Title

UNFOLD---A Computer Program for Analyzing Linear Systems by Calculation in the Time Domain

Authors

Horovitz, Mark W.
Nooney, Grove C.

Publication Date

1963-07-01

UCRL-10927

University of California

Ernest O. Lawrence Radiation Laboratory

UNFOLD--A Computer Program for Analyzing Linear Systems by Calculation in the Time Domain

Berkeley, California

DISCLAIMER

Non-Standard

UCRL-10927

UNIVERSITY OF CALIFORNIA

Lawrence Radiation Laboratory
Berkeley, California

Contract No. W-7405-eng-48

UNFOLD--A Computer Program for Analyzing Linear Systems by Calculation in the Time Domain

Mark W Horovitz and Grove C. Nooney

July 1963

UNFOLD--A Computer Program for Analyzing Linear Systems by Calculation in the Time Domain

UCRL-10927

Mark W. Horovitz*
Grove C. Nooney*

Introduction

The kinetics of tracer experiments on steady-state biological systems can be represented, to a useful degree of approximation, by a system of simultaneous linear differential equations. Standard methods of computing the behavior of linear systems use either direct methods of solution of the differential equations or operational techniques (e.g., Laplace transform). In typical engineering problems, these two techniques are effective; in contrast, many biological phenomena, although behaving as linear systems, cannot be conveniently analyzed by such methods, because the input, output, and behavior of the system cannot be represented in terms of mathematical operators or algebraic functions of simple form. Our procedure can easily cope with these difficulties.

Statement of the Problem

In a linear time-invariant system with a single input function of time B(t) and a corresponding output function of time A(t), the B(t) and A(t) are related by the integral equation

$$A(t) = \int_{-\infty}^{t} G(t-T)\,B(T)\,dT, \qquad (1)$$

where G(t) is a weighting function (impulse-response function), characterizing the response of the system. Measured values $A(t_i)$, $B(s_i)$ are given for the times t_i (where $i = 1, 2, \cdots, n$) and s_i (where $i = 1, 2, \cdots, k$). Estimates of the errors on $A(t_i)$, $B(s_i)$ may also be supplied. Program UNFOLD computes a G(t) which gives the best-fitting solution of Eq. (1). Conversely the program can also be used to compute A(t) from B(t) and G(t).

Procedure

1. Expansion in terms of a set of functions. We first represent A(t) and B(t) as expansions in terms of a set of functions. The method is detailed here only for A(t); the procedure for B(t) is directly analogous. We assume the validity for A, B, and G, of expansions of the form

$$A(t) \approx \sum_{j=1}^{m} a_j F_j(t),$$

where $\{F_j\}$ is some possibly complete or orthonormal set of functions. For simplicity we assume that: (a) A and B are specified at the same n values of t, (b) the same number (m) of functions F_i, is used in the expansions for A, B, and G, and (c) all measured points are equally weighted. (These simplifying restrictions are not essential.) We write $A_i = A(t_i)$. The desired values of a_j yield the minimum of

$$X_A = \sum_{i=1}^{n} \left[A_i - \sum_{j=1}^{m} a_j F_j(t_i) \right]^2$$

At the minimum,

$$\frac{\partial X_A}{\partial a_k} = 0 \text{ for each k, that is,}$$

$$\frac{\partial X_A}{\partial a_k} = -\sum_{i=1}^{n} F_k(t_i) \left\{ A_i - \sum_{j=1}^{m} a_j F_j(t_i) \right\} = 0, \text{ for } k=1,2\cdots,m \qquad (2)$$

We now simplify the notation by writing

$$F_k(t_i) = F_{ki}.$$

Equation (2) then becomes

$$\sum_{i=1}^{n} \left[A_i F_{ki} - \sum_{j=1}^{m} a_j F_{ki} F_{ji} \right] = 0, \text{ for } k=1,2,\cdots,m. \qquad (3)$$

Let the n-by-m-dimensional matrix of elements F_{ij} be denoted by F, the n-dimensional column vector of elements A_i by $\vec{A}$, the m-dimensional column vector of elements a_i by $\vec{a}$, and the transposed matrix of F by F^T. We can then rewrite (3) as

$$F\vec{A} = FF^T\vec{a} \qquad (4)$$

or

$$\vec{a} = (FF^T)^{-1} F\vec{A}. \qquad (5)$$

Equation (5) represents our solution for the a_i. We note that (FF^T) is a square, symmetric matrix.

2. Solution for G(t). Having determined the a_i and b_i, we now determine G(t). We prefer to replace Eq. (1) by the equivalent form

$$A(t) = \int_{0}^{t} G(T)\,B(t-T)\,dT, \qquad (6)$$

where we have assumed

$$G(t) = B(t) = 0 \text{ for } t < 0.$$

The function G(t) is replaced by $G(t) = \sum_{i=1}^{m} g_i F_i(t)$.

*Lawrence Radiation Laboratory, University of California, Berkeley, California.

The best-fitting solution is obtained by finding the values of g_i that minimize the expression

$$Z = \int_0^t \left[A(t) - \int_0^t G(T)\,B(t-T)\,dT\right]^2 dt$$

or

$$Z=\int_0^T \left\{\sum_{i=1}^m a_i F_i(t) - \int_0^t \sum_{i=1}^m \sum_{j=1}^m g_i b_j F_i(T) F_j(t-T) dT\right\}^2 dt. \quad (7)$$

The value of T is chosen so that the time from 0 to T includes the interval of interest. At the minimum we have

$$\frac{\partial Z}{\partial g_k} = 0 \quad \text{for } k = 1, 2, \cdots, m.$$

This condition leads to

$$\int_0^T \left\{\sum_{i=1}^m a_i F_i(t) - \int_0^t \sum_{i=1}^m \sum_{j=1}^m g_i b_j F_i(T) F_j(t-T) dT\right\}$$

$$\times \left\{\int_0^t \sum_{j=1}^m b_j F_k(T) F_j(t-T) dT\right\} dt = 0, \quad \text{for } k=1,2,\cdots,m. \quad (8)$$

We now write

$$c_k(t) = \int_0^t \sum_{j=1}^m b_j F_k(T) F_j(t-T) dT \quad (9)$$

and rewrite (8) as

$$\int_0^T \sum_{i=1}^m a_i\, F_i(t)\, c_k(t)\, dt$$

$$= \int_0^T \left[\int_0^t \sum_{i=1}^m \sum_{j=1}^m g_i b_j F_i(T) F_j(t-T) dT\right] c_k(t) dt, \quad \text{for } k=1, 2, \cdots, n. \quad (10)$$

We now write

$$D_k = \int_0^T \sum_{i=1}^m a_i\, F_i(t)\, c_k(t)\, dt,$$

and

$$E_{ki} = \int_0^T \left[\int_0^t \sum_{j=1}^m b_j F_i(T) F_j(t-T)\, dT\right] c_k(t)\, dt \quad (11)$$

Notice that owing to the variable upper limit of integration, Eqs., (9), (10), and (11) do not reduce to a simpler form when the functions $F_i(t)$ are orthogonal (over a specified range).

Then (10) becomes

$$D_k = \sum_{i=1}^m g_i E_{ki} \text{ or } \vec{D} = E\vec{g} \text{ for } k=1, 2, \cdots, m.$$

The solution is

$$\vec{g} = (E^T E)^{-1} E^T \vec{D}. \quad (12)$$

We then reconstruct G(t) by writing

$$G(t) = \sum_{i=1}^n g_i\, F_i(t). \quad (13)$$

3. Determination of A(t). As a check on our solution for G(t), we use B(t) and the computed G(t) to evaluate $A^*(T)$ by

$$A^*(t) = \int_0^t B(t-T)\, G(T)\, dT; \quad (14)$$

$A^*(t)$ should equal A(t).

When B(t) and G(t) are known, we can use Eq. (14) to determine A(t).

Options

Flexibility in the method of calculation is provided by permitting the user to choose one of several alternative modes of operation of the program. These options provide a choice of: (a) set of expansion functions, (b) form of the error weighting function, (c) integration procedure, (d) fitting criteria, (e) output format, (f) solving for G, given A and B, or solving for A, given G and B.

Applications

The method described can be used to analyze systems of linear differential equations; it yields a single function characterizing the response of the entire system. It is useful even when the usual treatment, involving direct solution of the differential equations or operator techniques, cannot be easily applied. The method can also be applied to the solution of general convolution-type integral equations of the first kind. This technique is suitable for analyzing experiments on tracer kinetics of steady-state systems, and may also be of value in the study of electrical networks, adaptive control systems, or instrumental resolution functions.

Acknowledgment

We are indebted to Dr. Aldo Rescigno (Donner Laboratory of Biophysics and Medical Physics) for suggesting this problem and for his advice, support, and encouragement.

Summary

The integral equation

$$A(t) = \int_{-\infty}^t G(t-T)\, B(T)\, dT$$

relates the input B(t) to the output A(t) in a linear time-invariant system. G(t) is a weighting function (impulse-response function), characterizing the response of the system. From given values of B(t) and A(t) at several values of t, Program UNFOLD computes a G(t) that best fits the weighting function, in a certain sense. Conversely, when the values of B(t) and G(t) are given the program can be used to compute A(t). UNFOLD proceeds by first expanding A(t), B(t), and G(t) in terms of a set of functions; G(t) is then obtained by a least-squares procedure.