$

where

$$ {}_1E_2 = - s_1\, {}_1E_1 + \sum_{k=N}^{\infty} s_{-k+2}\, g_k - {}_1E_0 \tag{4-26} $$

In general, at $t = t_o + iT$, the actual value of the output c_i is

$$c_i = T\left(\sum_{k=1}^{i-1} r_{i-k}\, g_k + r_o g_i + \sum_{k=i+1}^{N-1} r_{-k+i}\, g_k + \sum_{k=N}^{\infty} r_{-k+i}\, g_k \right)$$

or

$$g_i = \frac{c_i}{Tr_o} - \sum_{k=1}^{i-1} s_{i-k}\, g_k - \sum_{k=i+1}^{N-1} s_{-k+i}\, g_k - \sum_{k=N}^{\infty} s_{-k+i}\, g_k$$

But using the finite approximation and the previously computed values of $\bar{g}_j$, for j = 1, 2, ..., (i-1), the value of the computed $\bar{g}_i$ is

$$\bar{g}_i = \frac{c_i}{Tr_o} - \sum_{k=1}^{i-1} s_{i-k}\, \bar{g}_k - \sum_{k=i+1}^{N-1} s_{-k+i}\, g_k - {}_1E_0$$

$$= \frac{c_i}{Tr_o} - \sum_{k=1}^{i-1} s_{i-k}\, g_k - \sum_{k=i+1}^{N-1} s_{-k+i}\, g_k - \sum_{k=1}^{i-1} s_{i-k}\, {}_1E_k - {}_1E_0$$

$$= g_i + {}_1E_i$$

where

$$_1E_i = - \sum_{k=0}^{i-1} s_{i-k}\, {}_1E_k + \sum_{k=N}^{\infty} s_{-k+i}\, g_k - {}_1E_0 \qquad (4\text{-}27)$$

The expressions for each ${}_1E_i$, (i = 1, 2, ... N-1) can be written in the form of a matrix equation

$$\begin{bmatrix} 1 & 0 & 0 & --- & 0 & 0 \\ s_1 & 1 & 0 & --- & 0 & 0 \\ s_2 & s_1 & 1 & --- & 0 & 0 \\ | & | & | & & | & | \\ s_{N-3} & s_{N-4} & s_{N-5} & --- & 1 & 0 \\ s_{N-2} & s_{N-3} & s_{N-4} & --- & s_1 & 1 \end{bmatrix} \begin{bmatrix} {}_1E_1 \\ {}_1E_2 \\ {}_1E_3 \\ | \\ {}_1E_{N-2} \\ {}_1E_{N-1} \end{bmatrix} = \begin{bmatrix} H_1 - {}_1E_0 \\ H_2 - {}_1E_0 \\ H_3 - {}_1E_0 \\ | \\ H_{N-2} - {}_1E_0 \\ H_{N-1} - {}_1E_0 \end{bmatrix} \qquad (4\text{-}28)$$

$$\text{where } H_i = \sum_{k=N}^{\infty} s_{-k+i}\, g_k \qquad (4\text{-}29)$$

To facilitate analysis, both sides of Eq. (4-28) are pre-multiplied by the nonsingular transformation matrix [M] given in Eq. (4-7) to give

$$\begin{bmatrix} 1 & 0 & 0 & --- & 0 & 0 \\ \Delta_1 & 1 & 0 & --- & 0 & 0 \\ \Delta_2 & \Delta_1 & 1 & --- & 0 & 0 \\ | & | & | & & | & | \\ \Delta_{N-3} & \Delta_{N-4} & \Delta_{N-5} & --- & 1 & 0 \\ \Delta_{N-2} & \Delta_{N-3} & \Delta_{N-4} & --- & \Delta_1 & 1 \end{bmatrix} \begin{bmatrix} {}_1E_1 \\ {}_1E_2 \\ {}_1E_3 \\ | \\ {}_1E_{N-2} \\ {}_1E_{N-1} \end{bmatrix} = \begin{bmatrix} H_1 - H_0 \\ H_2 - H_1 \\ H_3 - H_2 \\ | \\ H_{N-2} - H_{N-3} \\ H_{N-1} - H_{N-2} \end{bmatrix} \qquad (4\text{-}30)$$

where $\Delta_i = s_i - s_{i-1}$ represents the difference between the normalized input function at successive intervals.

Upon examination of the right-hand side of Eq. (4-30), it is observed that, for the i-th term

$$H_i = \sum_{k=N}^{\infty} s_{-k+i}\, g_k = s_{-N+i}\, g_N + s_{-N+i-1}\, g_{N+1} + \cdots$$

and

$$H_{i-1} = \sum_{k=N}^{\infty} s_{-k+i-1}\, g_k = s_{-N+i-1} g_N + s_{-N+i-2}\, g_{N+1} + \cdots$$

and therefore

$$H_i - H_{i-1} = (s_{-N+i} - s_{-N+i-1}) g_N + (s_{-N+i-1} - s_{-N+i-2}) g_{N+1} + \cdots$$

$$= \Delta_{-N+i}\, g_N + \Delta_{-N+i-1}\, g_{N+1} + \cdots$$

$$= \sum_{k=N}^{\infty} \Delta_{-k+i}\, g_k$$

Eq. (4-30) can now be written as

$$\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 0 \\ \Delta_1 & 1 & 0 & \cdots & 0 & 0 \\ \Delta_2 & \Delta_1 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ \Delta_{N-2} & \Delta_{N-3} & \Delta_{N-4} & \cdots & 1 & 0 \\ \Delta_{N-1} & \Delta_{N-2} & \Delta_{N-3} & \cdots & \Delta_1 & 1 \end{bmatrix} \begin{bmatrix} {}_1E_1 \\ {}_1E_2 \\ {}_1E_3 \\ \vdots \\ {}_1E_{N-2} \\ {}_1E_{N-1} \end{bmatrix} = \begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ \vdots \\ p_{N-2} \\ p_{N-1} \end{bmatrix} \qquad (4\text{-}31)$$

TH

where

$$p_i = \sum_{k=N}^{\infty} \Delta_{-k+i}\, g_k$$

The right-hand side of Eq. (4-31) can be expressed as a product of two infinite matrices

$$\begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ \vdots \\ p_{N-2} \\ p_{N-1} \end{bmatrix} = \begin{bmatrix} \Delta_{-N+1} & \Delta_{-N} & \Delta_{-N-1} & \cdots \\ \Delta_{-N+2} & \Delta_{-N+1} & \Delta_{-N} & \cdots \\ \Delta_{-N+3} & \Delta_{-N+2} & \Delta_{-N+1} & \cdots \\ \vdots & \vdots & \vdots & \\ \Delta_{-2} & \Delta_{-3} & \Delta_{-4} & \cdots \\ \Delta_{-1} & \Delta_{-2} & \Delta_{-3} & \cdots \end{bmatrix} \begin{bmatrix} g_N \\ g_{N+1} \\ g_{N+2} \\ \vdots \\ \\ \\ \end{bmatrix} \tag{4-32}$$

or symbolically

$$[P]_1 = [F]_1 \; [G] \tag{4-33}$$

The coefficient matrix on the left-hand side of Eq. (4-31) is exactly the [A] matrix defined in Eq. (4-13) and therefore Eq. (4-31) can be expressed in the symbolic form.

$$[A]_1 \; [E]_1 = [F]_1 \; [G] \tag{4-34}$$

The system of Eq. (4-34) can be solved for $[E]_1$, giving the errors in the first iteration as an explicit function of the normalized input differences Δ_i and the values of the truncated portion of g(t).

$$[E]_1 = [A]_1^{-1} \; [F]_1 \, [G] \qquad (4\text{-}35)$$

During the second iteration, begun at some time $t \geq t_o + NT$, the computation is carried out with values of $\bar{g}_i$ which have an error ${}_1E_i$. In addition,truncation errors also accumulate. Therefore the total error propogated through the second iteration is the sum of the two contributions

$$[A]_2 \; [E]_2 = [B]_2 \; [E]_1 + [F]_2 \, [G] \qquad (4\text{-}36)$$

where $[F]_2$ is of the same form as $[F]_1$ in Eq. (4-34).

Substituting Eq. (4-35) into Eq. (4-36) and solving for the errors $[E]_2$ there results

$$[E]_2 = [A]_2^{-1} \, [B]_2 \, [A]_1^{-1} \, [F]_1 \, [G] + [A]_2^{-1} \, [F]_2 \, [G] \qquad (4\text{-}37)$$

During the third iteration the values of $\bar{g}_i$ used in computation are in error by ${}_2E_i$. Including these errors along with the truncation errors, the errors propogated through the third iteration of the deconvolution procedure are related to previous errors by

$$[A]_3 \, [E]_3 = [B]_3 \, [E]_2 + [F]_3 \, [G] \qquad (4\text{-}38)$$

Substituting the expression for $[E]_2$ from Eq. (4-37) and solving for $[E]_3$ gives

$$[E]_3 = [A]_3^{-1} [B]_3 [A]_2^{-1} [B]_2 [A]_1^{-1} [F]_1 [G] + [A]_3^{-1} [B]_3 [A]_2^{-1} [F]_2 [G] + [A]_3^{-1} [F]_3 [G] \tag{4-39}$$

Letting

$$[C]_j = [A]_j^{-1} [B]_j$$

and

$$\prod_{j=n}^{n-k} [C]_j = [C]_n [C]_{n-1} \cdots [C]_{n-k+1} [C]_{n-k}$$

the total error in the third iteration can be expressed as

$$[E]_3 = \prod_{j=3}^{2} [C]_j [A]_1^{-1} [F]_1 [G] + \prod_{j=3}^{3} [C]_j [A]_2^{-1} [F]_2 [G] + [A]_3^{-1} [F]_3 [G]$$

$$= \sum_{k=1}^{2} \left\{ \prod_{j=3}^{k+1} [C]_j \right\} [A]_k^{-1} [F]_k [G] + [A]_3^{-1} [F]_3 [G] \tag{4-40}$$

If this procedure is carried out through the i-th iteration, it can be shown that the expression for $[E]_i$ will have a form similar to Eq. (4-40), namely

$$[E]_i = [A]_i^{-1} [F]_i [G] + \sum_{k=1}^{i-1} \left\{ \prod_{j=i}^{k+1} [C]_j \right\} [A]_k^{-1} [F]_k [G] \tag{4-41}$$

Now, using the notation

$$[D]_n = \prod_{j=i}^{i-n+1} [C]_j$$

and

$$[P]_n = [F]_n \, [G]$$

the expression for the total error in the i-th iteration as an explicit function of the differences Δ_i between adjacent values of the normalized input function and the values of the truncated portion of g(t) is

$$[E]_i = [A]_i^{-1} \, [P]_i + \sum_{k=1}^{i-1} [D]_{i-k} \, [A]_k^{-1} \, [P]_k \qquad (4\text{-}42)$$

<u>Convergence of the Truncation Errors</u>

In order to analyze the convergence of the truncation errors, it is first necessary to investigate the nature of the matrices $[P]_j$.

From Eq. (4-32)

$$[P]_j = [F]_j \, [G]$$

or in detail,

$$\begin{bmatrix} {}_jP_1 \\ {}_jP_2 \\ {}_jP_3 \\ \vdots \\ {}_jP_{N-2} \\ {}_jP_{N-1} \end{bmatrix} = \begin{bmatrix} {}_j\Delta_{-N+1} & {}_j\Delta_{-N} & {}_j\Delta_{-N-1} & \cdots \\ {}_j\Delta_{-N+2} & {}_j\Delta_{-N+1} & {}_j\Delta_{-N} & \cdots \\ {}_j\Delta_{-N+3} & {}_j\Delta_{-N+2} & {}_j\Delta_{-N+1} & \cdots \\ \vdots & \vdots & \vdots & \\ {}_j\Delta_{-2} & {}_j\Delta_{-3} & {}_j\Delta_{-4} & \cdots \\ {}_j\Delta_{-1} & {}_j\Delta_{-2} & {}_j\Delta_{-3} & \cdots \end{bmatrix} \begin{bmatrix} g_N \\ g_{N+1} \\ g_{N+2} \\ \vdots \\ \\ \\ \end{bmatrix}$$

where the pre-subscripts j identify quantities associated with the j-th iteration.

Each entry of $[P]_j$ thus has the following form (dropping the pre-subscripts to simplify notation)

$$p_i = \sum_{k=N}^{\infty} \Delta_{-k+i} \, g_k$$

Now, if $\Delta_{max} \geq |\Delta_j|$ for all j, it follows that

$$|p_i| \leq \Delta_{max} \sum_{k=N}^{\infty} |g_k| \qquad (4\text{-}43)$$

for all i = 1, 2, ..., (N-1)

Furthermore, it is shown in Appendix B, that for an absolutely stable system, given $\epsilon > 0$, there exists an integer **P**, such that for any $N > P$

$$\sum_{k=N}^{\infty} |g_k| < \epsilon$$

or, from Eq. (4-43)

$$|p_i| < \Delta_{max} \, \epsilon \qquad (4\text{-}44)$$

Now, let

$$[Q]_j = [A]_j^{-1} \, [P]_j \qquad (4\text{-}45)$$

then Eq. (4-42) can be written as follows

$$[E]_i = [Q]_i + \sum_{k=1}^{i-1} [D]_{i-k} [Q]_k \qquad (4\text{-}46)$$

where from Appendix A

$$[A]_j^{-1} = \begin{bmatrix} 1 & 0 & 0 & --- & 0 \\ {}_j a_1 & 1 & 0 & --- & 0 \\ {}_j a_2 & {}_j a_1 & 1 & --- & 0 \\ | & | & | & & | \\ {}_j a_{N-2} & {}_j a_{N-3} & {}_j a_{N-4} & --- & 1 \end{bmatrix}$$

Therefore, the i-th entry of [Q] in Eq. (4-46) is of the form

$$q_i = p_i + \sum_{k=1}^{i-1} p_k \, a_{i-k} \qquad (4\text{-}47)$$

To establish a bound on the magnitude of q_i, let

$$\left| {}_j a_i \right| \leq a_{max} \qquad \text{for } i = 1, 2, \ldots, (N-1)$$
$$j = 1, 2, \ldots, \infty$$

then, from Eq. (4-44) and Eq. (4-47)

$$\left| {}_j q_i \right| \leq [1 + (i-1)\, a_{max}]\, \Delta_{max} \; \epsilon \qquad (4\text{-}48)$$

for all j.

Forming the matrix product indicated in Eq. (4-46), the n-th entry of $[E]_i$ representing the error associated with g_n in the i-th iteration is

$$ {}_iE_n = {}_iq_n + \sum_{k=1}^{i-1} \left[\sum_{j=1}^{N-1} ({}_kd_{nj})({}_kq_j) \right] \qquad (4\text{-}49)$$

The bound of this error is

$$ \left| {}_iE_n \right| < q_{max} \left[1 + \sum_{k=1}^{i-1} \sum_{j=1}^{N-1} \left| {}_kd_{nj} \right| \right] \qquad (4\text{-}50)$$

where from Eq. (4-48)

$$ q_{max} = [1 + (N-2)\, a_{max}]\, \Delta_{max}\, \epsilon \geq \left| {}_jq_i \right| $$

for all j, and all i = 1, 2, ... , N-1.

Now let all $[D]_k$ be such that, for any row n

$$ \sum_{j=1}^{N-1} \left| {}_1d_{nj} \right| \leq 1 - \beta_n $$

$$ \sum_{j=1}^{N-1} \left| {}_2d_{nj} \right| \leq (1 - \beta_n) \sum_{j=1}^{N-1} \left| {}_1d_{nj} \right| $$

$$ \vdots $$

$$ \sum_{j=1}^{N-1} \left| {}_kd_{nj} \right| \leq (1 - \beta_n) \sum_{j=1}^{N-1} \left| {}_kd_{nj} \right| $$

$$ \vdots $$

where

$$0 < \beta_n < 1 \tag{4-51}$$

If the conditions of Eq. (4-51) are satisfied, Eq. (4-50) can be written as

$$\left|{}_iE_n\right| < q_{max} \sum_{k=0}^{i-1} (1 - \beta_n)^k \tag{4-52}$$

In the limit, as the number of iterations, i, increases without bound,

$$\lim_{i \to \infty} \left|{}_iE_n\right| < q_{max} \sum_{k=0}^{\infty} (1 - \beta_n)^k$$

or

$$\lim_{i \to \infty} \left|{}_iE_n\right| < \frac{q_{max}}{\beta_n} \tag{4-53}$$

for $n = 1, 2, \ldots, (N-1)$

Thus, if the conditions of Eq. (4-51) are satisfied, the accumulated error due to truncation approaches a limit. Furthermore, this limit may be made arbitrarily small by choice of N.

Comparing the conditions of Eq. (4-23) of Section 4.1 with the conditions of Eq. (4-51) it is observed that the latter are much more restrictive, and in fact imply the former. Specifically, from Eq. (4-51)

$$\sum_{j=1}^{N-1} \left|{}_kd_{nj}\right| \leq (1 - \beta) \Rightarrow \left|{}_kd_{nj}\right| < 1$$

for all k,

and therefore

$$ {}_k d_{nj}^2 < \left| {}_k d_{nj} \right| $$

and

$$ \sum_{j=1}^{N-1} {}_k d_{nj}^2 < \sum_{j=1}^{N-1} \left| {}_k d_{nj} \right| $$

It should be pointed out, that the conditions of Eq. (4-51) are sufficient conditions for convergence of the truncation error, and that in practice a less restrictive condition may bring about convergence.

4.3 Discussion of the Error Analysis

In the preceding sections, the conditions for the convergence of the errors caused by inaccurate initial estimates and truncation were derived in terms of a coefficient matrix [D]. This matrix is a product of matrices $[C]_j$, the entries of which are complicated functions of differences of the adjacent values of the normalized input function s(t) over the entire interval of time during which the deconvolution procedure is carried out. In order to investigate the behavior of errors for any given input function, it is necessary to derive the [A] and [B] matrices given in Eq. (4-13) for each period of iteration, invert the [A] matrices, compute the coefficient matrices $[C] = [A]^{-1} [B]$ and, finally, form the products of the [C] matrices to obtain the [D] matrices which contain the information required for application of the convergence criteria.

Admittedly, the preceding is cumbersome to apply and requires considerable computation. However, attempts to obtain convergence

conditions in terms of quantities more simply related to the input function $r(t)$ have not been successful, primarily because of the unwieldly form of the coefficients of $[A]^{-1}$ (see Appendix A).

Furthermore, since the convergence criteria are based on the behavior of the input function $r(t)$ over the entire interval of time over which the deconvolution computation is iterated, they cannot be applied in advance of the actual computation without knowledge of $r(t)$ for the entire interval of interest. Thus, these criteria are of limited value for investigating the convergence of errors caused by truncation and inexact initial knowledge of the weighting function in advance of the actual computation. However, the application of the error convergence criteria simultaneously with each iteration of the deconvolution computation would furnish information on the behavior of the error while the computation is being carried out.

Thus, while in the general case, evaluation of the effects of the two types of inherent errors is difficult, in practical applications of the deconvolution procedure some quite feasible simplifying restrictions can be imposed. Several of these are discussed next.

4.4 Periodic Input Functions

The error analysis is considerably simplified if the input function $r(t)$ is known to be periodic, and if the period is an integral multiple of the time NT, i.e.,

$$r(t_o + kNT) = r(t_o), \qquad k = 1, 2, \ldots. \qquad (4\text{-}54)$$

Under these conditions

$$r_1 = r(t_o + T) = r(t_o + T + kNT)$$

$$= {}_k r_1$$

and in general,

$$r_i = r(t_o + iT) = r(t_o + iT + kNT) \quad (4\text{-}55)$$

$$= {}_k r_i$$

$$\text{for } i = 0, \pm 1, \pm 2, \ldots, \pm (N-1)$$

where the pre-subscript k indicates the value of r_i used in the k-th iteration of the deconvolution computation. Thus, under this condition, the values of r_i occurring in every iteration of the deconvolution procedure are identical to the corresponding r_i occurring during the first iteration, i.e.,

$${}_1 r_i = {}_2 r_i = {}_3 r_i = \ldots\ldots\ldots = {}_k r_i = \ldots\ldots \quad (4\text{-}56)$$

where 1, 2, ... , k, ... is the number of the iteration.

Now, since ${}_k s_i = \dfrac{{}_k r_i}{{}_k r_0}$, from Eq. (4-56) it follows that, for $i = 0, \pm 1, \pm 2, \ldots \pm N-1$

$${}_1 s_i = {}_2 s_i = {}_3 s_i = \ldots\ldots\ldots = {}_k s_i = \ldots\ldots \quad (4\text{-}57)$$

Using the definition of Eq. (4-9), it follows that

for all $i = 0, \pm 1, \pm 2, \ldots \pm N-1$,

$$ {}_1\Delta_i = {}_2\Delta_i = {}_3\Delta_i = \ldots\ldots = {}_k\Delta_i = \ldots\ldots \qquad (4\text{-}58) $$

Referring to the form of the [A] and [B] matrices shown in Eq. (4-13), it may be noted that since the form of these matrices does not change from iteration to iteration, and since the Δ_i which constitute the coefficients of these matrices are identical for each iteration, the matrices [A] and [B] respectively will be identical for each iteration. Thus

$$ [A]_1 = [A]_2 = [A]_3 = \ldots\ldots = [A]_k = \ldots\ldots $$

and

$$ [B]_1 = [B]_2 = [B]_3 = \ldots\ldots = [B]_k = \ldots\ldots \qquad (4\text{-}59) $$

Since $[C]_j$ is defined as

$$ [C]_j = [A]_j^{-1} \, [B]_j $$

it follows from Eq. (4-59) that

$$ [C]_1 = [C]_2 = [C]_3 = \ldots\ldots = [C]_k = \ldots\ldots \qquad (4\text{-}60) $$

and

$$ [D]_j = \prod_{k=1}^{j} [C]_k = [C]_1^j \qquad (4\text{-}61) $$

Now, since the convergence conditions in Eq. (4-23) and Eq. (4-51) are stated in terms of the coefficients of the [D] matrix, the application of these conditions becomes considerably simpler when the [D] matrix can be evaluated as in Eq. (4-61).

Under these conditions, the computation of the [C] matrix, which requires the inversion of the [A] matrix, must only be performed once, and the [D] matrices for successive iterations are found simply by performing the multiplication of [C] into itself the required number of times. This not only leads to a considerable computational simplification, but also allows the convergence conditions to be applied in advance of the actual deconvolution computation.

4.5 Initially Quiescent Systems

From the viewpoint of the practical application of the deconvolution procedure, a very important simplification is obtained when the system can be considered to be quiescent prior to the commencement of the deconvolution computation.

For the purposes of this discussion, a system will be considered to be quiescent if the input function $r(t)$ has a constant value or is equal to zero for all time t such that $(t_o - NT) \leq t < t_o - T$, where t_o is the time at which the deconvolution computation is to be started.

Let the input function $r(t)$ be of the form shown in Figure 6

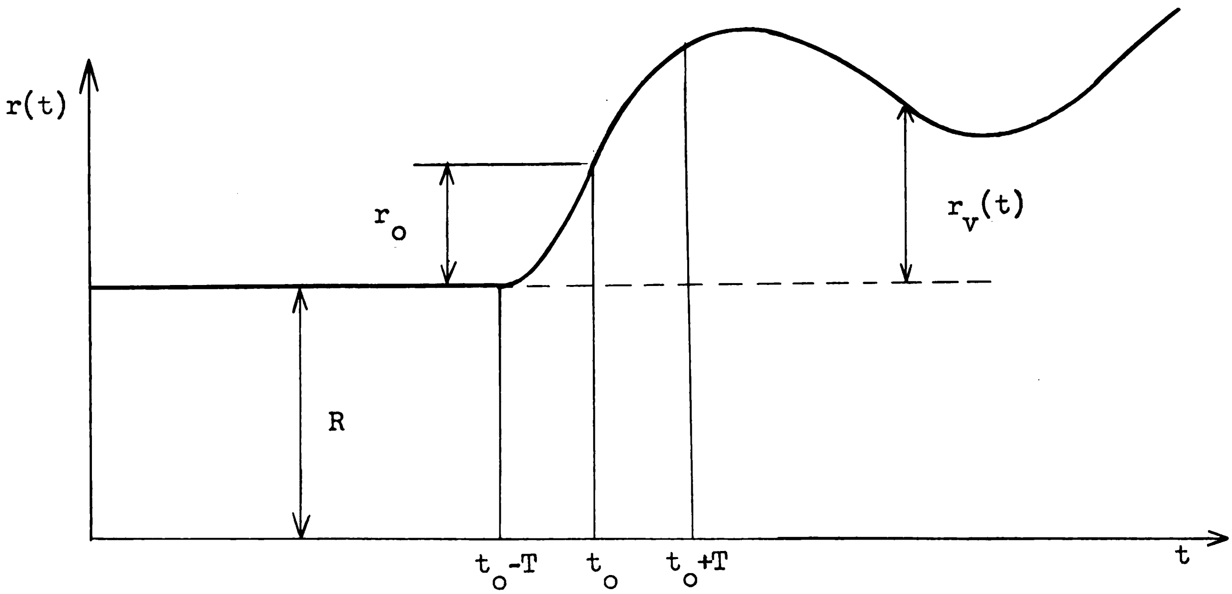

Figure 6 The form of the input function for an initially quiescent system

Analytically, the form of such an input function is

$$r(t) = R \qquad \text{for } (t_o - NT) \leq t \leq (t_o - T) \qquad (4\text{-}62)$$

$$r(t) = R + r_v(t) \quad \text{for } t > t_o - T$$

Using the convolution summation, the output c(t) of the system at any sampling instant $(t_o + nT)$, where n = 0, 1, ... , N-1, is given by

$$c(t_o + nT) = T \sum_{k=0}^{N-1} g(kT)\, r\,[t_o + (n-k)T] \qquad (4\text{-}63)$$

but, since r(t) is of the form shown in Eq.(4-62)

$$c(t_o + nT) = TR \sum_{k=0}^{N-1} g(kT) + T \sum_{k=0}^{n} g(kT)\, r_v[t_o + (n-k)T] \qquad (4\text{-}64)$$

or

$$c(t_o + nT) = C + c_v(t_o + nT)$$

where

$$c_v(t_o + nT) = T \sum_{k=0}^{n} g(kT) \quad r_v[t_o + (n-k)T] \qquad (4\text{-}65)$$

From Eq. (4-64) it can be seen that the output c(t) prior to $t = t_o$ consists only of the constant C, and the varying part of the output $c_v(t)$ is related to the varying part of the input, $r_v(t)$, by a convolution summation of the same form as Eq. (4-63), but having only n + 1 terms. Thus, considering only the varying parts of the output and input functions, the deconvolution computation may be started at $t = t_o$ without knowledge of the initial values of g(kT), since

$$r_v(t_o + kT) = 0 \qquad \text{for } k < 0.$$

Using the notation

$$c_v(t_o + nT) = c_n \qquad (4\text{-}66)$$

and

$$r_v(t_o + nT) = r_n$$

then, from the summation in Eq. (4-65), at $t = t_o + T$, the output c_1 is

$$c_1 = T \sum_{k=0}^{1} g_k \, r_{1-k}$$

$$= T[g_o r_1 + g_1 r_o]$$

but $g_o = 0$ for all systems under consideration, and hence;

$$g_1 = \frac{c_1}{Tr_o} \tag{4-67}$$

Similarly, for $t = t_o + 2T$,

$$c_2 = T \sum_{k=1}^{2} g_k \, r_{2-k}$$

$$= T[g_1 r_1 + g_2 r_o]$$

and it follows that:

$$g_2 = \frac{c_2}{Tr_o} - \frac{r_1}{r_o} g_1 \tag{4-68}$$

In general, for $t = t_o + nT$, where $n = 1, 2, 3, \ldots, N-1$

$$c_n = T \sum_{k=1}^{n} g_k \, r_{n-k} = T \, g_n r_o + T \sum_{k=1}^{n-1} g_k \, r_{n-k}$$

TH

but, since all g_k for k = 1, 2, ... , n-1 are available from previous computation, g_n can be computed as

$$g_n = \frac{c_n}{Tr_o} - \sum_{k=1}^{n-1} g_k \, s_{n-k} \qquad (4\text{-}69)$$

where

$$s_n = \frac{r_n}{r_o}$$

Thus, for an initially quiescent system, all values of g_n may be computed sequentially without any previous knowledge of the values of g_k. These computed values are then available for use in subsequent iterations of the deconvolution procedure, and it is not necessary to use estimated values of g_k. Under these conditions the procedure is free of the effects of the error that is associated with estimation.

However, the special nature of the input function r(t) which is necessary to obtain an initially quiescent system raises another problem. It may be observed from the expressions for the computation of any g_k, that in each case the computation requires multiplication by the term $\frac{1}{r_o}$, where r_o is the value of $r_v(t)$ at $t = t_o$. It is also required that $r_v(t_o - T) = 0$, thus r_o represents the change in the input function r(t) from $t = t_o - T$ to $t = t_o$, a period of time equal to the sampling interval T. If this change is small, then the reciprocal $(\frac{1}{r_o})$ is subject to large inaccuracy for even a small error in the measurement of the value of r_o. Thus, the change in r(t) from $t = t_o - T$ to $t = t_o$ must be large in order to make the computation of $\frac{1}{r_o}$ as accurate as possible. For

TH

example, if 1% accuracy is desired in $\frac{1}{r_o}$, and the precision of measurement of r_o is ϵ, then

$$\frac{1}{r_o} - \frac{1}{r_o + \epsilon} \leq \frac{.01}{r_o}$$

or

$$r_o \geq 100\epsilon$$

This implies that in order to benefit from the simplification brought about by the consideration of an initially quiescent system, the deconvolution computation must be started immediately after introducing a large disturbance into the input of the system which had been quiescent for a time greater than NT previous to that time. This requirement, while restrictive, is not unreasonable in applications to some practical systems.

The second type of error that has been considered, the truncation error, is not eliminated by the restriction to initially quiescent systems. However, as shown in Appendix B, the contribution of the truncation error can be made arbitrarily small by choice of a large enough time NT. Thus, if in a practical application NT is chosen such that $g(nT)$ for $n > N$ is much smaller than the precision of measurement, ϵ, of the system, the contribution of the truncation error may be made negligible.

It may be therefore concluded, that while the effects of the truncation and estimation error are difficult to evaluate in general, the process of deconvolution may be carried out essentially free of these types of error if the system is initially quiescent in the sense of the preceding discussion, and if the process is truncated only after a sufficiently large time NT.

V. CONCLUSION

In the preceding sections, the method of deconvolution was investigated as a solution to the identification problem inherent in any approach to an adaptive control system. The implementation of this method would make continuously available a representation of the weighting function of a slowly time-varying system, which then might be used in some adaptation scheme to render the overall system independent of the variation.

To compute one point on the weighting function, N-1 multiplication operations and N addition operations are required, where N is the number of sampling intervals. This computation is shown schematically in Figure 7. Thus, a total of 2N-1 arithmetical operations must be performed for each point. The number of samples, N, depends on the value of T chosen in the approximation of the convolution integral, since the total time NT is approximately constant for any one system under consideration. The storage requirements also depend on N, since 2N numbers must be in storage at all times. Thus, both the complexity of computation and the storage requirements are directly proportional to the number of samples, N.

However, the accuracy of the approximation, and hence the accuracy of the results, becomes better as T is made smaller, and consequently, as the number N is increased. As a result, in any application of the deconvolution procedure, a compromise must be made between accuracy and the size and complexity of the required computation. No attempt has been made to prepare specific computation techniques for the implementation of the deconvolution method, as that is not the purpose of this work.

Th

The error analysis, which comprises the greater portion of this work, is important because of the iterative nature of the deconvolution method and the errors arising from the finite approximation (truncation) and the inexact knowledge of the system characteristics prior to the beginning of the computation. The results of the error analysis are expressed as a set of indirect restrictions on the allowable variation of the input signal to insure convergence of these propogated errors. Since these restrictions are expressed as conditions applying to highly complicated functions of the variation of the input signal, direct conditions applying to the input signal could not be derived analytically, and the practical application of these conditions in the general case is difficult. However, certain simplifications are introduced by imposing restrictions on the input function $r(t)$.

In particular, if the input function $r(t)$ is periodic with a period kNT, the computation of the quantities required for the application of the error convergence criteria is greatly simplified. In addition, this restriction makes possible the application of the convergence criteria prior to the deconvolution computation.

Finally, in the special case in which the system can be considered to be initially quiescent, the deconvolution computation is free of the errors caused by inexact knowledge of the initial values of the weighting function. If in addition the number of sampling intervals N is chosen to be such that $g(NT)$ is much less than the precision of measurement, the effect of truncation is negligible, and the deconvolution computation can be carried out without the effect of these two types of inherent error.

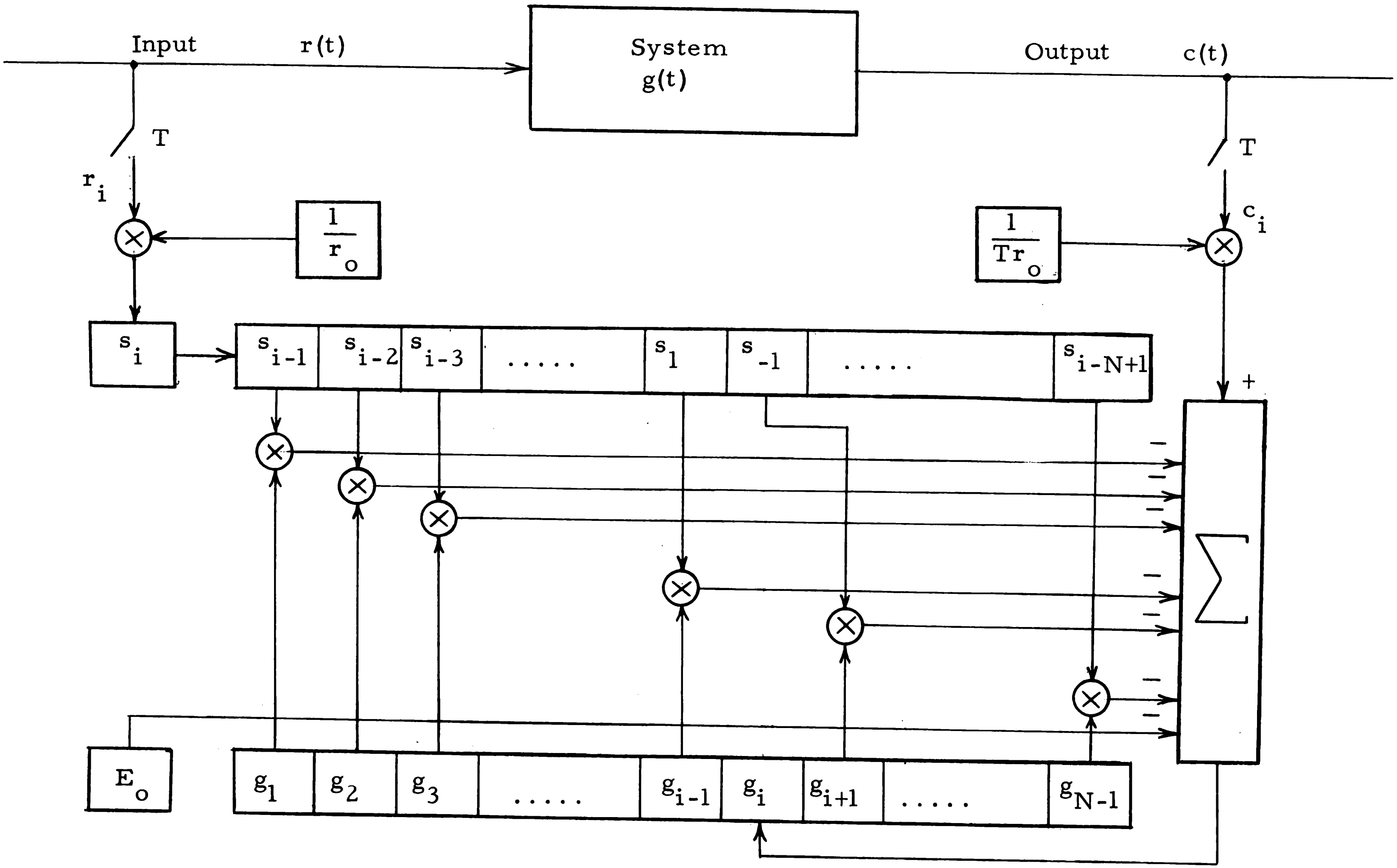

Figure 7. Computation of a single point g_i

APPENDIX A

INVERSION OF THE [A] MATRIX

It is desired to obtain the inverse of the matrix **[A]** shown below

$$[A] = \begin{bmatrix} 1 & 0 & 0 & --- & 0 & 0 \\ \Delta_1 & 1 & 0 & --- & 0 & 0 \\ \Delta_2 & \Delta_1 & 1 & --- & 0 & 0 \\ \Delta_3 & \Delta_2 & \Delta_1 & --- & 0 & 0 \\ | & | & | & & | & | \\ \Delta_{N-3} & \Delta_{N-4} & \Delta_{N-5} & --- & 1 & 0 \\ \Delta_{N-2} & \Delta_{N-3} & \Delta_{N-4} & --- & \Delta_1 & 1 \end{bmatrix} \qquad \text{(A-1)}$$

This matrix is unit lower triangular,[6] thus its inverse is also unit lower triangular and it can be obtained in a straightforward manner. By definition of an inverse, if

$$[A]^{-1} = \begin{bmatrix} 1 & 0 & 0 & --- & 0 & 0 \\ a_1 & 1 & 0 & --- & 0 & 0 \\ a_2 & a_1 & 1 & --- & 0 & 0 \\ | & | & | & & | & | \\ a_{N-3} & a_{N-4} & a_{N-5} & --- & 1 & 0 \\ a_{N-2} & a_{N-3} & a_{N-4} & --- & a_1 & 1 \end{bmatrix} \qquad \text{(A-2)}$$

then it follows that:

$$[A]^{-1} \; [A] = [I] \tag{A-3}$$

or in detail

$$\begin{bmatrix} 1 & 0 & 0 & --- & 0 \\ a_1 & 1 & 0 & --- & 0 \\ a_2 & a_1 & 1 & --- & 0 \\ | & | & | & & | \\ a_{N-2} & a_{N-3} & a_{N-4} & --- & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -- & 0 \\ \Delta_1 & 1 & 0 & -- & 0 \\ \Delta_2 & \Delta_1 & 1 & -- & 0 \\ | & | & | & & | \\ \Delta_{N-2} & \Delta_{N-3} & \Delta_{N-4} & -- & 1 \end{bmatrix} =$$

$$\begin{bmatrix} 1 & 0 & 0 & --- & 0 \\ 0 & 1 & 0 & --- & 0 \\ 0 & 0 & 1 & --- & 0 \\ | & | & | & & | \\ 0 & 0 & 0 & --- & 1 \end{bmatrix}$$

From Eq. (A-3) it follows that

$$a_1 + \Delta_1 = 0, \text{ or } a_1 = -\Delta_1 \tag{A-4}$$

and

$$a_2 + a_1 \Delta_1 + \Delta_2 = 0, \qquad \text{or} \quad a_2 = -\Delta_2 + \Delta_1^2 \tag{A-5}$$

Similarly,

$$a_3 + a_2 \Delta_1 + a_1 \Delta_2 + \Delta_3 = 0$$

or

$$a_3 = - \Delta_3 + 2 \Delta_1\Delta_2 - \Delta_1^3 \qquad \text{(A-6)}$$

In the same manner, multiplication of the fifth row of $[A]^{-1}$ into the first column of [A] yields:

$$a_4 + a_3 \Delta_1 + a_2 \Delta_2 + a_1 \Delta_3 + \Delta_4 = 0$$

and thus $a_4 = - \Delta_4 + 2 \Delta_1\Delta_3 - 3\Delta_1^2\Delta_2 + \Delta_2^2 + \Delta_1^4$ (A-7)

Employing the same procedure, it can be shown that[19]

$$a_5 = - \Delta_5 + 2\Delta_4 \Delta_1 + 2\Delta_3\Delta_2 - 3\Delta_3\Delta_1^2 - 3\Delta_2^2\Delta_1 + 4\Delta_2\Delta_1^3 - \Delta_1^5 \qquad \text{(A-8)}$$

$$a_6 = \Delta_6 + 2\Delta_5\Delta_1 + 2\Delta_4\Delta_2 - 3\Delta_4\Delta_1^2 + \Delta_3^2 - 6\Delta_3\Delta_2\Delta_1$$

$$+ 4\Delta_3\Delta_1^3 + \Delta_2^3 + 6\Delta_2^2\Delta_1^2 - 5\Delta_2\Delta_1^4 + \Delta_1^6 \qquad \text{(A-9)}$$

$$a_7 = - \Delta_7 + 2\Delta_6\Delta_1 + 2\Delta_5\Delta_2 - 3\Delta_5\Delta_1^2 + 2\Delta_4\Delta_3$$

$$- 6\Delta_4\Delta_2\Delta_1 + 4\Delta_4\Delta_1^3 - 3\Delta_3^2\Delta_1 - 3\Delta_3\Delta_2^2 + 12\Delta_3\Delta_2\Delta_1^2$$

$$- 5\Delta_3\Delta_1^4 + 4\Delta_2^3\Delta_1 - 10\Delta_2^2\Delta_1^3 + 6\Delta_2\Delta_1^5 - \Delta_1^7 \qquad \text{(A-10)}$$

$$a_8 = -\Delta_8 + 2\Delta_7\Delta_1 + 2\Delta_6\Delta_2 - 3\Delta_6\Delta_1^2 + 2\Delta_5\Delta_3$$

$$- 6\Delta_5\Delta_2\Delta_1 + 4\Delta_5\Delta_1^3 + \Delta_4^2 - 6\Delta_4\Delta_3\Delta_1$$

$$- 3\Delta_4\Delta_2^2 + 12\Delta_4\Delta_2\Delta_1^2 - 5\Delta_4\Delta_1^4 - 3\Delta_3^2\Delta_2$$

$$+ 6\Delta_3^2\Delta_1^2 + 12\,\Delta_3\Delta_2^2\Delta_1 - 20\Delta_3\Delta_2\Delta_1^3$$

$$+ 6\Delta_3\Delta_1^5 - \Delta_2^4 - 10\Delta_2^3\Delta_1^2 + 15\Delta_2^2\Delta_1^4 - 7\Delta_2^2\Delta_1^6 - \Delta_1^8 \qquad \text{(A-11)}$$

The form of the expression for terms up to a_{10} may be found in Reference 18 and for terms greater than a_{10} in Reference 19.

APPENDIX B

BEHAVIOR OF THE ABSOLUTE SUM OF THE TRUNCATED VALUES OF g(t)

It was shown in Section II, that for any absolutely stable linear system, g(t) will, in the general case, have the following form when t is sufficiently large

$$g(t) = Kt^n \exp(-at) \exp(j\omega_i t) \tag{B-1}$$

where n is a positive integer or zero; a, ω_i are positive constants.

Under the assumption that g(t) is of the above form for $t \geq NT$, it follows from Eq. (B-1) that, since $\left|\exp(j\omega_i t)\right| = 1$,

$$\left|g(t)\right| = K\, t^n \exp(-at) \tag{B-2}$$

and

$$\left|g_k\right| = K(kT)^n \exp(-akT) \tag{B-3}$$

Consequently the summation of the absolute values of the truncated portion of g(t) is

$$\sum_{k=N}^{\infty} \left|g_k\right| = K \sum_{k=N}^{\infty} (kT)^n \exp(-akT) \tag{B-4}$$

In order to evaluate Eq. (B-4) it is necessary to apply Maclaurin's Integral Theorem[7]

To apply this theorem, the following must hold:

$$\frac{|g_{k+1}|}{|g_k|} \leq 1 \qquad \text{(B-5)}$$

Substituting Eq. (B-3) into Eq. (B-5)

$$\frac{[(k+1)T]^n \ \exp[-a(k+1)T]}{(kT)^n \ \exp(-akT)} \leq 1$$

and it follows that

$$\left(\frac{k+1}{k}\right)^n \leq \exp(aT)$$

or

$$\ln\left(\frac{k+1}{k}\right) \leq \frac{aT}{n} \qquad \text{(B-6)}$$

If Eq. (B-6) is satisfied for all k $\geq$ N, the hypothesis of Maclaurin's Integral Theorem is satisfied and

$$\sum_{k=N}^{\infty} (kT)^n \exp(-akT) \leq T^n \int_N^{\infty} x^n \exp(-aTx)\, dx \qquad \text{(B-7)}$$

$$\leq \frac{\exp(-aNT)}{a^{n+1}\ T} \left[(aNT)^n + n(aNT)^{n-1} + \dots + n! \right]$$

For any finite n, the bracketed expression in Eq. (B-7) is of less than exponential order in (aNT), and thus the term exp(-aNT) dominates. Thus, for any $\frac{\epsilon}{K} > 0$, there exists an integer P such that for any $N > P$

$$\frac{\exp(-aNT)}{a^{n+1}\,T}\left[(aNT)^n + n(aNT)^{n-1} + \ldots + n!\right] < \frac{\epsilon}{K}$$

or

$$\sum_{k=N}^{\infty} |g_k| < \epsilon \qquad \text{(B-8)}$$

Thus, Eq. (B-8) states that for any absolutely stable linear system,

$\sum_{k=N}^{\infty} |g_k|$ is bounded and can be made arbitrarily small by choice of N.

REFERENCES

Books

1. E. Mishkin and L. Braun, Jr., "Adaptive Control Systems," McGraw-Hill Book Company, New York, 1961.

2. W. W. Seifert and C. W. Steeg, "Control Systems Engineering," McGraw-Hill Book Company, New York, 1960.

3. M. F. Gardner and J. L. Barnes, "Transients in Linear Systems," J. Wiley and Sons, New York, 1942.

4. J. G. Truxal, "Control System Synthesis," McGraw-Hill Book Company, New York, 1955.

5. S. Seshu and N. Balabanian, "Linear Network Analysis," J. Wiley and Sons, New York, 1953.

6. A. S. Householder, "Principles of Numerical Analysis," McGraw-Hill Book Company, New York, 1959.

7. F. H. Miller, "Calculus," J. Wiley and Sons, New York, 1948

8. R.S. Burington, "Handbook of Mathematical Tables and Formulas," Handbook Publishers, Inc., Sandusky, Ohio, 1953.

9. D.A.S. Fraser, "Statistics - An Introduction," J. Wiley and Sons, New York, 1958.

Bibliographies

10. J. A. Aseltine, et al., "A Survey of Adaptive Control Systems," I. R. E. Transactions on Automatic Control, PGAC-6 December, 1958, p. 102.

11. R. Stromer, "Adaptive or Self-Optimizing Control Systems - A Bibliography", I. R. E. Transaction on Automatic Control, PGAC-4, May 1959, p. 65.

Periodicals and Technical Reports

12. G. W. Anderson, et al., "A Self-Adjusting System for Optimum Dynamic Performance," IRE National Convention Record, part 4, pp. 182-190, 1958.

13. D. C. Kowalski, "Statistical Stability Monitor," Report No. 11307R, Bendix Corporation - Research Laboratories Division, May, 1960.

14. S. S. Osder, "Adaptive Flight Control System," Proceeding of the Self Adaptive Flight Control Systems Symposium, WADC T. R. 59-49, pp. 81-122, January, 1960.

15. R. E. Kalman, "Design of a Self-Optimizing Control System," Trans. A.S.M.E., Vol. 80, p. 468, February, 1958.

16. L. Braun, Jr., "On Adaptive Control Systems," IRE Transactions on Automatic Control, PGAC-4, No. 2, p. 30, November, 1959.

17. E. Mishkin and R. A. Haddad, "On the Identification and Command Problems in Computer-Controlled Adaptive Systems," Report No. R-767-59, Microwave Research Institute, Polytechnic Institute of Brooklyn, September, 1959.

18. E. I. Jury and F. W. Semelka, "Time-Domain Synthesis of Sampled Data Control Systems," Trans. A.S.M.E., Vol. 80, No. 8, November, 1958.

19. E. I. Jury and F. W. Semelka, "A Method for Determining the Coefficients of the Modified Z-Transform Expansion," Internal Memo No. 1, Sampled-Data Systems Group, Electronics Research Laboratory, University of California, Berkeley, California, August, 1956.