AD-A051 060 UNIVERSITY OF SOUTHERN CALIFORNIA LOS ANGELES DEPT O--ETC F/G 9/2
COMPUTATIONAL SOLUTIONS TO A NON-UNIFORM TIME-DELAY LINEAR SYST--ETC(U)
1977 W M CHAN, N E NAHI, J M MENDEL F44620-71-C-0067
UNCLASSIFIED AFOSR-TR-78-0201 NL

| OF |
AD
A051060

END
DATE FILMED
4 -78
DDC

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

AD A051060

DDC FILE COPY

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AFOSR-TR-78-0201
2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle): COMPUTATIONAL SOLUTIONS TO A NON-UNIFORM TIME-DELAY LINEAR SYSTEM.

5. TYPE OF REPORT & PERIOD COVERED: Interim rept.
6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s): W. M./Chan, N. E./Nahi [redacted] J. M./Mendel

8. CONTRACT OR GRANT NUMBER(s): F44620-71-C-0067, AFOSR-75-2797

9. PERFORMING ORGANIZATION NAME AND ADDRESS
University of Southern California
Department of Electrical Engineering
Los Angeles, CA 90007

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
61102F 2304/A1

11. CONTROLLING OFFICE NAME AND ADDRESS
Air Force Office of Scientific Research/NM
Bolling AFB, DC 20332

12. REPORT DATE: 1977
13. NUMBER OF PAGES: 9 (11p.)

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)

15. SECURITY CLASS. (of this report): UNCLASSIFIED
15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

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)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Two computational methods are considered for a class of non-uniform, continuous, multiple time-delay linear systems, which have been developed in the modeling of lossless layered media. In the first approach, we discretize the time axis and insert states of intermediate delays, to arrive at a set of standard finite-difference equations. For our particular system, matrix multiplications can be reduced to simple scalar multiplications. In the second approach, we define mapping rules for the transformation of states at an interfaces, and keep a state reference table for look-up and branching. The procedure

20. Abstract

is similar to ray-tracing. Several experimental results are presented to show the trade-off between storage requirement and CPU time-spent for the two methods.

Presented at Symposium on Applications of Computer Methods in Engineering, University of Southern California, Los Angeles, California 90007, August 23-26, 1977.

COMPUTATIONAL SOLUTIONS TO A NON-UNIFORM TIME-DELAY LINEAR SYSTEM

W. M. Chan,[I] N. E. Nahi[I] and J. M. Mendel[I]

Summary

Two computational methods are considered for a class of non-uniform, continuous, multiple time-delay linear systems, which have been developed in the modeling of lossless layered media. In the first approach, we discretize the time axis and insert states of intermediate delays, to arrive at a set of standard finite-difference equations. For our particular system, matrix multiplications can be reduced to simple scalar multiplications. In the second approach, we define mapping rules for the transformation of states at an interfaces, and keep a state reference table for look-up and branching. The procedure is similar to ray-tracing. Several experimental results are presented to show the trade-off between storage requirement and CPU time-spent for the two methods.

1. Introduction

In the development of time-domain state space models for lossless layered media, Mendel et. al. [1] have arrived at a class of non-uniform, multiple, continuous, time-delay linear equations. These equations are termed Causal Functional Equations. In this paper we will discuss two computational methods to obtain a solution for this class of equations.

The immediate application of this solution is to generate synthetic seismograms for geophysical study (e.g., in testing deconvolution algorithms and inversion techniques). As mentioned in the paper by Mendel et. al. [1], this class of equations can also be used to model lossless transmission lines and certain acoustic, optical and electromagnetic systems.

In the next section we will give a formulation of this class of equations and discuss some of its characteristics. In Section 3 we discuss the method of discretization of the time-axis and the proce-

---

I. Department of Electrical Engineering-Systems, University of Southern California, Los Angeles, California 90007.

dure to obtain a standard unit-delay finite-difference equations. A ray tracing technique using mapping rules for states is discussed in Section 4. In Section 5 we make a comparison between the two methods in terms of simplicity, storage requirement and CPU time-spent. Some experimental results are presented. We conclude with a summary of results and a sample plot of unit pulse response for layered earth media.

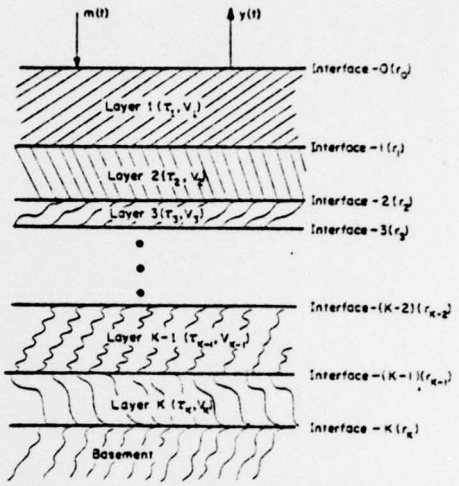


Figure 1. System of K layered media.

2. Causal Functional Equations

A system of K layered media is depicted in Figure 1. For each layer, we define two states: $u_j(t)$ and $d_j(t)$, which are the upgoing and downgoing states at the (j-1)th and jth interfaces respectively. The reflection and transmission characteristics at an interface are modeled by the reflection coefficients, r_j. Within the layer, the state only suffer a time delay, τ_j. A more detailed description can be found in Mendel et. al. [1].

The Causal Functional Equations are:

$$\underline{d}(t+\underline{\tau}) = A_1\,\underline{d}(t) + A_2\,\underline{u}(t) + \underline{g}\,m(t) \qquad (1)$$

$$\underline{u}(t+\underline{\tau}) = A_3\,\underline{d}(t) + A_4\,\underline{u}(t) \qquad (2)$$

$$\underline{y}(t) = \underline{h}'\,\underline{u}(t) + r_0\,m(t) \qquad (3)$$

where

$$\left.\begin{aligned} \underline{d}(t+\underline{\tau}) &\triangleq \mathrm{col}(d_1(t+\tau_1), d_2(t+\tau_2), \ldots, d_K(t+\tau_K)) \\ \underline{u}(t+\underline{\tau}) &\triangleq \mathrm{col}(u_1(t+\tau_1), u_2(t+\tau_2), \ldots, u_K(t+\tau_K)) \\ \underline{d}(t) &\triangleq \mathrm{col}(d_1(t), d_2(t), \ldots, d_K(t)) \\ \underline{u}(t) &\triangleq \mathrm{col}(u_1(t), u_2(t), \ldots, u_K(t)) \end{aligned}\right\} \qquad (4)$$

and

$$A_1 = \text{sub-diag}\,(1+r_1, 1+r_2, \ldots, 1+r_{K-1}) \qquad (5)$$

$$A_2 = -\text{diag}\,(r_0, r_1, \ldots, r_{K-1}) \qquad (6)$$

$$A_3 = \text{diag}\,(r_1, r_2, \ldots, r_K) \qquad (7)$$

$$A_4 = \text{super-diag}\,(1 - r_1, 1 - r_2, \ldots, 1 - r_{K-1}) \qquad (8)$$

$$\underline{g} = \text{col}\,(1 + r_0, 0, 0, \ldots, 0) \qquad (9)$$

$$\underline{h} = \text{col}\,(1 - r_0, 0, 0, \ldots, 0) \qquad (10)$$

The notation is:

$$\text{sub-diag} = \begin{pmatrix} 0 & & \cdots & & 0 \\ & \ddots & & & \\ & & \ddots & & \\ 0 & & & & 0 \end{pmatrix}, \quad \text{super-diag} = \begin{pmatrix} 0 & & & & 0 \\ & \ddots & & & \\ & & \ddots & & \\ 0 & & \cdots & & 0 \end{pmatrix}, \quad \text{diag} = \begin{pmatrix} \diagdown & & & & 0 \\ & \ddots & & & \\ & & \ddots & & \\ 0 & & & & \diagdown \end{pmatrix}$$

with the non-zero entries given in the above equations. Matrices A_1, A_2, A_3, A_4 are $K \times K$, $\underline{g}, \underline{h}$ are $K \times 1$. Physically, Eqs. (1) - (3) describe the interactions and transformations of the downgoing and upgoing states in a layered media. These equations are continuous-time equations with non-uniform, multiple time-delays. They are linear in the states; hence, we may decompose them in suitable form and apply the principle of superposition to obtain a solution. The output response, y(t), can be obtained by the convolution of the input wave form, m(t), and the impulse response of the system, which is itself a sequence of non-uniformly spaced impulses along the time-axis. In general, those impulses will become more densely populated on the time-axis as time increases.

3. Finite-Difference Technique

The system of equations in Section 2 can be converted into standard finite-difference equations (FDE) by discretization of time. Since the delay terms are non-uniform and not multiple of one another, we may have to choose a very small time interval, Δ, for discretization. Below, we show the procedure to arrive at the desired FDE:

First, let us find Δ, the greatest common factor of the delay terms, such that $\tau_i = k_i \Delta$, $i = 1, 2, \ldots, K$. The k_i are positive integers. Let $t = k\Delta$ and define $x(k) = x(k\Delta)$. We make the following transformations in Equ. (1) and (2):

$$\underline{d}(t + \underline{\tau}) = \underline{d}(k + \underline{k}_i)$$

$$\underline{u}(t + \underline{\tau}) = \underline{u}(k + \underline{k}_i)$$

Next, for each state with $k_i > 1$, we insert states of intermediate

delays to obtain a unit-delay system. Specifically, we make the following changes:

$$d_i(k+k_i) \rightarrow \text{col}\,(d_i(k+k_i), d_i(k+k_i-1), \ldots, d_i(k+1)),\quad i = 1, 2, \ldots, K.$$

$$u_i(k+k_i) \rightarrow \text{col}\,(u_i(k+1), u_i(k+2), \ldots, u_i(k+k_i)),\quad i = 1, 2, \ldots, K.$$

The above ordering of $d_i(k)$ and $u_i(k)$ preserves the structure of the matrices A_1, A_2, A_3 and A_4. After staking up the states into vectors $\underline{d}(k+1)$ and $\underline{u}(k+1)$, we arrive at the following equations:

$$\underline{d}(k+1) = A_1^1\,\underline{d}(k) + A_2^1\,\underline{u}(k) + \underline{g}^1\,m(k) \qquad (11)$$

$$\underline{u}(k+1) = A_3^1\,\underline{d}(k) + A_4^1\,\underline{u}(k) \qquad (12)$$

$$\underline{y}(k) = \underline{h}^{1'}\,\underline{u}(k) + r_0\,m(k) \qquad (13)$$

where

$$\underline{d}(k+1) \triangleq \text{col}\,(d_1(k+k_1), \ldots, d_1(k+1) \,|\, d_2(k+k_2), \ldots, d_2(k+1) \,|\, \ldots \,|\, d_K(k+k_K), \ldots, d_K(k+1)) \qquad (14)$$

$$\underline{u}(k+1) \triangleq \text{col}\,(u_1(k+1), \ldots, u_1(k+k_1) \,|\, u_2(k+1), \ldots, u_2(k+k_2) \,|\, \ldots \,|\, u_K(k+1), \ldots, u_K(k+k_K)) \qquad (15)$$

and

$$A_1^1 = \text{sub-diag}\,(1, 1, \ldots, 1 \,|\, (1+r_1), 1, \ldots, 1 \,|\, \ldots \,|\, (1+r_{K-1}), 1, \ldots, 1) \qquad (16)$$

$$A_2^1 = -\text{diag}\,(r_0, 0, \ldots, 0 \,|\, r_1, 0, \ldots, 0 \,|\, \ldots \,|\, r_{K-1}, 0, \ldots, 0) \qquad (17)$$

$$A_3^1 = \text{diag}\,(0, \ldots, 0, r_1 \,|\, 0, \ldots, 0, r_2 \,|\, \ldots \,|\, 0, \ldots, 0, r_K) \qquad (18)$$

$$A_4^1 = \text{super-diag}\,(1, \ldots, 1, (1-r_1) \,|\, \ldots \,|\, 1, \ldots, 1, (1-r_{K-1}) \,|\, 1, \ldots, 1) \qquad (19)$$

$$\underline{g}^1 = \text{col}\,(1+r_0, 0, \ldots, 0 \,|\, 0, \ldots, 0 \,|\, \ldots \,|\, 0, \ldots, 0) \qquad (20)$$

$$\underline{h}^1 = \text{col}\,(1-r_0, 0, \ldots, 0 \,|\, 0, \ldots, 0 \,|\, \ldots \,|\, 0, \ldots, 0) \qquad (21)$$

Let $L = \sum_{i=1}^{K} k_i$, then, $\underline{d}(k), \underline{u}(k), \underline{g}^1, \underline{h}^1$ are $L \times 1$, $A_1^1, A_2^1, A_3^1, A_4^1$ are $L \times L$.

To obtain the system's unit pulse response we set $m(0) = 1$, and $m(k) = 0$, $k \geq 1$. Our computational algorithm for observation time up to TMAX is:

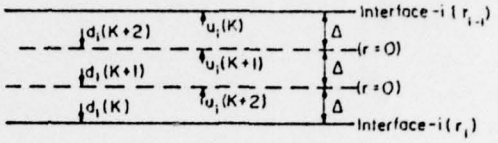


Figure 2. Illustration to show insertion of new state variable for the ith layer where $k_i = 3\Delta$.

(1) Initialization: $k = 0$

$$\left.\begin{aligned} \underline{d}(1) &= \underline{g}^1 \\ \underline{u}(1) &= \underline{0} \\ y(0) &= r_0 \end{aligned}\right\} \quad (22)$$

(2) Recursive Step: $k = 1, 2, \ldots, \frac{TMAX}{\Delta}$

$$\left.\begin{aligned} \underline{d}(k+1) &= A_1^1\, \underline{d}(k) + A_2^1\, \underline{u}(k) \\ \underline{u}(k+1) &= A_3^1\, \underline{d}(k) + A_4^1\, \underline{u}(k) \\ y(k) &= \underline{h}^{1\prime}\, \underline{u}(k) \\ &\text{Repeat.} \end{aligned}\right\} \quad (23)$$

Since each row of matrices A_1, A_2, A_3 and A_4 has only one non-zero element, the multiplication indicated in Eq. (23) is just a collection of scalar multiplications. The non-zero entries of the four A_i^1 matrices are related and can be obtained from a single array, A, with elements $(r_0 : A_3^1)$. New values of $\underline{u}(k)$ and $\underline{d}(k)$ can be stored back into the state vectors; thus, we only need three arrays for A, $\underline{u}(k)$, and $\underline{d}(k)$.

For our particular system, the insertion of states corresponds to the dividing of a layer into equal sub-layers by inserting interfaces whose reflection coefficients are zero. This phenomenon can be observed from the A^1 matrices in Eqs. (16)-(19). The particular ordering of the states $\underline{u}(k)$ and $\underline{d}(k)$ can be understood from the way the states are defined and is illustrated for the case $k_i = 3\Delta$ in Figure 2.

4. Ray Tracing Technique

In this section, we utilize the properties of linearity and the causal nature of our equations to develop a ray tracing technique. Instead of focusing on the interaction of states, we decompose Eqs. (1)-(3) and define the following mapping rules to track how a state propagates at an interface:

$$
d_j(t) \begin{cases} r_0\, m(t) & \rightarrow \quad y(t) & j = 0 \\ r_j\, d_j(t) & \rightarrow \quad u_j(t+\tau_j) & j = 1,\ldots,K \\ (1+r_j)\, d_j(t) & \rightarrow \quad d_{j+1}(t+\tau_{j+1}) & j = 0,\ldots,K-1 \end{cases} \qquad (24)
$$

$$
u_j(t) \begin{cases} (1-r_0)\, u_1(t) & \rightarrow \quad y(t) & j = 1 \\ (1-r_{j-1})\, u_j(t) & \rightarrow \quad u_{j-1}(t+\tau_{j-1}) & j = 2,\ldots,K \\ -r_{j-1}\, u_j(t) & \rightarrow \quad d_j(t+\tau_j) & j = 1,\ldots,K \end{cases} \qquad (25)
$$

We illustrate the interpretation of these mapping rules for $d_j(t)$, in Eq. (24) in Figure 3.

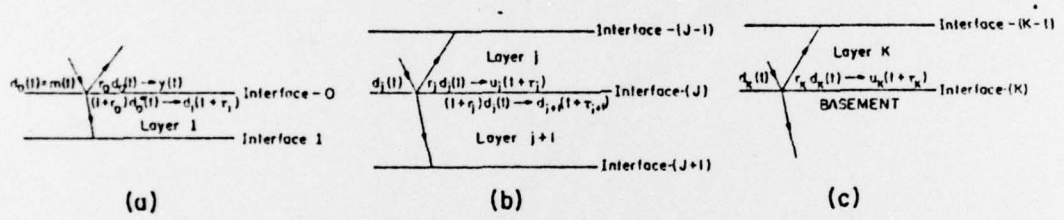


Figure 3. The transformation of $d_j(t)$ into neighboring states at (a) the surface, (b) jth interface, and (c) bottom interface.

In order to use Eqs. (24) and (25), we need to have a state reference table. Each element of the states in the table, called an event, is characterized by its time and amplitude. A subroutine, ISMAL, picks up the event of the next smallest time and processes it according to the mapping rules, which are coded into the UP and DOWN subroutines. The new events generated are stored back into the table. As each event branches into two new events, the table grows geometrically. To restrict the growth of the table, we use two tolerance parameters, AMIN and δT, and collapse events that occur at the same time point before we store them back into the table. AMIN controls the amplitude of the computed event. If that amplitude is less than AMIN, it is set to zero. δT controls the time separation of two events below which they are considered to occur at the same time. The output is observed up to a pre-determined time, TMAX.

5. Comparison of the Two Methods

If we expand the matrix multiplications in the Section 3 method, we obtain the Section 4 mapping rules, from step k to k+1; hence, the two methods are related. In the discrete method we keep the history in the state vectors, and in the ray-tracing technique we use a state reference table.

The discrete method is simple, easy to implement and recursive; however, due to the non-uniform nature of the delay-terms, we may have to use a Δ small enough such that it is a submultiple of all the delay terms. The storage requirement is usually not excessive. However, we may be computing zero for a lot of points on the time axis where no events occur. Thus the CPU time may be larger than for the ray-tracing technique.

The ray-tracing technique requires a slightly more complex program to implement it. Its major disadvantage is the huge storage requirement for the state reference table. Since this technique computes only the actual event points, it can be very efficient and fast for short observation time.

We present below three experimental results in testing the two methods, using a PDP-10 KI interactive system:

Experiment 1: 3 layers, TMAX = 5.00 minutes, $\Delta = 0.001$ minute, AMIN = 10^{-20}, $\delta T = 10^{-5}$ minutes.

τ = (0.3, 0.001, 0.5) minutes

r = (0.8, -.3, .3, .5)

This experiment illustrates the potential problem of computing artificial zero points for FDE. The first non-zero data occurs at k = 600!

Experiment 2: 3 layers, TMAX = 10 minutes, $\Delta = 0.01$ minute, AMIN = 10^{-7}, $\delta T = 10^{-5}$ minutes.

τ = (0.03, 0.01, 0.05) minutes

r = (0.8, -.3, .3, .5)

This experiment illustrates storage problem for ray tracing technique.

Experiment 3: 19 layers, TMAX = 2.20 minutes, $\Delta = 0.001$, AMIN = 10^{-5}, $\delta T = 10^{-5}$ minutes.

τ = (.016, .050, .004, .023, .022, .015, .042, .028, .006, .038, .003, .006, .007, .072, .005, .030, .027, .038, .014) minutes

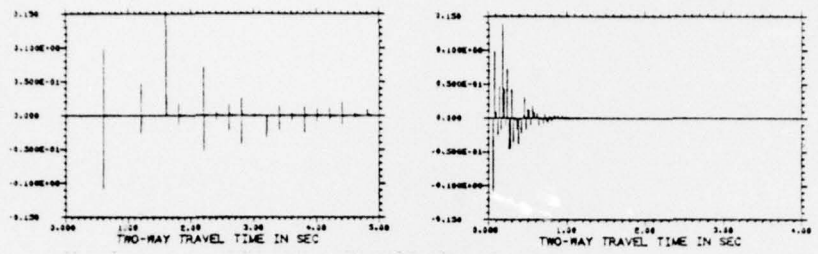
r = (.99, .028, -.061, .082, .034, -.068, -.016, .168, -.008, .108, .058, .114, -.057, .026, -.112, -.220, .076, .156, .039, -.229)

This experiment illustrates the problems of CPU and storage for a large number of layers case.

Experiments	FDE	Ray Tracing
(1) CPU	4 min. 46.05 sec.	4.70 sec.
Storage	7400	8726
Number of Output	5000	436
(2) CPU	2.35 sec.	2.85 sec.
Storage	1027	3434
Number of Output	1000	163
(3) CPU	1 min. 20.80 sec.	1 min. 4.56 sec.
Storage	3538	73902
Number of Output	2200	976

6. Conclusions

In this paper we have presented two computational methods for a class of causal functional equations. They are the discrete method of finite-difference equations and the continuous method of ray-tracing. The FDE has low storage requirement but may run into high CPU time-spent if the sampling interval, Δ, is too small. The ray tracing technique has a large storage requirement but can be very efficient for short observation time and small number of layers cases. The actual choice of a particular method will depend on the input data set. A sample plot of the unit impulse response for Experiments 1 and 2 is illustrated in Figure 4.



(1) (2)

Figure 4. Unit Impulse Response for Experiments 1 & 2. Observe the rapid decay in amplitudes of the impulses.

Acknowledgement

The work reported in this paper was performed at the University of Southern California, Los Angeles, California, under National Science Foundation Grants NSF ENG 74-02297A01 and NSF-ENG 75-03423, Air Force Office of Scientific Research Grant AFOSR 75-2797, Chevron Oil Field Research Co. Contract-76, Teledyne Exploration Co. Contract TEC-76 and JSEP through AFOSR/AFSC under Contract F44620-71-C-0067. The first author wishes to thank the Rockwell International Fellowship for his support, and Mr. M. Steinberger, for stimulating discussion, Mr. A. Yu, for the computer programming, and Mr. J. Kormylo, for the plotting subroutines. All of them are graduate students at the University of Southern California.

References

[1] J. M. Mendel, N. E. Nahi, L. M. Silverman and H. D. Washburn, "State Space Models of Lossless Layered Media," presented at Joint Automatic Control Conference, San Francisco, California, June 1977.