

AD-A092 226

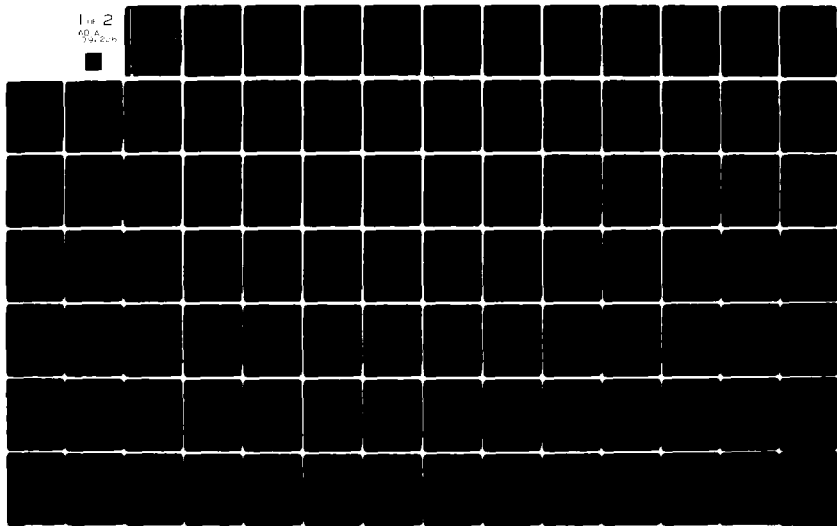
ROCHESTER INST OF TECH NY DEPT OF ELECTRICAL ENGINEERING F/6 9/3
EXTENSION OF PENCIL-OF-FUNCTIONS METHOD TO REVERSE-TIME PROCESS--ETC(U)
AUG 80 V K JAIN, T K SARKAR, D D WEINER N00014-79-C-0598
TR-80-4 ML

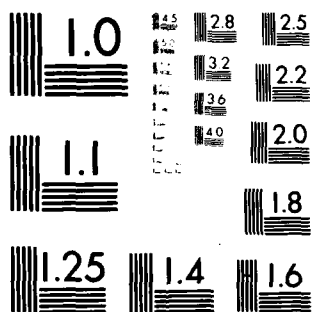
UNCLASSIFIED

1 of 2

AD-A092 226

78-0000





MICROCOPY RESOLUTION TEST CHART

NATIONAL BUREAU OF STANDARDS-1963-A

AD A092226

LEVEL

10
B.S.

EXTENSION OF PENCIL-OF-FUNCTIONS
METHOD TO REVERSE-TIME
PROCESSING WITH FIRST-ORDER DIGITAL FILTERS

BY

Vijay K. Jain¹
Tapan K. Sarkar²
Donald D. Weiner³

PTIC
ELECTE
DEC 1 1980

THIS DOCUMENT IS BEST QUALITY PRACTICABLE.
THE COPY FURNISHED TO DDC CONTAINED A
SIGNIFICANT NUMBER OF PAGES WHICH DO NOT
REPRODUCE LEGIBLY.

1. Department of Electrical Engineering
University of South Florida
Tampa, Florida 33620

3. Department of Electrical Engineering
Syracuse University
Syracuse, New York 13210

2. Department of Electrical Engineering
Rochester Institute of Technology
Rochester, New York 14623

DISTRIBUTION STATEMENT A
Approved for public release
Distribution Unlimited

FILE COPY

80 10 14 119

DISCLAIMER NOTICE

**THIS DOCUMENT IS BEST QUALITY
PRACTICABLE. THE COPY FURNISHED
TO DTIC CONTAINED A SIGNIFICANT
NUMBER OF PAGES WHICH DO NOT
REPRODUCE LEGIBLY.**

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

12 97

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER (14) TR-80-4, TR-3 ✓ AD A092226	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) (6) Extension of Pencil-of-Functions Method to Reverse-Time Processing with First-Order Digital Filters	5. TYPE OF REPORT & PERIOD COVERED (4) Technical Report No. 3	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) (10) Vijay K. Jain, Tapan K. Sarkar and Donald D. Weiner	8. CONTRACT OR GRANT NUMBER(s) (15) N00014-79-C-0598	
9. PERFORMING ORGANIZATION NAME AND ADDRESS Dept. of Electrical Engineering ✓ Rochester Institute of Technology Rochester, New York 14623	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS (17) 61153N RR021-01 RR021 01 01 NR371-014	
11. CONTROLLING OFFICE NAME AND ADDRESS Department of the Navy Office of Naval Research Arlington, Virginia 22217	12. REPORT DATE (11) August 1980	13. NUMBER OF PAGES 91
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) (16) RR02101	15. SECURITY CLASS. (of this report) UNCLASSIFIED	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) DISTRIBUTION STATEMENT A Approved for public release; Distribution Unlimited		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Pencil-of-functions System Identification Target Identification		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) In this presentation, the data signal is processed in reverse-time by a cascade of first order digital filters to yield a family of information signals. The Gram matrix of these information signals is shown to contain the essential information on the poles of the signal. The entire procedure of the application of pencil-of-function method is thus noniterative. Examples presented demonstrate (i) noiseworthiness in the representation problem when data are corrupted by noise, and (ii) the effectiveness of the method in the approximation problem.		

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

411977 HW

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Comparison of the method with the maximum entropy method (or linear predictor) and the Prony method is also included.

Accession For	
NTIS GRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification <i>Per on file</i>	
By _____	
Distribution/	
Availability Codes	
Avail and/or	
Dist	Special
<i>A 23</i>	<i>40</i>

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

TABLE OF CONTENTS

<u>Section</u>	<u>Title</u>	<u>Page</u>
I	Introduction	1
II	First-Order Fiter Based Pencil-of-Functions	
	Method for Modeling Impulse Responses	3
III	Parallelism and Orthogonality of Information	
	Signals	9
IV	Program Description	16
V	Application Examples	22
	References	34
	Appendix A -- Details of Program	35
	POF-FILTER	
	Appendix B -- Modeling of a Noisy	79
	Test Signal	
	Appendix C -- Modeling of a Scatterer	88
	Response	

LIST OF FIGURES

<u>Figure</u>	<u>Title</u>	<u>Page</u>
1	Generation of information signals by use of first-order filters	9
2	Magnitude characteristic of $\mu_1(z) = \frac{1-q}{1-qz^{-1}}$	10
3	Normalized cutoff frequency vs. parameter q	10
4	Magnitude vs Freq. of connection filters	11
5	Idealized connection filters and BP constituents	12
6	Comparison of true and model impulse responses	24
7	Comparison of true and model impulse responses	25
8	Comparison of true and model impulse responses	26
9	Comparison of true and model impulse responses	27
10	Comparison of true and model impulse responses	28
11	Comparison of true and model impulse responses	29
12	Comparison of measured data (of the response of a conducting pipe) and model impulse responses	32
13	Comparison of the derivative of measured data and the corresponding model impulse response	33
B1	True impulse response of a third order transfer function	80
B2	A simulated noisy signal under test	81

EXTENSION OF PENCIL OF FUNCTIONS METHOD TO REVERSE-TIME PROCESSING WITH FIRST-ORDER DIGITAL FILTERS

I. INTRODUCTION

Signal representation and approximation is basic to (a) time-domain extraction of singularities of a scatterer's field pattern. It is also useful in (b) bandwidth compression of signals, and (c) time-domain measurement and testing of networks/channels. This report discusses a unified approach to representing or approximating a given empirical signal by sum of exponentials, i.e., for finding the right hand side of

$$h_d(t) \cong h(t) = \sum_{i=1}^n W_i e^{\lambda_i t} \leftrightarrow H(s) = \sum_{i=1}^n \frac{W_i}{s - \lambda_i}$$

$$= \frac{\beta_{n-1}s^{n-1} + \beta_{n-2}s^{n-2} + \dots + \beta_0}{s^n + \alpha_{n-1}s^{n-1} + \dots + \alpha_0}$$

or, equivalently, the right hand side of the sampled version

$$h_d(k) = h(k) = \sum_{i=1}^n R_i (z_i)^k \leftrightarrow H(z) = \sum_{i=1}^n \frac{R_i}{(1 - z_i z^{-1})}$$

$$= \frac{b_0 + b_1 z^{-1} + \dots + b_{n-1} z^{-n+1}}{1 + a_1 z^{-1} + \dots + a_n z^{-n}}$$

The poles λ_i (or γ_i in z-domain) are either real, or they occur in complex conjugate pairs.

In the method described here, the data signal is processed in reverse-time by a cascade of first order digital filters -- each $\mu(z) = 1/(1 - qz^{-1})$, to yield a family of information signals. The Gram matrix G of these information signals is shown to contain the essential information on the denominator parameters of $H(z)$. Specifically, it is shown that $A(z)$ is determined as

$$A(z) = z^{-n} \left[\sum_{i=1}^{n+1} \sqrt{D_i} (z-1)^{n+1-i} \right] / \sqrt{D_1}$$

where D_i are the diagonal cofactors of the matrix G . The numerator parameters are then determined using a least-squares fit, i.e., $\underline{b} = -P^{-1}\underline{v}$, where P and $\underline{v}$ are defined in the paper.

The entire procedure is thus noniterative and computationally efficient. It is a further generalization of the method developed in [4]. Examples presented demonstrate (i) noiseworthiness in the representation problem when data are corrupted by noise and (ii) the effectiveness of the method in the approximation problem. Comparison of the method with the maximum entropy method (or linear predictor) and the Prony method is also included in the report.

The structure of the report is as follows. The fundamental results relating to the signal representation/approximation problem, through the use of reverse-time processing by first-order digital filters, are obtained in Chapter II. The important question of parallelism and orthogonality of the information signals is explored in Section III. Chapter IV gives the input description and user instructions for the program POF-FILTER which implements the method of Chapter II. Application examples as well as comparison with other methods, are given in Section V. Appendices A gives listing and description of program POF-FILTER and its routines. Appendices B and C contain the computer outputs relating to Example 1 and Example 2, respectively, of Section V.

SECTION II

FIRST-ORDER FILTER BASED PENCIL-OF-FUNCTIONS

METHOD FOR MODELING IMPULSE RESPONSES

We shall be interested in modeling the impulse response [1]-[3]

$$y(k) = \sum_{\ell=1}^n R_{\ell} (z_{\ell})^k \leftrightarrow \frac{B(z)}{A(z)} = \frac{b_1 z^{-1} + \dots + b_n z^{-n}}{1 + a_1 z^{-1} + \dots + a_n z^{-n}} \quad (1)$$

from its numerical data. Suppose a suitable K has been selected such that $y(k) = 0$ for $k > K$ (so that use of the upper limit K instead of ∞ on summations may be permitted). We define the reverse-time first-order filtered signals as

$$\begin{aligned} y_1(k) &= y(k) \\ y_2(k) &= qy_2(k+1) + y_1(k) \\ &\vdots \\ y_N(k) &= qy_N(k+1) + y_n(k) \end{aligned} \quad (2)$$

where $N=n+1$, and $y_i(K+1)=0$ for $i=1,2,\dots,N$. Further, $0 < q < 1$.

This family of signals, which we shall call information signals, exhibits the interesting property stated below.

Lemma 1

$$y_{i+1}(k) = \sum_{\ell=1}^n \frac{R_{\ell}}{(1-q z_{\ell})^i} (z_{\ell})^k \quad (3)$$

Proof: We prove this by induction. For $i=0$ the statement is trivially true since it is identical to (1) for this case. Assuming it to be true for $i-1$, let us proceed to prove it is true for i .

From (2)

$$y_{i+1}(k) = qy_{i+1}(k+1) + y_i(k) \quad (4)$$

which is readily shown to be equivalent to

$$\begin{aligned}
 y_{i+1}(k) &= \sum_{v=k}^{\infty} q^{v-k} y_i(v) \\
 &= \sum_{\ell=1}^n \frac{R_{\ell}}{(1-q z_{\ell})^{i-1}} \sum_{v=k}^{\infty} q^{v-k} (z_{\ell})^v \quad (\text{from induction hypothesis}) \\
 &= \sum_{\ell=1}^n \frac{R_{\ell}}{(1-q z_{\ell})^{i-1}} z_{\ell}^k \sum_{v=k}^{\infty} q^{v-k} (z_{\ell})^{v-k}
 \end{aligned}$$

The result of equation (3) follows immediately by observing that

$$\sum_{v=k}^{\infty} q^{v-k} (z_{\ell})^{v-k} = \frac{1}{(1-q z_{\ell})}$$

This Lemma leads us to the crucial observation stated next.

Lemma 2

The set

$$(qz_m - 1)y_2 + y_1, (qz_m - 1)y_3 + y_2, \dots, (qz_m - 1)y_N + y_n \quad (5)$$

is linearly dependent for $m=1,2,\dots,n$ where z_m are the poles of the right hand side of (1).

Note: We have used the notation y_i to denote the sequence $\{y_i(k)\}$,

$k=0,1,2,\dots$

Proof: In view of (3) we find

$$(qz_m - 1)y_{i+1}(k) + y_i(k) = \sum_{\substack{\ell=1 \\ \ell \neq m}}^n \frac{q(z_m - z_{\ell})R_{\ell}}{(1-q z_{\ell})^i} (z_{\ell})^k \quad (6)$$

for $i=1,2,\dots,n$. Clearly, the sequences $(qz_m - 1)y_{i+1} + y_i$, $i=1,\dots,n$ each contain only $n-1$ modes $(z_{\ell})^k$, $\ell \neq m$ hence are linearly dependent.

We can now apply the pencil-of-functions theorem of reference [4] to obtain the central theoretical result of this section.

We will call z_ℓ (see (1)) the poles of the impulse response, R_ℓ the corresponding residues, and $p_\ell = \{(z_\ell)^k\}$ the associated modes. Note that the poles occur in conjugate pairs whenever complex, as do the residues, since y is real.

Define the $N \times N$ dimensional Gram matrix (recall, $N=n+1$) [6]

$$F = \begin{bmatrix} \langle y_1, y_1 \rangle & \dots & \langle y_1, y_N \rangle \\ \vdots & \ddots & \vdots \\ \langle y_N, y_1 \rangle & \dots & \langle y_N, y_N \rangle \end{bmatrix}, \quad \langle y_i, y_j \rangle = \sum_{k=1}^K y_i(k) y_j(k) \quad (7a)$$

or, equivalently,

$$F = \sum_{k=1}^K \underline{f}(k) \underline{f}^T(k) \quad (7b)$$

where $\underline{f}^T(k) = [y_1(k) \ y_2(k) \ \dots \ y_N(k)]$.

Theorem 1

The poles of the impulse response $y(k)$ must satisfy the equation

$$\sum_{i=1}^N \sqrt{D_i} (qz-1)^{N-1} = 0 \quad (8)$$

where D_i are the diagonal cofactors of the Gram matrix F (defined in (7))

Proof: The theorem follows immediately upon application of the pencil of functions theorem (reference [4]) to the set (5).

Note that the denominator of transform of the impulse response is given by

$$A(z) = D_1^{-1/2} (qz)^{-n} \sum_{i=1}^N \sqrt{D_i} (qz-1)^{N-1} \quad (9)$$

This follows from (8) by dividing through by z^N and by normalizing the coefficients so that the leading coefficient becomes unity.

Determination of the denominator polynomial $A(z)$ completes the first of the two steps in the decoupled pencil-of-functions method. The next step, finding the numerator coefficients in $B(z)$, or equivalently finding the residues R_i , can be accomplished in two alternative ways.

The first method consists in solving for the residues from the equation¹

$$\begin{bmatrix} \frac{1}{1-qz_1} & \frac{1}{1-qz_2} & \cdots & \frac{1}{1-qz_n} \\ \frac{1}{(1-qz_1)^2} & \frac{1}{(1-qz_2)^2} & \cdots & \frac{1}{(1-qz_n)^2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{(1-qz_1)^N} & \frac{1}{(1-qz_2)^N} & \cdots & \frac{1}{(1-qz_n)^N} \end{bmatrix} \begin{bmatrix} R_1 \\ R_2 \\ \vdots \\ R_n \end{bmatrix} = \begin{bmatrix} y_2(0) \\ y_3(0) \\ \vdots \\ y_N(0) \end{bmatrix} \quad (10)$$

This equation follows from (3) upon setting $k=0$ and letting i range from 1 to n . Clearly, the use of this equation requires that the poles z be determined from the denominator $A(z)$ by use of a root-finding routine. This requirement is of no consequence if the final answers are needed in the s -domain, for conversion to s -domain involves finding the roots of $A(z)$ anyway.

The alternative approach is to find the optimum least-squares numerator coefficients (given the denominator of (9)) through the equation

¹ The equation corresponding to $i=0$ is ignored in formulating (10) because of the relatively poor signal/noise statistics of $y_1(0)$ (compared to, say, $y_2(0)$) when the impulse response data contain additive wideband noise. This statement assumes that the bandwidth of $y(k)$ is much smaller than that of the additive noise.

$$\begin{bmatrix} \langle w_1, w_1 \rangle & \langle w_1, w_2 \rangle & \dots & \langle w_1, w_n \rangle \\ \langle w_2, w_1 \rangle & \langle w_2, w_2 \rangle & \dots & \langle w_2, w_n \rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle w_n, w_1 \rangle & \langle w_n, w_2 \rangle & \dots & \langle w_n, w_n \rangle \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} = \begin{bmatrix} \langle y, w_1 \rangle \\ \langle y, w_2 \rangle \\ \vdots \\ \langle y, w_n \rangle \end{bmatrix} \quad (11)$$

where w_1 denotes the impulse response of $z^{-1}/A(z)$. Note that $w_1(k)=w(k-1)$ where $w(k)$ is the impulse response (i.e., inverse z-transform) of $1/A(z)$. All inner products are summed from $k=0$ to K .

Discussion

Equations (9) and (11) have been implemented in a computer program "POF-FILTER" written in FORTRAN IV. The program is presented in Section III.

The idea of reverse-time integration was proposed by Carr in [7] and Jain in [8]. Here, we have generalized the concept of reverse-time processing to the case of first-order filter processing. Note that the first order filter $1/(1-qz^{-1})$, used above, encompasses integration; just let $q=1$.

It should be borne in mind that the approach developed above is applicable to impulse responses only. With some effort, it may be modified for use to step responses and square-pulse responses. For more general inputs, however, one must use the coupled approach discussed in [9] (which of course involves greatly increased computations, e.g., the Gram matrix involved is of an order twice as high as in the decoupled procedure).

The decoupled approach can be used only if the data is of sufficient length K such that $y(k \geq K)=0$ for all practical purposes.

The reverse-time processing of $y(k)$ through the first order filters can be interpreted as forward-time processing of the signal $h(k)=y(K-k)$ through the same filters.

The use of square-roots of cofactors of the Gram matrix is analogous to the use of square-root factorization [10] of the Gram matrix. Attendant advantages are therefore expected to be realized. A more detailed analysis of this connection will be discussed elsewhere.

The transfer function of the first order filter used in equation (2) is $\mu(z) = 1/(1-qz^{-1})$. Instead, we could use filters with transfer function $\mu_1(z) = (1-q)/(1-qz^{-1})$; these filters have a d.c. gain equal to unity and the ratio of the output power to the input power is a direct measure of the extent of the rejection of higher frequencies. Equation (9) remains valid even when these unity d.c. gain filters are used.

Note that the first-order filtering (in reverse time) is achieved in (2) recursively, without the need to carry out discrete convolution.

SECTION III

PARALLELISM AND ORTHOGONALITY OF INFORMATION SIGNALS

Here we consider the important matter of parallelism and orthogonality of the information signals. In the last section we processed the signal $h(k) = y(K-k)$ through first order filters $\mu_1(z) = (1-q)/(1-qz^{-1})$. This is depicted in Fig. 1. Note that forward time processing of $h(k)$ is equivalent to reverse-time processing of $y(k)$. Because of familiarity with forward-time processing we will carry out the discussion below in terms of the signal $h(k)$. Also note that $h_i(k) = y_i(K-k)$, $i=1,2,\dots,N$. Finally, we remark that the Gram matrix and the properties of parallelism/orthogonality of the two families of signals $h_1, h_2, \dots, h_N$ and $y_1, y_2, \dots, y_N$ are identical. Indeed, $\langle h_i, h_j \rangle = \langle y_i, y_j \rangle$ and $\langle h_i, h_j \rangle = \langle y_i, y_j \rangle$ for $i, j=1,2,\dots,N$.

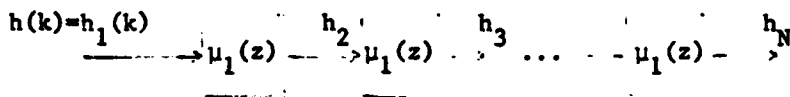


Fig. 1. Generation of information signals by use of first-order filters.

The magnitude (vs. frequency) characteristic of the first order filter $\mu_1(z)$ is shown in Fig. 2. The cutoff frequency (- 3dB point) ω_c is related to the parameter q as [5]

$$\omega_c = \frac{1}{\Delta} \ln\left(\frac{1}{q}\right) \quad (12)$$

where Δ is the sampling interval. Defining Ω as the normalized frequency (i.e., the ratio ω/ω_s where ω_s is the sampling frequency in radian/s) we can express the normalized cutoff frequency Ω_c as

$$\Omega_c = \frac{1}{2\pi} \ln(1/q) \quad (13)$$

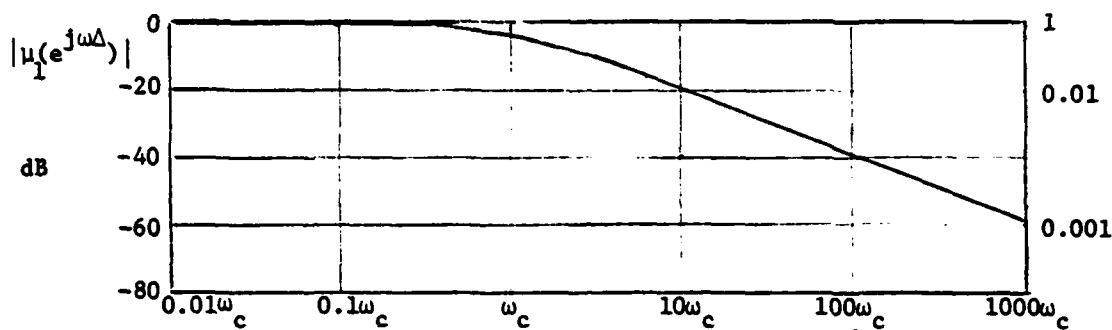


Fig. 2. Magnitude characteristic of $\mu_1(z) = \frac{1-q}{1-qz^{-1}}$

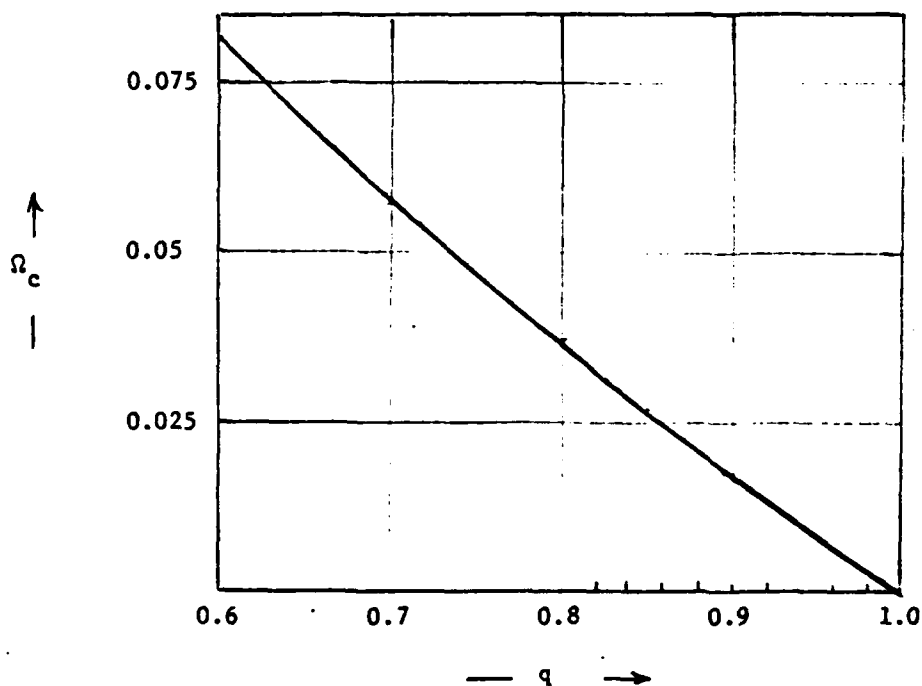


Fig. 3 Normalized cutoff frequency vs. parameter q

The normalized cutoff frequency Ω_c is plotted against the filter parameter q in Fig. 3.

To analyze the parallelism/orthogonality (PO) properties of the information signals let us define the connection filters

$$\begin{aligned} M_0(z) &= 1 \\ M_i(z) &= (\mu_1(z))^i, \quad i=1,2,\dots,n \end{aligned} \quad (14)$$

so that we may write

$$H_{i+1}(z) = M_i(z) H_i(z) \quad (15)$$

Note that $M_i(z)$ is an i-pole filter with a pole of multiplicity 1 at $z=q$.

The magnitude characteristic of the filters $M_0, \dots, M_4$ are shown in Fig. 4 for $q=0.8$.

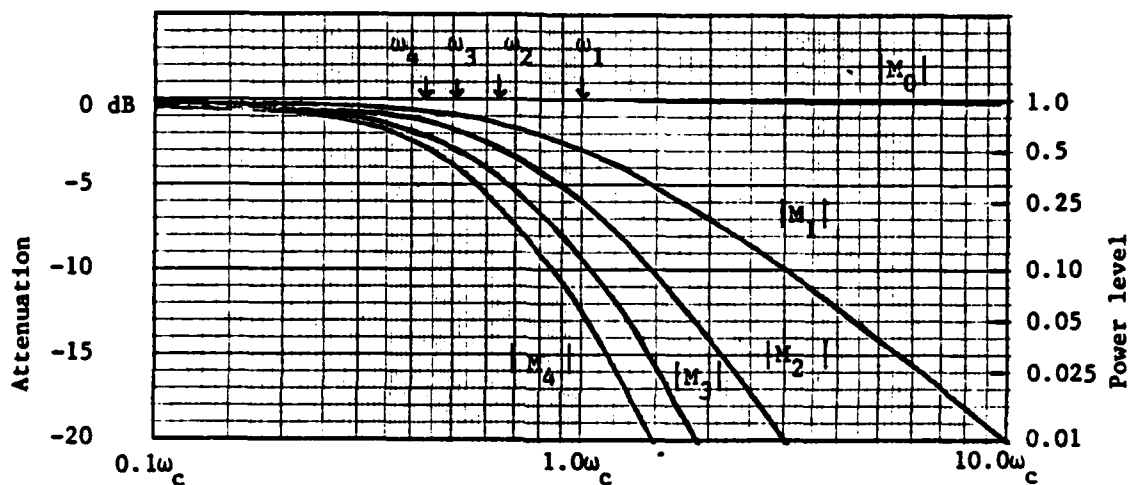


Fig. 4. Magnitude vs. Freq. of connection filters

We now present an approximate, and somewhat heuristic, analysis of the PO properties of the family of information signals.

Approximations -

We approximate the magnitude characteristics of the connection filters M_i as shown in Fig. 5a.

The phase properties of M_i will be ignored (assumed identically zero)

It is then possible to write

$$M_i(e^{j\omega\Delta}) = L_1(\omega) + \dots + L_n(\omega) \quad (16)$$

where $L_i(\omega)$ are the bandpass filter characteristics shown in Fig. 5b.

Further, (15) and (16) together yield

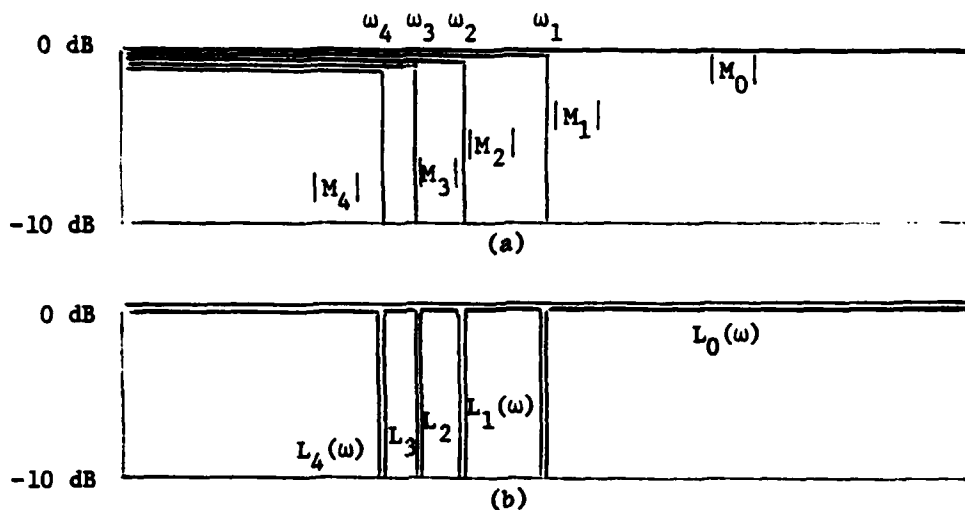


Fig. 5. Idealized connection filters and BP constituents.

$$H_{i+1}(e^{j\omega\Delta}) = (L_1(\omega) + \dots + L_n(\omega)) H_1(e^{j\omega\Delta}) \quad (17a)$$

$$= \phi_1(\omega) + \dots + \phi_n(\omega) \quad (17b)$$

where

$$\phi_i(\omega) = L_i(\omega) H_1(e^{j\omega\Delta}) \quad (18)$$

Clearly, $\phi_i(\omega)$, $i=0,1,\dots,n$ are orthogonal. Indeed,

$$\begin{aligned} \int_{-\omega_s/2}^{\omega_s/2} \phi_i(\omega) \phi_j^*(\omega) d\omega &= \int_{-\omega_s/2}^{\omega_s/2} |H_1(\omega)|^2 L_i(\omega) L_j^*(\omega) d\omega \\ &= 0 \text{ for } i \neq j \end{aligned} \quad (19)$$

because $L_i(\omega)=0$ wherever $L_j(\omega) \neq 0$ and vice-versa. Let us state this in the time domain as

$$\langle \phi_i, \phi_j \rangle = \sigma_{ii} \delta_{ij} \quad (20)$$

where δ_i is the Kronecker delta and $\{\phi_i(k)\} = \mathcal{F}^{-1} \phi_i(\omega)$.

We can summarize the above discussion as follows.

Theorem 2

The information signals are approximately tri-orthogonal.

Proof: From (18) and (20) we have

$$\begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ \vdots \\ \vdots \\ h_N \end{bmatrix} \approx \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 0 & 1 & 1 & \dots & 1 \\ 0 & 0 & 1 & \dots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \phi_0 \\ \phi_1 \\ \phi_2 \\ \vdots \\ \vdots \\ \phi_n \end{bmatrix} \quad (21)$$

where $\phi_0, \dots, \phi_n$ are a set of orthogonal signals. The approximation sign arises because of the assumptions made earlier.

Now reversing the time-indices about $k=K$, we have

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ \vdots \\ y_3 \end{bmatrix} \approx \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 0 & 1 & 1 & \dots & 1 \\ 0 & 0 & 1 & \dots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \psi_0 \\ \psi_2 \\ \psi_3 \\ \vdots \\ \vdots \\ \psi_n \end{bmatrix} \quad (22)$$

where, by definition $\psi_i(k) = \phi_i(K-k)$. Since

$$\begin{aligned} \langle \phi_i, \phi_j \rangle &= \langle \psi_i, \psi_j \rangle \\ &= \sigma_i \delta_{ij} \end{aligned} \quad (23)$$

the assertion of the theorem is proved.

Note that the weights σ_i can be calculated (in view of (18)) as

$$\sigma_i = \int_{-\omega_s/2}^{\omega_s/2} |L_i(\omega) H_1(e^{j\omega\Delta})|^2 d\omega$$

Else, they may be calculated in the time domain via (20).

Note that (22) may be written more compactly as

$$\underline{y} = U \underline{\psi} \quad (25)$$

where $\underline{y} = [y_1, y_2, \dots, y_N]^T$, $\underline{\psi} = [\sigma_0, \sigma_1, \dots, \sigma_n]^T$ and U is the upper triangular matrix of 1's. Secondly, we observe that the Gram matrix of the information signals can be written as

$$F = \langle \underline{y}, \underline{y}^T \rangle \quad (26)$$

where the inner-product is taken over each term of the matrix $\underline{y} \underline{y}^T$. In terms of the approximate tri-orthogonal characterization in (22), or (25), we have

$$\begin{aligned} F &= U \langle \underline{\psi}, \underline{\psi}^T \rangle U^T \\ &= U \Sigma U^T \end{aligned} \quad (26)$$

where Σ is the $n+1$ dimensional diagonal matrix $\text{diag}\{\sigma_0, \sigma_1, \dots, \sigma_n\}$.

The lemma below summarizes the PO properties of the information signals.

Lemma 3

The correlation coefficient of a pair of information signals y_i and y_j , $j > i$ is

$$\rho_{ij} = \sqrt{\frac{\sigma_{j-1} + \dots + \sigma_n}{\sigma_{i-1} + \dots + \sigma_n}}, \quad i, j = 1, \dots, n+1 \quad (27)$$

Proof - The relation follows readily from the approximate tri-orthogonal characterization of the information signals. Using (22) and (23), we have

$$\rho_{ij} = \frac{\sigma_{j-1} + \dots + \sigma_n}{\sqrt{\sigma_{i-1} + \dots + \sigma_n} \sqrt{\sigma_{j-1} + \dots + \sigma_n}}$$

which is the same as (27).

Discussion

In any system identification technique it is desirable to have basis functions that differ significantly from each other. Ideally, they should be orthogonal. By varying the parameter q between 0 and 1, it is possible to generate sets of basis functions that vary between a set whose elements are almost identical to approximately a set whose elements are orthogonal.

If $q = 0$ (recall that q is the z -domain pole of the first order filters $\mu_1(z)$), then the connection filters M_i all have a nearly all-pass characteristic, and as a result $\sigma_i \approx 0$ for $i \neq n$. Thus σ_n dominates all other σ_i and we have

$$\rho_{ij} \approx 1$$

This is undesirable; however, such is indeed the case when unit-delays are employed such as in methods like Prony, Linear Predictive analysis, Maximum-Entropy method, etc. [1]-[2]. The strong correlation between the basis functions leads to numerical difficulties.

On the other extreme, $q=1$ results in pure (digital) integration of $y(k)$, in reverse-time, for generation of the information signals. In this case the analysis of PO properties developed above is not applicable. Therefore, consider q close to one (from the left; e.g., $q = 0.99$). Now the cutoff frequencies $\omega_1, \omega_2, \dots, \omega_n$ become crowded near the zero frequency. Hence $\sigma_i \approx 0$ for $i \neq 0$ and σ_0 dominates all other σ_i . Therefore, $y_2, \dots, y_N$ have very small energy (thus very small fraction of signal information), and they are nearly orthogonal to y_1 .

Intermediate values of q lead to other useful sets of basis functions.

Example

Consider that the signal $y(k)$ has flat low-pass spectrum from $\Omega = 0$ to 0.05 with amplitude level 10. Then $q = 0.8$ and $n = 4$ lead to the values

$$\sigma_0 = 1.45, \sigma_1 = 1.28, \sigma_2 = 0.46, \sigma_3 = 0.31, \sigma_4 = 1.5$$

from which the correlation coefficients can be computed readily. For example $\rho_{23} = 0.893$, and $\rho_{14} = 0.548$.

SECTION IV

PROGRAM DESCRIPTION

The program POF-FILTER is a high accuracy FORTRAN IV program designed to implement the decoupled pencil-of-functions method. Specifically, it models the impulse response of a finite-order linear system by processing the given data through a cascade of first-order linear filters in reverse-time. Some of the features of the program are stated below.

- * Decoupled denominator and numerator determination. This permits fairly high order models to be determined, since the order of the Gram matrix is $n+1$ where n is the model order (or $n+2$ when data bias is also to be estimated. In contrast, the coupled procedure requires the use of a $2n+2$ order Gram matrix (or $2n+3$ when bias is also to be estimated)
- * First-order filter, rather than pure integrators, are used for generating the family of information signals. This results in a nearly tri-orthogonal set of signals which result in a better conditioned Gram matrix than with the use of pure integrators.
- * Bias correction option. Data bias can be estimated and thereby more accurate estimates for the model transfer function parameters obtained.
- * Noise correction option. A preliminary routine for estimation of noise effect, and correction thereof, has been included. Theoretical work and testing/improvement of this routine remains to be done.
- * Direct transmission option. Structural correctness of the model in either the presence or absence of the direct transmission term is preserved by exercising this option.
- * Results of modeling are obtained in the z -domain and on an optional

basis in the s-domain also.

- * Simulation option. The data modeled may be laboratory test data, or one may generate an impulse response within the program by specifying it either in the form $\sum_{k=1}^n R_k(z_k)^k$ or by the z-domain transfer function $H(z) = (b_0 + b_1 z^{-1} + \dots + b_n z^{-n}) / (1 + a_1 z^{-1} + \dots + a_n z^{-n})$.

In the simulation mode, a desired amount of bias and/or additive noise may be incorporated for test purposes.

In the laboratory-data case, a preliminary bias removal and a data scaling procedure (to maximize the effectiveness of the algorithm) has been incorporated.

- * The routine which finds the cofactors of the Gram matrix of the information signals has been optimized by incorporating a scaling and corresponding descaling stages.
 - * For comparison purposes the program provides the option to use two other modeling techniques, the linear predictor and the Prony method. The latter can be the classical Prony, which uses $2n+2$ data points, or the least-squares Prony.
- The provision of these methods within this single program is an essential step toward the evaluation of the pencil-of-functions method against other benchmark methods.

The input data cards on the subsequent pages give a description of all input variables, and in so doing provide an understanding of the program use.

INPUT DATA CARDS

CARD # 1 The first card is a title card.

CARD # 1 The first card is a title card. Columns 1 through 70 are available for alphanumeric title. This title is reproduced in the output

CARD # 2 Another title card (columns 1-70). Not reproduced in output

CARD # 3 Another title card (columns 1-70). Not reproduced in output

CARD # 4 First option card

Variable Name (Format)	Description	Columns	Preferred/default Value
NPT (I4)	Number of signal points used in analysis/modeling	1-4	-
IXX, IYY (2I2)	Blank (unused)	5-8	-
N (I2)	Model order	9-10	-
ISIM (I2)	Simulation mode option	11-12	-
	ISIM = -1 Real (laboratory) data in integer format (10I5)		
	= 0 Real (laboratory) data in real format (10F8.6)		
	= 1 simulation from digital transfer function		
	= 2 simulation from sums of exponential*sinusoid form (specified in continuous-time domain)		
NCOMP (I2)	Number of terms in the sums of exponential*sinusoid form	13-14	-
IPLT (I2)	Plot option	15-16	
	IPLT = 0 Plot routine not called		
	= 1 Plot of the original data and model response obtained		
	= m same as 1 except that every mth point of the signals are plotted (e.g., with m=2 alternate points are plotted)		
NNPT (I4)	Number of signal points read or through simulation. NNPT should be greater than or equal to NPT	17-20	NPT

YYYY (F10.0)	Blank (unused)	21-30	
DT (F10.0)	Sampling interval	31-40	
BIAS (F10.0)	Bias to be added to data This is for use in the simulation modes (ISIM=1 or 2) when it is desired to study the effect of bias on data	41-50	
ANBIAS (F10.0)	Number of points used for a preliminary estimate of bias. That is, NBIAS= Integer(ANBIAS) number of points from the right are used to find a crude estimate of bias; this crude bias is subtracted from data. This is useful only when real data is analyzed (ISIM= -1 or 0), and ignored when simulation data is generated.	51-60	Leaving this blank results in the use of 20% dta from the right for crude bias.
VAR (F10.0)	Variance of noise to data Must be used only in the simulation mode (ISIM=1 or 2) and should be left blank when real data is analyzed.	61-70	

CARD # 5.1-5.n If ISIM= -1 or 0 these cards contain 1-50 if ISIM=-1
real data 1-80 if ISIM=0

If ISIM= 1 these cards contain (in (5F10.0) format the coefficients of the z-domain transfer function; first denominator coefficients, and on succeeding card(s) the numerator coefficients.

If ISIM= 2 these cards contain (in (5F10.0) format the coefficients of the exponential*sinusoid terms; one such term on each card. Each card contains a) the weighting coefficient b) the exponent (real), c) the radian frequency of the sinusoid, and d) the phase of the sinusoid. If the phase is zero, the sinusoid is a sine wave with zero value at k=0.

Example: If $y(t) = e^{-t} + 7e^{-3t} \cos(2t)$ is to be simulated, then ISIM=2, NCOMP=2 and these cards are as follows

+1.00000 -1.00000
+7.00000 -3.00000 +2.00000 +1.57079

CARD # 6

Subtitle card. This alphanumeric subtitle is reproduced in the output

CARD # 7

Second option card. This numeric information is also reproduced on the output (for user convenience)

Variable Name (Format)	Description	Columns	Preferred/Default Value
IPR (I2)	Print option. Increasing value results in more printing. Use 0, 1 or 2 for normal use. Values 3, 4 and 5 are useful for diagnostic purposes, or when resut-s of intermediate computations are needed (for example, the first-order filtered signals, i.e., the information signals are printed only if IPR.GE.4)	1-2	0
IZTS (I2)	z-domain to s-domain conversion option. IZTS = -1 conversion to s-domain is not performed = 0 conversion performed; only the poles in s-domain printed = 1 conversion performed; in addition to poles in s-domain the s-domain denominator and numerator also printed.	3-4	
IREM (I2)	Number of coefficients in the numerator (of the z-domain model), counted from the right, and set to zero	5-6	0
ISPN	Method option ISPN = -1 pencil-of-functions method employed = 1 pencil-of-functions method employed; noise added to simulated data. Must be used only when ISIM=1 or 2 = 0 Analysis of noise only =-2 Covaria ce equations used (for LPC or Prony) =-3 Autocorrelation equations used (for LPC or Prony)	5-6	Warning: Do not use For LPC set IREM=19 For LPC set IREM=19

IFIX (I2)	Noise correction option (under development - ignore)	9-10	
NFIX (I2)	Auxiliary parameter for use with IFIX - ignore	11-12	
IBIAS (I2)	Bias <u>extraction</u> option	13-14	
	IBIAS = 0 Bias extraction not exercised		Warning: Do not use values other than 0 or 1
	= 1 bias extracted		
IB0 (I2)	Direct transmission term option	15-16	
	IB0 = 0 Constrains $b_0=0$ in numerator determination		
	IB0 = 1 Model assumes direct transmission is present, and b_0 is determined together with other coefficients		
MNPT (I4)	Controls the number of points used in the computation of error, and for printing (when IPR.GE.2) and plotting	17-20	NPT
QI (F10.0)	z-domain pole of the first order filter (i.e., the parameter q of Section I)	21-30	suggest QI=0.8
DFAC (F10.0)	Auxiliary variable for use with IFIX (under development)	31-40	

SECTION V

APPLICATION EXAMPLES

Two examples will be presented in this section. The first deals with a simulated noisy signal, $x(k) = y(k) + w(k)$ where $y(k)$ is the response of a third order transfer function and $w(k)$ is a zero mean noise component. The performance of the pencil-of-functions method, with and without Gram matrix enhancement, is compared with that of other methods [1]-[3]. The second example pertains to the transient response of a conducting pipe tested at the ATHAMAS-I EMP simulator. These examples demonstrate the effectiveness of the pencil-of-functions method as a practical modeling technique.

Example 1

Let

$$y(k) \longleftrightarrow \frac{1 - 1.92z^{-1} + z^{-2}}{1 - 2.68z^{-1} + 2.476z^{-2} - 0.782z^{-3}}$$

where the sequence y is truncated at $K=99$. The signal under test is obtained as

$$x(k) = y(k) + w(k)$$

where $w(k)$ is a zero mean, uncorrelated sequence with standard deviation equal to 0.0316. The energies of the signal and noise components are 1.82 and 0.1, respectively, hence the sequence under test has a signal-to-noise ratio of 12.6 dB. The numerical data and a plot of the signal under test are given in Appendix B.

The signal under test was analyzed by the following methods [2].

1. Pencil-of-functions method
 - a) without enhancement of the Gram matrix
 - b) with enhancement of the Gram matrix
2. Linear predictive coder (LPC)
 - a) using autocorrelation equations
 - b) using covariance equations

3. Prony's method [11]

- a) using autocorrelation equations
- b) using covariance equations [3]

The computer output for these six runs is given in Appendix B. Here

we summarize the fractional energy error, defined as

$$\nu = \frac{\sum_{k=0}^K (y(k) - \hat{y}(k))^2}{\sum_{k=0}^K y^2(k)} \quad (28)$$

where $\hat{y}(k)$ is the impulse response of the model. Note that in simulation mode the model response can be, and is, compared with the true signal. When real data is tested the true signal is not available, hence $x(k)$ must be used in this equation instead of $y(k)$.

For the present example the true signal is available, hence the fractional energy error is computed by exact application of (28). The values for the various methods are tabulated in Table 1.

TABLE 1

Method	Fractional energy error
Pencil-of-functions	0.0167
Pencil-of-functions with enhancement	0.0016
Linear Predictor (autocorrelation eqn)	0.1444
Linear Predictor (covariance eqn)	0.1567
Prony (autocorrelation eqn)	0.1431
Prony (covariance eqn)	0.1397

The model responses are compared with the true signal $y(k)$ in Figures 6 through 11. The reader is cautioned that the solid line in each of these figures represents the true signal, and not the noisy signal under test. The latter is shown in Fig. B2 in Appendix B. The dotted line in each of these figures represents the model response.

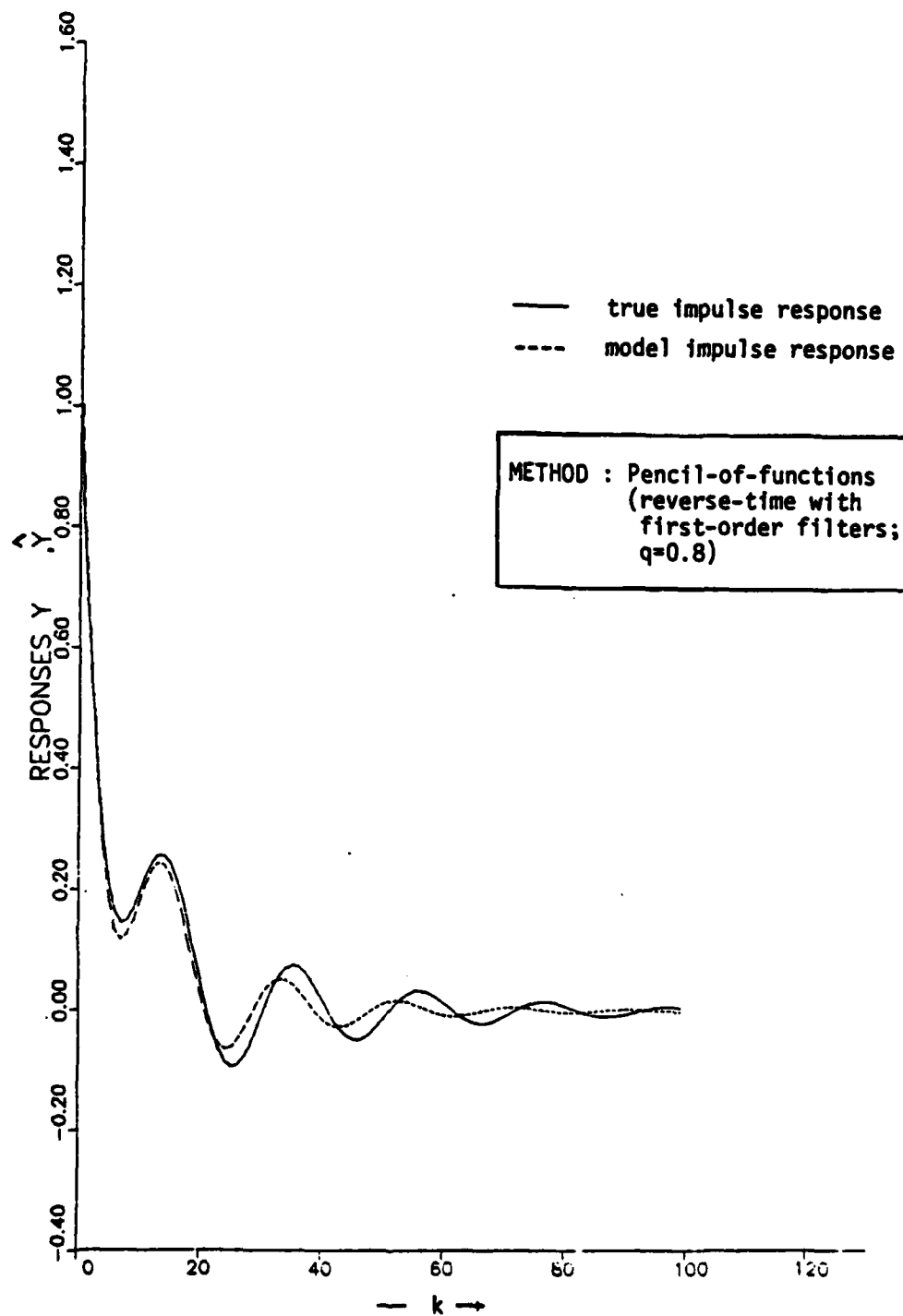


Fig. 6. Comparison of true and model impulse responses

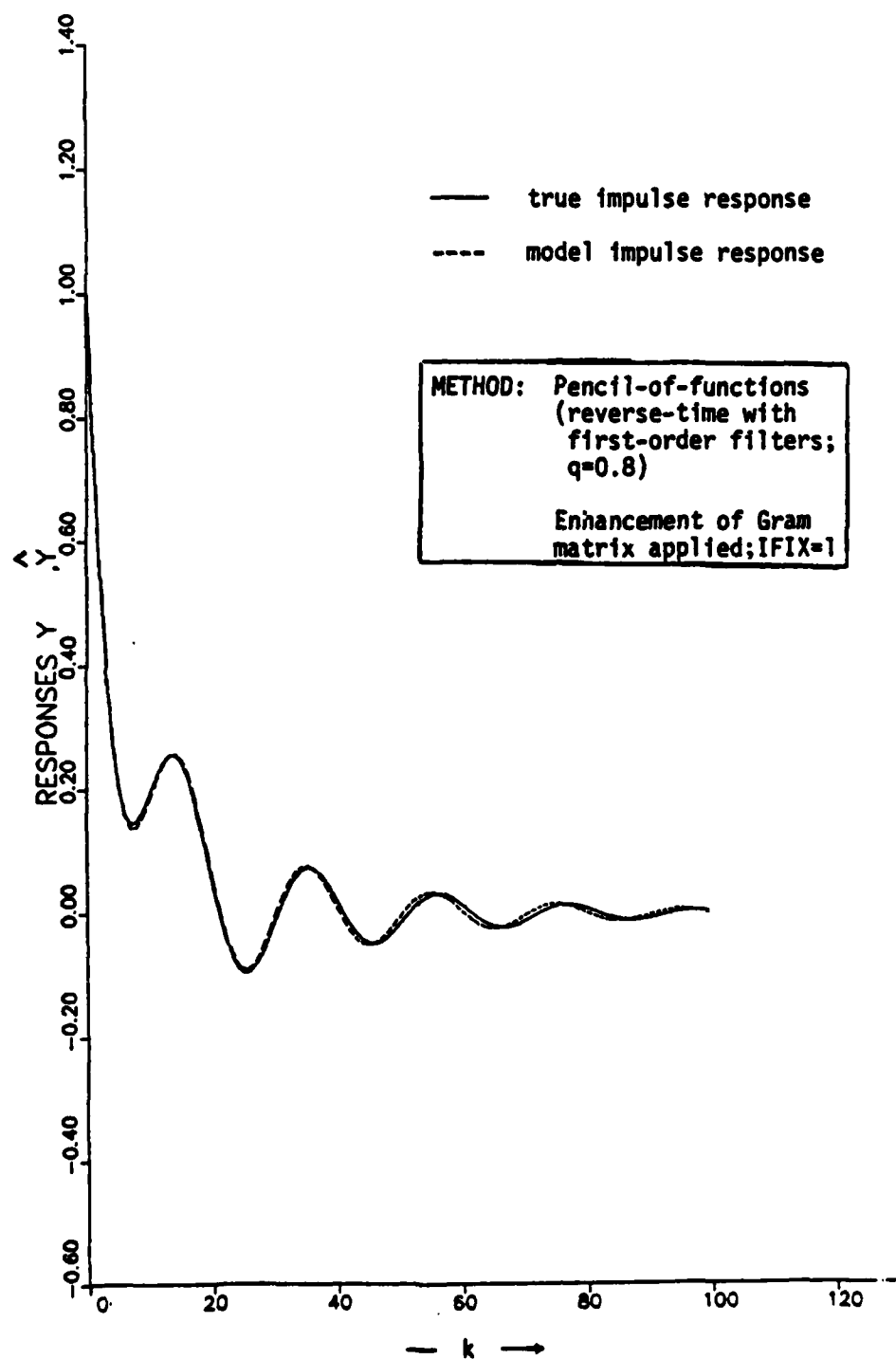


FIG. 7. Comparison of true and model impulse responses

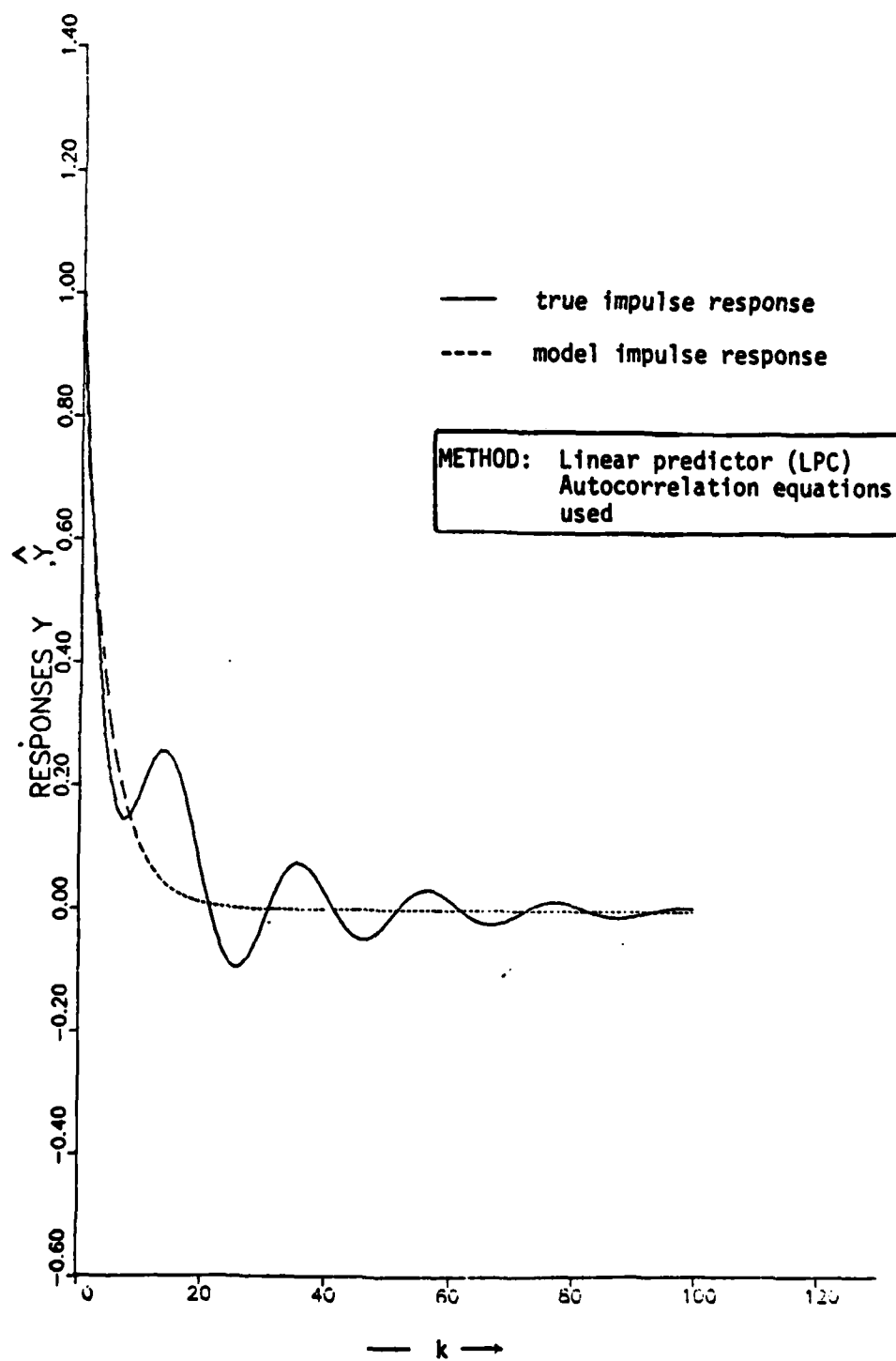


Fig. 8. Comparison of true and model impulse responses

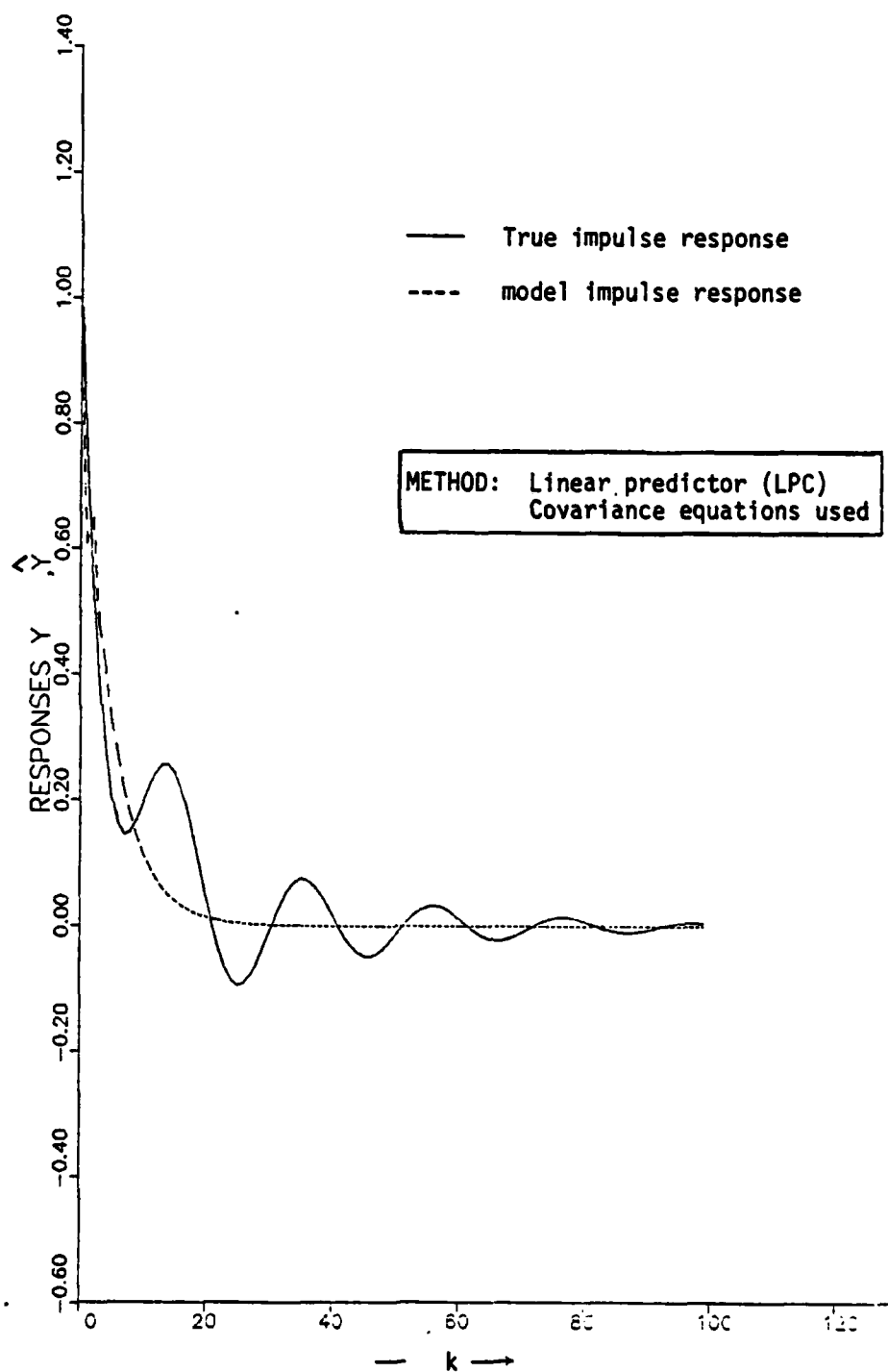


Fig. 9. Comparison of true and model impulse responses

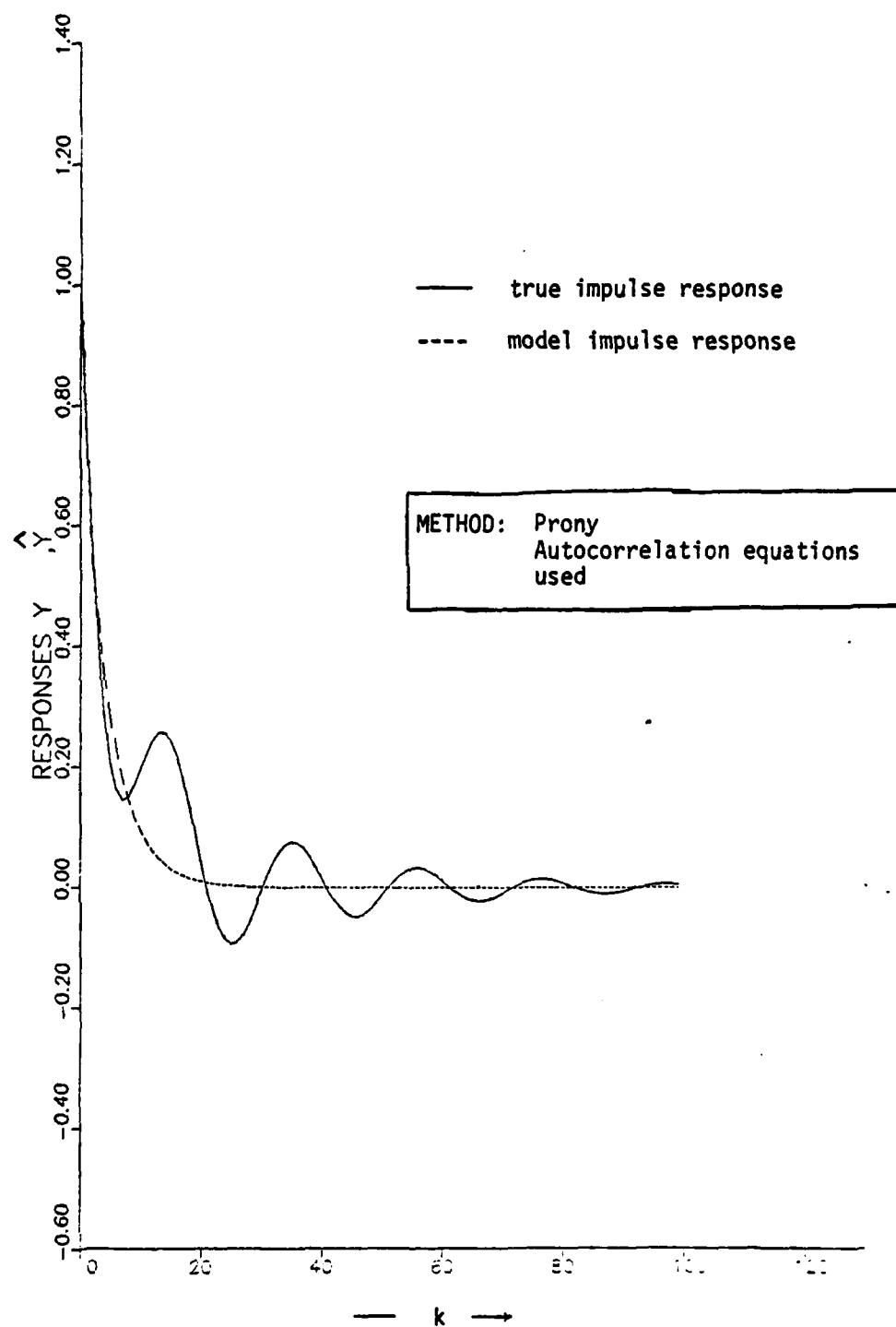


Fig. 10. Comparison of true and model impulse responses

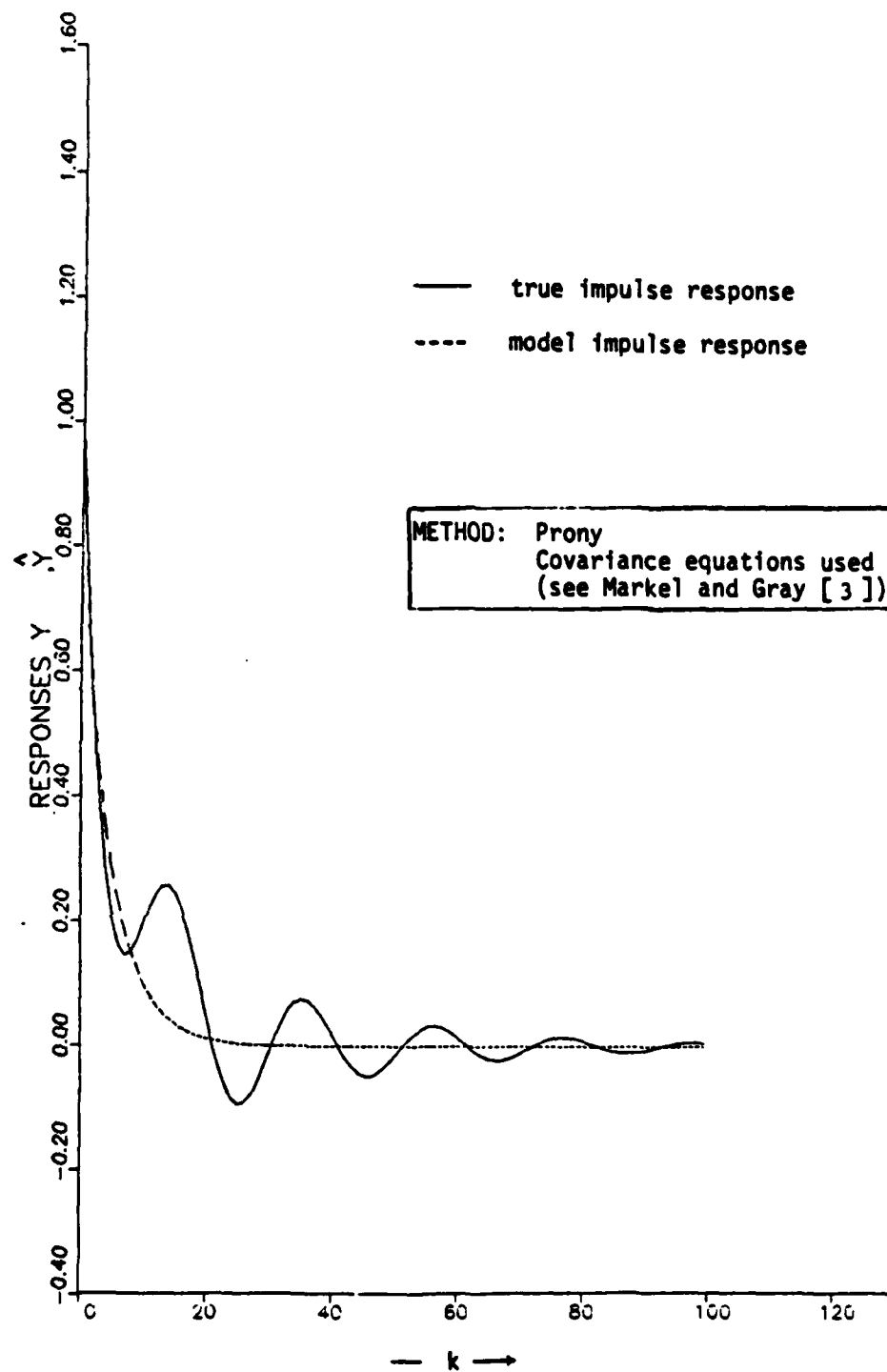


Fig. 11. Comparison of true and model impulse responses

This example demonstrates the superiority of the pencil-of-functions method over two other widely used methods (for modeling impulse responses), namely the all pole linear predictor and the prony method.

Example 2

As a real world application we consider the use of pencil-of-functions method to the transient response of a conducting pipe tested at the ATHAMAS-I EMP simulator. The conducting pipe is 10m long and 1 m in diameter. Hence, the true resonance of the pipe is expected to be in the neighborhood of 14 MHz. Also, the pipe has been excited in such a way that it is reasonable to expect only odd harmonics at the scattered fields. The data measured are the integral of the E-field; t.e., the measured variable is a voltage. The transient response used for analysis is shown in Fig. 12 by the solid line. The results of analysis by the pencil-of-functions method are given in Appendix C for the case of an 8th order model; the model response, with an error of 0.0125, is shown in Fig. 12 by the dotted line. Model poles are:

fundamental	$-4.280 \pm j 67.686$	Mrad/s	(=10.794 MHz)
3rd harmonic	$-22.470 \pm j 218.200$	"	(=34.911 MHz)
curve-fit pair	$-2.543 \pm j 12.890$	"	(= 2.051 MHz)
curve-fit pair	$-16.547 \pm j 88.981$	"	(=14.404 MHz)

Note that a pole at the origin (due to the integrator) has not been obtained because we have used the bias extraction option. On the other hand, the curve-fit pole pair arises because the data do not truly pertain to the impulse response of a finite order (lumped) linear system.

Next the data were differentiated (actually differenced), and analyzed by the pencil-of-functions method. The results are given in Appendix C for 8th order analysis; the model response, with a fractional

energy error of 0.0369 is shown in Fig 13 by the dotted line (the solid line depicts the differentiated data). The model poles are:

fundamental	$-9.453 \pm j 71.596$	Mrad/s	(=11.494 MHz)
3rd harmonic	$-25.996 \pm j 222.340$	"	(=35.628 MHz)
5th harmonic	$-113.303 \pm j 617.095$	"	(=99.855 MHz)
curve-fit pair	$-22.268 \pm j 59.213$	"	(=10.068 MHz)

Here again a curve-fit pole pair arises because the data do not truly pertain to the impulse response of a finite order (lumped) linear system.

We also give below the poles arising from a 6th order model (the computer output is not given, nor the graphical display of the model response):

fundamental	$-9.575 \pm j 75.106$	Mrad/s	(=12.050 MHz)
3rd harmonic	$-14.866 \pm j 221.693$	"	(=35.363 MHz)
curve-fit pair	$-14.123 \pm j 34.045$	"	(= 5.866 MHz)

The fractional energy error in modeling is 0.0566.

Here again a curve-fit pole pair arises because the data do not truly pertain to the impulse response of a finite order (lumped) linear system.

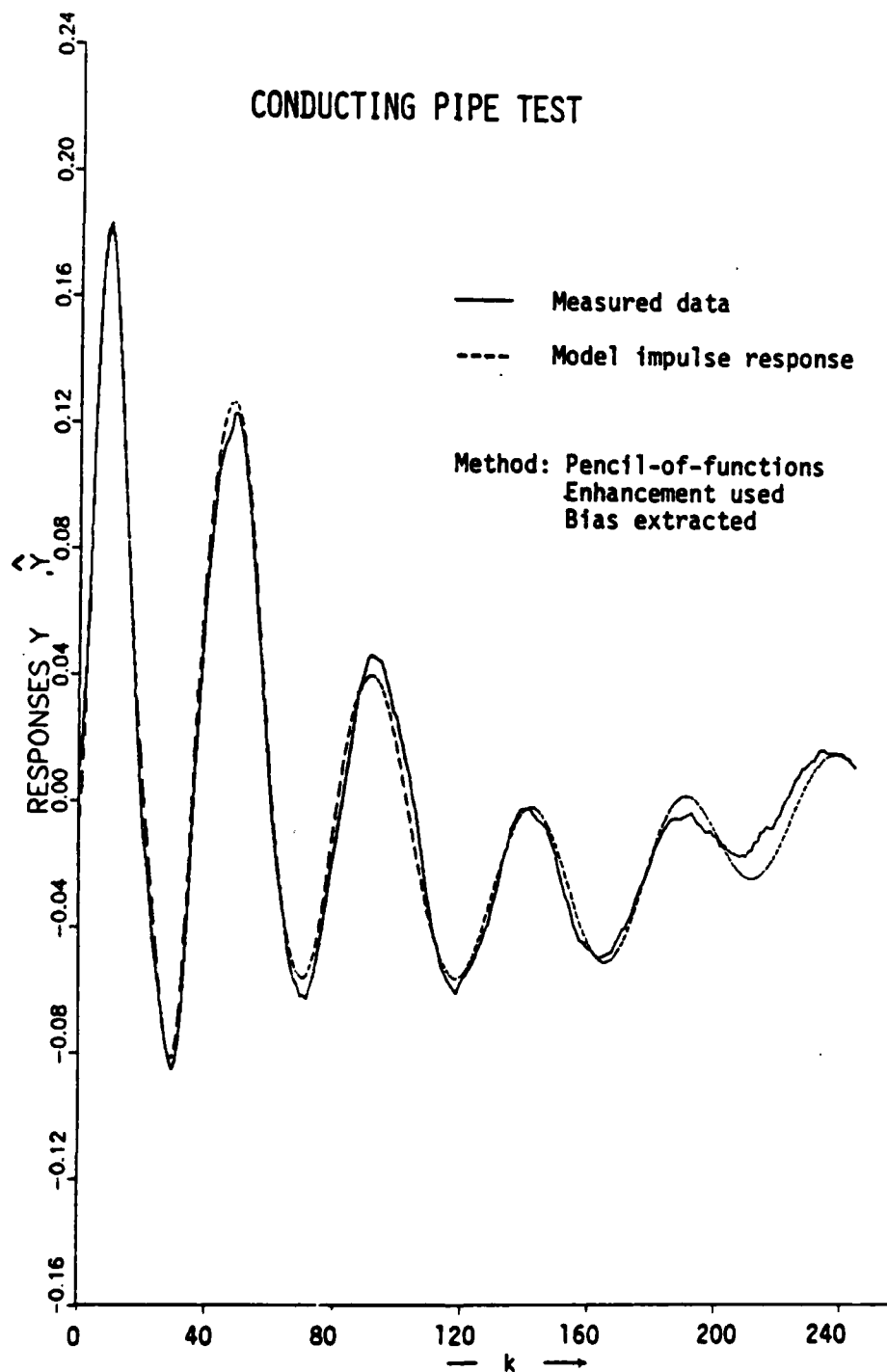


Fig. 12. Comparison of measured data (of the response of of a conducting pipe) and model impulse response

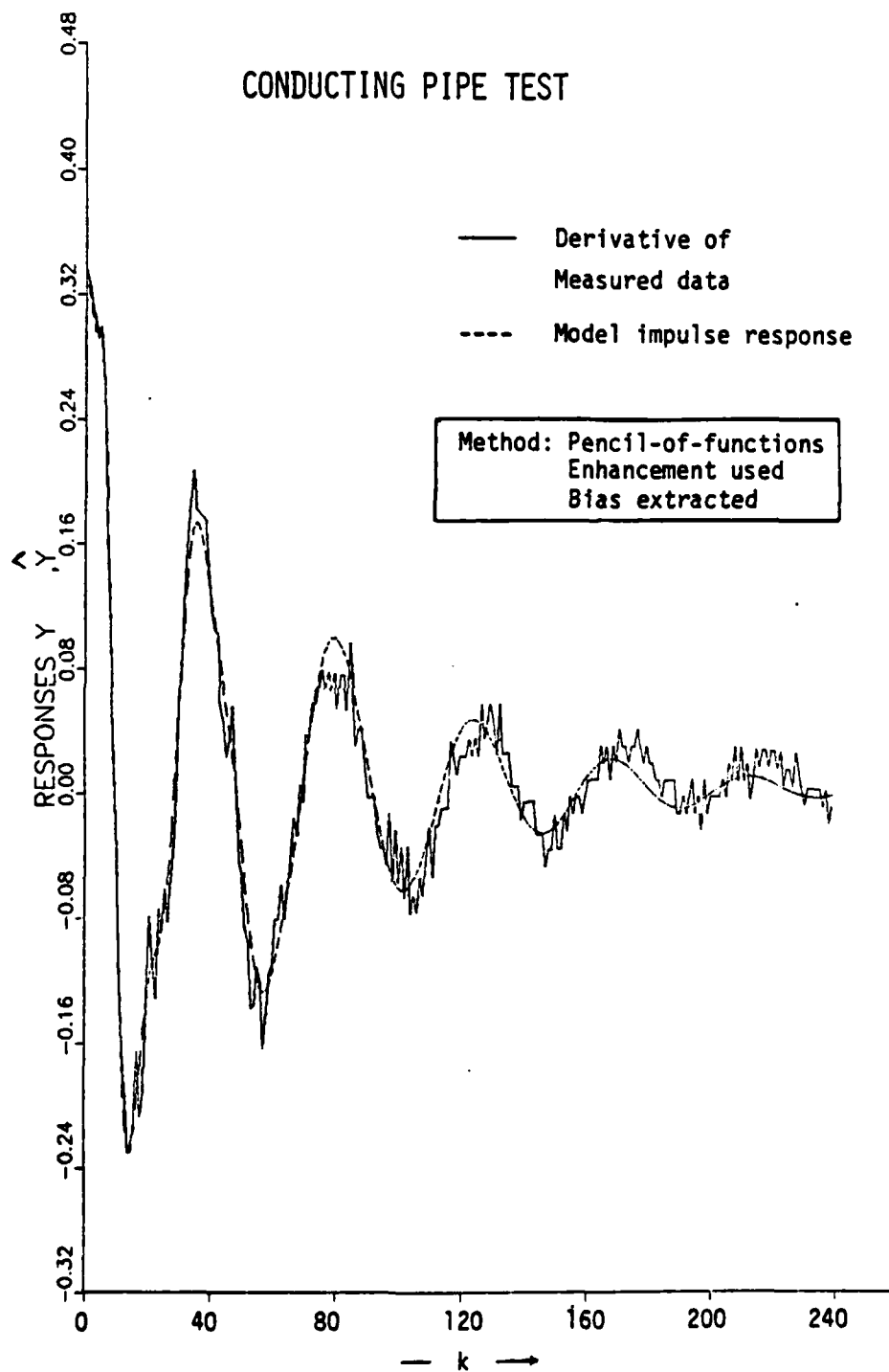


Fig. 13. Comparison of the derivative of measured data and the corresponding model impulse response

REFERENCES

- [1] J. Makhoul, "Linear prediction: A tutorial review," IEEE Proc., Vol. 63, pp. 561-580, April 1980.
- [2] T. K. Sarkar, V. K. Jain, J. Nebat, D. D. Weiner, "A comparison of the pencil-of-functions method with Prony's method Wiener filters and other identification techniques," Technical Rep. to Air Force Weapons Lab. (Contract F 33615-77-C-2059, Task 12), December 1977.
- [3] J. D. Markel and A. H. Gray, Linear Prediction of Speech. Berlin: Springer-Verlag, 1976.
- [4] V. K. Jain, "Filter analysis by use of pencil-of-functions: Part I," IEEE Trans. Circuits and Systems, Vol. CAS-21, pp. 574-579, September 1974.
- [5] B. Gold, and C. M. Rader, Digital Processing of Signals. New York: McGraw-Hill, 1969.
- [6] E. W. Cheney, Introduction to Approximation Theory. New York: McGraw-Hill, 1966.
- [7] J. H. Carr, III, "An analytical investigation of transients synthesis by exponentials," M.S. Thesis, Mass. Inst. Tech., Cambridge, 1949.
- [8] V. K. Jain, and R. D. Gupta, "Identification of linear systems through a Grammian technique," Intntl. J. Control, Vol. 12, pp. 421-431, 1970.
- [9] V. K. Jain and J. S. Osman, "Computer routines for use in second-order Volterra identification of EMI," RADC-TR-79-266, October 1979.
- [10] G. J. Bierman, Factorization Methods for Discrete Sequential Estimation. New York: Academic Press, 1977.
- [11] M. L. Van Blaricum, "PSPL: A computer program for extracting the poles and residues from a transient signal using Prony's method," Scientific Report No. 76-2, Electromagnetics Laboratory, University of Illinois, Urbana, 1976.

APPENDIX A

DETAILS OF PROGRAM POF-FILTER

LSITINGS

PURPOSE, EQUATIONS

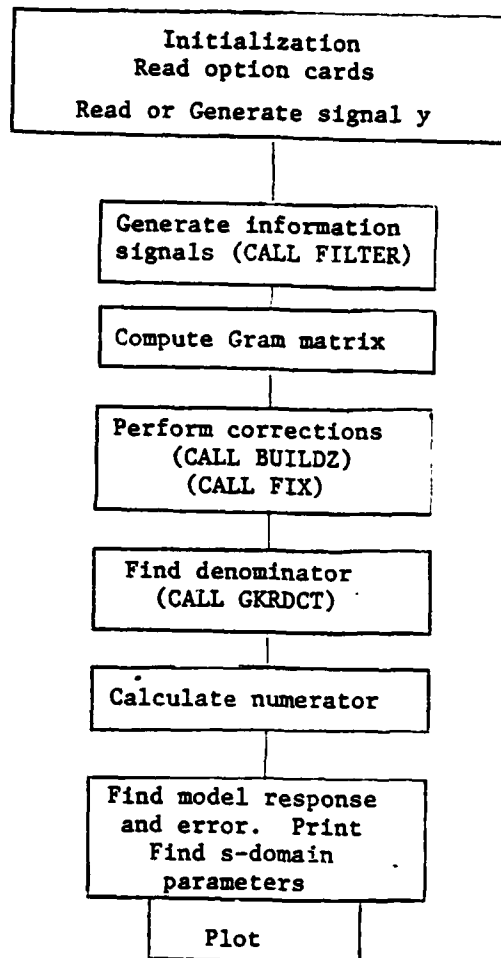
FLOWCHART

VARIABLES

FOR THE MAIN AND SUBROUTINES

MAIN PROGRAM

FLOWCHART



LISTING OF POF-FILTER

```

PROGRAM "POF-FILTER"
IMPULSE-RESPONSE MODELING
BY PENCIL-OF-FUNCTIONS METHOD
APRIL 1980
* DECOUPLED DENOM. AND NUM. DETERMINATION*
* FIRST-ORDER FILTERS USED*
* NOISE CORRECTION OPTION*
* BIAS CORRECTION OPTION*
* DIRECT TRANSMISSION OPTION*
* RESULTS IN BOTH Z- AND S- DOMAINS*

```

```

"POF-FILTER" MODELS IMPULSE RESPONSE OF A SCATTERER/CHANNEL/NETWORK
IT CAN BE USED IN SIMULATION MODE
OR ON EXPERIMENTALLY RECORDED RESPONSES.
FOR COMPARISON PURPOSES IT ALSO PROVIDES
ON AN OPTIONAL BASIS THE FOLLOWING METHODS
# LINEAR PREDICTOR (COV OR AUTOCORR)
# PRONY
# LEAST-SQUARES PRONY

```

NOTE: 1350 LINES OF CODE. THIS CAN BE REDUCED SUBSTANTIALLY
 FOR PARTICULAR APPLICATIONS. IN PARTICULAR, ROUTINES "PLOP"(47),
 "ZTOS"(157), AND "FCLRT"(201) MAY BE ELIMINATED
 IF ONLY Z-DCMAIN RESULTS NEEDED

```

*****
DIMENSION F(2500),U(800),LU(800),X(800,11),G(11,11),AM(11,11)
DIMENSION GN(11,11),GEST(11,11),GDJM(11,11),E(11,11),EN(11,11)
DIMENSION V(22),VV(22),AMP(11),SR(11),SI(11),SPH(11)
DIMENSION TITLE(70),IBUF(512),IDUM(10)
DOUBLE PRECISION DT,AC,BD,ERROR
COMMON /DA0/ISP,DELTA,SIG2,DT,QI,BIAS,IBIAS,DFAC
COMMON /DA1/FBAR,EBAR,FESUM,EESUM
COMMON /IO/IR,ILT,IPR,ITFR,IZPR,IROND,IPLT
REWIND5
MAXPL=800
MAX=11
MAX2=2*MAX
IR=5
ILT=6
ISKIP=0
CALL VEQUAT(MAXPL,U,F,0,10)
CALL VEQUAT(MAXPL,UU,F,0,11)
CALL VEQUAT(MAX2,V,VV,0,0)
WRITE(ILT,2)
READ(IR,6)(TITLE(I),I=1,70)
WRITE(ILT,16)(TITLE(I),I=1,70)
READ(IR,6)(TITLE(I),I=1,70)
READ(IR,6)(TITLE(I),I=1,70)
READ(IR,4)NPT,IXX,IYY,N,ISIP,NCCMP,IPLT,NNPT,
+YYYY,DT,BIAS,ANBIAS,VAR
NP1=N+1
NP2=NP1+1
NP3=N+3
NPNP2=N+N+2
NPNP1=N+N+1
IF (NNPT.EQ.0) NNPT=NPT
IF (DT.EQ.0.0) DT=1.0
IFUN=2

```

```

IF (IPLT.LT.0) IFUN=1
IF (IPLT.LT.0) IPLT=-IPLT
FMAX=1.0
IF (ISIM.EQ.3) GO TO 63
IF (ISIM.NE.-1) GO TO 59
IMAX=0
DO 58 I=1,NNPT,10
READ (IR,161) (IDUM(J),J=1,10)
DO 53 J=1,10
IF (IABS(IDUM(J)).GT.IMAX) IMAX=IABS(IDUM(J))
K=I-1+J
3 F(K)=IDUM(J)
8 CONTINUE
FMAX=IMAX
GO TO 45
9 CONTINUE
IF (ISIM.NE.0) GO TO 49
READ (IR,163) (F(K),K=1,NNPT)
FMAX=0.0
DO 41 K=1,NNPT
IF (ABS(F(K)).GT.FMAX) FMAX=ABS(F(K))
CONTINUE
5 NBIAS=ANBIAS
IF (NBIAS.EQ.0) NBIAS=0.2*NNPT
NDIE=NNPT+1-NBIAS
FB=0.0
DO 57 K=NDIE,NNPT
FB=FB+F(K)
FB=FB/NBIAS
DO 56 K=1,NNPT
F(K)=(F(K)-FB)/FMAX
GO TO 61
CONTINUE
IF (ISIM.EQ.1) READ (IR,160) (V(I),I=1,NNP1)
IF (ISIM.EQ.1) READ (IR,160) (V(I),I=NNP2,NNP2)
CALL VEQUAT(NNP1,V(NNP2),-1.0,0.3)
IF (ISIM.EQ.1) CALL RESPON(F,L,N,V,VV,NNPT)
IF (ISIM.EQ.1) GO TO 61
DO 60 I=1,NCOMP
READ (IR,5) AMP(I),SR(I),SI(I),SPH(I)
WRITE (ILT,11) I,AMP(I),SR(I),SI(I),SPH(I)
CALL SIGNAL(F,NNPT,AMP,SR,SI,SPH,DT,NCOMP)
CONTINUE
IF (IPLT.GE.1) CALL PLOTS(IBUF,512,9)
11 READ (IR,8) (TITLE(I),I=1,70)
IF (EOF(IR).NE.0) GO TO 998
WRITE (ILT,3)
WRITE (ILT,18) (TITLE(I),I=1,70)
READ (IR,8) (TITLE(I),I=1,70)
WRITE (ILT,18) (TITLE(I),I=1,70)
BACKSPACE 5
READ (IR,6) IPR,IZTS,IEM,ISPN,IFIX,NFIX,IEIAS,I30,MNPT,QI,DFAC
NSTRT=2
IF (ISPN.EQ.-2) NSTRT=NP1
IF (ISPN.EQ.-3) NSTRT=1
IZPR=0
IF (IPR.GE.20.AND.IPR.LE.39) IZPR=(IPR-10)/10
ITPR=0
IF (IPR.GE.90) ITPR=1
IPR=IPR-10*(IPR/10)

```

```

IF (ISPN.LE.-2.AND.IREM.GT.N) IFEM=N
IF (DFAC.EQ.0) DFAC=0.1
IBS=1
IF (IBIAS.EQ.0) IBS=0
IF (IBIAS.LE.0) IBIAS=0
IROUND=0

```

CORRUPT SIGNAL IF DESIRED.
PROCESS WITH FIRST ORDER FILTERS

```

10 CONTINUE
IF (VAR.GE.1.0E-6) CALL CORUPT(F,X,VAR,NPT,MAXPL)
3 CONTINUE
IF (VAR.GE.1.0E-6) GO TO 99
DO 30 K=1,NPT
1 X(K,1)=F(K)+BIAS
3 CONTINUE
INT=-1
IF (ISPN.LE.-2) INT=2
IF (NP1.GT.1) CALL FILTER(X,NPT,NP1,MAXPL,INT)

```

COMPUTE GRAM MATRIX

```

NPP=NP1
IF (IBIAS.NE.0) NPP=NP2
DO 44 I=1,NPP
DO 44 J=1,NPP
AD=0.0
IF (ISPN.EQ.0.AND.IROUND.EQ.0) GO TO 43
DO 42 K=NSTRT,NPT
AD=AD+X(K,I)*X(K,J)
GN(I,J)=AD*DT

IJ1=IABS(I-J)+1
IF (ISPN.EQ.-3.AND.I.GE.2) GN(I,J)=GN(1,IJ1)

GDUM(I,J)=GN(I,J)
CONTINUE
IF (IROUND.EQ.0) G(I,J)=GN(I,J)
CONTINUE
IF (ISPN.NE.0.OR.IROUND.NE.0)
1CALL GKROCT(GN,E,DET,V,NPP,NPP,MAX,1)
IF (IROUND.EQ.0) WRITE(ILT,171) DET
IF (IROUND.EQ.1) WRITE(ILT,172) DET
IF (IPR.GE.1) CALL PRTHAT(GN,NPP,NPP,MAX,-1)
WRITE(ILT,1)
IRO=IROUND
IF (IROUND.EQ.0) IROUND=IROUND+1
IF (IRO.EQ.0.AND.ISPN.GT.-1) GO TO 410
IF (IFIX.EQ.-1) GO TO 203

```

ESTIMATE OF ** G

```

6 CALL BU10Z(AM,V,NP1,NPT,MAX,NFIX)
-----NF1 REPLACED BY NPP NEXT 3 CARDS--
CALL FIX(GDUM,AM,GEST,E,V,NPP,NPP,SIG2,MAX,IFIX)
IF (IFIX.EQ.1) WRITE(ILT,462) SIG2
CALL GKROCT(GEST,E,DET,V,NPP,NPP,MAX,1)
WRITE(ILT,162) DET
IF (IPR.GE.1) CALL PRTHAT(GEST,NP1,NP1,MAX,0)
DO 154 I=1,NP1
DO 154 J=1,NP1
GDUM(I,J)=GEST(I,J)
NFIX=NFIX-1

```

```

IF(NFIX.GE.1)GO TO 156
ISKIP=1
IROUND=0

```

DETERMINE NUMERATOR

```

13  CONTINUE
    IF(IBS.EQ.1)IBIAS=1
    IF(ISPN.EQ.0)GO TO 998
    CALL VEQLAT(NP1,V(NP2),VV,0,0)
    V(NP2)=-1.0
    CALL RESPON(X(1,1),U,N,V,VV,NPT)
    IF(IPR.GE.4)WRITE(ILT,174)
    IF(IPR.GE.4)WRITE(ILT,210)(X(K,1),K=1,NPT)
    CALL FILTER(X,NPT,NP1-IREM,MAXPL,2)
    L=N-IREM
    IF(IBIAS.NE.0)L=L+1
    LP1=L+1
    LP2=L+2
    IF(IBIAS.NE.0)CALL VEQUAT(NPT,X(1,LP1),U,0,11)
    CALL VEQUAT(NPT,X(1,LP2),F,0,1)
    CALL VEQUAT(NPT,X(1,LP2),BIAS,0,4)
    IF(IPR.GE.5)WRITE(ILT,110)((X(K,I),K=1,NPT),I=1,LP2)
    L=L+I80
    LP1=L+1
    DO 216 I=1,L
    DO 216 J=1,LP1
    AD=0.0
    DO 215 K=1,NPT
5    AD=AD+X(K,I+1-I80)*X(K,J+1-I80)
6    G(I,J)=AD*DT
5    IF(IPR.GE.5)CALL PRMAT(G,L,LP1,MAX,205)
    CALL GKROCT(G,E,DET,VV,L,L,MAX,0)
7    IF(IPR.GE.5)CALL PRMAT(E,L,L,MAX,207)
    CALL VEQLAT(NP1,VV,AMP,0,0)
    DO 220 I=1,L
    AD=0.0
    DO 219 J=1,L
9    AD=AD+E(I,J)*G(J,LP1)
0    VV(I)=AD
    FMEAN=0.0
    IF(IBIAS.NE.0)FMEAN=VV(L)
    CALL VEQUAT(IREM,VV(N+I80-IREM+1),AMP,0,0)
    V(NP2)=0.0
    CALL VEQLAT(N+I80,V(NP3-I80),VV,0,1)
    IF(IPR.GE.2)WRITE(ILT,12)FMAX
    WRITE(ILT,303)FMAX
    WRITE(ILT,210)(V(I),I=1,NP1)
    WRITE(ILT,210)(V(I),I=NP2,NPNP2)
    IF(IBIAS.NE.0)WRITE(ILT,305)FMEAN

```

MODEL RESPONSE, AND ERROR.

```

IF(MNPT.EQ.0)MNPT=NNPT
CALL VEQLAT(NP1,V(NP2),-1.0,0,3)
CALL RESPON(X(1,2),U,N,V,VV,MNPT)
CALL VEQLAT(MNPT,X(1,2),FMEAN,0,4)
ERROR=0.0
FFSUM=0.0
DO 213 K=1,MNPT
X(K,1)=F(K)*FMAX+BIAS

```

```

X(K,2)=X(K,2)*FMAX
FFSUM=FFSUM+X(K,1)*X(K,1)
X(K,3)=X(K,1)-X(K,2)
13 ERROR=ERROR+X(K,3)*X(K,3)
FFSUM=FFSUM*DT
ERROR=ERROR*DT
RATIO=ERROR/FFSUM
WRITE(ILT,304)ERROR,FFSUM,RATIO
IF(IPR.GE.2)WRITE(ILT,112)
IF(IPR.GE.2)WRITE(ILT,110)(X(K,1),K=1,MNPT)
IF(IPR.GE.2)WRITE(ILT,113)
IF(IPR.GE.2)WRITE(ILT,110)(X(K,2),K=1,MNPT)
IF(IPR.GE.2)WRITE(ILT,114)
IF(IPR.GE.2)WRITE(ILT,110)(X(K,3),K=1,MNPT)
DELT=DT
IF(IZTS.GE.0)CALL ZTOS(V(1),V(NP2),N,DELT,IZTS)

IF(IPLT.EQ.0)GO TO 239
IF(IPLT.EQ.1)GO TO 238
DO 230 I=1,2
KK=0
DO 230 K=1,MNPT,IPLT
KK=KK+1
10 X(KK,I)=X(K,I)
MNPT=MNPT/IPLT
16 T0=0.0
DELT=DT*IPLT
CALL PLOP(MNPT,IFUN,X,MAXFL,T0,DELT,1HY,1HT,IBUF)
19 CONTINUE
FORMAT STATEMENTS

FORMAT(I4,6I2,I4,6F10.0)
FORMAT(5F10.3)
FORMAT(8I2,I4,4F10.0)
FORMAT(70A1)
FORMAT(2X,70A1)
3 FORMAT(10(5X,F5.0))
4 FORMAT(10(5X,I5))
FORMAT(2X,I2,*AMP=*,F8.2,*S=*,F10.4,*J=*,F10.4,
1*PHASE=*,F10.4)
FORMAT(2X,*WAVEFORMS AND NUMER. SCALED BY XMAX=*,F12.5)
0 FORMAT(10X,8HG MATRIX)
0 FORMAT(10X,8HM MATRIX)
2 FORMAT(10X,11HGEST MATRIX,* (DET=*,G13.6,*)*)
0 FORMAT(2X,5(2X,G13.6))
10 FORMAT(20(1X,F5.2))
0 FORMAT(2X,10(F7.4))
2 FORMAT(2X,*ORIGINAL SIGNAL (INCLUDES BIAS, IF ANY)*,/)
3 FORMAT(2X,*IMPL.RESP OF MODEL (INC.3-HAT, IF IBIAS.NE.0)*,/)
4 FORMAT(2X,*ERROR=F(K)-FREC(K)*,/)
8 FORMAT(10X,14HNOISY X MATRIX)
9 FORMAT(10X,8HX MATRIX)
1 FORMAT(10X,16HTRUE GRAM MATRIX,* (DET=*,G13.6,*)*)
2 FORMAT(10X,17HNOISY GRAM MATRIX,* (DET=*,G13.6,*)*)
4 FORMAT(2X,*IMPULSE RESPONSE OF 1/A(Z)*
J FORMAT(5F10.0)
1 FORMAT(10I5)
3 FORMAT(10F8.6)
3 FORMAT(2X,*EST TF B(Z)/A(Z)*, (FMAX=*,E12.5,*)*)
+ FORMAT(/,2X,*SS ERROR=*,G13.6,*SS SIGNAL=*,G13.6,
+*RATIO=*,G13.6,/)
5 FORMAT(2X,*ESTIMATED MEAN=*,G13.5)
2 FORMAT(*ESTIMATED NOISE VAR=*,G12.5)
FORMAT(/)
41 (MAIN-6)

```

FORMAT (1F1)
FORMAT (////)

98 GO TO 1111
CONTINUE
CALL PLOT(0.,0.,999)
STOP
END

SUBROUTINE: BUILDR

PURPOSE: To generate a conversion matrix to go from $\sqrt{D}_1$ to parameters a_i of $A(z)$. See equation (9) of Section I

EQUATIONS:

Conversion matrix (shown below for $n=3$)

$$= \text{Diag}\{q^3, q^2, q, 1\} \begin{bmatrix} 1 & & & \\ -3 & 1 & & \\ 3 & -2 & 1 & \\ -1 & 1 & -1 & 1 \end{bmatrix}$$

ROUTINE VARIABLES

- A Conversion matrix (not to be confused with the denominator polynomial)
- X Vector which brings in $[\sqrt{D}_1 \ \sqrt{D}_2 \ \dots \ \sqrt{D}_N]$ to the routine and takes back the denominator parameters $[a_0 \ a_1 \ \dots \ a_n]$
- N System order plus one
- MAX Maximum dimensionality of matrix A

FURTHER DESCRIPTION:

SUBROUTINE BUILDER(A,X,N,MAX)

CONVERSION MATRIX: REVERSE FOR PROCESSING -- I.R. MODELING

DIMENSION A(MAX,1),X(1),Y(20)

DOUBLE PRECISION DT,Y

COMMON /DA0/ISPN,DELTA,SIG2,DT,QI,BIAS,ISIAS,DFAC

COMMON /IO/IR,ILT,IPR,ITPR,IZPR,IRJUND,IPLT

NM1=N-1

DO 11 I=1,N

Y(I)=0.0

DO 11 J=1,N

A(I,J)=0.0

A(N,N)=1.0

DO 20 JJ=1,NM1

J=N-JJ

DO 15 KK=1,2

K=KK-1

DO 15 I=J,NM1

A(I+K,J)=A(I+K,J)+A(I+1,J+1)*(1.0-K-K)

CONTINUE

QQ=1.0

CHANGED THRU 22 3-17-80

DO 22 II=1,NM1

I=N-II

QQ=QQ*CI

DO 22 J=1,N

A(I,J)=QQ*A(I,J)

DO 25 I=1,N

IF(IPR.GE.3)WRITE(ILT,5)(A(I,JJ),JJ=1,N)

DO 25 J=1,N

Y(I)=Y(I)+A(I,J)*X(J)

DO 28 I=1,N

X(I)=Y(I)/Y(1)

IF(IPR.GE.3)WRITE(6,7)(X(I),I=1,N)

FORMAT(2X,10G12.5)

FORMAT(* ESTIMATED PARAMETER VECTOR*,/,10G13.6)

RETURN

END

SUBROUTINE: BUILDZ

PURPOSE: To calculate unit noise covariance matrix
for reverse-time first-order filtering case
(under further development)

EQUATIONS:

$$Z = \sum_{k=1}^K (K-k+1) \underline{m}(k) \underline{m}^T(k)$$

where $\underline{m}(k)$ is the vector of unit-pulse responses
of the connection filters

ROUTINE VARIABLES

Z	Covariance matrix for unit noise
R	Work vector
NP1	system order plus one
NDIM	Maximum dimension of the matrix Z
NPT	Number of points in signal
NFIX	Not used (blank)

FURTHER DESCRIPTION:

SUBROUTINE BUILDZ(Z,R,NP1,NPT,NDIM,NFIX)

ALTERNATIVE NOISE COV PGM FOR *GNN*

DIMENSION Z(NDIM,1),R(1)

DOUBLE PRECISION DT

COMMON /DA0/ISPN,DELTA,SIG2,DT,GI,BIAS,IBIAS,DFAC

COMMON /IO/IR,ILT,IPR,ITPR,IZPR,IROUND,IPLT

Q=QI

TIME=DT*NPT

IOPT=NFIX+1

GO TO(201,101,101,201),IOPT

11 SC=DT

N=NP1-1

R(1)=1.0

DO 1 I=1,NP1

IF(I.GE.2)R(I)=R(I-1)

DO 1 J=1,NP1

Z(I,J)=0.0

DO 2 K=1,NPT

NPTK=NPT+1-K

DO 3 J=1,NP1

DO 3 I=J,NP1

Z(I,J)=Z(I,J)+R(I)*R(J)*NPTK

R(1)=0.0

DO 4 I=1,N

R(I+1)=2*R(I+1)+R(I)

CONTINUE

DO 7 J=1,NP1

DO 7 I=J,NP1

Z(I,J)=Z(I,J)+DT** (I+J-1)

GO TO 301

1 CONTINUE

DO 210 J=1,NP1

DO 210 I=J,NP1

IF(I.EQ.1)Z(1,1)=TIME

0 Z(I,J)=(TIME** (I+J-1))/(I+J-1)

WRITE(ILT,161)

1 CONTINUE

DO 174 I=1,NP1

DO 168 J=I,NP1

6 Z(I,J)=Z(J,I)

IF(IPR.GE.2)WRITE(ILT,220)(Z(I,J),J=1,NP1)

4 CONTINUE

3 FORMAT(10X,* QUANT. NOISE *)

1 FCRMAT(10X,*BIAS EFFECT*)

3 FORMAT(2X,5(2X,G13.6))

RETURN

END

SUBROUTINE: CORUPT

PURPOSE: This routine, useful in simulation mode, can be called when it is desired to add random noise of given variance to the signal

EQUATIONS:

ROUTINE VARIABLES

F	Input signal
X	First column of X would contain the corrupted signal; the second column temporarily contains the noise added to signal
VAR	Variance of noise added to signal
NPT	Number of signal points
NDIM	Maximum column dimensionality of X

FURTHER DESCRIPTION:

This routine needs a library routine to produce random numbers (gaussian, zero mean and uncorrelated)

SUBROUTINE CORUPT (F,X,VAR,NPT,NDIM)

 ADDS NOISE

DIMENSION F(1),X(NDIM,1)

DOUBLE PRECISION DT,AD,3D

COMMON /DA0/ISPN,DELTA,SIG2,DT,QI,BIAS,IBIAS,DFAC

COMMON /DA1/FBAR,EBAR,FESUM,EESUM

COMMON /IO/IR,ILT,IPR,ITPR,IZPR,IRJUND,IPLT

FBAR=0.

EBAR=0.

FESUM=0.

EESUM=0.

WRITE (ILT,469)VAR

IS=2458169

IS2=397665

SIGMA=SQRT (VAR)

CALL NRML(NPT,1,1,0.,SIGMA,IS,IS2,X(1,2),0)

DO 26 K=1,NPT

5 X(K,1)=F(K)+BIAS+X(K,2)

DO 211 K=1,NPT

FB=F(K)+BIAS

FBAR=FBAR+FB

EBAR=EBAR+X(K,2)

EESUM=EESUM+X(K,2)*X(K,2)

FESUM=FESUM+FB*X(K,2)

IF (ISPN.EQ.0) X(K,1)=X(K,2)

1 CONTINUE

EESUM=EESUM*DT

FESUM=2.0*FESUM*DT

FBAR=FBAR/NPT

EBAR=EBAR/NPT

WRITE (ILT,482)FBAR,EBAR,FESUM,EESUM

IF (IPR.LE.2) GO TO 411

WRITE (ILT,8)

WRITE (ILT,110) (X(K,1),K=1,NPT)

IF (ISPN.EQ.0) GO TO 411

WRITE (ILT,18)

WRITE (ILT,115) (X(K,2),K=1,NPT)

WRITE (ILT,1)

1 CONTINUE

9 CONTINUE

FCRMT STATEMENTS

FORMAT (10X,16HROUNDED F SIGNAL)

FCRMT (10X,16HROUNDOFF ERROR E)

3 FCRMT (2X,5 (2X,G11.4))

3 FCRMT (20(1X,F5.2))

3 FCRMT (1X,20(1X,F5.3))

2 FCRMT (2X,5HFBAR=,E11.4,6H EBAR=,E11.4,5H FE2=,E11.4,4H EE=,E11.4)

3 FCRMT (2X,*VARIANCE OF NOISE=*,E12.4)

FCRMT (/)

RETURN

END

SUBROUTINE: FILTER

PURPOSE: To produce the information signals. Specifically, this routine performs reverse-time first order filtering upon the given signal (stored in the first column of the matrix X)

EQUATIONS:

$$X(k,i+1) = q X(k+1,i+1) + X(k,i)$$

ROUTINE VARIABLES

X	Matrix of information signals. First column brings in the signal to be processed
NPT	Number of signal points
NP1	Number of information signals (model order plus one)
NDIM	Maximum column dimensionality of X
INT	Option parameter. -1 for reverse-time filtering, +2 for unit-shifts (or delays)

FURTHER DESCRIPTION:

SUBROUTINE FILTER(X,NPT,NP1,NDIM,INT)

DIMENSION X(NDIM,1)

DOUBLE PRECISION DT,SC,BD

COMMON /DA0/ISPA,DELTA,SIG2,DT,QI,BIAS,IBIAS,DFAC

COMMON /ID/IR,ILT,IPR,ITPR,IZPR,IROUND,IPLT

GENERATE FIRST-ORDER FILTER PROCESSED SIGNALS FROM DATA IN X(K,1)

INT=1 OR 0 FOR FORWARD, -1 FOR REVERSE TIME

FIRST-ORDER FILTERING

INT=2 FOR UNIT DELAYS (X(K,I+1)=X(K-1,I))

N=NP1-1

NP2=NP1+1

IOPT=INT+2

GO TO(51,11,11,91),IOPT

FORWARD FIRST-ORDER FILTERING

CONTINUE

DO 40 J=1,N

JJ=J+1

X(1,JJ)=X(1,1)

DO 40 K=2,NPT

K1=K-1

X(K,JJ)=QI*X(K1,JJ)+X(K,J)

CONTINUE

GO TO 70

REVERSE-TIME FIRST-ORDER FILTERING

CONTINUE

DO 60 J=1,N

JJ=J+1

X(NPT,JJ)=X(NPT,1)

X(NPT,JJ)=0.0

BD=X(NPT,JJ)

DO 60 KK=2,NPT

K=NPT+1-KK

K1=K+1

CHANGED NEXT CARD 3/17/80

BD=BD+QI*X(K,J)

BD=QI*BD+X(K,J)

X(K,JJ)=BD

CONTINUE

IF(IBIAS.EQ.0)GO TO 62

IPWR=IBIAS-1

DO 61 KK=1,NPT

TIME=DT*KK

K=NPT+1-KK

X(K,NP2)=TIME**IPWR

CONTINUE

GO TO 70

GENERATE UNIT DELAYS

CONTINUE

DO 93 I=2,NP1

I1=I-1

X(1,I)=0.0

DO 93 K=2,NPT

K1=K-1

X(K,I)=X(K1,I1)

GO TO 81

CONTINUE

SC=1.0

DO 90 I=2,NP1

SC=SC*DT

DO 90 K=1,NPT

X(K,I)=SC*X(K,I)

50 (FILTER 1)

```

1  CONTINUE
   IF (IPR.LT.4) GO TO 99
   IF (IROLND.EQ.1) WRITE (ILT,176) INT
   IF (IROLND.EQ.0) WRITE (ILT,179) INT
   DO 180 I=1,NP2
80  WRITE (ILT,110) (X(K,I),K=1,NPT)
   WRITE (ILT,1)
9   CONTINUE

110  FORMAT(4(1X,F12.6))
10  FCRMAT(10(1X,F7.3))
79  FCRMAT(10X,*FILTER,INT=*,I2,*   FOF PROCESSED X*)
78  FCRMAT(10X,*FILTER,INT=*,I2,*   NOISY FOF PROCESSED X*)
   FORMAT (/)
   RETURN
   END

```

SUBROUTINE:

FIX

PURPOSE:

To enhance the Gram matrix of the information signals
and to effect noise corrections
(Under further development)

EQUATIONS:

$$F = G - \sigma_{est} P$$

where P is the unit noise covariance matrix

ROUTINE VARIABLES

G	Noisy Gram matrix of reverse-time first-order filtered signals
P	Covariance matrix of unit noise (also reverse-time first-order filtered)
C	Corrected matrix
D	Work matrix
N	Dimensionality of the matrices
NC	Not used
SIG	Estimated noise variance
NDIM	Maximum dimensionality of the matrices
IFIX	Option parameter (Use IFIX=1)

FURTHER DESCRIPTION:

Under further development

SUBROUTINE FIX(G,P,C,D,X,N,NC,SIG,NDIM,IFIX)

ESTIMATE NOISE INTENSITY SIG (ASSUME WHITE NOISE)

CORRECT NOISY MATRIX= C

[P] DENOTES NOISE MATRIX FOR UNIT NOISE

NC IS THE NONZERO SUBMATRIX OF P =CCV OF NOISE

DIMENSION G(NDIM,1),P(NDIM,1),C(NDIM,1),D(NDIM,1),X(1)

COMMON /DA0/ISPN,DELTA,SIG2,DT,QI,3IAS,IEIAS,DFAC

COMMON /IO/IR,ILT,IPR,ITFR,IZPR,IRJUND,IPLT

DOUBLE PRECISION SUMDET,DT

SI=SIG

IF(IFIX.EQ.0)GO TO 51

GDP=G(1,1)/P(1,1)

JCT=0

SIG=0.0

JCT=JCT+1

SUMDET=0.0

CALL GKROCT(G,D,GDET,X,0,N,NDIM,0)

IF(JCT.EQ.1)DETG=GDET

DO 7 I=1,N

DO 7 J=1,N

SUMDET=SUMDET+D(I,J)*P(I,J)

IF(SUMDET.LT.0.0.AND.IFIX.NE.2)GO TO 11

SI=1.000/SUMDET

WRITE(ILT,32)JCT,ICT,GDET,SUMDET,SI

IF(SI/GDP.GT.0.1)WRITE(ILT,31)

ICT=0

CONTINUE

DO 9 I=1,N

DO 9 J=1,N

C(I,J)=G(I,J)-SI*P(I,J)

IF(IFIX.EQ.0)GO TO 11

CALL GKROCT(C,D,CDET,X,0,N,NDIM,0)

IF(CDET.LT.0.0.OR.CDET.GT.GDET)ICT=ICT+1

IF(ICT.GT.5)GO TO 10

IF(ICT.GT.0)SI=SI/2.0

IF(ICT.GT.0)GO TO 51

IF(JCT.GE.5)GO TO 11

THR=DETG*DFAC

IF(JCT.EQ.1)WRITE(ILT,33)DETG,DFAC,THR

SIG=SIG+SI

IF(CDET.GT.THR)CALL MEQUAT(N,N,G,C,NDIM,1)

WRITE(ILT,34)JCT,ICT,GDET,SI,CDET

IF(CDET.GT.THR)GO TO 3

FORMAT(2X,*NOISE VAR EXCESSIVE, SIG/GDP.GT.0.1*)

FORMAT(1X,*J, I GDET, SUMDET, SI:*,2I2,4E11.2)

FORMAT(1X,*GDET, DFAC, THR:*,6E11.2)

FORMAT(1X,*J, I SI, CDET*,2I2,4E11.2)

RETURN

END

SUBROUTINE:

GKRDCT

PURPOSE:

Basically, this routine finds the cofactors and/or the inverse of a square matrix. It also calculates the denominator parameters through pencil-of-functions method (if ISPN.GE.-1) or the LPC/Prony methods (ISPN.LE.-2)

EQUATIONS:

$$A(z) = D_1^{-1/2} (qz)^{-n} \sum_{i=1}^N \sqrt{D_i} (qz - i)^{N-i}$$

ROUTINE VARIABLES

X Gram matrix of information signals
Y The adjoint or the inverse matrix of X is returned in Y
DET The determinant of X is returned in this variable
XLAMDA Vector of computed denominator parameters
NN Not used (blank)
N Dimensionality of X and Y
MAX Maximum dimensionality of X and Y
IOPT Option parameter; 0 for finding the inverse and determinant of the matrix X, 1 to find the denominator parameters

FURTHER DESCRIPTION:

Scaling has been introduced in this routine to enable accurate computations even for high order modeling (say 6 to 10).

This routine calls BUILD to go from D_1 to the parameters a_1 of $A(z)$.

SUBROUTINE GKRDCT (X,Y,DET,XLAMDA,NN,N,MAX,IOPT)

DIMENSION XLAMDA(1)
DIMENSION X(MAX,1),Y(MAX,1)
DOUBLE PRECISION A,B,C,D,E
DIMENSION NUM(2,10),SCAL(10),RSC(10),Z(10,10)
DOUBLE PRECISION DT,SC,AD,BD
COMMON /DA0/ISPN,DELTA,SIG2,DT,QI,BIAS,IBIAS,DFAC
COMMON /IO/IR,ILT,IPR,ITPR,IZPR,IROUND,IPLT
IGKR=1
IF(N.NE.1)GO TO 3
Y(1,1)=1.0/X(1,1)
DET=X(1,1)
GO TO 61
CONTINUE
NMAT=N
LPC=0
IF(ISPN.LE.-2.AND.IOPT.EQ.1)LPC=1
IF(LPC.EQ.1)NMAT=N-1
-----SCALE---
DO 11 I=1,NMAT
SCAL(I)=1.0
IF(X(I+LPC,I+LPC).GT.0.1E-20)SCAL(I)=SQRT(X(I+LPC,I+LPC))
RSC(I)=1.0/SCAL(I)
CONTINUE
DO 6 I=1,NMAT
DO 6 J=1,NMAT
Y(J,I)=X(J+LPC,I+LPC)*RSC(I)*RSC(J)
Z(J,I)=Y(J,I)
IF(ITPR.GE.1)CALL PRMAT(Z,NMAT,NMAT,10,0)

---BEGIN GK REDUCTION---

A=1.0
DO 43 I=1,NMAT
B=0.0
L=I
M=I

FIND LARGEST ENTRY A(L,M) IN THE LOWER DIAGONAL SUBMATRIX

DO 18 J=I,NMAT
DO 18 K=I,NMAT
IF(ABS(Y(K,J)).LE.DABS(B))GO TO 18
B=ABS(Y(K,J))
L=K
M=J
CONTINUE

INTERCHANGE ROWS

IF(L.EQ.I)GO TO 24
DO 23 J=1,NMAT
C=Y(L,J)
Y(L,J)=Y(I,J)
Y(I,J)=C

INTERCHANGE COLUMNS

IF(M.EQ.I)GO TO 29

```

DO 20 J=1,NMAT
C=Y(J,M)
Y(J,M)=Y(J,I)
2 Y(J,I)=C
9 NUM(1,I)=L
NUM(2,I)=M
B=Y(I,I)
Y(I,I)=A
DO 42 J=1,NMAT
IF(J.EQ.I)GO TO 42
C=-Y(I,J)
Y(I,J)=0.0
DO 41 K=1,NMAT
D=C*Y(K,I)
E=B*Y(K,J)+D
IF(DABS(E).LT.1.00-10*DABS(D))E=0.0
1 Y(K,J)=E/A
2 CONTINUE
3 A=B

```

RESTORE COLUMNS

```

DO 56 I=2,NMAT
J=NMAT+1-I
K=NUM(2,J)
IF(K.EQ.J)GO TO 52
DO 51 L=1,NMAT
C=Y(K,L)
Y(K,L)=Y(J,L)
Y(J,L)=C
K=NUM(1,J)

```

RESTORE ROWS

```

IF(K.EQ.J)GO TO 56
DO 57 L=1,NMAT
C=Y(L,K)
Y(L,K)=Y(L,J)
Y(L,J)=C
CONTINUE
DET=A
DSCAL=1.0
DO 59 I=1,NMAT
DSCAL=SCAL(I)*DSCAL
DET=DET*SCAL(I)*SCAL(I)
IF(ITPR.GE.1)WRITE(ILT,337)DET,A,(RSC(I),I=1,NMAT)
7 FORMAT(1X,'DET,A, RSC(I): ',7E11.4)
IF(IOPT.EQ.1.AND.ISPN.GE.-1)GO TO 61
IF(ITPR.GE.1)CALL PRMAT(Y,N,N,MAX,0)
IF(ITPR.GE.1)CALL PROMAT(Z,Y,NMAT,10,MAX,0)
DO 60 I=1,NMAT
DO 60 J=1,NMAT
Y(I,J)=Y(I,J)+RSC(I)*RSC(J)/A
CONTINUE
IF(IOPT.NE.1)GO TO 1000
IF(ISPN.LE.-2)GO TO 440
IF(Y(1,1).LT.0.0)GO TO 1000
.....
SC=1.0
DO 200 I=2,NMAT
SC=SC*CT
RSC(I)=RSC(I)/RSC(1)
IF(IGKR.EQ.0)XLAMDA(I)=RSC(I)*Y(I,1)/Y(1,1)
IF(IGKR.EQ.0)GO TO 199

```

```

A=Y(I,I)
IF (Y(I,I).LT.(0.0)) A=0.0
IF (IGKR.EQ.2) A=ABS(Y(I,I))
XLAMDA(I)=RSC(I)*DSQRT(A/Y(1,1))
IF (Y(I,1).LT.0.0) XLAMDA(I)=-XLAMDA(I)
99 XLAMDA(I)=SC*XLAMDA(I)
00 CONTINUE
XLAMDA(1)=1.000
AT=RSC(1)*OSCAL*SQRT(Y(1,1))
IF (IPR.GE.1) WRITE (6,106) (XLAMDA(I), I=1,NMAT), AT
NPP=N
IF (IBIAS.NE.0) NPP=N-1
CALL BUILDY(Y,XLAMDA,NPP,MAX)
16 FORMAT(5X,*SYNTHETIC VECTOR, AND SQRT(Y11)*,/,10G12.5)
GO TO 1000
10 CONTINUE
IF (ITPR.LE.0) GO TO 449
WRITE (ILT,448)
CALL PRMAT(Y,N,N,MAX,0)
DO 442 I=1,NMAT
DO 442 J=1,NMAT
2 Z(I,J)=X(I+1,J+1)
CALL PRMAT(Z,Y,NMAT,10,MAX,0)
7 FCRMAT(2X,9G11.4)
8 FORMAT(2X,* INVERSE AND PRODUCT MATRICES*)
9 CONTINUE
XLAMDA(1)=1.0
DO 450 I=2,N
XLAMDA(I)=0.0
DO 450 J=1,NMAT
0 XLAMDA(I)=XLAMDA(I)-Y(I-1,J)*X(J+1,1)
00 CONTINUE
RETURN
END

```

SUBROUTINE MEQUAT

PURPOSE: To set an $m \times n$ dimensional matrix B equal to a matrix A of the same dimensionality

Equation: $B = A$

ROUTINE VARIABLES: M Row dimensionality of B and A
 N Column dimensionality of B and A
 B Matrix to be set
 A Matrix to which the matrix B is set
 NDIM Maximum number of rows permissible
 IOPT Print option; 0 for no printing, 2 or greater for printing

SUBROUTINE MEQUAT(M,N,B,A,NDIM,IOPT)

 IOPT=0 SET B TO ZERO

1 B EQUAL TO A

10 B TO IDENTITY

DIMENSION A(NDIM,1),B(NDIM,1)

DO 33 I=1,M

DO 33 J=1,N

IF(IOPT.NE.1) B(I,J)=0.0

IF(IOPT.EQ.10.AND.I.EQ.J) B(I,J)=1.0

IF(IOPT.EQ.1) B(I,J)=A(I,J)

CONTINUE

RETURN

END

SUBROUTINE: PLOP

PURPOSE: To plot a pair of columns of the array X
(This routine may be substituted by user's own routine)

SUBROUTINE PLOP(NPT,NF,Y,NDIM,T0,DT,LABEL,INDEP,IBUF)

NPT=NUMB OF TIME PTS (WARNING: NDIM SHOULD BE GE. NPT+2)

NF=NUMBER OF FUNS

Y(K,) DATA ARRAY OF DIMENSION NDIM,NF

T0=INITIAL TIME, DT=TIME INCREMENT

LABEL, INDEP = TITLES FOR Y AND X AXES

DIMENSION Y(NDIM,NF),YY(2),LABEL(1),INDEP(1)

DIMENSION X(512),IBUF(512)

COMMON /IO/IR,ILT,IPR,ITPR,IZPR,IROUND,IPLT

M=NF*NDIM

M1=M+1

M2=M+2

NPT1=NPT+1

NPT2=NPT+2

X(1)=T0

DO 9 K=2,NPT

X(K)=X(K-1)+DT

DO 8 I=1,NF

DO 8 K=NPT1,NDIM

Y(K,I)=Y(NPT,I)

INITIALIZE(LIQ,INK, 12IN,PAPER) MAX.LENGTH=60 IN

CALL PLOTMX(60.0)

SET ORIGIN

CALL PLOT(0.,-.5,3)

CALL FACTOR(5.0/6.5)

BEGIN PLOTTING

CALL SCALE(X,6.5,NPT,1)

CALL SCALE(Y(1,1),10.0,M,1)

CALL AXIS(0.,0.,11*TIME (SEC.),

*-16,6.5,0.,X(NPT1),X(NPT2))

CALL AXIS(0.,0.,16*RESPONSES Y ,Y,

*16,10.,30.,Y(M1),Y(M2))

WRITE(6,6)X(NPT1),X(NPT2)

WRITE(6,7)Y(M1),Y(M2)

FORMAT(1X,*T0,DIV (6.5 DIV)*,4(1X,F7.3))

FORMAT(1X,*Y0,DIV (10 DIV)*,4(1X,F7.3))

DO 10 I=1,NF

Y(NPT1,I)=Y(M1)

Y(NPT2,I)=Y(M2)

IF(I.EQ.1)CALL LINE(X,Y(1,I),NPT,1,I-1,I)

IF(I.EQ.2)CALL DASHLN(X,Y(1,2),NPT,1)

CONTINUE

CALL PLOT(10.,0.,-3)

RETURN

END

SUBROUTINE: POLCON

PURPOSE: To combine the factors of a polynomial in order to produce the coefficients.

SUBROUTINE POLCON(C,R2,K,N)

A POLYNOMIAL CONSTRUCTION PROGRAM NEEDED FOR ZTOS

DIMENSION C(1),R2(1)

COMPLEX C,R2,COMP

DIMENSION DC(2)

EQUIVALENCE (COMP,DC)

NP1=N+1

DO10 I=2,NP1

R2(I)=0.0000

R2(1)=1.0000

DO4 I=1,N

COMP=C(I)

IF (I.EQ.K.OR. (DC(1).EQ.0.000.AND.DC(2).EQ.0.000)) GO TO 4

DO2JJ=1,I

J=I-JJ+1

R2(J+1)=R2(J+1)*C(I)+R2(J)

R2(1)=R2(1)*C(I)

CONTINUE

RETURN

END

EQUATION: $(x - c_1)(x - c_2) \dots (x - c_n)$

$$= r_1 + r_2x + \dots + r_nx^{n-1} + r_{n+1}x^n$$

VARIABLES:

C Vector containing the roots of the factors
R2 Vector returning the coefficients of the polynomial
K Exclude Kth factor if K#0
N Number of roots contained in C

SUBROUTINE:

POLRT

PURPOSE:

To find the roots of a polynomial

EQUATIONS:

$$a_1 + a_2x + \dots + a_{n+1}x^n$$

$$(x - p_1 - jq_1)(x - p_2 - jq_2) \dots (x - p_n - jq_n)$$

ROUTINE VARIABLES

XCOF	Coefficients of the polynomial (XCOF(1)=a ₁)
COF	Work vector
M	Order of polynomial
ROOTR	Real parts of the roots are returned in this vector
ROOTI	Imaginary parts of the roots are returned in this vector
IER	Type of error, if any, returned in this integer variable

FURTHER DESCRIPTION:

SUBROUTINE POLRT(XCOF,COF,M,ROOTR,ROOTI,IER)

COMPUTES THE REAL AND COMPLEX ROOTS OF A REAL POLYNOMIAL

DESCRIPTION OF PARAMETERS

XCOF -VECTOR OF M+1 COEFFICIENTS OF THE POLYNOMIAL
ORDERED FROM SMALLEST TO LARGEST POWER

COF -WORKING VECTOR OF LENGTH M+1

M -ORDER OF POLYNOMIAL

ROOTR-RESULTANT VECTOR OF LENGTH M CONTAINING REAL ROOTS
OF THE POLYNOMIAL

ROOTI-RESULTANT VECTOR OF LENGTH M CONTAINING THE
CORRESPONDING IMAGINARY ROOTS OF THE POLYNOMIAL

IER -ERROR CODE WHERE

IER=0 NO ERROR

IER=1 M LESS THAN ONE

IER=2 M GREATER THAN 36

IER=3 UNABLE TO DETERMINE ROOT WITH 500 ITERATIONS
ON 5 STARTING VALUES

IER=4 HIGH ORDER COEFFICIENT IS ZERO

DIMENSION XCOF(1),COF(1),ROOTR(1),ROOTI(1)

DOUBLE PRECISION XO,YO,X,Y,XPR,YPR,UX,UY,V,YT,XT,U,XT2,YT2,SUMSQ,
1 DX,DY,TEMP,ALPHA,XCOF,COF,ROOTR,ROOTI,ER1,ER2,XSS,XS,YSS,YS,TCL
COMMON /IO/IR,ILT,IPR,ITPR,IZPR,ROUND,IPLT

LIMITED TO 36TH ORDER POLYNOMIAL OR LESS.

FLOATING POINT OVERFLOW MAY OCCUR FOR HIGH ORDER

POLYNOMIALS BUT WILL NOT AFFECT THE ACCURACY OF THE RESULTS.

METHOD

NEWTON-RAPHSON ITERATIVE TECHNIQUE. THE FINAL ITERATIONS
ON EACH ROOT ARE PERFORMED USING THE ORIGINAL POLYNOMIAL
RATHER THAN THE REDUCED POLYNOMIAL TO AVOID ACCUMULATED
ERRORS IN THE REDUCED POLYNOMIAL.

ER2=1.00+50

TCL=1.00-8

IFIT=0

N=M

IER=0

IF(XCOF(N+1)) 10,25,10

10 IF(N) 15,15,32

SET ERROR CODE TO 1

15 IER=1

20 IF(IER)200,201,200

1 WRITE(6,203)IER

3 FORMAT(1X,'*ERROR CALLED FROM POLRT, IER = ',I3)

1 RETURN

SET ERROR CODE TO 4

25 IER=4

GO TO 20

SET ERROR CODE TO 2

```

30 IER=2
   GO TO 20
32 IF (N-36) 35,35,30
35 NX=N
   NXX=N+1
   N2=1
   KJ1 = N+1
   DO 40 L=1,KJ1
   MT=KJ1-L+1
40 COF(MT)=XCOF(L)

```

SET INITIAL VALUES

```

45 XO=.00500101
   YO=0.01000101

```

ZERO INITIAL VALUE COUNTER

```

-----BEGIN ITERATION-----
X AND Y ARE THE REAL AND IMAG PARTS OF ROOT
IN=0
50 X=XO

```

INCREMENT INITIAL VALUES AND COUNTER

```

XO=-10.0*YO
YO=-10.0*X

```

SET X AND Y TO CURRENT VALUE

```

X=XO
Y=YO
IN=IN+1
GO TO 59
55 IFIT=1
   XPR=X
   YPR=Y

```

EVALUATE POLYNOMIAL AND DERIVATIVES

```

59 ICT=0
60 UX=0.0
   UY=0.0
   V =0.0
   YT=0.0
   XT=1.0
   U=COF(N+1)
   IF (L) 65,130,65
65 DO 70 I=1,N
   L =N-I+1
   TEMP=COF(L)
   XT2=X*XT-Y*YT
   YT2=X*YT+Y*XT
   U=U+TEMP*XT2
   V=V+TEMP*YT2
   FI=I
   UX=UX+FI*XT*TEMP
   UY=UY-FI*YT*TEMP
   XT=XT2

```

```

70 YT=YT2
   SUMSQ=LX*JX+LY*JY
   IF (SUMSQ) 75,110,75
75 DX=(V*LY-U*JX)/SUMSQ
   X=X+DX
   DY=-(U*JY+V*JX)/SUMSQ
   ILOC=75
   IF (ITPR.GE.1) WRITE (ILT,442) ILOC
   Y=Y+DY
   XSS=X
   YSS=Y
   IF (YSS.EQ.0.000) YSS=1.000
   IF (XSS.EQ.0.000) XSS=1.000
   RMAG=SQRT(XSS*XSS+YSS*YSS)
   MODIF,APR,80  *SS TO *SS+.00001*RMAG IN NEXT CARD
   ER1=A3S(DX/(XSS+.00001*RMAG))+A3S(DY/(YSS+.00001*RMAG))
   IF (ITPR.GE.1) WRITE (ILT,444) CX,XSS,DY,YSS,ER1,ER2
   ILOC=77
   IF (ITPR.GE.1) WRITE (ILT,442) ILOC
   IF (ER1.GT.ER2) GO TO 78
   ER2=ER1
   XS=XSS
   YS=YSS
   IF (ER1-TCL) 100,80,80

```

STEP ITERATION COUNTER

```

80 ICT=ICT+1
   IF (ICT-500) 60,85,85
85 IF (IFIT) 100,90,100
90 IF (IN-5) 50,95,95
   -----EXIT FROM ITERATIONS-----

```

SET ERROR CODE TO 3

```

95 IER=3
   X=XS
   Y=YS
   ER1=ER2
100 DO 105 L=1,NXX
   MT=KJ1-L+1
   TEMP=XCOF(MT)
   XCOF(MT)=COF(L)
105 COF(L)=TEMP
   ITEMP=N
   N=NX
   NX=ITEMP
   IF (IFIT) 120,55,120
110 IF (IFIT) 115,50,115
115 X=XPR
   Y=YPR
20 IFIT=0
   IF (ABS(Y)-1.00-8*A3S(X)) 135,125,125
125 ALP=A=X+X
   SUMSQ=X*X+Y*Y
   N=N-2
   GO TO 140
30 X=0.0
   NX=NX-1
   NXX=NXX-1

```

```

135 Y=0.0
    SUMSQ=0.0
    ALPHA=X
    N=N-1
140 COF(2)=COF(2)+ALPHA*COF(1)
145 DO 150 L=2,N
150 COF(L+1)=COF(L+1)+ALPHA*COF(L)-SUMSQ*COF(L-1)
155 ROOTI(N2)=Y
    ROOTR(N2)=X
    IF(ER1.GT.TCL)WRITE(6,554)N2,ER1
554 FCRMAT(1X,*ERROR ON *,I3,* TH ROOT IS *,D10.3)
    ER2=1.00+50
    N2=N2+1
    IF(SUMSQ) 160,165,160
160 Y=-Y
    SUMSQ=0.0
    GO TO 155
165 IF(N) 20,20,45
,2 FCRMAT(2X,*TEST IN POLRT*,I5)
,4 FCRMAT(1X,*DX,XSS,DY,YSS,ER1-2*,2E8.1,2X,2E8.1,2X,2E9.2)
    RETURN
    END

```

SUBROUTINE:

PRDMAT

PURPOSE:

This subroutine computes the product of two square matrices.

EQUATIONS:

$A \leftarrow A * B$

ROUTINE VARIABLES: A, B N x N matrices

NDIM1 Maximum row dimension of A

NDIM2 Maximum row dimension of B

LOC An integer which is printed

SUBROUTINE PROMAT(A,B,N,NDIM1,NDIM2,LOC)

COMPLETES PRODUCT A = A*B

DIMENSION A(NDIM1,1),B(NDIM2,1),C(10,10)

IF(LOC.GE.1)WRITE(6,5)LOC

DO 31 I=1,N

DO 31 J=1,N

C(I,J)=0.0

DO 21 K=1,N

C(I,J)=C(I,J)+A(I,K)*B(K,J)

CONTINUE

DO 41 I=1,N

DO 41 J=1,N

A(I,J)=C(I,J)

DO 45 I=1,N

WRITE(6,15)(A(I,J),J=1,N)

FORMAT(* LOCATION/INTEGER=*,I5)

FORMAT(2X,10G13.6)

RETURN

END

FUNCTION COMB(N,M)

CALCULATES COMBINATION M CUT OF N

IF(N.LE.0)GO TO 99

L=1

LD=1

IF(M.EQ.0)GO TO 8

MN1=N-M+1

DO 5 I=MN1,N

L=L*I

DO 7 I=1,M

LD=LD*I

COMB=L/LD

RETURN

END

SUBROUTINE: PRTMAT, PRTVEC, PRCVEC, PRVEC

PURPOSE: These four subroutines perform printing of arrays,
 PRTMAT of matrices and the other three of vectors.
 See subroutines for comments.

EQUATIONS:

ROUTINE VARIABLES

Subroutine PRTMAT

A	Matrix being printed
M	Its row dimensionality
N	Its column dimensionality
NDIM	Maximum number of rows permitted
LOC	Use LOC=0. If nonzero, this same number is printed.

FURTHER DESCRIPTION:

SUBROUTINE PRMAT(A,M,N,NDIM,LOC)

PPRINTS A MATRIX, AND AN INTEGER (PERHAPS A LOCATION) IF LOC.GE.1

DIMENSION A(NDIM,1)
IF (LOC.GE.1) WRITE(6,5) LOC
DO 31 I=1,M
WRITE(6,15) (A(I,J),J=1,N)
FORMAT(* LOCATION/INTEGER=*,I5)

FORMAT(2X,10G13.6)
RETURN
END

SUBROUTINE PRVVEC(A,N)

PRINTS A COMPLEX VECTOR, MAX N=5
WHEN VARIABLE IS SINGLE PRECISION

COMPLEX A(5)
WRITE(6,1) (A(I),I=1,N)
FORMAT(1X,2E12.5,4(3X,2E12.5))
RETURN
END
SUBROUTINE PRVVEC(A,N)

PRINTS A COMPLEX VECTOR, MAX N=5
WHEN VARIABLE IS DOUBLE PRECISION

COMPLEX A
DIMENSION A(1)
WRITE(6,1) (A(I),I=1,N)
FORMAT(1X,2D12.5,4(3X,2D12.5))
RETURN
END
SUBROUTINE PRVVEC(A,N)

THIS SUBROUTINE OUTPUTS DOUBLE PRECISION SINGLE DIMENSIONED ARRAY

DIMENSION A(1)
WRITE(6,1) (A(I),I=1,N)
FORMAT(1X,6D16.6)
RETURN
END

SUBROUTINE: RESPON

PURPOSE: To determine the response of the digital transfer function $H(z)$ to an input sequence $v(k)$. The coefficients of $H(z)$ are given to the routine in the NPNP2=N+1 vector GAMMA

EQUATIONS:

$$H(z) = \frac{b_0 + b_1 z^{-1} + \dots + b_n z^{-n}}{1 + a_1 z^{-1} + \dots + a_n z^{-n}}$$

$$x_k = -a_1 x_{k-1} - \dots - a_n x_{k-n} + b_0 v_k + \dots + b_n v_{k-n}$$

ROUTINE VARIABLES

X	The vector which returns the response of $H(z)$
V	Vector containing the input sequence
N	order of transfer function H
GAMMA	Vector of coefficients of H GAMMA = (1, a_1 , ..., a_n , $-b_0$, $-b_1$, ..., $-b_n$)
XLAMDA	Work vector
MP1	Number of response points generated

FURTHER DESCRIPTION:

The routine assumes zero initial conditions

SUBROUTINE RESPON(X,V,N,GAMMA,XLAMDA,MP1)

DIMENSION X(1),V(1),GAMMA(1),XLAMDA(1)

DOUBLE PRECISION XSAV,AC,20

NM1=N-1

NP1=N+1

NPNP1=N+N+1

NPNP2=N+N+2

DO 19 I=1,NPNP1

XLAMDA(I)=0.0

XSAV=0.0

DO 20 K=1,MP1

IF(N.EQ.1)GO TO 25

DO 21 I=1,NM1

J=NP1-I

XLAMDA(J)=XLAMDA(J-1)

CONTINUE

DO 22 I=1,N

J=NPNP2-I

XLAMDA(J)=XLAMDA(J-1)

XLAMDA(1)=XSAV

XLAMDA(NP1)=V(K)

XSAV=0.0

DO 23 I=1,NPNP1

XSAV=XSAV-GAMMA(I+1)*XLAMDA(I)

IF(DABS(XSAV).GE.1.0E10)XSAV=0.0

X(K)=XSAV

RETURN

END

SUBROUTINE: SIGNAL

PURPOSE: This routine generates a signal which is a weighted sum of exponential*sinusoid terms

EQUATIONS:

$$f(t) = \sum_{i=1}^m W_i e^{-\alpha_i t} \sin(\beta_i t + \phi_i)$$

$$F(k) = f(k\Delta)$$

ROUTINE VARIABLES

F	Vector returning the generated signal
NPT	Number of signal points generated
AMP	Vector of weights associated with each exp*sinusoid term
SR	Vector of exponents "
SI	Vector of radian frequencies "
SPH	Vector of phases "
DT	Sampling interval
NCOMP	Number of terms

FURTHER DESCRIPTION:

This routine is useful only in the simulation mode and is called when ISIM=2

SUBROUTINE SIGNAL(F,NPT,AMP,SR,SI,SPH,DT,NCOMP)

DIMENSION F(1),AMP(1),SR(1),SI(1),SPH(1)
COMMON /IO/IR,ILT,IPR,ITPR,IZPR,IROUND,IPLT
DOUBLE PRECISION A,B,C,X

DO 12 K=1,NPT

F(K)=0.0

DO 20 I=1,NCOMP

A=SR(I)*DT

B=SI(I)*DT

C=SPH(I)

DO 15 KK=1,NPT

K=KK-1

X=AMP(I)

IF(A.NE.0.0)X=X*DEXP(A*K)

IF(B.NE.0.0)X=X*DSIN(B*K+C)

F(KK)=X+F(KK)

CONTINUE

IF(IPR.LT.2)GO TO 30

WRITE(ILT,9)

WRITE(ILT,6)(F(K),K=1,NPT)

WRITE(ILT,1)

CONTINUE

FORMAT(/)

FORMAT(20(1X,F5.2))

FORMAT(10X,* F SIGNAL*)

RETURN

END

SUBROUTINE: VEQUAT

PURPOSE: To equate a vector Y to another suitable vector

EQUATIONS: Y(k) = 0 vector if IOPT=0
 = X(k) =1 or 2 (print also)
 = Y(k)*X(1) =3
 = Y(k)+X(1) =4
 = δ_k (1,0,0,0,...) =10
 = 1 (1,1,1,1,...) =11

VARIABLES:

NPT Dimensionality of X
Y The vector to be set
X Auxiliary vector
NPUL Not used
IOPT Option parameter (see above)

SUBROUTINE VEQUAT(NPT,Y,X,NPUL,IOPT)

IOPT=0 SET Y TO ZERO
1 OR 2 SET Y=X (PRINT IF 2)
3 SET Y= Y* CONST X(1)
4 SET Y= Y+ CONST X(1)
9 SET Y TO ZERO
10 SET Y=IMPULSE
11 SET Y=STEP
DIMENSION X(1),Y(1)
IF (IOPT.EQ.0) IOPT=9
DO 33 K=1,NPT
IF (IOPT.EQ.1.OR.IOPT.EQ.2) Y(K)=X(K)
IF (IOPT.EQ.3) Y(K)=Y(K)*X(1)
IF (IOPT.EQ.4) Y(K)=Y(K)+X(1)
IF (IOPT.GE.9) Y(K)=0.0
IF (IOPT.EQ.11) Y(K)=1.0
CONTINUE
IF (IOPT.EQ.2) WRITE (6,6) (Y(K),K=1,NPT)
FORMAT (2X,10G12.5)
IF (IOPT.EQ.10) Y(1)=1.0
RETURN
END

SUBROUTINE: ZTOS

PURPOSE: Given the z-domain transfer function $H(z)$ this routine computes the equivalent s-domain transfer function $H(s)$

EQUATIONS:	$H(z)$	$H(s)$	Impulse invariant if IZTS=0
			Pulse invariant " =1
			Trapezoid invariant " =2

ROUTINE VARIABLES

B Vector of denominator parameters
A Vector of numerator parameters

Comment - unfortunately, these names are opposite to the convention used in the rest of the report. However, the reader need not be concerned about this unless he wishes to study this routine; in the latter case, he should bear this in mind.

N	Order of the transfer function
DELTA	Sampling interval
IZTS	Option for type of conversion (see above)

FURTHER DESCRIPTION:

See Gold and Rader, or Oppenheim and Schafer, or Stanley for the theory of z-domain to s-domain converison.

SUBROUTINE ZTOS (B,A,N,DELTA,IZTS)

GIVEN THE DISCRETE DESCRIPTION THIS SUBROUTINE COMPUTES THE EQUIVALENT CONTINUOUS DOMAIN DESCRIPTION OF A LINEAR DYNAMIC SYSTEM

THE INPUT ARRAYS A AND B ARE FILLED ACCORDING TO THE DIFFERENCE EQUATION

$$B(1)*Y(K)+B(2)*Y(K-1)+...+B(N+1)*Y(K-N) - A(1)*U(K)-A(2)*U(K-1)-...-A(N+1)*U(K-N) = 0$$

B(1) MUST EQUAL 1

POLES OF THE CONTINUOUS DOMAIN MUST BE DISTINCT AND NON-ZERO FOR THE TRANSFORMATION TO BE VALID

UPON RETURNING ARRAYS A AND B CONTAIN THE EQUIVALENT CONTINUOUS DESCRIPTION STORED ACCORDING TO THE DIFFERENTIAL EQUATION

$$B(1)*Y(T)+B(2)*D(1,Y(T))+...+B(N+1)*D(N,Y(T)) - A(1)*U(T)-A(2)*D(1,U(T))-...-A(N+1)*D(N,U(T)) = 0$$

WHERE D(M,F(T)) = THE MTH TIME DERIVATIVE OF FUNCTION, F

B(N+1) ALWAYS IS 1

N = ORDER OF SYSTEM

N (MAXIMUM) = ONE LESS THAN THE DIMENSION SUBSCRIPT

IZTS=0 ---> IMPULSE

IZTS=1 ---> PULSE

IZTS=2 ---> TRAPAZOIDAL

DELTA = SAMPLING INTERVAL = 1/(SAMPLING FREQUENCY)

COMPLEX CR,CA,CF,CG,CF1,CON1,CON2,CONT

COMPLEX POLE(20),ZRO(20)

DIMENSION B(1),A(1),TEMP(20),RR(20),RI(20),CR(20),CA(20),CF(20),
ICG(20),CF1(20),ZR(20),ZI(20)

COMMON /IO/IR,ILT,IPR,ITPR,IZPR,IROUND,IPLT

CONT=0.0000

NP1=N+1

IF(A(NP1))410,411,410

1 ICHECK=2

GO TO 401

1 ICHECK=1

GO TO 400

1 CONT=A(NP1)/B(NP1)

DO402I=1,N

1 A(I)=A(I)-CONT*B(I)

IF(ITPR.GE.1)WRITE(ILT,441)

1 CALL POLRT(B,TEMP,N,RR,RI,IER)

IF(ITPR.GE.1)WRITE(ILT,442)IER

FORMAT(2X,*GOING INTO POLRT*)

1 FORMAT(2X,*RETURNED FROM POLRT, IER=*,I2)

DO6I=1,N

CR(I)=CMPLX(RR(I),RI(I))

CF(I)=1.0000/CR(I)

IF(IZPR.GE.1)WRITE(6,1002)

2 FORMAT(* THE POLES OF THE Z-DOMAIN*)

IF(IZPR.GE.1)CALL PROVEC(CF,N)

IF(IZPR.EQ.1)GO TO 1101

DO3I=1,N

CON2=1.0000

```

CON2=0.0000
DO4J=1,N
CON2=CON2*CR(I)+A(N-J+1)
IF(I-J)5,4,5
CON1=CON1*(1.0000-CR(I)+CF(J))
CONTINUE
CA(I)=CON2/CON1
IF(IZTS-1)225,224,230
25 DO222I=1,N
   CA(I)=CA(I)/DELTA
22 CR(I)=CLOG(CR(I))/DELTA
   GO TO 226
24 DO2I=1,N
   CON1=CLOG(CR(I))/DELTA
   CA(I)=CA(I)*CR(I)*CON1/(CR(I)-1.0000)
   CR(I)=CON1
   GO TO 226

10 ICHECK=2
DO231I=1,N
CON1=CLOG(CR(I))/DELTA
CON2=CA(I)*CR(I)/((1.0000-CR(I))*(1.0000-CR(I)))
CONT=CCNT-CON2*(1.0000+CON1*DELTA-CR(I))
CA(I)=CON2*CON1*CCN1*DELTA
1 CR(I)=CON1
  WRITE(6,2131)
31 FORMAT(2X,*TIME SCALED FOR SCATTERER*)
6  WRITE(6,1004) IZTS
04 FORMAT(1X,*S-POLES*,5X,*SR*,9X,*SI*,9X,*SMAG*,9X,*FR*,7X,
+*IZTS=*,I2)
DT=0.001953125
RDT=DELTA/DT
RDT=1.0
DO 226 I=1,N
SSR=-REAL(CR(I))*RDT
SSI=AIMAG(CR(I))*RDT
SSM=CABS(CR(I))*RDT
SSFR=(SSM/6.2831853)
8  WRITE(6,1010) I,SSR,SSI,SSM,SSFR
  IF(IPR.LE.0)GO TO 1101
10 FORMAT(3X,I2,5F12.4)
  IF(IPR.GE.1)WRITE(6,1003)
13 FORMAT(* NUMERATOR CONSTANTS OF FACTORIZED H(S)*)
  IF(IPR.GE.1)CALL PROVEC(CA,N)
DO 240 I=1,N
1  POLE(I)=CR(I)
  CALL POLCON(CR,CG,0,N)
DO7I=1,NP1
CF(I)=0.0000
DO9K=1,N
  CALL POLCON(CR,CF1,K,N)
DO9J=1,N
  CF(J)=CF(J)+CF1(J)*CA(K)
CF(NP1)=0.0000
GO TO (403,404),ICHECK
CF(NP1)=CONT
DO405I=1,N
CF(I)=CF(I)+CONT*CG(I)
CONTINUE
IF(IPR.LE.1)GO TO 520

```

FIND ZEROS

```

07 DO 507 I=1,NP1
   A(I)=CF(I)
   NC=0
   DO 510 I=1,NP1
     IF (ABS(A(I)).GT.1.D-6) NC=I-1
10 CONTINUE
   IF (NC.EQ.0) GO TO 520

   N1=NC+1
   AK=A(N1)
   DO 515 I=1,N1
     A(I)=A(I)/AK
15 CALL PCLRT(A,TEMP,NC,RR,RI,IER)
   DO 517 I=1,N0
     ZR(I)=RR(I)
     ZI(I)=RI(I)
     ZRO(I)=CMPLX(ZR(I),ZI(I))
   .7 CONTINUE
     IF (IPR.GE.2) WRITE (6,1007) AK

07 FORMAT(* ZEROS OF H(S), NUMERATOR CONSTANT =*,E13.6)
   IF (IPR.GE.2) CALL PRTEC(ZRO,N0)

0 CONTINUE
   DO 20 I=1,NP1
     B(I)=CG(I)
     A(I)=CF(I)
     IF (IPR.GE.1) WRITE (6,1005)
05 FORMAT(* S-DOMAIN DENOMINATOR*)
     IF (IPR.GE.1) CALL PRVEC(B,NP1)
     IF (IPR.GE.1) WRITE (6,1006)
06 FORMAT(* S-DOMAIN NUMERATOR*)
     IF (IPR.GE.1) CALL PRVEC(A,NP1)
01 RETURN
   END

```

APPENDIX B

Modeling of a Noisy Test Signal

The signal considered in example 1 of Section IV is

$$x(k) = y(k) + w(k)$$

where $y(k)$ is the impulse response of $(1 - 1.92z^{-1} + z^{-2})/(1 - 2.68z^{-1} + 2.476z^{-2} - 0.782z^{-3})$ and $w(k)$ is additive white noise. The true signal of interest, $y(k)$, is shown in Fig. B1 and the signal under test, $x(k)$, is shown in Fig. B2.

Given below are the (card deck) input to program POF-FILTER and, succeeding it, the printer output from the program.

INPUT CARDS

```

EXAMPLE 1      LPC. PROXY VS. PENCIL-OF-FNS.
NPT      IP_LSP IPLT  XMSE      DT      BIAS      ANBIAS      SIG2
      N      +POOMP NNPT      DT      ANBIAS
0100      5 1+2 1 100 -5.000000 -1.000000 +0.000000 +0.000000 +0.0001
+1.000000 -2.680000 +2.476000 -0.782000
+1.000000 -1.920000 -1.000000
IPR IPRM ISPN IFIX NFIX -- PENCIL-OF-FUNCTIONS--
2-1 1-1-1 1+0 1 100 +0.8
IPR IPRM ISPN IFIX NFIX -- PENCIL-OF-FUNCTIONS: LINEAR--
2-1 1-1 1+0 1 100 +0.000000 +0.00011
IPR IPRM ISPN IFIX NFIX -- LPC: AUTOCORR--
2-1 0-3-1 1+0 1 100 +1.0
IPR IPRM ISPN IFIX NFIX -- LPC: COV--
2-1 0-2-1 1+0 1 100 +1.0
IPR IPRM ISPN IFIX NFIX -- PROXY. AUTOCORR--
2-1 1-3-1 1+0 1 100 +1.0
IPR IPRM ISPN IFIX NFIX -- PROXY. COV--
2-1 1-2-1 1+0 1 100 +1.0

```

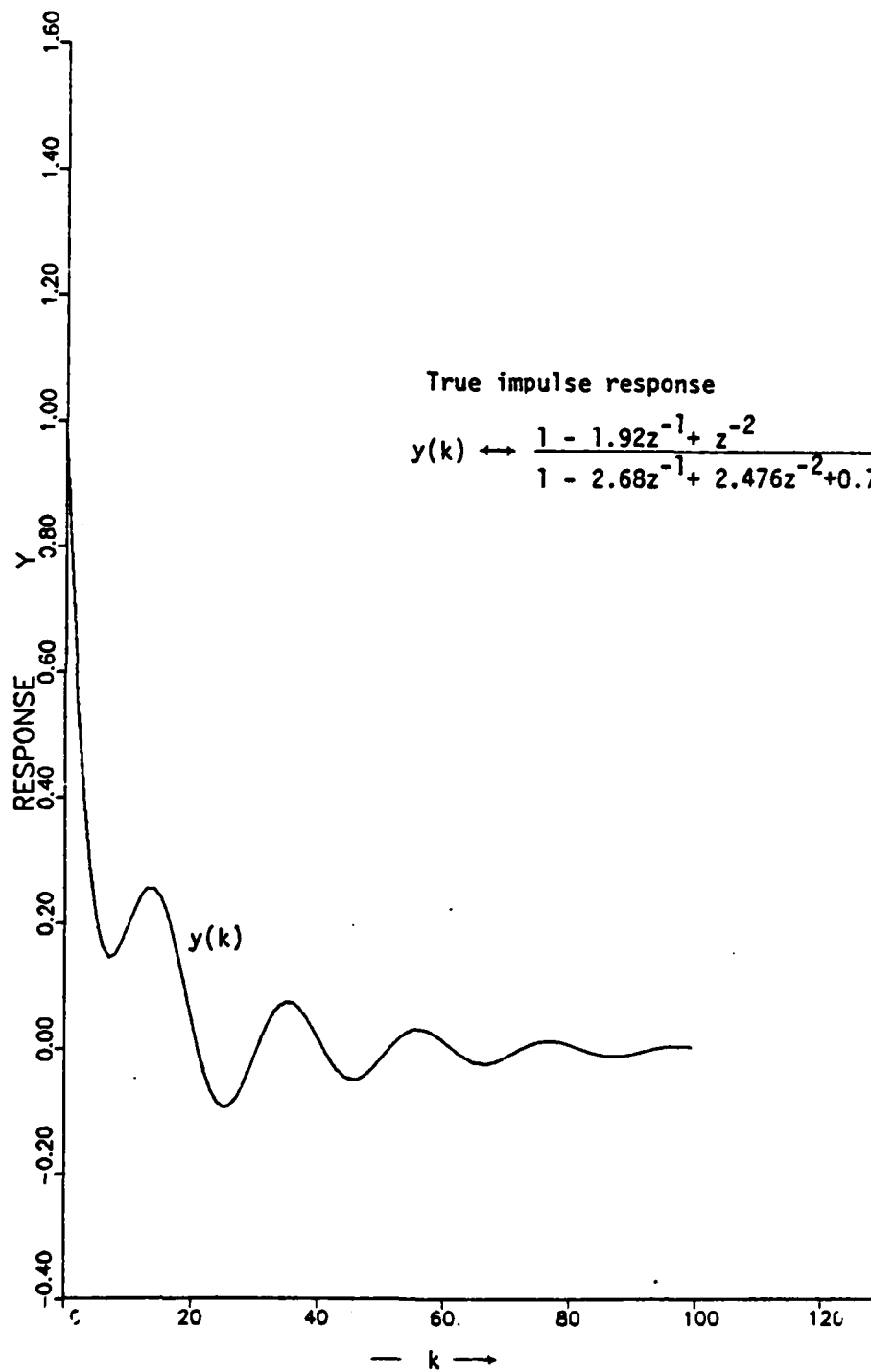


Fig. B1 True impulse response of a third order transfer function

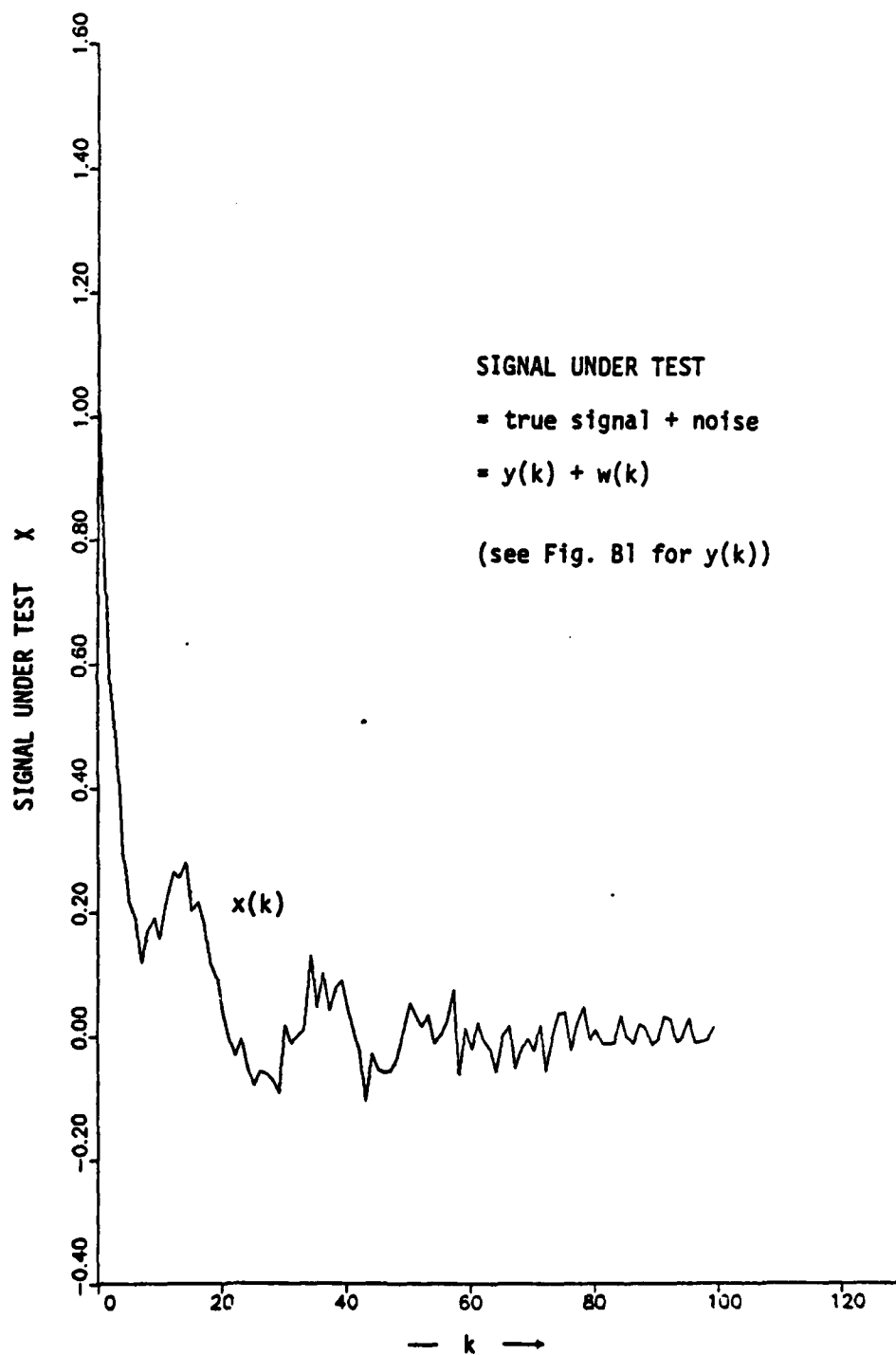


Fig. B2. A simulated noisy signal under test

PRINTER OUTPUT

EXAMPLE 1 LFC. PROMY VS PENCIL-OF-P.A.S.

IPP TRIN ISP: IPIX MPX -- PENCIL-OF-FUNCTIONS--

3-1 1-1-1 1-3 1 100 -0.0

VARIANCE OF NOISE = .1000E-02

FEAR = .5724E-01 LEAR = .5354E-02 FL2 = .4557E-01 DE = .3172E-01

SYNTHETIC VECTOR, AND SGP(Y11)

1.0000 .5252 .52575 .45636E-01 127.20

TPC: GRAM MATRIX (DET = 367.801)

1.82921 5.69161 10.1984 60.8396

5.69161 10.3332 64.3907 215.144

10.1984 64.3907 233.303 809.572

60.8396 215.144 809.572 2928.24

WAVEFORMS AND NUMER. SCALED BY XMAX = 1.00000

EST TF S(Z)/A(Z) (FMAX = .10000E+01)

1.00000 -2.63435 2.40737 -.757030

1.03949 -1.95233 1.00140 0.

SD ERROR = .470150E-01SS SIGNAL = 2.82033 RATIO = .166697E-01

ORIGINAL SIGNAL (INCLUDES BIAS, IF ANY)

1.0000 .7600 .5630 .4032 .2963 .2076 .1627 .1459 .1500 .1696
 .1950 .2221 .2437 .2559 .2583 .2437 .2169 .1633 .1400 .0926
 .0443 .0002 -.0379 -.0669 -.0556 -.0932 -.0901 -.0770 -.0502 -.0339
 -.0074 .0104 .0412 .0591 .0707 .0754 .0733 .0649 .0515 .0347
 .0161 -.0025 -.0193 -.0330 -.0426 -.0476 -.0479 -.0437 -.0359 -.0254
 -.0133 -.0009 .0106 .0204 .0276 .0319 .0325 .0307 .0261 .0194
 .0119 .0032 -.0045 -.0113 -.0172 -.0206 -.0219 -.0211 -.0135 -.0144
 -.0099 -.0037 .0018 .0067 .0106 .0133 .0146 .0145 .0130 .0105
 .0073 .0030 -.0001 -.0036 -.0064 -.0055 -.0096 -.0090 -.0091 -.0076
 -.0059 -.0031 -.0005 .0010 .0034 .0054 .0063 .0066 .0063 .0054
 IMPL. RESP OF MODEL (INC.B-BAT, IF BIAS.NE.0)

1.0394 .7571 .5715 .3979 .2603 .1816 .1335 .1179 .1265 .1505
 .1412 .2109 .2382 .2439 .2405 .2232 .1937 .1540 .1106 .0653
 .0220 -.0130 -.0401 -.0569 -.0533 -.0610 -.0409 -.0321 -.0125 .0075
 .0259 .0596 .0407 .0322 .0503 .0437 .0335 .0212 .0032 .0140
 -.0142 -.0217 -.0259 -.0269 -.0246 -.0200 -.0137 -.0069 .0006 .0069
 .0110 .0150 .0163 .0157 .0135 .0102 .0061 .0019 -.0021 .0034
 -.0079 -.0091 -.0092 -.0094 -.0067 -.0046 -.0021 .0003 .0024 .0040
 .0050 .0054 .0052 .0044 .0035 .0019 .0005 -.0000 -.0019 .0027
 -.0031 -.0031 -.0028 -.0022 -.0015 -.0007 .0001 .0000 .0014 .0017
 .0019 .0017 .0015 .0011 .0006 .0001 -.0003 -.0007 -.0009 .0010
 ERROR=F(K)-FALO(K)

-.0099 -.0271 -.0117 .0052 .0160 .0260 .0292 .0281 .0242 .0191
 .0144 .0113 .0105 .0121 .0139 .0205 .0251 .0245 .0234 .0273
 .0219 .0139 .0023 -.0100 -.0023 -.0032 -.0113 -.0457 -.0457 -.0414
 -.0329 -.0212 -.0075 .0069 .0204 .0317 .0393 .0407 .0433 .0306
 .0308 .0192 .0067 -.0061 -.0193 .0276 .0342 .0372 .0369 .0320
 -.0232 -.0160 -.0067 .0047 .0141 .0216 .0267 .0289 .0232 .0249
 .0198 .0122 .0044 -.0034 -.0104 .0160 .0199 .0214 .0209 .0164
 -.0143 -.0041 -.0034 .0023 .0073 .0114 .0141 .0158 .0150 .0132
 .0173 .0067 .0027 -.0010 -.0050 .0079 .0099 .0107 .0106 .0045
 -.0074 -.0049 -.0021 .0007 .0032 .0052 .0066 .0070 .0072 .0065

IPP IRL4 ISRA IFIX IFIX -- FUNCTIONS: ENHANCED --

2=1 1=1 1 1+1 1 100 +0.800000 +0.00011
 VARIANCE OF NOISE = .1000E-02
 F01F= .5724E-01 F02F= .335E-02 F03F= .4557E-01 SE= .5172E-01
 SYNTHETIC VECTOR, AND SQR(Y11)

1.0000 .39252 .32575 .4563E-01 127.2E

TRUE GRAM MATRIX (DET= 367.001)

1.42931	5.69161	18.1984	60.8396
5.69161	18.1984	64.3907	215.144
18.1984	64.3907	233.303	609.572
60.8396	215.144	609.572	2928.24
100.000	100.000	100.000	100.000
100.000	272.540	744.170	2029.04
100.000	744.170	3313.90	12533.0
100.000	2029.04	12533.0	55255.5

J,I GDET,SUMDET,SI: 1 0 .37E+03 .15E+04 .66E-03
 GDET,OFAC,TR: .37E+03 .11E-03 .40E-01
 J,I SI,COET 1 0 .37E+03 .66E-03 .42E+02
 J,I GDET,SUMDET,SI: 2 0 .42E+02 .10E+05 .97E-04
 J,I SI,COET 2 0 .42E+02 .97E-04 .84E+00
 J,I GDET,SUMDET,SI: 3 0 .42E+00 .49E+06 .20E-05
 J,I SI,COET 3 0 .42E+00 .20E-05 .36E-03

ESTIMATED NOISE VAR= .75673E-03

SYNTHETIC VECTOR, AND SQR(Y11)
 1.0000 .35133 .29774 .40624E-01 117.03

GLST MATRIX (DET= .364275E-03)

LOCATION/INTGR= 5

1.45354	5.61594	18.1227	60.7640
5.61594	18.1227	63.8275	213.609
18.1227	63.8275	233.756	600.000
60.7640	213.609	600.000	2883.40

WAVEFORMS AND NUMBER, SCALED BY XMAX= 1.00000

EST IF B(Z)/A(Z) (FMAX= .10000E+01)
 1.00000 -0.65558 2.49231 -.792556
 .956405 -1.88710 .961501 0.

SS ERROR= .475636E-02 SS SIGNAL= 2.82039 RATIO= .158642E-02

ORIGINAL SIGNAL (INCLUDES BIAS, IF ANY)

1.0000	.7500	.5600	.4032	.2863	.2070	.1627	.1459	.1506	.1656
.1956	.2221	.2437	.2559	.2563	.2437	.2109	.1833	.1400	.0926
.0448	.0000	-.0378	-.0669	-.0656	-.0932	-.1901	-.3778	-.0552	-.0339
-.0074	.0154	.0412	.0531	.0707	.0754	.0733	.0649	.0515	.0347
.0161	-.0025	-.0143	-.0330	-.0426	-.0476	-.0479	-.0437	-.0359	-.0294
-.0133	-.0309	.0106	.0204	.0276	.0310	.0326	.0307	.0261	.0194
.0115	.0032	-.0049	-.0113	-.0172	-.0206	-.0219	-.0211	-.0155	-.0144
-.0094	-.0037	.0018	.0067	.0104	.0133	.0146	.0145	.0130	.0109
.0070	.0035	-.0001	-.0030	-.0064	-.0055	-.0096	-.0099	-.0091	-.0076
-.0055	-.0031	-.0006	.0017	.0034	.0054	.0063	.0066	.0063	.0054

IMPL. NOISE OF 100.0 (I.O. = -AT, IF BIAS, NE, 0)

.4564	.7522	.5700	.4230	.2941	.2105	.1506	.1096	.1410	.1990
.1564	.2151	.2347	.2394	.2340	.2491	.2259	.1913	.1481	.1000
.0512	.0053	-.0383	-.0626	-.0613	-.0952	-.1940	-.3703	-.0452	-.0237
.0033	.0297	.0503	.0669	.0759	.0775	.0720	.0613	.0443	.0249
.0051	-.0137	-.0154	-.0421	-.0479	-.0444	-.0460	-.0354	-.0276	-.0143
-.0019	.0110	.0116	.0223	.0337	.0346	.0321	.0267	.0151	.0100
.0011	-.0075	-.0147	-.0200	-.0229	-.0235	-.0217	-.0191	-.0124	-.0066
-.0015	.0057	.0102	.0133	.0151	.0161	.0149	.0123	.0087	.0040

.0971	.1431	.1003	.2051	.2155	.2110	.1925	.1622	.1231	.0790
.0339	-.0009	-.0449	-.0728	-.0401	-.0968	-.0930	-.0802	-.0601	-.0354
-.0085	.0174	.0404	.0595	.0702	.0750	.0730	.0647	.0513	.0345
.0157	-.0026	-.0194	-.0331	-.0427	-.0476	-.0479	-.0436	-.0359	-.0254
-.0133	-.0010	.0106	.0204	.0276	.0317	.0324	.0307	.0261	.0194
.0115	.0031	-.0048	-.0117	-.0173	-.0206	-.0219	-.0211	-.0185	-.0144
-.0093	-.0037	.0018	.0067	.0106	.0133	.0146	.0145	.0130	.0105
.0073	.0038	-.0001	-.0038	-.0064	-.0095	-.0096	-.0098	-.0091	-.0076
-.0055	-.0031	-.0006	.0018	.0038	.0054	.0063	.0066	.0063	.0054

IPR IREM ISPR IFIX NFIX -- LPC: COV--

2-1 2-1 1+3 1 100 +1.0

VARIANCE OF NOISE= .1000E-01

FBAR= .5724E-01 BAR= .3358E-02 F12= .4557E-01 E1= .6172E-01

TRUE GRAM MATRIX (OIT= .-15503E-01)

1.03466	1.03584	1.24678	1.46306
1.03584	1.35865	1.51621	1.62233
1.24678	1.51621	1.92593	2.27764
1.46306	1.62233	2.27764	2.94736

WAVEFORMS AND NOISE SCALED BY XMAX=

1.00000

EST OF $\hat{S}(Z)/A(Z)$ (FMAX= .10000E+01)

1.00000	-.653233	-.270334	.116396
.430611	0.	0.	0.

SS ERROR= .441846 SS SIGNAL= 2.02039 RATIO= .156661

ORIGINAL SIGNAL (INCLUDES BIAS, IF ANY)

1.0000	.7600	.9600	.4832	.2863	.2076	.1627	.1459	.1506	.1696
.1956	.2221	.2437	.2559	.2563	.2437	.2168	.1833	.1400	.0926
.0446	.0002	-.0378	-.0669	-.0856	-.0932	-.0901	-.0778	-.0582	-.0339
-.0074	.0194	.0412	.0591	.0707	.0754	.0733	.0649	.0515	.0347
.0161	-.0025	-.0193	-.0330	-.0426	-.0476	-.0479	-.0437	-.0359	-.0254
-.0133	-.0007	.0106	.0204	.0276	.0317	.0324	.0307	.0261	.0194
.0115	.0031	-.0048	-.0117	-.0173	-.0206	-.0219	-.0211	-.0185	-.0144
-.0093	-.0037	.0018	.0067	.0106	.0133	.0146	.0145	.0130	.0105
.0073	.0038	-.0001	-.0038	-.0064	-.0095	-.0096	-.0098	-.0091	-.0076
-.0055	-.0031	-.0006	.0018	.0038	.0054	.0063	.0066	.0063	.0054

IMPL. PLSP OF MODEL (INC. B-MAT, IF BIAS.NE.0)

.9305	.6375	.6416	.4797	.4179	.3272	.2709	.2160	.1767	.1425
.1156	.0935	.0759	.0613	.0496	.0402	.0325	.0263	.0213	.0173
.0140	.0113	.0092	.0074	.0060	.0049	.0039	.0032	.0026	.0021
.0017	.0014	.0011	.0009	.0007	.0006	.0005	.0004	.0003	.0003
.0002	.0002	.0001	.0001	.0001	.0001	.0001	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000

END OF F(K) = F(K)

.0646	.1521	-.0078	-.0705	-.1316	-.1196	-.1032	-.0708	-.0261	.0270
.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000
.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000
.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000
.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000
.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000
.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000
.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000
.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000
.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000	.1000

-.0136	-.0022	-.0095	-.0106	-.0078	-.0030	.0021	.0063	.0091	.0100
-.0002	.0075	.0035	.0005	-.0027	-.0054	-.0072	-.0060	-.0081	-.0074
-.0064	-.0063	-.0045	-.0041	-.0042	-.0050	-.0061	-.0078	-.0090	-.0102
-.0105	-.0106	-.0097	-.0075	-.0052	-.0021	.0013	.0046	.0075	.0097
-.0110	-.0111	-.0101	-.0081	-.0053	-.0016	-.0016	-.0053	-.0083	-.0106
-.0115	-.0121	-.0113	-.0090	-.0061	-.0026	-.0007	.0041	.0070	.0092
-.0104	-.0106	-.0099	-.0072	-.0055	-.0024	-.0002	-.0031	-.0057	-.0076
-.0097	-.0099	-.0085	-.0071	-.0051	-.0023	-.0003	-.0022	-.0043	-.0060
-.0070	-.0073	-.0060	-.0055	-.0044	-.0025	-.0005	-.0015	-.0032	-.0046
-.0054	-.0057	-.0055	-.0046	-.0036	-.0022	-.0006	-.0005	-.0023	-.0034

$$2-1 \quad 4-3-1 \quad 1+3 \quad + \quad 100 \quad +1.0$$

F319= .572E-01 F328= .335E-02 F12= .4557E-01 F3= .6274E-01

2.2222	2.2776	2.2222	2.4630
2.2776	2.2776	2.2776	2.2776
2.2222	2.2776	2.2776	2.2776
2.4630	2.2222	2.2776	2.2776

EST TF S(Z)/A(Z) (FMAX= .10000E+01)

SS = 2.02639 SS SIGNAL = 2.02639 RATIO = .144437

[illegible]

0.034 0.014 0.012 0.007 0.014 0.014 0.075 0.044 0.002 0.001

.0015	.0032	-.0048	-.0111	-.0172	-.0006	-.0019	-.0211	-.0185	-.0144
-.0003	-.0037	.0011	.0067	.0106	.0133	.0146	.0145	.0130	.0105
.0073	.0036	-.0001	-.0036	-.0064	-.0095	-.0096	-.0091	-.0091	-.0076
-.0055	-.0031	-.0006	.0011	.0038	.0054	.0063	.0066	.0063	.0054

IPP FROM ISPA IFIX WFIX -- FROMY, AUTOCCRA--

2-1 1-3-1 1+1 1 100 -1.0

VARIANCE OF NOISE = .1000E-02

FB1F= .6724E-01 FB2F= .3358E-02 FL2= .4557E-01 IF= .6172E-01

TRUE GRAM MATRIX (DET= 4.14352)

1.42941	2.27761	1.82222	1.46306
2.27761	2.94741	2.27766	1.82233
1.82222	2.27766	2.94740	2.27764
1.46306	1.82233	2.27764	2.94736

WAVEFORMS AND NOISE, SCALED BY XMAX= 1.00000

BST IF 1(Z)/1(Z) (FMAX= .10000E+01)

1.00000	-.731971	-.465346E-01	-.738759E-02
.999999	.271631E-01	-.535506E-01	0.

SS ERROR= .403609 SS SIGNAL= 2.82039 RATIO= .143132

ORIGINAL SIGNAL (INCLUDES BIAS, IF ANY)

1.0000	.7500	.5536	.4032	.2663	.2076	.1627	.1459	.1506	.1696
.1956	.2201	.2437	.2559	.2563	.2437	.2185	.1833	.1430	.0926
.0441	.0002	-.0373	-.0669	-.0956	-.0932	-.0901	-.1770	-.0542	-.0339
-.0074	.0184	.0412	.0591	.0707	.0754	.0753	.0649	.0515	.0347
.0181	-.0025	-.0193	-.0330	-.0426	-.0476	-.0479	-.0437	-.0359	-.0254
-.0133	-.0009	.0105	.0204	.0276	.0318	.0320	.0307	.0261	.0194
.0115	.0032	-.0048	-.0111	-.0172	-.0206	-.0219	-.0211	-.0185	-.0144
-.0003	-.0037	.0011	.0067	.0106	.0133	.0146	.0145	.0130	.0105
.0073	.0036	-.0001	-.0036	-.0064	-.0095	-.0096	-.0091	-.0091	-.0076
-.0055	-.0031	-.0006	.0011	.0038	.0054	.0063	.0066	.0063	.0054

IMPL. RESP OF MODUL (INC. B-FAT, IF BIAS.NE.0)

.9999	.7500	.5536	.4032	.2663	.2076	.1627	.1459	.1506	.1696
.0947	.0757	.0637	.0467	.0391	.0313	.0251	.0202	.0162	.0130
.0104	.0034	.0067	.0054	.0043	.0035	.0023	.0022	.0016	.0014
.0011	.0004	.0007	.0006	.0006	.0004	.0003	.0002	.0002	.0002
.0001	.0001	.0001	.0001	.0001	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000

ERROR=F(K)-FAC(K)

.0002	.0011	.0174	-.0673	-.0677	-.0763	-.0651	-.0355	.0041	.0310
.0017	.1465	.1830	.2072	.2172	.2114	.1937	.1631	.1236	.0790
.0344	-.0031	-.0045	-.0713	-.0669	-.0956	-.0932	-.0901	-.0642	-.0339
.0185	.0175	.0005	.0545	.0702	.0751	.0750	.0647	.0513	.0345
.0159	-.0026	-.0194	-.0330	-.0426	-.0476	-.0479	-.0437	-.0359	-.0254
.0133	-.0009	.0105	.0204	.0276	.0318	.0320	.0307	.0261	.0194
.0115	.0032	-.0048	-.0111	-.0172	-.0206	-.0219	-.0211	-.0185	-.0144
.0003	-.0037	.0011	.0067	.0106	.0133	.0146	.0145	.0130	.0105
.0073	.0036	-.0001	-.0036	-.0064	-.0095	-.0096	-.0091	-.0091	-.0076
.0055	-.0031	-.0006	.0011	.0038	.0054	.0063	.0066	.0063	.0054

IPX IACH ISPX IFIX XFIX -- P-ONLY . COV--

2-1 1-2-1 1-0 1 100 -1.0

VARIANCE OF NOISE= .1000E-02

FBAF= .5724E-01 FBAF= .3358E-02 F12= .4557E-01 F11= .5172E-01

TRAIL GRAM MATRIX (DIT= .515509E-01)

1.03480 1.08894 1.24878 1.46306

1.08894 1.55965 1.51621 1.82238

1.24878 1.51621 1.92853 2.27764

1.46306 1.82238 2.27764 2.54736

WAVEFORMS AND NOISE. SCALED BY XMAX= 1.00000

EST TP S(Z)/A(Z) (FMAX= .10000E-01)

1.00000 -.553233 -.270334 .116398

1.00147 .580729E-01 -.213690 0.

SS ERROR= .594049 SS SIGNAL= 2.82039 RATIO= .139714

ORIGINAL SIGNAL (INCLUDES BIAS, IF ANY)

1.0000 .7500 .5600 .4032 .2965 .2076 .1627 .1459 .1506 .1696
.1356 .2221 .2437 .2559 .2563 .2437 .2189 .1838 .1400 .0926
.0441 .0002 -.0375 -.0589 -.0696 -.0932 -.0901 -.0778 -.0562 -.0339
-.0074 .0184 .0412 .0591 .0707 .0754 .0733 .0649 .0515 .0347
.0161 -.0025 -.0193 -.0330 -.0426 -.0476 -.0479 -.0437 -.0359 -.0294
-.0133 -.0009 .0196 .0204 .0276 .0318 .0329 .0307 .0261 .0194
.0119 .0082 -.0048 -.0111 -.0172 -.0206 -.0219 -.0211 -.0185 -.0144
-.0093 -.0037 .0018 .0067 .0106 .0133 .0146 .0145 .0130 .0105
.0073 .0038 -.0001 -.0036 -.0064 -.0085 -.0096 -.0097 -.0091 -.0076
-.0055 -.0031 -.0006 .0013 .0038 .0054 .0063 .0066 .0063 .0054
IMPL. PLSE OF MODEL (INC. B-MAT, IF BIAS. IS 0)

1.0015 .7523 .5483 .4449 .3513 .2959 .2300 .1866 .1506 .1222
.0989 .0301 .0648 .0525 .0425 .0344 .0278 .0225 .0152 .0148
.0120 .0097 .0079 .0063 .0051 .0042 .0034 .0027 .0022 .0018
.0014 .0012 .0009 .0006 .0006 .0005 .0004 .0003 .0003 .0002
.0001 .0001 .0001 .0001 .0001 .0001 .0000 .0000 .0000 .0000
.0000 .0000 .0000 .0000 .0000 .0000 .0000 .0000 .0000 .0000
.0000 .0000 .0000 .0000 .0000 .0000 .0000 .0000 .0000 .0000
.0000 .0000 .0000 .0000 .0000 .0000 .0000 .0000 .0000 .0000
.0000 .0000 .0000 .0000 .0000 .0000 .0000 .0000 .0000 .0000
ERROR=F(K)-F10(K)

-.0015 .0377 .0125 -.0413 -.0650 -.0784 -.0673 -.0407 -.0602 .0474
.0567 .1421 .1709 .2038 .2138 .2034 .1910 .1607 .1218 .0778
.0321 -.0094 -.0457 -.0733 -.0907 -.1073 -.0935 -.0800 -.0654 -.0357
.0059 .0172 .0403 .0588 .0701 .0749 .0723 .0646 .0513 .0344
.0159 -.0025 -.0194 -.0331 -.0427 -.0477 -.0479 -.0436 -.0359 -.0294
.0188 -.0010 .0196 .0204 .0276 .0317 .0329 .0307 .0261 .0194
.0119 .0082 -.0048 -.0111 -.0172 -.0206 -.0219 -.0211 -.0185 -.0144
.0093 -.0037 .0018 .0067 .0106 .0133 .0146 .0145 .0130 .0105
.0073 .0038 -.0001 -.0036 -.0064 -.0085 -.0096 -.0097 -.0091 -.0076
.0055 -.0031 -.0006 .0013 .0038 .0054 .0063 .0066 .0063 .0054

APPENDIX C

MODELING OF A SCATTERER RESPONSE

The signal of interest is the recorded response of a conducting pipe, considered in Example 2 of Section IV. On site digital sampling of the response was carried out as follows:

k=0 to k=300	Sampling interval	= 0.390625 ns
k=301 to 302	"	= 0.703125 ns
k=303 to end	"	= 1.953125 ns

For analysis purposes we resampled the first 300 data points by picking up every 5th point; the remaining data were used as such to gather a total of 245 data points. Note that we have ignored the intermediate sampling rate of the original data points 301-302.

The reconstituted signal was shown in Fig. 12 by the solid line.

Two runs will be presented below. First pertaining to the signal obtained above (from original data). The second pertains to differentiated (actually, differenced) signal produced from the resampled response (Fig. 13).

Given below are the (card deck) input to program POF-FILTER and, succeeding it, the printer output from the program; first for the signal itself and next for the differentiated signal.

INPUT CARDS (RESAMPLED SIGNAL)

```

RESPONSE OF A SCATTERER.
NPT      TRISP  IFLT  XMSB      DT      ANBIAS
      +-----+-----+-----+-----+-----+-----+
0245      8  0  0  1  245  +5.000000  +0.100000  +0.000000  +.44
0.000000 .007201 .027343 .055857 .084552 .113956 .139694 .156347 .157316 .170496
.165947 .153401 .136615 .117101 .093741 .070678 .044999 .031981 .010793 .009001
-.001955-.003731-.004137-.005593-.006346-.007251-.007915-.007902-.003343-.006849
-.003543-.007207-.009103-.009577-.004599-.001342-.003693 .013457 .001133 .003162
.001661 .002857 .001799 .000369 .007865 .001991 .003474 .000819 .001923 .001409
.006699 .000000 .001916 .001916 .007090 .003671 .000000 .004553 .002678 .001679
.001764-.003511-.007255-.002165-.004601-.007034-.005578-.007569-.009621-.002751
.002673-.007701-.007434-.007203-.001861-.005663-.006216-.000439-.000543-.000304
.003241-.001575-.001617-.000878 .003851 .000120 .001732 .004491 .000912 .000652
.000000 .000000 .000000 .000000 .000000 .000000 .000000 .000000 .000000 .000000

```

```

.009-10 .00777- .00102- .00241- .00957- .01377- .02187- .028212- .036960- .042557
-.049791- .054524- .057311- .062557- .065360- .067845- .069223- .070971- .071909- .068535
-.068960- .066355- .064118- .062122- .060119- .057015- .054901- .051847- .049753- .044466
-.041240- .037020- .031666- .027877- .025228- .019836- .017666- .015473- .013276- .013214
-.013135- .013041- .012980- .012835- .012683- .012474- .012376- .011795- .024810- .027766
-.032361- .036168- .040383- .042626- .046591- .050447- .052161- .054943- .055551- .057223
-.057820- .058403- .060047- .060619- .060104- .059591- .059073- .058552- .058030- .054284
-.051600- .051071- .044357- .046777- .043803- .040317- .037623- .036016- .033325- .030646
-.028594- .024235- .022652- .019916- .018409- .016834- .016336- .015835- .016406- .015906
-.015404- .014901- .014396- .016036- .017630- .018234- .019869- .023414- .019865- .021502
-.022044- .024741- .024202- .025928- .026371- .026913- .027457- .028004- .027483- .028053
-.028463- .028669- .023383- .022873- .020275- .019945- .011345- .019030- .018642- .016113
-.014632- .012208- .009755- .008395- .005960- .003800- .002316- .000029- .000189- .001405
.002596 .002689 .004911 .006042 .006066 .004995 .004932 .004952 .004906 .004945
.004769 .003601 .003434 .001216 .000017- .001147- .002432- .001509- .002231- .003232
IPR IPFM IPSN IFIX NFIX IBILS ISG MNPRT -PENCIL OF FUNCTIONS, ENHANCED-
100 0-1 1 1+1 0 000 +0.000000 +0.00011

```

PRINTER OUTPUT (CONDUCTING PIPE RESPONSE ANALYSIS)

RESPONSE OF A SCATTERER.

IPR IPFM IPSN IFIX NFIX IBILS ISG MNPRT -PENCIL OF FUNCTIONS, ENHANCED
 000 0-1 1 1+1 0 000 +0.000000 +0.00011

TRUE GRAM MATRIX (DLT= .351620E-51)

```

J,1 GDET,SCMDET,SI: 1 0 .35E-51 .29E+05 .34E-04
GDET,DFAC,TR: .35E-51 .11E-03 .34E-55
J,1 SI,CDLT 1 0 .35E-51 .34E-04 .22E-52
J,1 GDET,SCMDET,SI: 2 0 .22E-52 .40E+06 .25E-05
J,1 SI,CDLT 2 0 .22E-51 .25E-05 .12E-54
J,1 GDET,SCMDET,SI: 3 0 .12E-54 .77E+08 .13E-07
J,1 SI,CDLT 3 0 .12E-54 .13E-07 .39E-57

```

ESTIMATED NOISE VAP= .30469E-04

GLST MATRIX (DLT= .351620E-51)

LOT TF 3(Z)/A(Z) (FMAX= .17050E+00)

```

1.00000 -7.60367 25.5007 -49.2445 59.9213
-47.0545 23.2899 -6.64323 .639351
0. .614727E-01 -.252311 .323601 .519693E-01
-.544243 .679150 -.267869 .482362E-01
ESTIMATED MEAN= -.28425E-01

```

SC LRROR= .647E-01-03SS SIGNAL= .67805E-01RATIO= .12503E-01

S-FOLLS	SR	SI	SMAG	FR	IZTS= 0
1	-19.5472	66.5315	90.5073	14.4046	
2	-19.5472	-66.5315	90.5073	14.4046	
3	-2.6432	-12.6372	12.6906	2.0516	
4	-2.6432	12.6372	12.6906	2.0516	
5	-22.4736	-219.3543	219.3543	34.9113	
6	-22.4736	219.3543	219.3543	34.9113	
7	-4.2738	-67.6359	67.6211	10.7941	
8	-4.2738	67.6359	67.6211	10.7941	

S-COMPAI. CALCULATOR

```

.637360+01 .123800+02 .110880+03 .+63320+02 .102140+03 .234840+02
.242160+02 .178170+01 .100100+01
S-COMPAI. CALCULATOR
-.773600+02 -.194430+03 -.488470+03 -.+32210+03 -.297760+03 -.446610+02
-.103540+02 .473530+00 .676110+01

```

INPUT CARDS (DIFFERENTIATED SIGNAL)

DIFFERENTIATED RESPONSE OF A SCATTERER.

IPES	IFLT	XMSE	BIAS	ANBIAS
5	0+0 1 250	+5.000000	+1.953125	+0.000000 +4.
30000	.022010	.006480	.000000	.236950 .234040 .257380 .166530 .109680 .031830
40000	-.120170	-.173750	-.193240	-.234200 -.231030 -.216750 -.170310 -.211780 -.197910
20000	-.064250	-.115990	-.136140	-.079550 -.079680 -.066450 -.057430 -.053470 -.031000
24000	.067400	.121030	.132180	.162060 .202070 .177390 .175180 .172760 .170290
35100	.111770	.099330	.097790	.052940 .041210 .019300 .029450 .051040 -.004340
47910	-.058040	-.090030	-.099760	-.142200 -.140190 -.116910 -.125670 -.167750 -.143560
19760	-.114300	-.085490	-.085620	-.063590 -.051000 -.063420 -.041930 -.020520 -.031300
00700	-.010290	.032810	.032320	.053440 .055050 .063380 .073760 .062460 .072590
61430	.072180	.030860	.071290	.070740 .048870 .051580 .069650 .026210 .036700
36320	.014720	.007010	.007120	.006710 .023420 .039450 .039510 .050180 .017730
60740	-.024360	.060460	.038590	.070380 .038410 .061140 .070150 .050680 .053770
69840	-.047630	.025670	.057860	.025030 .024850 .013750 .013510 .013530 .030040
19450	.031740	.019680	.019560	.020030 .031040 .028340 .031340 .020640 .053150
31700	.042700	.043540	.042190	.021510 .053900 .021680 .021950 .021970 .000620
00790	.000940	.020390	.014550	.004260 .019110 .004910 .030300 .030050 .029650
51000	-.033940	.009690	.017580	.039260 .039960 .017140 .027820 .006080 .016720
05870	-.005000	.016470	.005880	.005110 .005130 .005180 .005210 .026720 .015960
06340	.005290	.028940	.016100	.037740 .026560 .026940 .016070 .026910 .026770
37540	.026590	.013530	.016860	.015770 .015750 .004980 .005010 .005730 .005020
05020	.005030	.015050	.016400	.016440 .005540 .016350 .005450 .005310 .016190
05420	-.026970	.005590	.016200	.005430 .005420 .005440 .005470 .005210 .005700
15800	.004940	.026710	.004650	.025580 .004300 .015000 .006650 .003680 .025290
14210	.004940	.024530	.013570	.024180 .023800 .012840 .023450 .001600 .012180
11910	.000980	.022800	.011250	.000240 .010710 .000130 .000300 .000460 .000610
00760	-.011600	.001070	.022760	.012010 .012140 .012350 .009230 .007220 .010010
1900	IPES IFLT XMSE	BIAS	ANBIAS	WITH BIAS= +1, ENHANCED
0 0-1	1 1+1 1 245	+0.700000	+0.00011	

AD-A092 226

ROCHESTER INST OF TECH NY DEPT OF ELECTRICAL ENGINEERING F/6 9/3
EXTENSION OF PENCIL-OF-FUNCTIONS METHOD TO REVERSE-TIME PROCESS-ETC(U)
AUG 80 V K JAIN, T K SARKAR, D D WEINER N00014-79-C-0598

UNCLASSIFIED

TR-80-4

NL

2 of 2
5/10/10



END

DATE

FILED

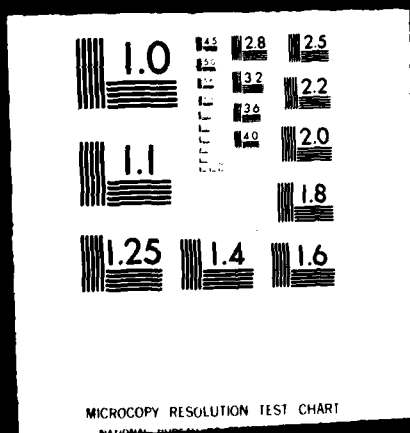
1-81

DTIC

OF 2

D A

092226



PRINTER OUTI

DIFFERENTIATED

IFR IFEM ISPA
 001 0-1 1 1+1
 TRUE 3
 J,I GDET,SUMOLT
 GDET,DFAC,TRR
 J,I SI,CDLT 1
 J,I CDLT,SUMOLT
 J,I SI,CDLT 2
 J,I GDET,SUMOLT
 J,I SI,CDLT 3
 J,I GDET,SUMOLT
 J,I SI,CDLT 4
 J,I GDET,SUMOLT
 ESTIMATED MOIS.
 GDET 14
 EST IF 3(Z)/A
 1.00000
 -22.4908
 1.0015
 -3.42449
 ESTIMATED MOIS.
 SS ERROR= .131

S-POLES SR
 1 -22.267
 2 -22.267
 3 -1.4521
 4 -1.4521
 5 -25.115
 6 -29.115
 7 -111.302
 8 -111.302
 S-DOMIN. BLNDMI
 .411710+02 .
 .413270+02 .
 S-DOMIN. ALBLR
 -.263910+02 -.
 -.251410+02 -.

PRINTER OUTPUT (DIFFERENTIATED SIGNAL)

DIFFERENTIATED RESPONSE OF A SCATTERER.

IPR IFEM ISOR IFIX .FIX IILS ISO MORT - WITH IELIS = +1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445, 446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589, 590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 600, 601, 602, 603, 604, 605, 606, 607, 608, 609, 610, 611, 612, 613, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623, 624, 625, 626, 627, 628, 629, 630, 631, 632, 633, 634, 635, 636, 637, 638, 639, 640, 641, 642, 643, 644, 645, 646, 647, 648, 649, 650, 651, 652, 653, 654, 655, 656, 657, 658, 659, 660, 661, 662, 663, 664, 665, 666, 667, 668, 669, 670, 671, 672, 673, 674, 675, 676, 677, 678, 679, 680, 681, 682, 683, 684, 685, 686, 687, 688, 689, 690, 691, 692, 693, 694, 695, 696, 697, 698, 699, 700, 701, 702, 703, 704, 705, 706, 707, 708, 709, 710, 711, 712, 713, 714, 715, 716, 717, 718, 719, 720, 721, 722, 723, 724, 725, 726, 727, 728, 729, 730, 731, 732, 733, 734, 735, 736, 737, 738, 739, 740, 741, 742, 743, 744, 745, 746, 747, 748, 749, 750, 751, 752, 753, 754, 755, 756, 757, 758, 759, 760, 761, 762, 763, 764, 765, 766, 767, 768, 769, 770, 771, 772, 773, 774, 775, 776, 777, 778, 779, 780, 781, 782, 783, 784, 785, 786, 787, 788, 789, 790, 791, 792, 793, 794, 795, 796, 797, 798, 799, 800, 801, 802, 803, 804, 805, 806, 807, 808, 809, 810, 811, 812, 813, 814, 815, 816, 817, 818, 819, 820, 821, 822, 823, 824, 825, 826, 827, 828, 829, 830, 831, 832, 833, 834, 835, 836,

001 0-1 1 1+1 1 245 +0.70000 +0.00011

TRU: JRM: 42TRIX (OUT= .249313-42)

J, I GET, SUMO_T, SI : 1 0 .25E-42 .21E+04 .43E-03

GDET,DFAC,TF-4: .272-42 .112-03 .272-46

J.I. SINGH, CO-1 T 1 C .254-42 .472-63 .702-43

J, I CD = T, S, MD = T, SI : 2 0 .78 = -43 .23 = -34 .35 = -03

J, I SI, COL T 2 3 .74-43 .357-03 .243-43

J, I GDE T. S. MD = T, Si : 3 C .24E-43 .41E+04 .25E-03

J. I. S. I. C. O. L. T. 3 3 .242-43 .252-03 .72E-44

J, I 6JLT, SUMOCT, SI : 4 0 .72E-44 .50E+04 .27E-03

J, I SI, CD-T 4 3 .725-44 .172-03 .225-44

J, I GOVT, SUPDCT, SI: 5 0 .22L-44 .56L+04 .12L-03

ESTIMATED NOISE VAR= .124011-02

GAST HATFIELD (D-T-063-2032-45)

EST TF 3(Z)/4(Z) (FMAX= .330000-00)

1.00000 -6.14306 16.9495

-27.0042

30.1500

-22.4709 11.3529 -3.53420

• 512711

1.03151 -5.29485 12.0095

-16.3523

24.6775

-1.92049 3.56436 -0.003455

6-11635-01

SECRETED NAME .140352-01

SS ERROR= .136053E-01 SS SIGNAL= .352405 RATIO= .365044E-01

S-POL=5 SR SI SMAG FR IZTS= 0

1	-22.2679	-73.2127	63.2615	10.0654
---	----------	----------	---------	---------

2 -22.2679 54.2129 63.2616 10.0664

5 -71.4527 -71.5455 72.2170 11.4937

4	-1,4527	71,5456	72,2173	11,4937
---	---------	---------	---------	---------

5 -23.435 -222.3415 223.553 35.6276

222.3405	223.6550	35.6476
----------	----------	---------

7	-111.3076	-617.0446	627.4039	66.6554
---	-----------	-----------	----------	---------

2230.7015	627.0015	627.0015	627.0015
2230.7015	627.0015	627.0015	627.0015

3-00411. D. ADMIRALTY

.421710+02 .675130+02 .217730+03 .171760+03 .264340+03 .638320+02

• 3270+02 • 3420+01 • 100000+01

S-DOMAIN ANALYSIS

- .2603910+02 - .363640+03 - .512710+03 - .134520+04 - .357940+03 - .223590+03

- .253610+12 - .431550+01 - .125050+00