

AD-A108 500

NEBRASKA UNIV LINCOLN DEPT OF MATHEMATICS

F/G 20/14

TRANSIENT FIELDS PRODUCED IN HETEROGENEOUS BODIES BY ELECTROMAG--ETC(U)

OCT 81 R J KRUEGER

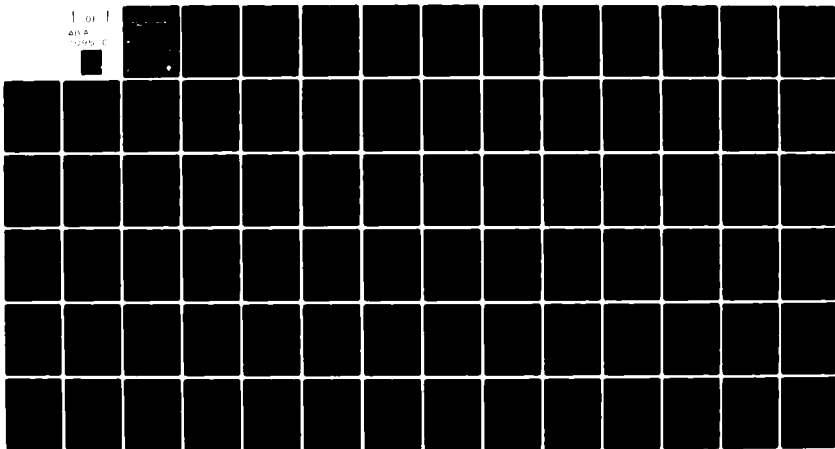
F33615-78-D-0629

NL

UNCLASSIFIED

SAM-TR-80-12

1 OF 1
AD-A
108500



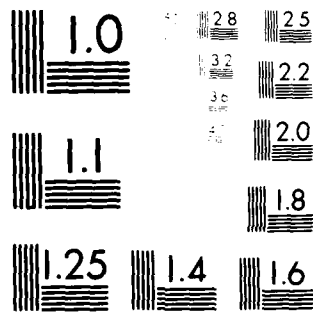
END

DATE

FILED

01-82

DTIC



MICROCOPY RESOLUTION TEST CHART
NAT. BUREAU OF STANDARDS-1963-A

(12)

LEVEL II

40

Report SAM-TR-81-12

AD A108500

TRANSIENT FIELDS PRODUCED IN HETEROGENEOUS BODIES BY ELECTROMAGNETIC PULSES

Robert J. Krueger, Ph.D.
Department of Mathematics
University of Nebraska
Lincoln, Nebraska 68588

DTIC
ELECTE
DEC 10 1981
S B

October 1981

Final Report for Period 8 May 1980 - 30 September 1980

Approved for public release; distribution unlimited.

Prepared for
USAF SCHOOL OF AEROSPACE MEDICINE
Aerospace Medical Division (AFSC)
Brooks Air Force Base, Texas 78235



JEP

81 12 10 015

DTIC FILE COPY

NOTICES

This final report was submitted by the Department of Mathematics, University of Nebraska, Lincoln, Nebraska, under contract F33615-78-D-0629, Task 0022, job order 7757-01-83, with the USAF School of Aerospace Medicine, Aerospace Medical Division, AFSC, Brooks Air Force Base, Texas. Earl L. Bell (USAFSAM/BRM) was the Laboratory Project Scientist-in-Charge.

When U.S. Government drawings, specifications, or other data are used for any purpose other than a definitely related Government procurement operation, the Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise, as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Office of Public Affairs (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

Earl L. Bell
EARL L. BELL, M.S.
Project Scientist

Richard A. Albanese MD
RICHARD A. ALBANESE, M.D.
Supervisor

Roy L. DeHart
ROY L. DEHART
Colonel, USAF, MC
Commander

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER SAM-TR-81-12	2. GOVT ACCESSION NO. AD-710850	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) TRANSIENT FIELDS PRODUCED IN HETEROGENEOUS BODIES BY ELECTROMAGNETIC PULSES		5. TYPE OF REPORT & PERIOD COVERED Final Report 8 May 1980 - 30 Sep 1980
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Robert J. Krueger, Ph.D.		8. CONTRACT OR GRANT NUMBER(s) F33615-78-D-0629 0022
9. PERFORMING ORGANIZATION NAME AND ADDRESS Department of Mathematics University of Nebraska Lincoln, Nebraska 68588		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 62202F 7757-01-83
11. CONTROLLING OFFICE NAME AND ADDRESS USAF School of Aerospace Medicine (RZP) Aerospace Medical Division (AFSC) Brooks Air Force Base, Texas 78235		12. REPORT DATE October 1981
		13. NUMBER OF PAGES 80
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) USAF School of Aerospace Medicine (BRM) Aerospace Medical Division (AFSC) Brooks Air Force Base, Texas 78235		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) Electromagnetic field Conductivity Electromagnetic pulses Scattered waves Internal field Scattering operators Transient Computer program Permittivity		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) An exact analytical technique for determining induced electromagnetic fields in a one-dimensional medium is developed. The incident field, which impinges normally on the medium, is an electromagnetic pulse of arbitrary shape and duration. The medium is of finite depth and can have arbitrarily specified and nonconstant permittivity and conductivity. This analytical result is implemented in a computer program called HATS (High Accuracy Temporal Solution), which is virtually free of numerical dispersion.		

UNCLASSIFIED

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

20. ABSTRACT (continued)

Several sample problems are formulated and then solved via this program. Appendix B supplies a manual that details use of the program and includes a source listing of HATS, sample input, and plots.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

TABLE OF CONTENTS

	<u>Page</u>
STATEMENT OF THE PROBLEM.....	5
SOLUTION TECHNIQUE.....	8
SAMPLE PROBLEM.....	8
CONCLUSIONS.....	17
REFERENCES.....	17
APPENDIXES:	
A - OUTLINE OF SOLUTION TECHNIQUE.....	19
B - USE OF THE HATS PROGRAM.....	25

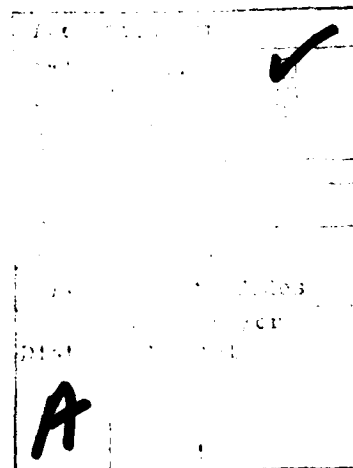
LIST OF ILLUSTRATIONS

Figure

1.	Example of permittivity and conductivity profiles for a one-dimensional, three-layer medium.....	6
2.	A sinusoidal incident pulse of finite duration.....	7
3.	A spike incident field.....	7
4.	Permittivity profile for the sample problem.....	9
5.	Incident field for the sample problem. (20-ns pulse, 1000 MHz).....	9
6.	Internal field (z=0.714 cm) vs. time (ns), zero conductivity.....	10
7.	Internal field (z=1.067 cm) vs. time (ns), zero conductivity.....	11
8.	Internal field (z=2.100 cm) vs. time (ns), zero conductivity.....	12
9.	Conductivity profile for the sample problem.....	13
10.	Internal field (z=0.714 cm) vs. time (ns), nonzero conductivity.....	14
11.	Internal field (z=1.067 cm) vs. time (ns), nonzero conductivity.....	15

<u>Figure</u>	<u>Page</u>
12. Internal field ($z=2.100$ cm) vs. time (ns), nonzero conductivity.....	16
A-1. Space-time diagram showing domain of dependence along line $\tau=0$	22
A-2. Space-time diagram showing domain of dependence along line $x=1$	22
B-1. Permittivity and conductivity profiles for a two-layer medium.....	27
B-2. Some typical incident fields, $E^i(\frac{z}{c} - t)$, where c is the speed of light in free space.....	28
B-3. Permittivity and conductivity profiles for a three-layer medium.....	60
B-4. Internal, reflected, and transmitted fields for a medium with nonzero conductivity, time $t=6$ ns. Incident field is 1000-MHz pulse of 20-ns duration, impinging on the medium from the left at time $t=0$	69
B-5. Internal, reflected, and transmitted fields for a medium with nonzero conductivity, time $t=21$ ns. Incident field is 1000-MHz pulse of 20-ns duration, impinging on the medium from the left at time $t=0$	69
B-6. Comparison of transient and steady state intensities in a medium with nonzero conductivity. Incident field is 1000-MHz pulse.....	70
B-7. Internal field ($x=.15625$), nonzero conductivity.....	71
B-8. Internal field ($x=.25$), nonzero conductivity.....	72
B-9. Internal field ($x=.71875$), nonzero conductivity.....	73
B-10. Internal, reflected, and transmitted fields for a medium with zero conductivity, time $t=6$ ns. Incident field is 1000-MHz pulse of 20-ns duration, impinging on the medium from the left at time $t=0$	74
B-11. Internal, reflected, and transmitted fields for a medium with zero conductivity, time $t=21$ ns. Incident field is 1000-MHz pulse of 20-ns duration, impinging on the medium from the left at time $t=0$	74
B-12. Comparison of transient and steady state intensities in a medium with zero conductivity. Incident field is 1000-MHz pulse.....	75

<u>Figure</u>	<u>Page</u>
B-13. Internal field ($x=.15625$), zero conductivity.....	76
B-14. Internal field ($x=.25$), zero conductivity.....	77
B-15. Internal field ($x=.71875$), zero conductivity.....	78
B-16. Reflected and internal fields ($t=.181$ ns), zero conductivity, spike incident field.....	79
B-17. Reflected and internal fields ($t=.361$ ns), zero conductivity, spike incident field.....	79
B-18. Reflected and internal fields ($t=.542$ ns), zero conductivity, spike incident field.....	80
B-19. Reflected and internal fields ($t=.723$ ns), zero conductivity, spike incident field.....	80



TRANSIENT FIELDS PRODUCED IN HETEROGENEOUS BODIES BY ELECTROMAGNETIC PULSES

STATEMENT OF THE PROBLEM

This effort examines the question of electromagnetic field structure induced in heterogeneous bodies by pulse electromagnetic radiation. Of special interest is the case of a sinusoidally varying incident field of finite duration.

To determine the structure of induced electromagnetic fields, the problem was modeled mathematically. The assumptions made and the resulting model are given below. As the model is described, the nature of the media and incident fields to be considered will be made clear.

The medium considered in this work is one dimensional and of finite depth L . Thus, it is assumed to vary spatially only in the z direction, extending from $z=0$ to $z=L$. The permittivity, $\epsilon(z)$, and conductivity, $\sigma(z)$, of the medium are piecewise continuous functions as pictured, for example, in Figure 1. By allowing ϵ and σ to be piecewise continuous, a layered medium can be considered. Free space is assumed to exist outside of the medium, so $\epsilon(z)=\epsilon_0$ and $\sigma(z)=0$ for $z<0$ and $z>L$. Finally, the medium is assumed to be non-dispersive in that the permittivity and conductivity are independent of the induced field.

An electromagnetic wave propagating along the z -axis normal to the medium has transverse electric field, $E(z,t)$, satisfying

$$E_{zz} - \epsilon(z)\mu_0 E_{tt} - \sigma(z)\mu_0 E_t = 0, \quad -\infty < z < \infty, \quad -\infty < t < \infty \quad (1)$$

which follows from Maxwell's equations. Here, t denotes time, μ_0 is the constant magnetic permeability, and subscripts denote partial derivatives. The

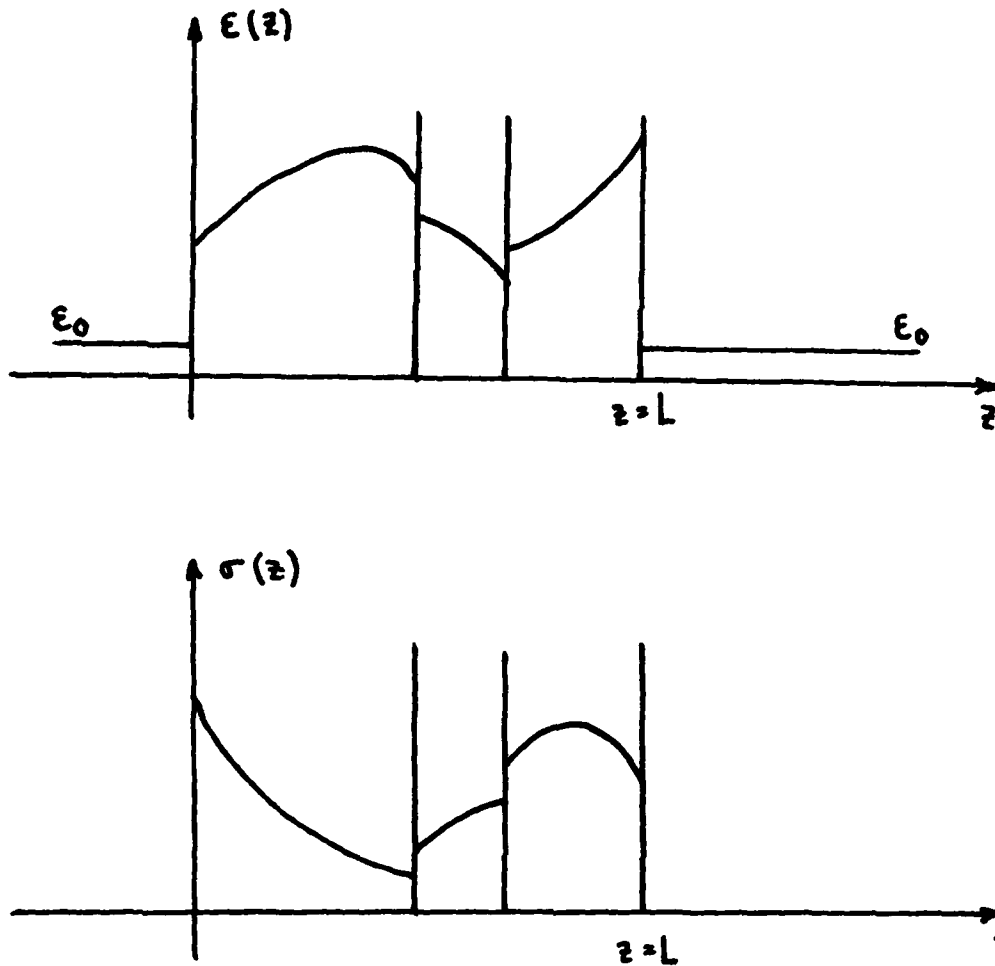


Figure 1. Example of permittivity and conductivity profiles for a one-dimensional, three-layer medium.

incident electromagnetic pulse, E^i , is assumed to propagate from left to right and first impinges on the medium at time $t=0$. Thus, the initial condition for equation (1) is

$$E(z,t)=E^i(z-ct) \text{ for } t<0 \quad (2)$$

where $E^i(z-ct)=0$ for $z-ct \geq 0$ and c is the speed of light in free space. Figures 2 and 3 show what some typical incident pulses may look like.

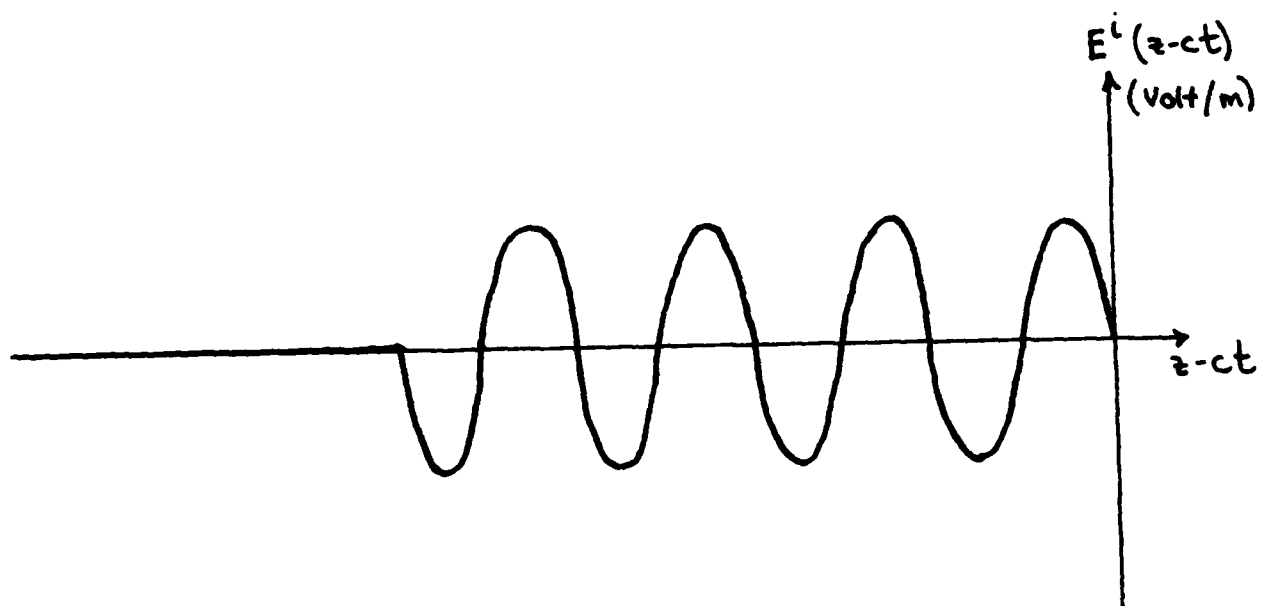


Figure 2. A sinusoidal incident pulse of finite duration.



Figure 3. A spike incident field.

The problem considered, then, is that of determining the solution $E(z,t)$ of the system of equations (1) and (2) for $0 \leq z \leq L$, $t > 0$ for any given $E^i(z-ct)$, $\epsilon(z)$, $\sigma(z)$.

SOLUTION TECHNIQUE

The problem modeled above has been solved numerically by the HATS computer program (High Accuracy Temporal Solution). A detailed description of the mathematical foundations of the numerical methods used in the program can be found in reference 8, and preliminary results are given in references 4-7. A brief summary of reference 8 is given in Appendix A. A description of the use and capabilities of this program is given in Appendix B, which is a self-contained user's manual for HATS.

Note that the numerical solution is based on exact analytical representations of the fields involved, with no recourse to low frequency approximations. The technique used has been found to be highly stable.

SAMPLE PROBLEM

As an example of the results that can be obtained via the HATS program, consider the three-layer medium shown in Figure 4, where it is first assumed that $\sigma(z) \equiv 0$. A 1000-MHz incident field of 20-ns duration impinges on the medium from the left at time $t=0$. Thus, at some time $t < 0$ the incident field looks like that shown in Figure 5.

The resulting field inside the medium was generated by the HATS program for the time interval $t=0$ to $t=25$ ns. Figures 6, 7, and 8 show the internal fields as a function of time at three specific locations ($z=.714$ cm, $z=1.067$ cm, $z=2.100$ cm) in the medium. The locations $z=.714$ cm and $z=2.100$ cm (Figs. 6 and 8) were chosen because they represent in some sense the "worst case" behavior, in that the transients at these points had the greatest amplitude relative to steady state amplitude of any points in the medium. Figure 7 ($z=1.067$ cm) represents the "best case" behavior in that the ratio of transient amplitude to steady amplitude was the smallest. At all points in the medium, transient amplitude was greater than steady state amplitude.

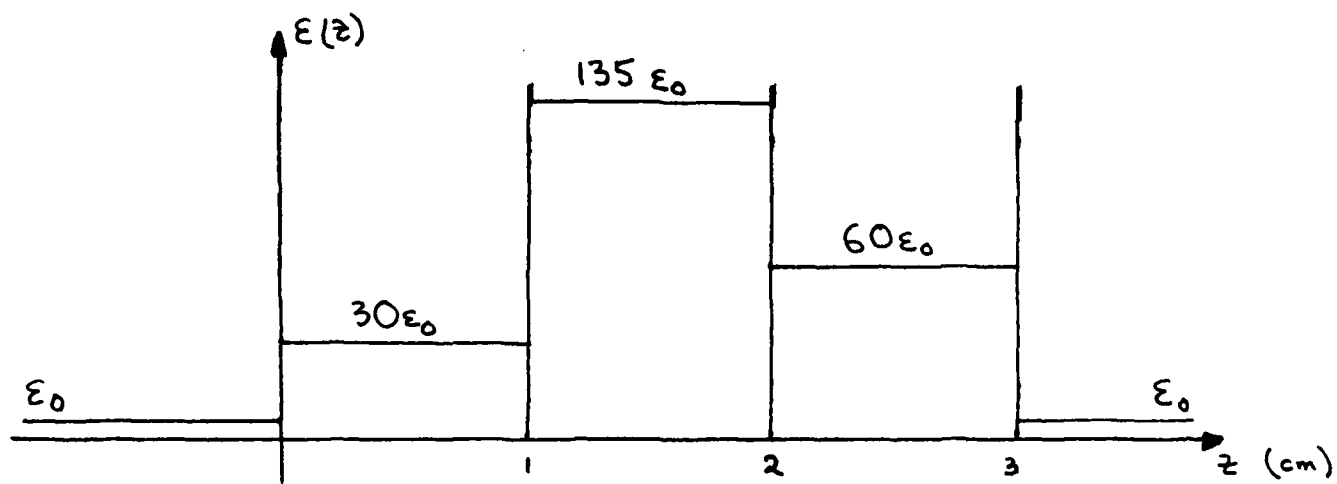


Figure 4. Permittivity profile for the sample problem.

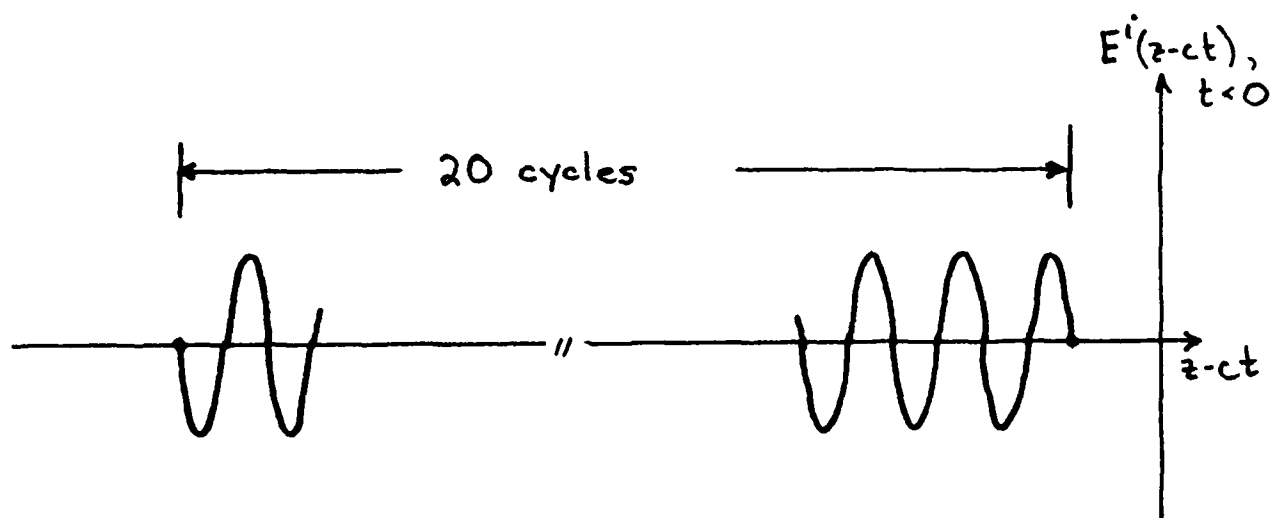


Figure 5. Incident field for the sample problem. (20-ns pulse, 1000 MHz)

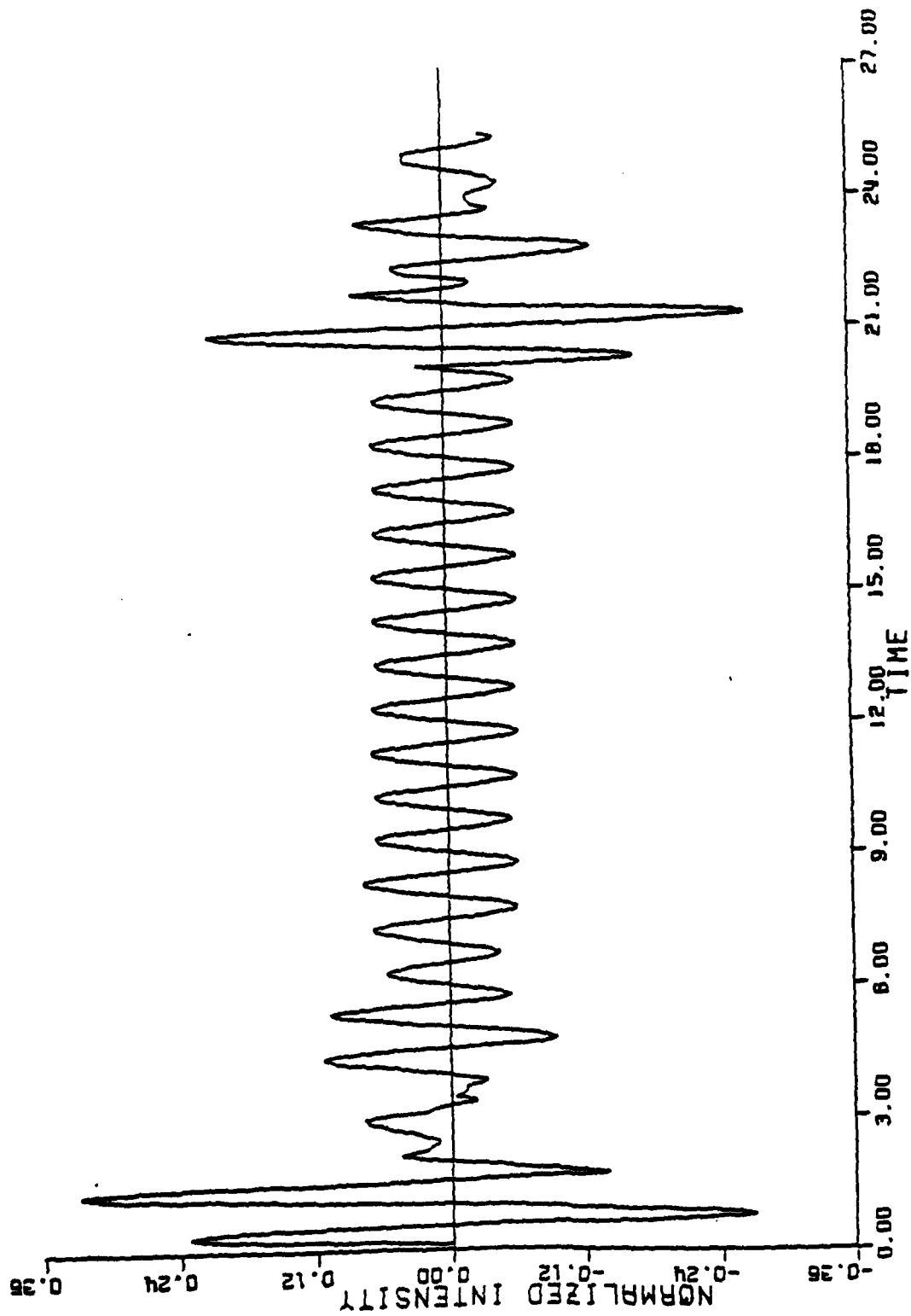


Figure 6. Internal field ($z=0.714$ cm) vs. time (ns), zero conductivity.

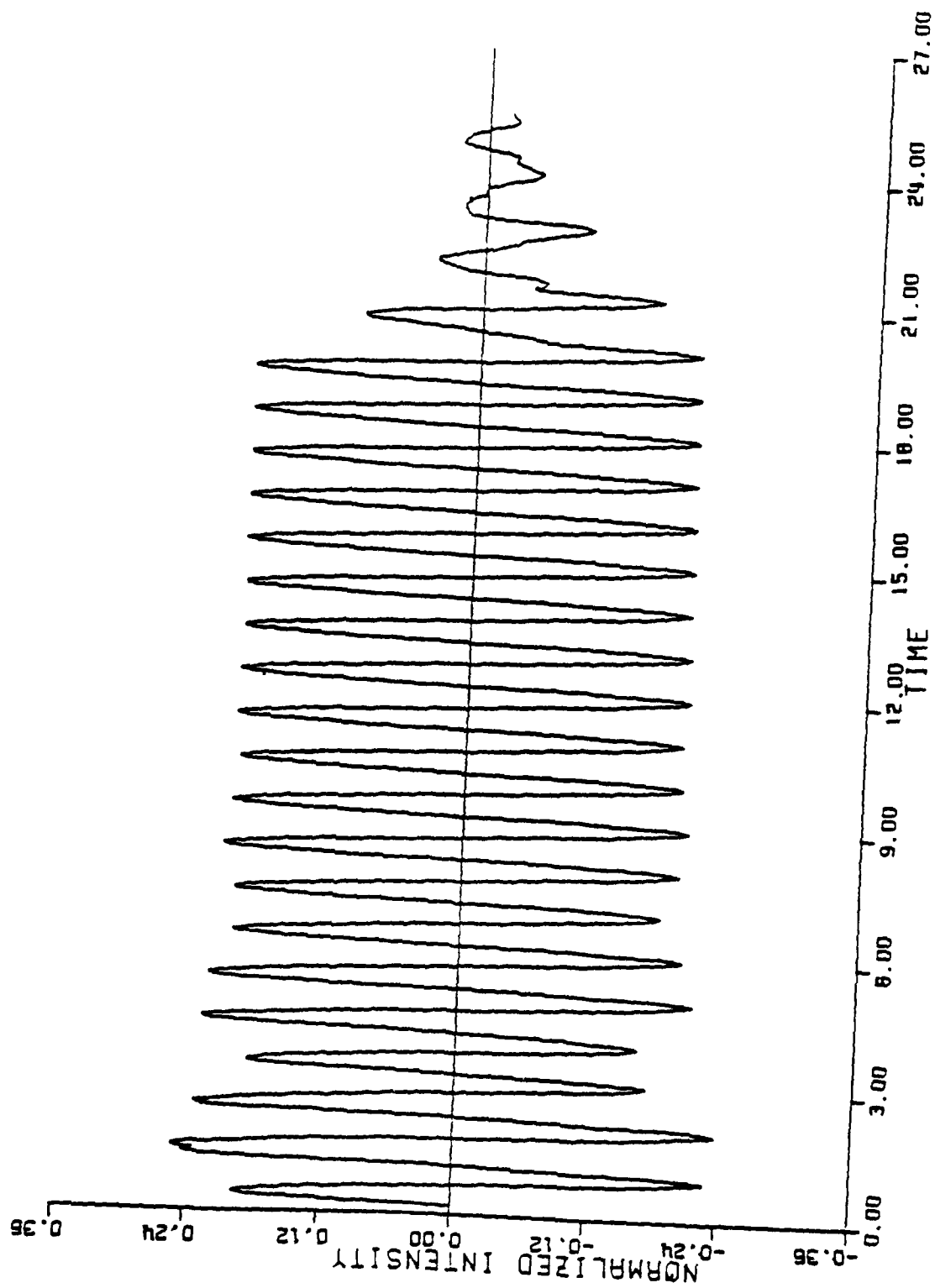


Figure 7. Internal field ($z=1.067$ cm) vs. time (ns), zero conductivity.

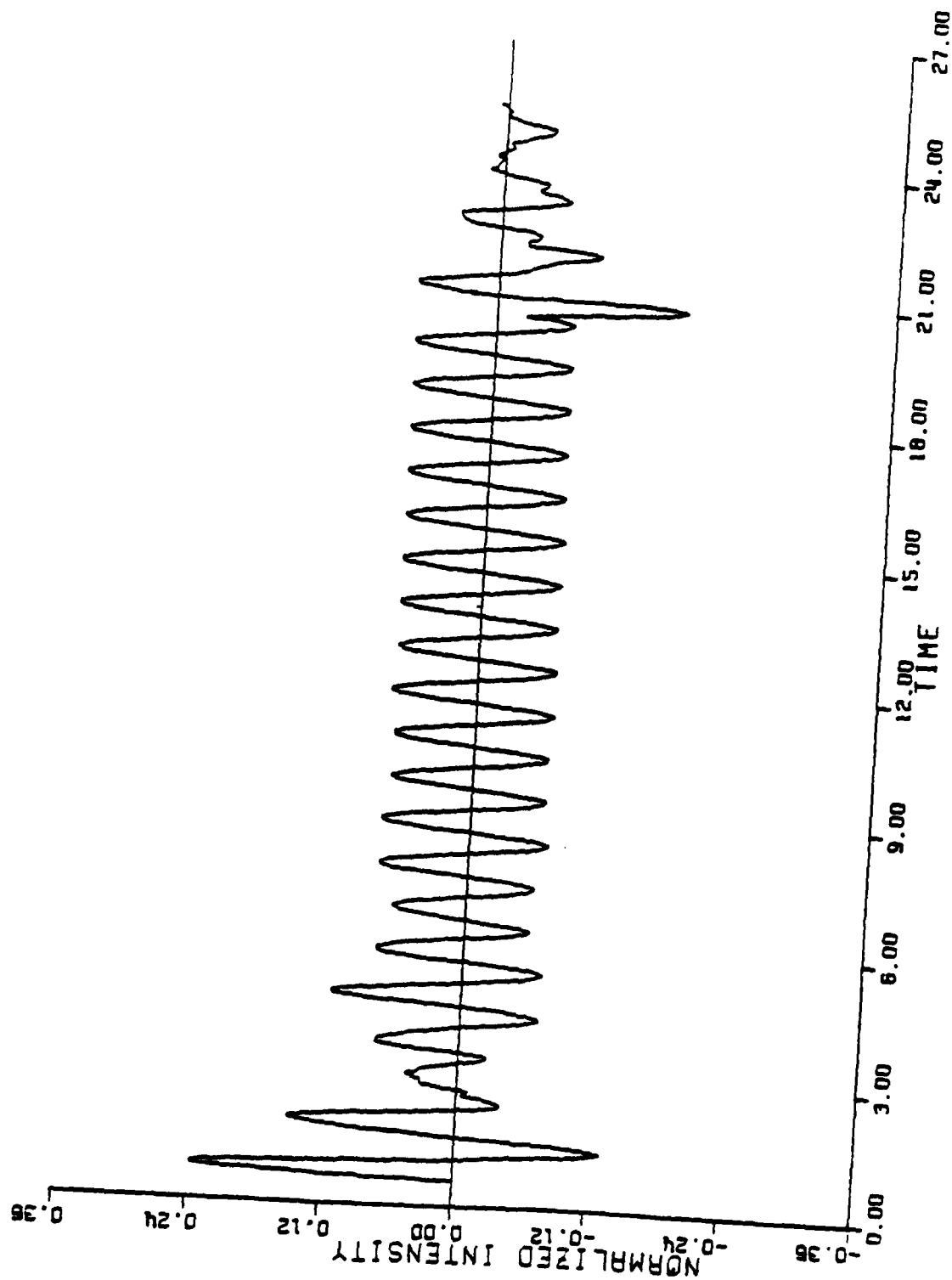


Figure 8. Internal field ($z=2.100$ cm) vs. time (ns), zero conductivity.

Now consider a medium with permittivity profile given in Figure 5 and conductivity profile given in Figure 9. The fields at $z=.714$ cm, $z=1.067$ cm, $z=2.100$ cm (Figs. 10-12) fairly accurately give worst and best behavior. Again, transient amplitude was everywhere greater than steady amplitude, although the effect of nonzero conductivity reduces the ratio of transient to steady amplitude.

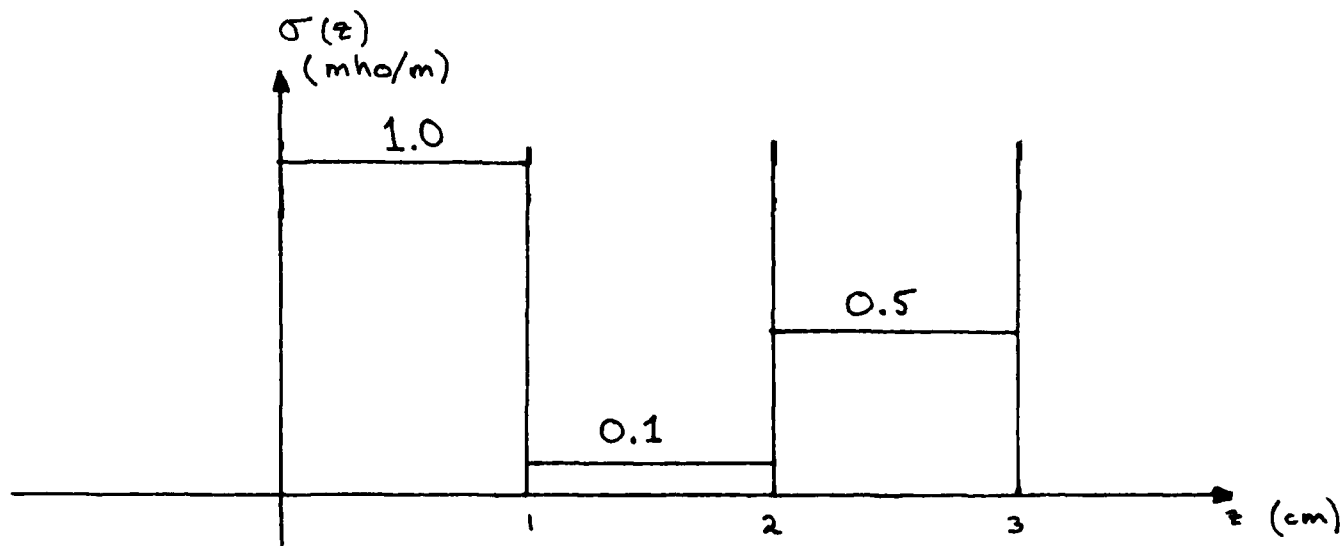


Figure 9. Conductivity profile for the sample problem.

Additional information regarding these plots (how the specific values of z were chosen, views of the entire field in the medium at particular times, etc.) and a detailed discussion of how they were obtained are given in Appendix B. The problem studied in this section is only a sample problem; any permittivity and conductivity profiles and incident field can be input into the HATS program, modulo the restrictions called out in Appendix B.

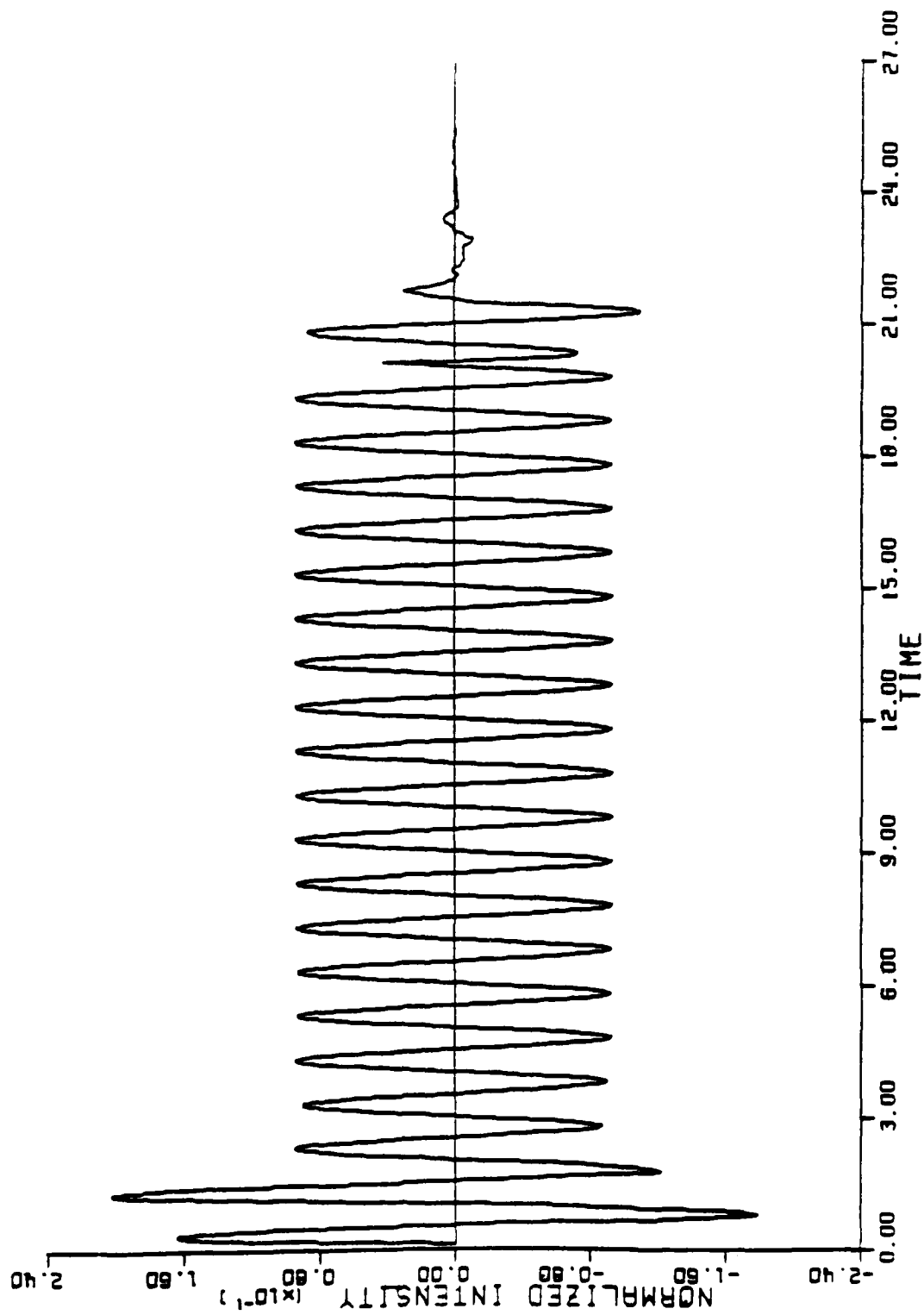


Figure 10. Internal field ($z=0.714$ cm) vs. time (ns), nonzero conductivity.

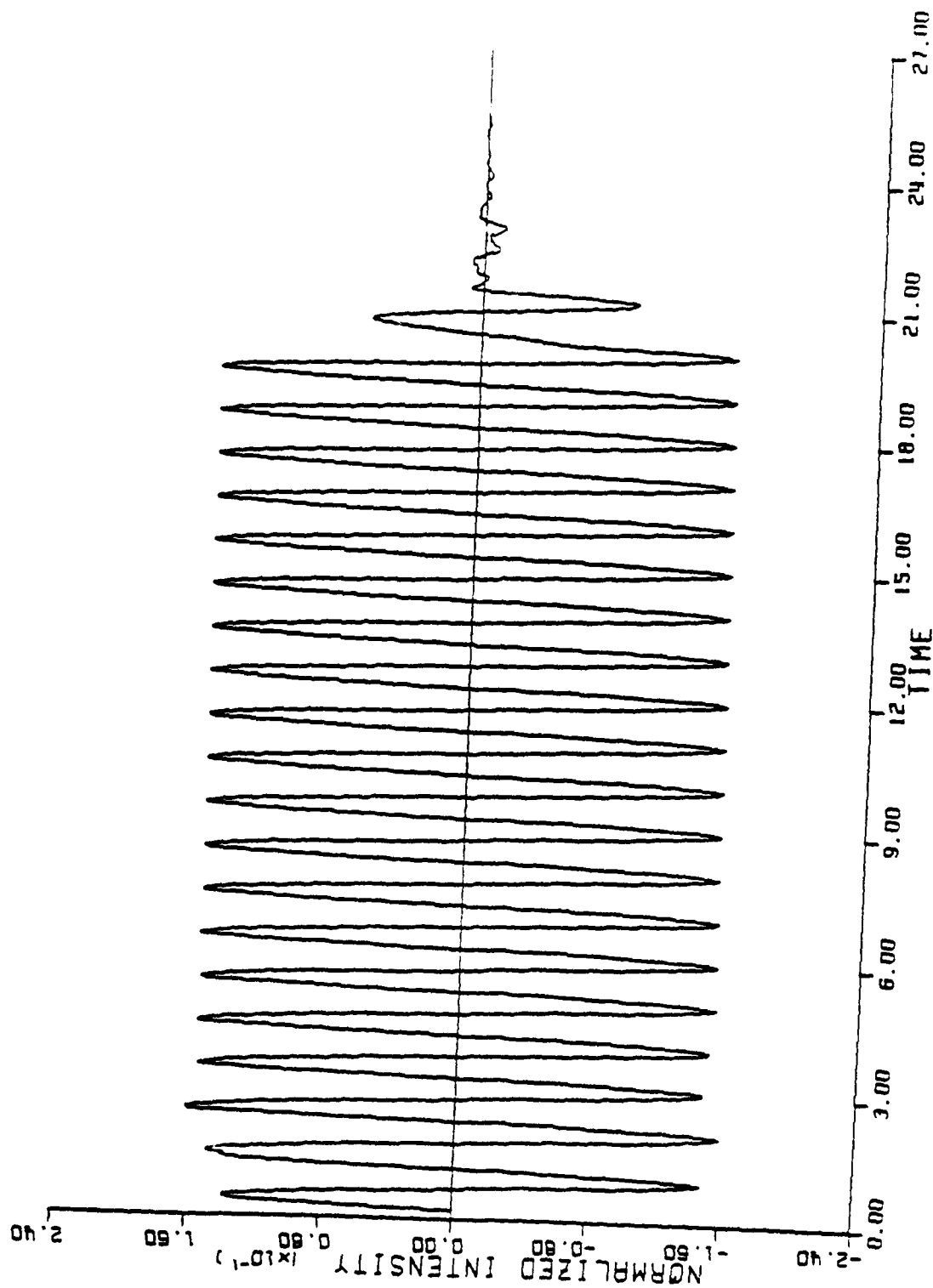


Figure 11. Internal field ($z=1.067$ cm) vs. time (ns), nonzero conductivity.

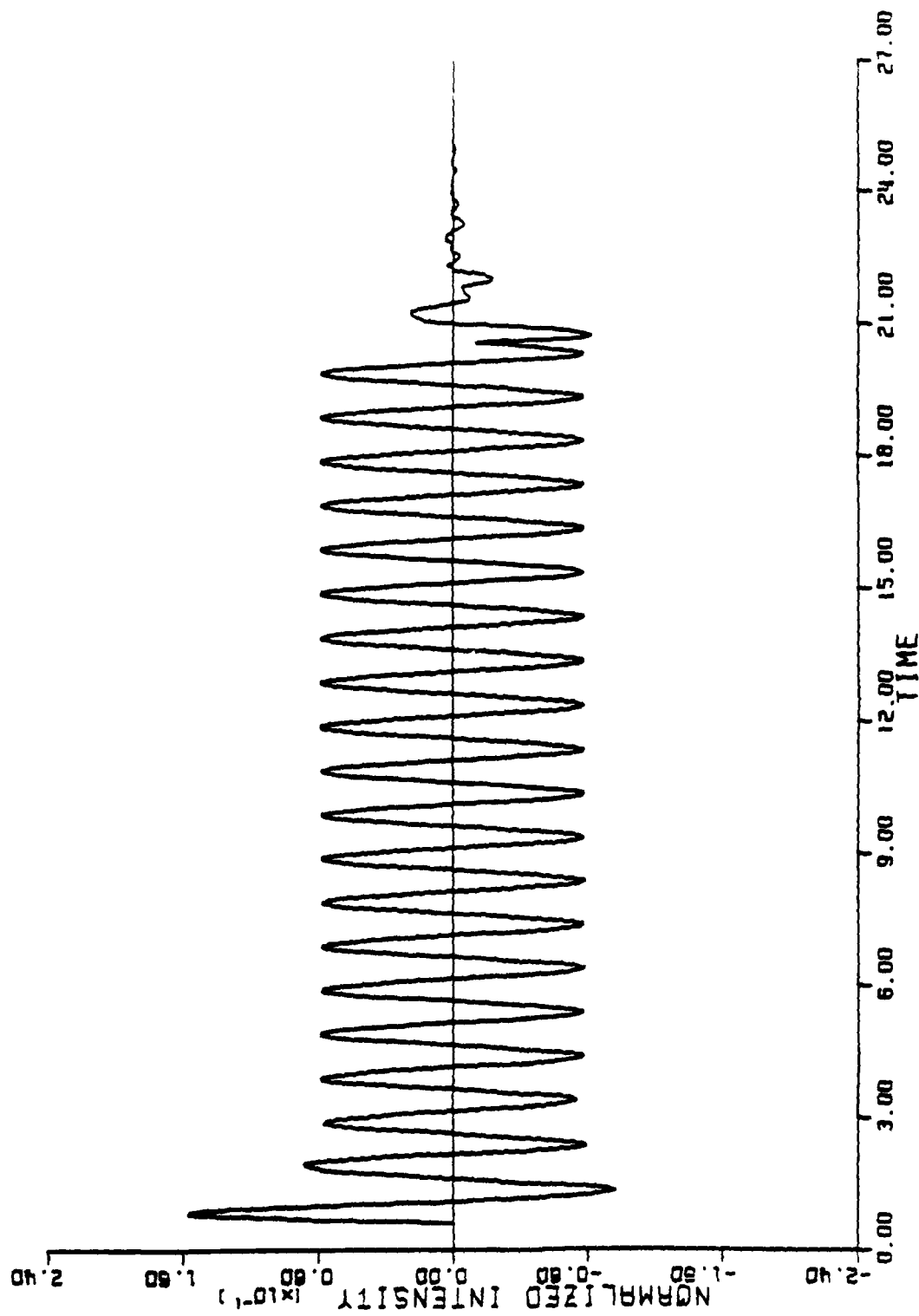


Figure 12. Internal field ($z=2.100$ cm) vs. time (ns), nonzero conductivity.

CONCLUSIONS

Use of the HATS computer program has shown that relatively large amplitude transients can occur in a medium when a transient signal, such as that from a phased array system, impinges on the medium. Although the model used to determine this is a fairly simple one, the pattern shown here should extend to more realistic situations. Nonetheless, further work is needed to definitely establish the presence and magnitude of these transients for other geometries and for the important case of a frequency dependent medium, such as human tissue.

REFERENCES

1. Ames, W. F. Numerical methods for partial differential equations. New York: Academic Press, 1977.
2. Baker, C. T. H. The numerical treatment of integral equations. Oxford: Oxford University Press, 1977.
3. Bolomey, J., C. Durix, and D. Lesselier. Time domain integral equation approach for inhomogeneous and dispersive slab problems. IEEE Trans Antennas Propagat AP-26:658-667 (1978).
4. Krueger, R. An inverse problem for a dissipative hyperbolic equation with discontinuous coefficients. Quart Appl Math 34:129-147 (1976).
5. Krueger, R. An inverse problem for an absorbing medium with multiple discontinuities. Quart Appl Math 36:235-253 (1978).
6. Krueger, R. Some results on linear delay integral equations. J Math Anal Appl 67:232-238 (1979).
7. Krueger, R. Numerical aspects of a dissipative inverse problem. To appear in IEEE Trans Antennas Propagat.
8. Krueger, R. A scattering operator approach to the numerical solution of time dependent wave equations. Unpublished article.

APPENDIX A. OUTLINE OF SOLUTION TECHNIQUE

The problem considered in this work is modeled by

$$E_{zz} - \epsilon(z)\mu_0 E_{tt} - \sigma(z)\mu_0 E_t = 0, \quad -\infty < z < \infty, \quad -\infty < t < \infty. \quad (\text{A.1})$$

It is assumed that $\epsilon(z)$ and $\sigma(z)$ are given and that $\epsilon(z) = \epsilon_0$, $\sigma(z) = 0$ for $z < 0$ and $z > L$. Finally, the solution of equation (A.1) is given by

$$E(z, t) = E^i(z - ct) \quad \text{for } z < 0 \quad (\text{A.2})$$

where $E^i(z - ct) = 0$ for $z - ct > 0$. The solution of the initial value problem (A.1, A.2) must be found for $0 \leq z \leq L$ and $t > 0$.

To facilitate the numerical solution of equation (A.1), the transformations

$$x = x(z) = \left(\int_0^z \{ \epsilon(s) \mu_0 \}^{\frac{1}{2}} ds \right) / \ell \quad (\text{A.3})$$

$$\tau = t / \ell \quad (\text{A.4})$$

$$u(x, \tau) = E(z, t) \quad (\text{A.5})$$

are used to convert equation (A.1) to

$$u_{xx} - u_{\tau\tau} + A(x)u_x + B(x)u_\tau = 0, \quad -\infty < x < \infty, \quad -\infty < \tau < \infty \quad (\text{A.6})$$

where

$$A(x) = -\ell \frac{d}{dz} \{ \epsilon(z) u_0 \}^{-\frac{1}{2}}$$

$$B(x) = -\ell \sigma(z) / \epsilon(z)$$

and ℓ is such that $x(L)=1$. In terms of this transformed problem, the solution $u(x, \tau)$ of equation (A.6) must be found for $0 \leq x \leq 1$, $\tau > 0$ given that

$$u(x, \tau) = u^i(x - \tau) \text{ for } \tau < 0 \quad (A.7)$$

where $u^i(\tau) = E^i(c\ell\tau)$.

The advantage of the change of variables (A.3)-(A.5) is that the characteristics of equation (A.6) are straight lines. On the other hand, while the solution E of equation (A.1) has continuous first derivatives, the solution u of equation (A.6) has discontinuities in its x derivative. To see this, assume that $\epsilon(z)$ is discontinuous at the points $z=z_i$, $i=0, 1, \dots, NL$ where $0=z_0 < z_1 < z_2 < \dots < z_{NL}=L$. Let $x_i = x(z_i)$ for $i=0, 1, \dots, NL$. Then the change of variable (A.3) implies that

$$c_i u_x(x_i +, \tau) = u_x(x_i -, \tau) \quad (A.8)$$

for $i=0, 1, \dots, NL$ where

$$c_i = \{ \epsilon(z_i +) / \epsilon(z_i -) \}^{\frac{1}{2}}.$$

One technique for solving the problem (A.6)-(A.8) is by using finite differences [1]. However, in using an explicit differencing scheme on this problem, roundoff error accumulated quickly and prevented accurate solutions for incident pulses of realistic duration. An alternate approach is to use the time-domain integral equation technique of Bolomey et al. [3] with suitable modifications to take into account the finite depth of the medium. However,

this technique proved to be susceptible to numerical dispersion in that truncation error propagates through the solution at nonphysical speeds, causing spreading of the fields. Again, for pulses of realistic duration, this creates highly unreliable solutions.

The above two solution techniques are based on the familiar space-time diagram shown in Figure A-1, which displays the fact that the solution u at the point (x, τ) depends on the solution in a certain interval (the domain of dependence) on the x -axis determined by the characteristic lines emanating from (x, τ) . Note, however, that an equally valid solution technique is afforded through the space-time diagram of Figure A-2. To understand the data required for the formulation of the problem suggested by Figure A-2, note that the incident electromagnetic field $E^i(z-ct)$ gives rise to a transmitted field $E^t(z-ct)$ for $z > L$. In the transformed problem (A.6)-(A.8), the incident plane wave $u^i(\tau) = E^i(c\ell\tau)$ produces a transmitted plane wave $u^t(x-\tau)$ where $u^t(\tau) = E^t(L-c\ell+c\ell\tau)$ and $u^i(\tau) = u^t(\tau) = 0$ for $\tau > 0$. Since $u(1, \tau) = u^t(1-\tau)$, it follows that the data required for the formulation of the problem shown in Figure A-2 is $u^t(\tau)$ for $\tau < 0$. The advantage of this formulation over that suggested by Figure A-1 is that exact analytical expressions exist for the solution $u(x, \tau)$ in terms of the data u^t and these expressions give rise to highly accurate numerical schemes.

Pursuing the approach suggested by Figure A-2, it is first necessary to determine the transmitted field u^t corresponding to a given incident field u^i . In reference 5 these fields are shown to be related by the equation

$$u^i(\tau) = \sum_{j=1}^{2NL} w_j u^t(\tau + r_j) + \int_{\tau}^0 W(\tau-s) u^t(s) ds, \quad \tau < 0. \quad (A.9)$$

The constants w_j and r_j are known and are determined by the properties of the medium (formulas are given in reference 5). The kernel function, W , also depends only on the properties of the medium and is independent of the fields u^i, u^t .

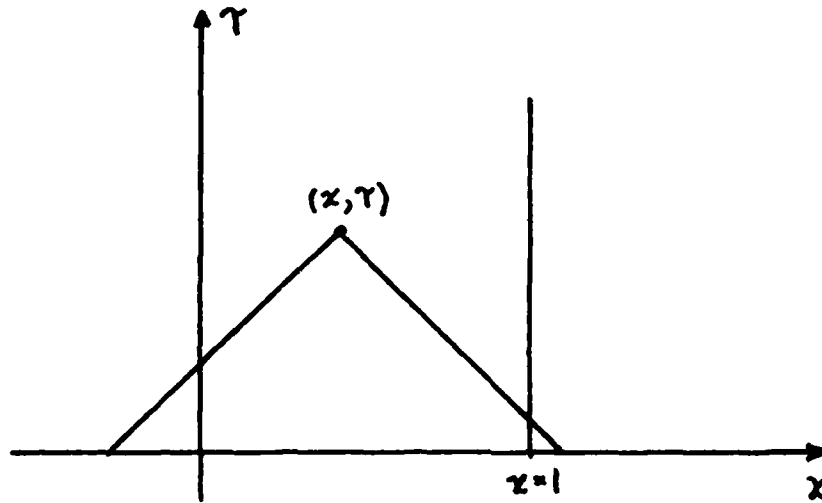


Figure A.1. Space-time diagram showing domain of dependence along line $\tau=0$.

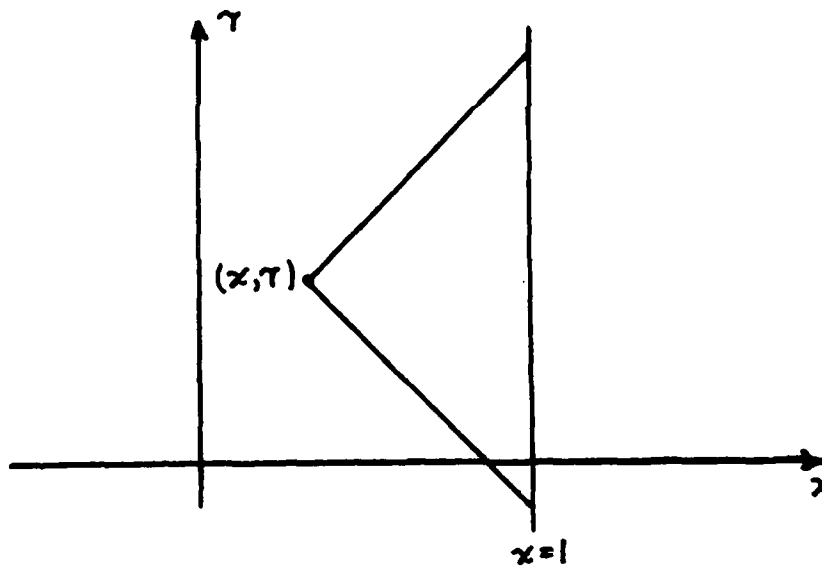


Figure A.2. Space-time diagram showing domain of dependence along line $x=1$.

Assuming for the moment that W is known, equation (A.9) is a delay Volterra integral equation for u^t . Thus, it is equivalent to a Volterra integral equation of the second kind (as shown in reference 6), so stable numerical techniques for solving equation (A.9) for u^t are available (see reference 2). Consequently, the data required for the formulation in Figure A-2 can be considered known.

Reference 8 shows that for any x in the interval $[x_{NL-1}, 1]$ and for any τ , the solution $u(x, \tau)$ of equation (A.6) is given by

$$\begin{aligned} 2u(x, \tau) = & \exp\left\{\frac{1}{2}\int_x^1 (A(s) - B(s))ds\right\} \{ (c_{NL} + 1)u^t(x - \tau) \\ & - (c_{NL} - 1)u^t(2 - x - \tau) \exp\left\{\int_x^1 B(s)ds\right\} \\ & + \int_x^{2-x} u^t(y - \tau) N(x, y, 1, \tau) dy \}. \end{aligned} \quad (A.10)$$

Here, $N(x, y, 1)$ is the solution of the characteristic initial value problem

$$N_{xx} - N_{yy} + B(x)(N_x + N_y) + D_+(x)N = 0, \quad x_{NL-1} \leq x \leq x_{NL} = 1, \quad (A.11)$$

$$2N(x, x, 1) = c_{NL} B(1) - A(1) - (c_{NL} + 1) \int_x^1 n_+(s) ds, \quad (A.12)$$

$$2N(x, 2 - x, 1) = (c_{NL} B(1) - A(1) + (c_{NL} - 1) \int_x^1 D_-(s) ds) \exp\left\{\int_x^1 B(s) ds\right\} \quad (A.13)$$

where

$$A(1) = A(1-), \quad B(1) = B(1-)$$

and

$$D_{\pm} = 1/4(B^2 - A^2) + 1/2(-A' \pm B').$$

The problem (A.11)-(A.13) can be easily solved numerically by the method of characteristics [1]. Thus, $u(x,\tau)$ can be determined from equation (A.10) for any x in $[x_{NL-1}, 1]$ by means of a simple quadrature. To determine $u(x,\tau)$ for x in $[x_{NL-2}, x_{NL-1}]$, reference 8 shows that a system of formulas analogous to equations (A.10)-(A.13) utilizes the data $u(x_{NL-1}, \tau)$. Continuing in this manner, quadrature formulas can be used to determine $u(x,\tau)$ for any x in the interval $[0,1]$. In the program HATS, the trapezoidal rule is used to perform these quadratures.

The remaining point to be considered is determination of the kernel function, W , which appears in equation (A.9). If the transmitted field is a Dirac delta function,

$$u^t(\tau) = \delta(\tau) \tag{A.14}$$

then, according to equation (A.9), the "incident field" that would generate such a transmitted field is the kernel W , modulo delta function singularities. But if the transmitted field is assumed to be given by equation (A.14), then the technique outlined above enables one to determine the corresponding field $u(0,\tau)$ for all τ . From this it is possible to separate out the incident field that generates (A.14) and hence obtain W . Again, details are given in reference 8.

APPENDIX B. USE OF THE HATS PROGRAM

	<u>Page</u>
I. Introduction.....	26
II. Inputting ϵ and σ	29
III. Inputting the Incident Wave.....	32
IV. Other Input Data.....	33
V. Output.....	37
VI. Precautions, Tricks, and General Observations.....	39
Extending the Maximum Time.....	40
Increasing the Number of Intervals Allowed.....	41
Decreasing the Step Size.....	41
Numerical Solution of Equation (B.1).....	42
VII. Source Listing of HATS.....	44
VIII. Setting Up Some Programs.....	59
IX. Source Listing of INCWAV for Field of Finite Duration.....	64
X. Source Listing of a Plotting Routine.....	64
XI. Searching for Maximum Intensity.....	66
XII. Plots from Sample Programs.....	68

I. INTRODUCTION

The following general description is meant to be a user's guide to the program HATS (High Accuracy Temporal Solution) and does not dwell on mathematical details other than what is needed to properly use the program.

Briefly stated, this program allows a user to cause an incident plane electromagnetic wave to impinge normally on a medium of finite depth and to determine the resulting scattered and internal fields. The medium is assumed to be one-dimensional (i.e., stratified in the z direction) and "linear" in that its permittivity, $\epsilon(z)$, and conductivity, $\sigma(z)$, are independent of frequency. Free space is assumed to exist on either side of the medium. The functions $\epsilon(z)$ and $\sigma(z)$ are to be piecewise continuously differentiable and, hence, may have jump discontinuities. If a layer is defined to be an interval in the medium on which ϵ and σ are continuously differentiable, then the program HATS is capable of handling a five-layer medium. A two-layer medium is pictured in Figure B-1.

The incident field can be of any form the user desires with the following two provisions:

1. The incident field must be a continuous function.
2. The incident field does not contact the medium with time $t=0$.
Thus, transient as well as steady state fields can be determined by this program.

See Figure B-2 for a picture of some typical incident fields.

Details regarding proper input into the program and form of the output will be given in the following sections.

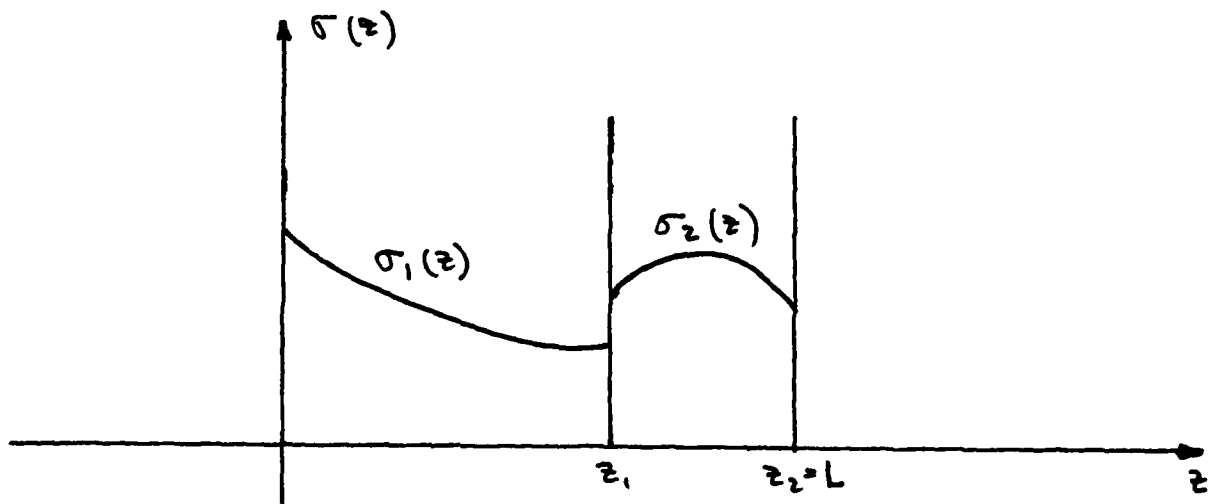
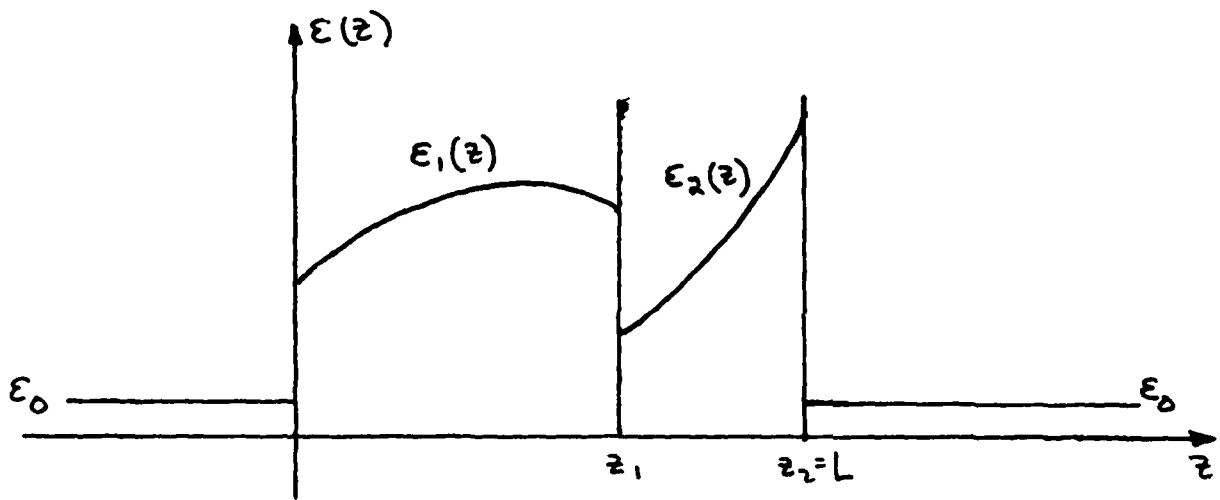


Figure B-1. Permittivity and conductivity profiles for a two-layer medium.

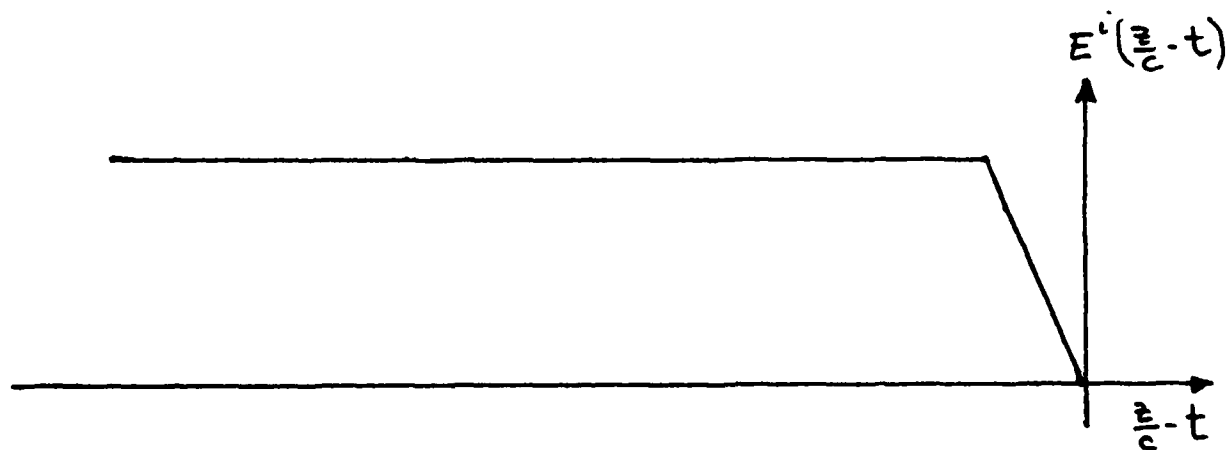
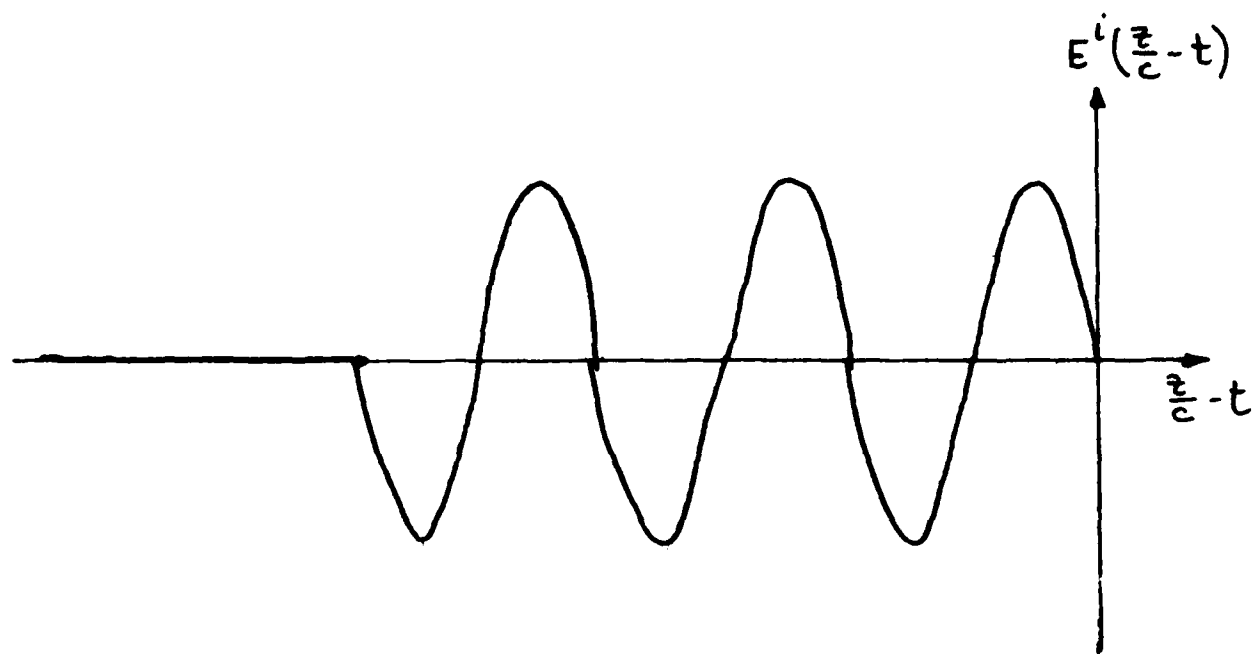


Figure B-2. Some typical incident fields, $E^i(\frac{z}{c} - t)$, where c is the speed of light in free space.

II. INPUTTING ϵ AND σ

The program presumes that a Liouville transform has been made on the spatial variable z and that a normalization has been performed on the time variable t . Therefore, a certain amount of massaging must be done before data and functions are put into the program, and some interpretation of the program results is necessary. All data are initially expressed in rationalized MKS units.

Consider a medium of NL layers, where $1 \leq NL \leq 5$. Layer i extends from $z=z_{i-1}$ to $z=z_i$, where $z_0=0$ and $z_{NL}=L$. Assume that the permittivity and conductivity are given on layer i by the functions $\epsilon_i(z)$ and $\sigma_i(z)$, $z_{i-1} < z < z_i$, and let

$$\epsilon(z) = \epsilon_i(z) \text{ for } z_{i-1} < z < z_i, \quad i=1, \dots, NL.$$

The Liouville transform

$$x = x(z) = \left(\int_0^z \{ \epsilon(s) \mu_0 \}^{\frac{1}{2}} ds \right) / \ell \quad (R.1)$$

where

$$\ell = \int_0^L \{ \epsilon(s) \mu_0 \}^{\frac{1}{2}} ds$$

and the normalization

$$\tau = t / \ell$$

is used to produce a problem in (x, τ) coordinates in which the velocity of propagation is unity.

In the (x, τ) coordinate system the scattering medium extends from $x=0$ to $x=1$ and is again divided into NL layers. Properties of each of these layers (in terms of the x variable) are put into the program as function statements

appearing in subroutine EVAL. For each layer, four functions must be specified. For layer I ($1 \leq I \leq NL$) these functions are

$$AI(x) = -\frac{d}{dz} \{ \epsilon_I(z) \mu_0 \}^{-\frac{1}{2}}, \quad z_{I-1} < z < z_I$$

$$AIP(x) = \frac{d}{dx} AI(x)$$

$$BI(x) = -\frac{d}{dz} \sigma_I(z) / \epsilon_I(z), \quad z_{I-1} < z < z_I$$

$$BIP(x) = \frac{d}{dx} BI(x).$$

These functions are to be entered in the appropriate lines in subroutine EVAL. If NL is less than five, the statement functions which appear in EVAL for layers NL+1, ..., 5 are not used. These unused statements may be left in the program.

As an example of how to determine these input functions, let NL=2, $z_0=0$, $z_1=.01$ m, $z_2=.03$ m, and

$$\epsilon_1(z) = \epsilon_1 \epsilon_0$$

$$\epsilon_2(z) = \epsilon_2 \epsilon_0$$

$$\sigma_1(z) = s_1 \text{ mho/m}$$

$$\sigma_2(z) = s_2 \text{ mho/m}$$

where ϵ_1 , ϵ_2 , s_1 , s_2 are constants and ϵ_0 is the permittivity of free space,

$$\epsilon_0 = 8.854 \cdot 10^{-12} \text{ coul}^2 / \text{rm}^2.$$

Then

$$\begin{aligned} L &= \int_0^{.01} \{ \epsilon_1 \epsilon_0 \mu_0 \}^{\frac{1}{2}} ds + \int_{.01}^{.03} \{ \epsilon_2 \epsilon_0 \mu_0 \}^{\frac{1}{2}} ds \\ &= (0.01 \sqrt{\epsilon_1} + 0.02 \sqrt{\epsilon_2}) / c \end{aligned}$$

where c is the speed of light in free space. It follows that for $0 < z < 0.01$ m,

$$x = x(z) = (z\sqrt{\epsilon_1})/(\ell c)$$

and for $0.01 < z < 0.03$,

$$x = x(z) = (0.01\sqrt{\epsilon_1} + (z - 0.01)\sqrt{\epsilon_2})/(\ell c).$$

Thus,

$$A1(x) = 0$$

$$A1P(x) = 0$$

$$B1(x) = -\ell s_1 / (\epsilon_1 \epsilon_0)$$

$$B1P(x) = 0$$

$$A2(x) = 0$$

$$A2P(x) = 0$$

$$B2(x) = -\ell s_2 / (\epsilon_2 \epsilon_0)$$

$$B2P(x) = 0.$$

As another example, consider a one-layer medium of depth 1 m (i.e., $z_0 = 0$, $z_1 = 1$) with

$$\epsilon(z) = 9(z+2)^2 \epsilon_0, \quad 0 < z < 1$$

$$\sigma(z) = (10 - (z+1)^2) \cdot 10^{-3} \text{ mho/m}, \quad 0 < z < 1.$$

Then

$$\ell = \frac{3}{c} \int_0^1 (s+2) ds = 15/2c,$$

$$x = x(z) = (z^2 + 4z)/5$$

and

$$z=z(x)=-2\sqrt{4+5x}.$$

It follows that

$$A1(x)=2.5/(z+2)^2=2.5/(4+5x)$$

$$A1P(x)=-12.5/(4+5x)^2$$

$$B1(x)=-10^{-3}(5+5x-2\sqrt{4+5x})/(6c\epsilon_0\{4+5x\})$$

$$B1P(x)=-5\cdot 10^{-3}(4+5x-\sqrt{4+5x})/(6c\epsilon_0\{4+5x\}^{5/2}).$$

For more general permittivity profiles, it may be necessary to solve equation (B.1) numerically for x as a function of z using, say, Newton's method. More will be said about this in Section VI.

III. INPUTTING THE INCIDENT WAVE

The "time" variable τ used in HATS has been normalized by setting

$$\tau=t/\ell$$

and hence, τ is a dimensionless quantity. Thus, if the incident field is $E^i(-t)$, $t>0$, then the corresponding incident field UINC to be used in the program is

$$UINC(-\tau)=E^i(-\tau\ell), \tau>0$$

or

$$UINC(\tau)=E^i(\tau\ell), \tau<0.$$

For example, consider an incident sinusoidal wave train of frequency f . Then

$$E^i(-t) = \sin(2\pi ft), \quad t > 0.$$

It is assumed that $E^i(-t) \equiv 0$ for $t < 0$.) The incident field to be used in the program is

$$UINC(\tau) = \sin(-2\pi f\ell\tau), \quad \tau < 0.$$

This function is to be inserted into the subroutine INCWAV. Depending on the complexity of the incident field, a single function statement may not suffice to define the incident wave. This calls for an obvious modification of the subroutine. See the source listing in Section IX for an example of such a modification. In that example, a wave train of frequency f and of finite duration is generated.

IV. OTHER INPUT DATA

In addition to the function statements that must be written into the program, certain constants are to be read in as data.

Line 1: N, NL, IFRQ, ITM, ISCAT, IOPT

Format: (6I5)

N: number of subintervals to be used in the x interval $[0,1]$. Since HATS solves partial differential equations and integral equations numerically, the x interval $[0,1]$ is split into N equal subintervals to establish a mesh of grid points. Thus, $H=1/N$ is the step size used in most of the calculations. To reduce the effects of roundoff error, it is suggested (but not necessary) that N be a power of 2, say $N=16, 32, 64$, or 128 . N must be an even integer not exceeding 128. Generally, the larger N is, the more accurate the program results, but this is at the expense of increased computation.

NL: number of layers in the medium. NL must be at least 1 and no greater than 5.

IFRQ: frequency of observation of the internal field. In addition to using a "spatial" (x) step size of H, HATS also uses a "temporal" (τ) step size of H. The internal field will be output at τ intervals of

$$\Delta\tau = 2H \cdot \text{IFRQ};$$

i.e., at time intervals of

$$\Delta\tau = 2H \cdot \text{IFRQ} \cdot \ell.$$

ITM: maximum "time" to be considered. ITM is the maximum number of τ intervals of length $2H$ to be used in the program; i.e., the maximum time t considered will be

$$t = 2H \cdot \text{ITM} \cdot \ell.$$

ITM must not exceed 2000 and should be an integral multiple of IFRQ.

ISCAT: output option. If ISCAT=1, the incident, reflected, and transmitted fields (i.e., field transmitted out the right-hand edge of the medium) will be printed out for $\tau=0, 2H, 4H, \dots, 2H \cdot \text{ITM}$. If ISCAT=0, this printout is omitted.

IOPT: output option. If IOPT=0, the field within the medium will be printed out at the $(1+N/2)$ grid points, $x=0, 2H, 4H, \dots, \ell$ for values of τ given by

$$\tau = 2H \cdot \text{IFRQ}, 4H \cdot \text{IFRQ}, 6H \cdot \text{IFRQ}, \dots, 2H \cdot \text{ITM}.$$

If IOPT=1, the field will be printed out for selected grid points (to be supplied later) at the above-mentioned τ values.

Line 2: X(1), X(2), ..., X(6)

Format: (6I5)

Assuming that layer i extends from $z=z_{i-1}$ to $z=z_i$, the Liouville transform (B.1) implies that in the x variable this layer extends from x_{i-1} to x_i , where

$$x_i = (\int_0^{z_i} \{\epsilon(s)\mu_0\}^{\frac{1}{2}} ds) / \ell, \quad i=0, 1, \dots, NL.$$

The quantities $X(i)$ are defined by

$$X(i) = x_{i-1} / H.$$

Notice that $X(1)=0$ and $X(NL+1)=N$. The $X(i)$'s are to be even integers satisfying

$$X(i) - X(i-1) \leq 32, \quad i=2, 3, \dots, NL+1.$$

If the medium has fewer than five layers, the values assigned to $X(NL+2)$, ..., $X(6)$ are immaterial.

Line 3: C(1), C(2), ..., C(6)

Format: (6E12.5)

If the permittivity in layer i is given by $\epsilon_i(z)$, then

$$C(i) = \{\epsilon_i(z_i+) / \epsilon_{i-1}(z_i-)\}^{\frac{1}{2}} \quad (B.2)$$

where, as usual, the z_i 's denote the boundaries of the layers.
For $i=1$, (B.2) is replaced by

$$C(1) = \{\epsilon_1(0+)/\epsilon_0\}^{\frac{1}{2}}$$

and for $i=NL+1$,

$$C(NL+1) = \{\epsilon_0/\epsilon_{NL}(L-)\}^{\frac{1}{2}}.$$

Again, if there are fewer than five layers, the values assigned to $C(NL+2)$, ..., $C(6)$ are immaterial.

The following data must also be supplied if $IOPT=1$.

Line 4: NOBPTS

Format: (I5)

If it is desired to determine the internal field only at selected spatial grid points, then NOBPTS is the number of points in the x interval $[0,1]$ at which the field should be determined.

Line 5: IX

Format: (I5)

IX is an integer specifying the position of the first grid point at which the field should be determined. It is defined by

$$IX = x/H$$

where x is the point where the solution is desired. IX must be an even integer, $0 \leq IX \leq N$.

Line 6: IX

Format: (I5)

This specifies the position of the second point of observation.

This same input is continued until all NORPTS positions have been specified.

V. OUTPUT

HATS automatically prints out the input data from lines 1, 2, and 3.

If ISCAT=1, the incident, reflected, and transmitted fields are printed next. The incident field is printed first in (1X,8E10.3) format for as many lines as is necessary to print the entire field. The array being printed out is

$$UI(J), J=1, 2, \dots, ITM+1$$

where

$$\begin{aligned} UI(J) &= UINC(-2(J-1)H) \\ &= E^i(-2(J-1)H\lambda) \end{aligned}$$

The reflected field is printed next, using the same format. The array being printed here is

$$UR(J) = E^r(2(J-1)H\lambda), J=1, 2, \dots, ITM+1.$$

Finally the transmitted field is printed where

$$UT(J) = E^t(-2(J-1)H\lambda), J=1, 2, \dots, ITM+1.$$

The remaining output depends on the value of IOPT. If IOPT=0, the next line of output is an integer IT (in I10 format) followed by an array of (1+N/2) numbers in (1X,8E10.3) format. The array being printed out is

$$U(J) = u(2(J-1)H, 2 \cdot IT \cdot H).$$

In other words, the field throughout the width of the medium is being printed for time $\tau = 2 \cdot IT \cdot H$. To relate this to the physical field E , note that

$$E(z, t) = u(x, \tau)$$

where, if $x_{i-1} \leq x \leq x_i$,

$$z = z_{i-1} + \left(\ell \int_{x_{i-1}}^x \exp \left\{ - \int_{x_{i-1}}^y A(s) ds \right\} dy \right) / \left(\epsilon_i (z_{i-1}^+) \mu_0 \right)^{\frac{1}{2}},$$

$$t = \tau \ell$$

and

$$A(s) = A_i(s) \text{ for } x_{i-1} \leq s \leq x_i.$$

This printout is continued for all values of IT given by

$$IT = IFTQ, 2 \cdot IFRQ, \dots, ITM.$$

On the other hand, if $IOPT = 1$, then the next line of output is an integer IX (in I10 format) followed by an array of $1 + (ITM - (IX/2)) / IFRQ$ numbers in (IX, 8310.3) format. The array being printed is

$$U(J) = u(IX \cdot H, 2(IFRQ \cdot (J-1) + IX/2)H).$$

In other words, the field at position $x = IX \cdot H$ is being printed at τ intervals of $2H \cdot IFRQ$ starting from $\tau = IX \cdot H$. This printout is continued for all NOBPTS values of IX specified by the user in the input data.

Finally, the message FINISHED is printed to indicate that the program has finished execution in a normal fashion.

In general, deducing any reasonable conclusion by looking at the printed output will be a formidable task. The user will probably find it much more

meaningful to plot the data, using, for example, a CALCOMP plotting routine. For this, a two-step process is suggested.

1. Run HATS, having the output transferred to a disk file.
2. Run the plotting routine, reading data from the disk file.

To facilitate step 1, all data is output from HATS via the subroutine PRNT. The comment statements in the subroutine are self-explanatory, indicating which blocks of code are used to output the various data. In addition, the commented write statements can be used to send a streamlined form of the output to a disk file. The user should note the format used so that the data is later read properly from the disk file. Also note that output device 8 needs to be defined via JCL.

An example of a plotting routine has been listed in Section X. The actual calls made to the graphing subroutines will have to be replaced with calls utilizing software at USAFSAM.

VI. PRECAUTIONS, TRICKS, AND GENERAL OBSERVATIONS

As with all numerical solutions, HATS is not infallible. Continuous but wildly varying conductivity and permittivity profiles within a layer will generally cause inaccurate solutions to be calculated. Also, if the conductivity is excessively large, the numerical solution of the partial differential equations will again prove unreliable. In this case internal and scattered fields will generally start to grow with time, so it will quickly become apparent to the user that the results are in error.

One method for trying to handle large conductivities or wildly varying conductivities and/or permittivities is to decrease step size H (i.e., increase N). Since this will generally cause the restriction

$$X(i) - X(i-1) \leq 32$$

to be violated, the following example shows how HATS can be "tricked."

Consider a two-layer medium and suppose $N=32$, with $X(1)=0$, $X(2)=12$, $X(3)=32$. If a smaller step size is needed and N should be 64, this will place the boundaries of the layers at $X(1)=0$, $X(2)=24$, $X(3)=64$; consequently

$$X(3)-X(2)=40>32.$$

To circumvent this problem, define the medium to consist of three layers instead of two, with boundaries, for example, at $X(1)=0$, $X(2)=24$, $X(3)=44$, $X(4)=64$. The function statements for $A3(x)$, $A3P(x)$, $B3(x)$, $B3P(x)$ will be exactly the same as those for layer 2. Further, $C(1)$ and $C(2)$ will be as in the $N=32$ case, and $C(4)$ will have the value that $C(3)$ has in the $N=32$ case. Finally, to indicate that there is indeed no discontinuity at $X(3)$, set $C(3)=1.0$.

If N needs to be increased beyond 128, consult the last portion of this Section.

If a large step size (small N) is used, then detail in the solution is lost and some accuracy in the solution values calculated will generally be lost. On the other hand, fewer calculations will be performed within the program and the solution can be monitored over a longer period of time since the maximum value of τ is $ITM \cdot 2H$.

Extending the Maximum Time

If necessary, values of ITM greater than 2000 can be considered relatively easily, at the expense of increased storage. It suffices to increase the dimension of the arrays

V , $BUFF$, UI , U , UR

which appear in `COMMON` statements. Let $ITM1=ITM+1$. The new dimensions for the above-mentioned arrays are to be

$V(5,ITM1,2)$, $BUFF(200+ITM1)$, $UI(ITM1)$, $U(ITM1)$, $UR(ITM1)$.

Incidentally, if there are fewer than five layers, storage can be reduced by changing the 5 in $V(5,ITM1,2)$ to the actual number of layers. (In subroutine FIELD there is an array $UT(2001)$ that contains the transmitted field. The dimension of this array need not be increased since UT is equivalenced to $BUFF$).

Increasing the Number of Intervals Allowed

If a value of NL greater than 5 is desired, it is necessary to

1. Change the dimension of all arrays in COMMON statements from 5 to the new value of NL (i.e., wherever a 5 appears, it should be replaced by NL).
2. Change the appropriate READ statements in the main program and WRITE statements in subroutine PRNT.

Decreasing the Step Size

If a value of N greater than 128 must be considered, it is necessary to

1. Change the dimension of all arrays appearing in COMMON statements from 129 to $N+1$.
2. Change the dimension of $BUFF$ to $N+ITM+3$.
3. Change the EQUIVALENCE statement at the beginning of subroutine FIELD to

EQUIVALENCE ($BUFF(N+2)$, $UT(1)$).

Numerical Solution of Equation (B.1)

For some permittivity profiles the Liouville transform (B.1) will need to be solved numerically for z as a function of x . This requires two new arrays to be generated, $Z(I)$ and $ZH(I)$, $I=1, 2, \dots, N$, where

$$(I-1)H = \left(\int_0^{Z(I)} \{\epsilon(s)\mu_0\}^{\frac{1}{2}} ds \right) / l$$

$$(I-.5)H = \left(\int_0^{ZH(I)} \{\epsilon(s)\mu_0\}^{\frac{1}{2}} ds \right) / l$$

and $Z(N+1)=L$. The above equations are readily solved via Newton's method.

Since the functions AI , AIP , BI , BIP in subroutine EVAL cannot be expressed in terms of the x variable, they must now be written as functions of z ,

$$AI(z) = -l \frac{d}{dz} \{ \epsilon_I(z) \mu_0 \}^{-\frac{1}{2}}, \quad z_{I-1} < z < z_I$$

$$AIP(z) = \left(\frac{d}{dz} AI(z) \right) / \left(\frac{dx}{dz} \right)$$

$$BI(z) = -l \sigma_I(z) / \epsilon_I(z), \quad z_{I-1} < z < z_I$$

$$BIP(z) = \left(l \frac{d}{dz} BI(z) \right) / \{ \epsilon_I(z) \mu_0 \}^{\frac{1}{2}}.$$

Finally, in subroutine EVAL the statements

$$XR = (I-1) * H$$

$$XR = (I-.5) * H$$

should be replaced with

$$XR = Z(I)$$

$$XR = ZH(I)$$

respectively, and the remaining eight statements of the form

$$XR=(X(J)+I-1)*H$$

$$XR=(X(J)+I-.5)*H$$

should be replaced with

$$XR=Z(NPTL(J-1)+I-1)$$

$$XR=ZH(NPTL(J-1)+I-1).$$

VII. SOURCE LISTING OF HATS

```

1. C.....MAIN PROGRAM
2.     INTEGER X
3.     COMMON /BLOCK1/H,NL,NTGP,NPTL(5),IFRG,ITM,IT,ISCAT,IOPT
4.     COMMON /BLOCK2/A(33),B(33),AP(33),BP(33),DT(33),AH(33),BH(33),
5.     1      APH(33),BPH(33),DTH(33),AI(33),BI(33),DTI(33),
6.     2      DP(33),EBI(33),AR(5),AL(5),BR(5),BL(5)
7.     COMMON /BLOCK3/X(6),C(6),DPI(33),DPBI(33),EAMB(5,33),EAPB(5,33),
8.     1      QCLD(33),QNEW(33),XP(5,129),XH(5,129),YP(5,129),
9.     2      YM(5,129),XPC(5),XMC(5),YPC(5),YMC(5),APBI(5),
10.    3      AMBI(5),P(5,2,33,33)
11.    COMMON /BLOCK4/JV(5,16),JW(5,16),SV(5,16),SW(5,16),V(5,2001,2),
12.    1      W(5,129,2),VS(129,2),WS(129,2)
13.    COMMON /BLOCK5/BUFF(2201),UI(2001),U(2001),UR(2001)
14.    READ (5,2000) N, NL, IFRQ, ITM, ISCAT, IOPT, (X(I), I = 1,6)
15.    READ (5,3000) (C(I), I = 1,6)
16.    NTGP = N + 1
17.    H = 1.0 / N
18.    CALL PRNT(1)
19.    CALL KERNEL
20.    CALL DELTA
21.    CALL INCWAV
22.    CALL FIELD
23.    CALL PRNT(6)
24.    STOP
25.    2000 FORMAT (6I5,/,6I5)
26.    3000 FORMAT (6E12.5)
27.    END

```

```

1. SUBROUTINE EVAL(LAYER,ICK)
2. C.....INSERT FUNCTION STATEMENTS FOR EACH LAYER
3.     A1(X) = 0.0 + 0. * X
4.     B1(X) = 0.0 + 0. * X
5.     A1P(X) = 0.0 + 0. * X
6.     B1P(X) = 0.0 + 0. * X
7.     A2(X) = 0.0 + 0. * X
8.     B2(X) = 0.0 + 0. * X
9.     A2P(X) = 0.0 + 0. * X
10.    B2P(X) = 0.0 + 0. * X
11.    A3(X) = 0.0 + 0. * X
12.    B3(X) = 0.0 + 0. * X
13.    A3P(X) = 0.0 + 0. * X
14.    B3P(X) = 0.0 + 0. * X
15.    A4(X) = 0.0 + 0. * X
16.    B4(X) = 0.0 + 0. * X
17.    A4P(X) = 0.0 + 0. * X
18.    B4P(X) = 0.0 + 0. * X
19.    A5(X) = 0.0 + 0. * X
20.    B5(X) = 0.0 + 0. * X
21.    A5P(X) = 0.0 + 0. * X
22.    B5P(X) = 0.0 + 0. * X
23. C.....
24.     INTEGER X
25.     COMMON /BLOCK1/H,NL,NTGP,NPTL(5),IFRG,ITM,IT,ISCAT,IOPT
26.     COMMON /BLOCK2/A(33),B(33),AP(33),BP(33),DT(33),AH(33),BH(33),
27.     1      APH(33),BPH(33),DTH(33),AI(33),BI(33),DTI(33),
28.     2      DP(33),EBI(33),AR(5),AL(5),BR(5),BL(5)

```



```

29.      COMMON /BLOCK3/X(6),C(6),DPI(33),DPBI(33),EAMB(5,33),EAPB(5,33),
30.      1      GOLD(33),QNEW(33),XP(5,129),XM(5,129),YP(5,129),
31.      2      YM(5,129),XPC(5),XMC(5),YPC(5),YMC(5),APBI(5),
32.      3      AMBI(5),P(5,2,33,33)
33.      HS = H / 6.
34.      NP1 = NPTL(LAYER)
35.      IF (ICK.EQ.2) GO TO 70
36.      IF (LAYER - 2) 10, 20, 30
37.      10 DO 15 I = 1, NP1
38.          XR = (I - 1.) * H
39.          A(I) = A1(XR)
40.          B(I) = B1(XR)
41.          AP(I) = A1P(XR)
42.          BP(I) = B1P(XR)
43.          DT(I) = (B(I) ** 2 - A(I) ** 2) / 4.0
44.          XR = (I - .5) * H
45.          AH(I) = A1(XR)
46.          BH(I) = B1(XR)
47.          APH(I) = A1P(XR)
48.          BPH(I) = B1P(XR)
49.          DTH(I) = (BH(I) ** 2 - AH(I) ** 2) / 4.0
50.      15 CONTINUE
51.      GO TO 67
52.      20 DO 25 I = 1, NP1
53.          XR = (X(2) + I - 1.) * H
54.          A(I) = A2(XR)
55.          B(I) = B2(XR)
56.          AP(I) = A2P(XR)
57.          BP(I) = B2P(XR)
58.          DT(I) = (B(I) ** 2 - A(I) ** 2) / 4.0
59.          XR = (X(2) + I - .5) * H
60.          AH(I) = A2(XR)
61.          BH(I) = B2(XR)
62.          APH(I) = A2P(XR)
63.          BPH(I) = B2P(XR)
64.          DTH(I) = (BH(I) ** 2 - AH(I) ** 2) / 4.0
65.      25 CONTINUE
66.      GO TO 67
67.      30 IF (LAYER - 4) 40, 50, 60
68.      40 DO 45 I = 1, NP1
69.          XR = (X(3) + I - 1.) * H
70.          A(I) = A3(XR)
71.          B(I) = B3(XR)
72.          AP(I) = A3P(XR)
73.          BP(I) = B3P(XR)
74.          DT(I) = (B(I) ** 2 - A(I) ** 2) / 4.0
75.          XR = (X(3) + I - .5) * H
76.          AH(I) = A3(XR)
77.          BH(I) = B3(XR)
78.          APH(I) = A3P(XR)
79.          BPH(I) = B3P(XR)
80.          DTH(I) = (BH(I) ** 2 - AH(I) ** 2) / 4.0
81.      45 CONTINUE
82.      GO TO 67
83.      50 DO 55 I = 1, NP1
84.          XR = (X(4) + I - 1.) * H
85.          A(I) = A4(XR)
86.          B(I) = B4(XR)
87.          AP(I) = A4P(XR)
88.          BP(I) = B4P(XR)

```

```

89.      DT(I) = (B(I) ** 2 - A(I) ** 2) / 4.0
90.      XR = (X(4) + I - .5) * H
91.      AH(I) = A4(XR)
92.      BH(I) = B4(XR)
93.      APH(I) = A4P(XR)
94.      BPH(I) = B4P(XR)
95.      DTH(I) = (BH(I) ** 2 - AH(I) ** 2) / 4.0
96.      55 CONTINUE
97.      GO TO 67
98.      60 DO 65 I = 1, NP1
99.          XR = (X(5) + I - 1.) * H
100.         A(I) = A5(XR)
101.         B(I) = B5(XR)
102.         AP(I) = A5P(XR)
103.         BP(I) = B5P(XR)
104.         DT(I) = (B(I) ** 2 - A(I) ** 2) / 4.0
105.         XR = (X(5) + I - .5) * H
106.         AH(I) = A5(XR)
107.         BH(I) = B5(XR)
108.         APH(I) = A5P(XR)
109.         BPH(I) = B5P(XR)
110.         DTH(I) = (BH(I) ** 2 - AH(I) ** 2) / 4.0
111.      65 CONTINUE
112.      67 CONTINUE
113.         AI(NP1) = 0.0
114.         BI(NP1) = 0.0
115.         DTI(NP1) = 0.0
116.         EBI(NP1) = 1.0
117.         N = NP1 - 1
118.         DO 68 K = 1, N
119.             J = NP1 - K
120.             JP = J + 1
121.             AI(J) = AI(JP) + HS * (A(J) + 4. * AH(J) + A(JP))
122.             BI(J) = BI(JP) + HS * (B(J) + 4. * BH(J) + B(JP))
123.             DTI(J) = DTI(JP) + HS * (DT(J) + 4. * DTH(J) + DT(JP))
124.             EBI(J) = EXP( BI(J) )
125.      68 CONTINUE
126.      GO TO 90
127.      70 DO 80 I = 1, NP1
128.          B(I) = - B(I)
129.          BP(I) = - BP(I)
130.          BH(I) = - BH(I)
131.          BPH(I) = - BPH(I)
132.          BI(I) = - BI(I)
133.          EBI(I) = 1.0 / EBI(I)
134.      80 CONTINUE
135.      90 CONTINUE
136.          AR(LAYER) = A(NP1)
137.          BR(LAYER) = B(NP1)
138.          AL(LAYER) = A(1)
139.          BL(LAYER) = B(1)
140.          DO 100 I = 1, NP1
141.              DP(I) = DT(I) - (AP(I) - BP(I)) / 2.
142.      100 CONTINUE
143.      RETURN
144.      END

```

```

1.      SUBROUTINE INC#AV
2.      C.....INSERT FUNCTION STATEMENT FOR INCIDENT FIELD
3.      UINC(X) = 0.0 + 0. * X
4.      C.....
5.      COMMON /BLOCK1/H,NL,NTGP,NPTL(5),IFRQ,ITM,IT,ISCAT,IOPT
6.      COMMON /BLOCK5/BUFF(2201),UI(2001),U(2001),UR(2001)
7.      H2 = H * 2.
8.      ITM1 = ITM + 1
9.      DO 10 J = 1, ITM1
10.     X = - (J - 1) * H2
11.     UI(J) = UINC(X)
12. 10 CONTINUE
13.     RETURN
14.     END

```

```

1.      SUBROUTINE PRNT(K)
2.      INTEGER X
3.      COMMON /BLOCK1/H,NL,NTGP,NPTL(5),IFRQ,ITM,IT,ISCAT,IOPT
4.      COMMON /BLOCK2/A(33),B(33),AP(33),BP(33),DT(33),AH(33),BH(33),
5.      1      APH(33),BPH(33),DTH(33),AI(33),BI(33),DI(33),
6.      2      DP(33),EBI(33),AR(5),AL(5),BR(5),BL(5)
7.      COMMON /BLOCK3/X(6),C(6),DPI(33),DPBI(33),EAMB(5,33),EAPB(5,33),
8.      1      QQLD(33),QNEW(33),XP(5,129),XM(5,129),YP(5,129),
9.      2      YM(5,129),XPC(5),XMC(5),YPC(5),YMC(5),APBI(5),
10.     3      AMBI(5),P(5,2,33,33)
11.     COMMON /BLOCK4/JV(5,16),JW(5,16),SV(5,16),SW(5,16),V(5,2001,2),
12.     1      W(5,129,2),VS(129,2),WS(129,2)
13.     COMMON /BLOCK5/BUFF(2201),UI(2001),U(2001),UR(2001)
14.     IF (K - 2) 1, 2, 50
15. 1 CONTINUE
16.     C.....PRINT OUT INPUT DATA FROM LINES 1, 2, 3
17.     N = NTGP - 1
18.     NLL = NL + 1
19.     WRITE (6,2000) N, NL, IFRQ, ITM, ISCAT, IOPT, (X(I), I = 1, NLL)
20.     WRITE (6,3000) (C(I), I = 1, NLL)
21.     C      WRITE (8,2000) N, NL, IFRQ, ITM
22.     RETURN
23. 2 CONTINUE
24.     RETURN
25. 50 CONTINUE
26.     IF (K - 4) 3, 4, 60
27. 3 CONTINUE
28.     C.....PRINT INCIDENT, REFLECTED AND TRANSMITTED FIELDS IF ISCAT = 1
29.     IF (ISCAT.EQ. 0) RETURN
30.     ITM1 = ITM + 1
31.     WRITE (6,5000)
32.     WRITE (6,1000) (UI(I), I = 1, ITM1)
33.     WRITE (6,5100)
34.     WRITE (6,1000) (UR(I), I = 1, ITM1)
35.     WRITE (6,5200)
36.     WRITE (6,1000) (V(NL,1,1), I = 1, ITM1)
37.     C      WRITE (8,1500) (V(NL,1,1), I = 1, ITM1)
38.     C      WRITE (8,1500) (UR(I), I = 1, ITM1)
39.     RETURN
40. 4 CONTINUE
41.     C.....PRINT FIELD AT ALL GRID POINTS INSIDE MEDIUM AT TIME TAU = IT
42.     WRITE (6,7000) IT
43.     WRITE (6,1000) (U(I), I = 1, NTGP, 2)

```

```

44. C      WRITE (8,6000) IT
45. C      WRITE (8,1500) (U(I),I = 1, NTGP, 2)
46.      RETURN
47.      50 CONTINUE
48.      IF (K - 6) 5, 6, 6
49.      5 CONTINUE
50. C.....PRINT FIELD AT POSITION X = IT * H FOR TIMES
51. C.....TAU = IT * H, (IT + IFREQ * 2) * H, ...
52.      IMAX = 1 + (ITM - (IT / 2)) / IFREQ
53.      WRITE (6,8000) IT
54.      WRITE (6,1000) (U(I),I = 1, IMAX)
55. C      WRITE (8,6000) IT
56. C      WRITE (8,1500) (U(I),I = 1, IMAX)
57.      RETURN
58.      5 CONTINUE
59.      WRITE (6,4000)
60.      RETURN
61.      1000 FORMAT (1X,8E10.3)
62.      1500 FORMAT (3E10.3)
63.      2000 FORMAT (5I5,/,6I5)
64.      3000 FORMAT (6E12.5)
65.      4000 FORMAT (///,1X,3HFINISHED)
66.      5000 FORMAT (///,1X,14HINCIDENT FIELD,/)
67.      5100 FORMAT (///,1X,15HREFLECTED FIELD,/)
68.      5200 FORMAT (///,1X,17HTRANSMITTED FIELD,/)
69.      6000 FORMAT (110)
70.      7000 FORMAT (///,1X,29HINTERNAL FIELD AT TIME TAU = ,I10.8H * 2 * H,/)
71.      8000 FORMAT (///,1X,21HFIELD AT POSITION X = ,I10.4H * H,/)
72.      END

```

```

1.      SUBROUTINE KERNEL
2. C.....THIS SUBROUTINE SOLVES SYSTEMS OF PARTIAL DIFFERENTIAL
3. C.....EQUATIONS TO OBTAIN RIEMANN FUNCTIONS FOR EACH LAYER
4. C.....OF THE MEDIUM
5.      INTEGER X
6.      COMMON /BLOCK1/H,NL,NTGP,NPTL(5),IFREQ,ITM,IT,ISCAT,IOPT
7.      COMMON /BLOCK2/A(33),B(33),AP(33),BP(33),DT(33),AH(33),BH(33),
8.      1      APH(33),BPH(33),DTH(33),AI(33),BI(33),DTI(33),
9.      2      DP(33),EBI(33),AR(5),AL(5),BR(5),BL(5)
10.     COMMON /BLOCK3/X(6),C(6),DPI(33),DPBI(33),EAMB(5,33),EAPB(5,33),
11.     1      QOLD(33),QNEW(33),XP(5,129),XM(5,129),YP(5,129),
12.     2      YM(5,129),XPC(5),XMC(5),YPC(5),YMC(5),APBI(5),
13.     3      AMBI(5),P(5,2,33,33)
14.     DIMENSION W(33)
15. C.....NL = NUMBER OF LAYERS
16. C.....NPTL(I) = NUMBER OF GRID POINTS IN LAYER I
17.     DO 10 I = 1, NL
18.       NPTL(I) = X(I+1) - X(I) + 1
19.     10 CONTINUE
20.     H4 = H/4
21.     H5 = H/6
22.     DO 110 I = 1, NL
23.       F1 = - (C(I+1) + 1.) / 2.0
24.       F2 = (C(I+1) + 1.) / 4.0
25.       F3 = (C(I+1) - 1.) / 2.0
26.       F4 = F3 / 4.0
27.       F5 = F3 / 2.0
28.       NP1 = NPTL(I)

```

```

29.      N = NP1 - 1
30.      DO 100 ICK = 1,2
31.      CALL EVAL(I,ICK)
32.      F6 = (AR(I) - BR(I)) / 4.0
33.      D1 = F3 * (AR(I) + BR(I)) / 2.0
34.      D2 = - F2 * F6 * 4.0
35.      D3 = F2 * DP(NP1)
36.      DPI(NP1) = 0.0
37.      DPBI(NP1) = 0.0
38.      AMB = (AR(I) - BR(I)) / 2.0
39.      DO 20 K = 1, N
40.      J = NP1 - K
41.      JP = J + 1
42.      DPI(J) = DTI(J) + (A(J) - B(J)) / 2.0 - AMB
43.      OPBI(J) = OPBI(JP) + H5 * (B(J) * (DT(J) - (AP(J) - BP(J)) / 2.0)
44.      1      + 4. * BH(J) * (DTH(J) - (APH(J) - BPH(J)) / 2.0)
45.      2      + B(JP) * (DT(JP) - (AP(JP) - BP(JP)) / 2.0))
46.      20 CONTINUE
47.      C.....COMPUTE BOUNDARY VALUES
48.      DO 25 K = 1, NP1
49.      P(I,ICK,K,1) = F1 * (DTI(K) + (A(K) - B(K)) / 2.0) + D1
50.      J = NP1 + 1 - K
51.      P(I,ICK,NP1,K) = EBI(J) * (F3 * (DTI(J) + (A(J) + B(J))/2.0) + D2)
52.      25 CONTINUE
53.      C.....GENERATE SOLUTION OF PDE
54.      DO 40 NCT = 2, N
55.      LN = N + 2 - NCT
56.      KMAX = NCT - 1
57.      DN = DP(LN)
58.      DNM = DP(LN-1)
59.      EN = EBI(LN)
60.      JCLD(1) = F2 * DN/EN
61.      ENM = EBI(LN-1)
62.      JCLD(NCT) = J3 + (F4 * DPI(LN) - F6)*DPI(LN) + F5 * DPBI(LN)
63.      EQ = ENM/EN
64.      H4 = H/4.
65.      HE = H/EJ
66.      DE = DN/EN
67.      HDE = H4 * DN * EQ
68.      DET = 1. + H4 * H * DNM
69.      DO 30 J = 1, KMAX
70.      M = LN + J
71.      M1 = M - 1
72.      J1 = J + 1
73.      P(I,ICK,M1,J1) = (P(I,ICK,M,J1) - H * (HDE * P(I,ICK,M1,J) +
74.      1      EN * JCLD(J1) + ENM * JCLD(J)))/DET
75.      QNEW(J1) = (JCLD(J) + H4 * (DE * P(I,ICK,M1,J) +
76.      1      DNM * (P(I,ICK,M,J1)/ENM - HE * JCLD(J1))))/DET
77.      30 CONTINUE
78.      DO 35 J = 2,NCT
79.      JCLD(J) = QNEW(J)
80.      35 CONTINUE
81.      40 CONTINUE
82.      QNEW(1) = F2 * DNM/ENM
83.      QNEW(NP1) = D3 + (F4 * DPI(1) - F6) * DPI(1) + F5 * DPBI(1)
84.      C.....COMPUTE INTEGRAL FOR SCATTERING KERNEL
85.      F7 = EBI(1) * 2.
86.      F8 = (AL(1) - BL(1)) / 2.0
87.      W(1) = 0.

```

```

88.      W1 = F7 * QNEW(1) - F8 * P(1,ICK,1,1)
89.      DO 50 K = 2, NP1
90.      W2 = F7 * QNEW(K) - F8 * P(1,ICK,K,K)
91.      W(K) = W(K-1) + H * (W1 + W2)
92.      W1 = W2
93. 50 CONTINUE
94.      IF (ICK.EQ.2) GO TO 70
95.      F9 = EBI(1)
96.      F11 = F8 * (C(I+1) + 1.)
97.      DO 60 K = 1, NP1
98.      XP(I,K) = ( -F11 + W(K)) / EBI(1)
99.      XM(I,K) = F11 + 2. * P(I,1,K,K) - W(K)
100. 60 CONTINUE
101.      XMC(I) = XM(I,NP1) - F3 * (AL(I) + BL(I)) * EBI(1)
102.      XPC(I) = - XMC(I) / EBI(1)
103.      GO TO 90
104. 70 G1 = F3 * (AL(I) + BL(I)) * EBI(1)
105.      G2 = -F1 * (AL(I) - BL(I))
106.      DO 80 KT = 1, NP1
107.      K = NP1 - KT + 1
108.      YP(I,KT) = G1 - 2.* P(I,2,NP1,NP1) + 2.* P(I,2,K,K) + W(NP1) - W(K)
109.      YM(I,KT) = ( -G1 + 2.* P(I,2,NP1,NP1) - W(NP1) + W(K)) / EBI(1)
110. 80 CONTINUE
111.      YPC(I) = G1 - 2.* P(I,2,NP1,NP1) - G2 + W(NP1)
112.      YMC(I) = - YPC(I) / EBI(1)
113. 90 CONTINUE
114. 100 CONTINUE
115.      DO 105 J = 1, NP1
116.      EAMB(I,J) = EXP( (AI(J) + BI(J)) / 2. ) / 2.0
117.      EAPB(I,J) = EXP( (AI(J) - BI(J)) / 2. ) / 2.0
118. 105 CONTINUE
119.      BR(I) = - BR(I)
120.      BL(I) = - BL(I)
121.      APSI(I) = 2. * EAPB(I,1)
122.      AMBI(I) = 2. * EAMB(I,1)
123. 110 CONTINUE
124.      RETURN
125.      END

```

```

1.      SUBROUTINE DELTA
2.      C.....THIS SUBROUTINE COMBINES THE RIEMANN FUNCTIONS FROM
3.      C.....SUBROUTINE KERNEL TO FORM THE SCATTERING KERNELS
4.      INTEGER X
5.      REAL*8 SUM1,SUM2,SUM3,SUM4
6.      COMMON /BLOCK1/H,NL,NTGP,NPTL(5),IFRQ,ITM,IT,ISCAT,IOP
7.      COMMGN /BLOCK3/X(6),C(6),DPI(33),DPBI(33),EAMB(5,33),EAPB(5,33),
8.      1          QOLD(33),QNEW(33),XP(5,129),XM(5,129),YP(5,129),
9.      2          Y4(5,129),XPC(5),XMC(5),YPC(5),YMC(5),APBI(5),
10.     3          AMBI(5),P(5,2,33,33)
11.     COMMON /BLOCK4/JV(5,16),JW(5,16),SV(5,16),SW(5,16),V(5,2001,2),
12.     1          W(5,129,2),VS(129,2),WS(129,2)
13.     N = NL
14.     NTGP = 0
15.     DO 10 I = 1, NL
16.     NTGP = NTGP + NPTL(I)
17. 10 CONTINUE
18.     NTGP = NTGP - NL + 1
19.     NTGP1 = NTGP - 1
20.     DO 30 I = 1, NL
21.     NP2 = NPTL(I) + 1
22.     DO 20 K = NP2, NTGP
23.     XP(I,K) = XPC(I)
24.     XM(I,K) = XMC(I)
25.     YP(I,K) = YPC(I)
26.     YM(I,K) = YMC(I)
27. 20 CONTINUE
28. 30 CONTINUE
29.     C.....CALCULATE LOCATION AND STRENGTH OF SINGULARITIES
30.     NM1 = N - 1
31.     JV(N,1) = 0
32.     JW(N,1) = 2 * X(N + 1)
33.     SV(N,1) = AMBI(N) * (C(N + 1) + 1.) / 2.
34.     SW(N,1) = - APBI(N) * (C(N + 1) - 1.) / 2.
35.     IF (N.EQ.1) GO TO 70
36.     DO 60 IC = 1, NM1
37.     NCL = N - IC
38.     NCK = NCL + 1
39.     NJ = 2 ** (N - NCL)
40.     NJ2 = NJ / 2
41.     DO 40 I = 1, NJ2
42.     JV(NCL,I) = JV(NCK,I)
43.     JW(NCL,I) = JW(NCK,I)
44.     SV(NCL,I) = AMBI(NCL) * (C(NCK) + 1.) * SV(NCK,I) / 2.
45.     SW(NCL,I) = APBI(NCL) * (C(NCK) + 1.) * SW(NCK,I) / 2.
46. 40 CONTINUE
47.     NJ3 = NJ2 + 1
48.     DO 50 I = NJ3, NJ
49.     JV(NCL,I) = 2 * X(NCK) - JW(NCK,I - NJ2)
50.     JW(NCL,I) = 2 * X(NCK) - JV(NCK,I - NJ2)
51.     SV(NCL,I) = - AMBI(NCL) * (C(NCK) - 1.) * SW(NCK,I - NJ2) / 2.
52.     SW(NCL,I) = - APBI(NCL) * (C(NCK) - 1.) * SV(NCK,I - NJ2) / 2.
53. 50 CONTINUE
54. 60 CONTINUE
55. 70 CONTINUE
56.     AP = APBI(N) / 4.0
57.     AM = AMBI(N) / 4.0
58.     DO 80 K = 1, NTGP
59.     V(N,K,1) = AM * XM(N,K)
60.     V(N,K,2) = V(N,K,1)

```

```

61.      W(N,K,1) = AP * XP(N,K)
62.      W(N,K,2) = W(N,K,1)
63.      80 CONTINUE
64.      V(N,1,1) = 0.
65.      W(N,1,1) = 0.
66.      NP1 = NPTL(N)
67.      V(N,NP1,2) = AM * XMC(N)
68.      W(N,NP1,2) = AP * XPC(N)
69.      IF (N.EQ.1) GO TO 240
70.      DO 230 IC = 1, NM1
71.      I = N - IC
72.      C.....CALCULATE INTEGRALS FOR LAYERS N-1,N-2,...2,1. IGNORING SINGULARIT
73.      NP1 = NPTL(I)
74.      NI = NP1 - 1
75.      AM = AMBI(I) / 4.
76.      AP = APBI(I) / 4.
77.      IP = I + 1
78.      CM = 2. * (C(IP) - 1.)
79.      CP = 2. * (C(IP) + 1.)
80.      V(I,1,1) = 0.
81.      V(I,1,2) = AM * (CP * V(IP,1,2) - CM * W(IP,1,2))
82.      W(I,1,1) = 0.
83.      W(I,1,2) = 0.
84.      DO 110 K = 1, NI
85.      SUM1 = 0.00
86.      SUM2 = 0.00
87.      DO 90 J = 1, K
88.      IK = K - J + 2
89.      JP = J + 1
90.      IL = IK - 1
91.      SUM1 = SUM1 + XM(I,J) * V(IP,IK,1) + XM(I,JP) * V(IP,IL,2)
92.      1      + YM(I,J) * W(IP,IK,1) + YM(I,JP) * W(IP,IL,2)
93.      SUM2 = SUM2 + XP(I,J) * V(IP,IK,1) + XP(I,JP) * V(IP,IL,2)
94.      1      + YP(I,J) * W(IP,IK,1) + YP(I,JP) * W(IP,IL,2)
95.      90 CONTINUE
96.      KP = K + 1
97.      DO 100 M = 1, 2
98.      V(I,KP,M) = AM * (CP * V(IP,KP,M) - CM * W(IP,KP,M) + M * SUM1)
99.      W(I,KP,M) = AP * M * SUM2
100.     CONTINUE
101.     110 CONTINUE
102.     W(I,NP1,2) = W(I,NP1,2) + AP * (CP * W(IP,1,2) - CM * V(IP,1,2))
103.     SUM3 = 0.00
104.     SUM4 = 0.00
105.     DO 140 K = NP1, NTGP1
106.     SUM1 = 0.00
107.     SUM2 = 0.00
108.     DO 120 J = 1, NI
109.     IK = K - J + 2
110.     JP = J + 1
111.     IL = IK - 1
112.     SUM1 = SUM1 + XM(I,J) * V(IP,IK,1) + XM(I,JP) * V(IP,IL,2)
113.     1      + YM(I,J) * W(IP,IK,1) + YM(I,JP) * W(IP,IL,2)
114.     SUM2 = SUM2 + XP(I,J) * V(IP,IK,1) + XP(I,JP) * V(IP,IL,2)
115.     1      + YP(I,J) * W(IP,IK,1) + YP(I,JP) * W(IP,IL,2)
116.     120 CONTINUE
117.     SUM3 = SUM3 + V(IP,K - NP1 + 2,1) + V(IP,K - NP1 + 1,2)
118.     SUM4 = SUM4 + W(IP,K - NP1 + 2,1) + W(IP,K - NP1 + 1,2)
119.     KP = K + 1
120.     KM = K - NP1 + 2

```



```

121.      DO 130 M = 1, 2
122.      V(I,KP,M) = AM * (CP * V(IP,KP,M) - CM * W(IP,KP,M)
123.      1      + H * (SUM1 + XMC(I) * SUM3 + YMC(I) * SUM4))
124.      W(I,KP,M) = AP * (CP * W(IP,KM,M) - CM * V(IP,KM,M)
125.      1      + H * (SUM2 + XPC(I) * SUM3 + YPC(I) * SUM4))
126.      130 CONTINUE
127.      140 CONTINUE
128.      C.....ADD IN CONTRIBUTIONS DUE TO SINGULARITIES
129.      NJ = 2 ** (N - I - 1)
130.      DO 220 K = 1, NJ
131.      JVK = JV(I + 1,K) / 2
132.      JWK = JW(I + 1,K) / 2
133.      SVK = SV(I + 1,K)
134.      SWK = SW(I + 1,K)
135.      V(I,1 - JVK,2) = V(I,1 - JVK,2) + AM * SVK * XM(I,1)
136.      W(I,1 - JVK,2) = W(I,1 - JVK,2) + AP * SVK * XP(I,1)
137.      NS = NTGP - 1 + JVK
138.      NF = 2 - JVK
139.      DO 160 KI = NF, NTGP
140.      DO 150 M = 1, 2
141.      V(I,KI,M) = V(I,KI,M) + AM * SVK * XM(I,KI + JVK)
142.      W(I,KI,M) = W(I,KI,M) + AP * SVK * XP(I,KI + JVK)
143.      150 CONTINUE
144.      160 CONTINUE
145.      IF (NS - N1) 180, 170, 170
146.      170 KKI = NP1 - JVK
147.      V(I,NP1 - JVK,2) = (V(I,NP1 - JVK,2) - AM * SVK * XM(I,NP1))
148.      1      + AM * SVK * XMC(I)
149.      W(I,NP1 - JVK,2) = (W(I,NP1 - JVK,2) - AP * SVK * XP(I,NP1))
150.      1      + AP * SVK * XPC(I)
151.      180 NS = NTGP - 1 + X(I + 1) - JWK
152.      NF = JWK - X(I + 1) + 2
153.      V(I,NF - 1,2) = V(I,NF - 1,2) + AM * SWK * YM(I,1)
154.      W(I,NF - 1,2) = W(I,NF - 1,2) + AP * SWK * YP(I,1)
155.      DO 200 KI = NF, NTGP
156.      KKI = KI + X(I + 1) - JWK
157.      DO 190 M = 1, 2
158.      V(I,KI,M) = V(I,KI,M) + AM * SWK * YM(I,KKI)
159.      W(I,KI,M) = W(I,KI,M) + AP * SWK * YP(I,KKI)
160.      190 CONTINUE
161.      200 CONTINUE
162.      IF (NS - N1) 220, 210, 210
163.      210 KKI = JWK - X(I) + 1
164.      V(I,KKI,2) = (V(I,KKI,2) - AM * SWK * YM(I,NP1)) + AM * SWK * YMC(I)
165.      W(I,KKI,2) = (W(I,KKI,2) - AP * SWK * YP(I,NP1)) + AP * SWK * YPC(I)
166.      220 CONTINUE
167.      230 CONTINUE
168.      240 CONTINUE
169.      C.....CALCULATE SCATTERING KERNELS WS AND VS
170.      CP = (C(1) + 1.) / 2.
171.      CM = (C(1) - 1.) / 2.
172.      DO 260 K = 1, NTGP
173.      DO 250 M = 1, 2
174.      VS(K,M) = - CM * V(1,K,M) + CP * W(1,K,M)
175.      WS(K,M) = CP * V(1,K,M) - CM * W(1,K,M)
176.      250 CONTINUE
177.      260 CONTINUE
178.      CALL PRNT(2)
179.      RETURN
180.      END

```

```

10.      SUBROUTINE FIELD
11.      C.....THIS SUBROUTINE COMPUTES SCATTERED AND INTERNAL FIELDS
12.      C.....THROUGH USE OF THE SCATTERING KERNELS OBTAINED IN
13.      C.....SUBROUTINE DELTA AND THE RIEMANN FUNCTIONS OBTAINED
14.      C.....IN SUBROUTINE KERNEL
15.      INTEGER X
16.      DIMENSION UT(2001),WW(129),S(32),JJ(32)
17.      COMMON /BLOCK1/H,NL,NTGP,NPTL(5),IFRQ,ITM,IT,ISCAT,IOP
18.      COMMON /BLOCK2/A(33),B(33),AP(33),BP(33),DT(33),AH(33),BH(33),
19.      1      APH(33),BPH(33),DTH(33),AI(33),BI(33),DTI(33),
20.      2      DP(33),EBI(33),AR(5),AL(5),BR(5),BL(5)
21.      COMMON /BLOCK3/X(6),C(6),DPI(33),DPBI(33),EAMB(5,33),EAPB(5,33),
22.      1      QOLD(33),QNEW(33),XP(5,129),XM(5,129),YP(5,129),
23.      2      YM(5,129),XPC(5),XMC(5),YPC(5),YMC(5),APBI(5),
24.      3      AMBI(5),P(5,2,33,33)
25.      COMMON /BLOCK4/JV(5,16),JW(5,16),SV(5,16),SW(5,16),V(5,2001,2),
26.      1      W(5,129,2),VS(129,2),WS(129,2)
27.      COMMON /BLOCK5/BUFF(2201),UI(2001),U(2001),UR(2001)
28.      EQUIVALENCE (BUFF(130),UT(1))
29.      REAL*8 S1,S2,S3,S4,S5,S6
30.      DO 10 I = 1, 2200
31.      BUFF(I) = 0.
32.      10 CONTINUE
33.      C.....COMPUTE TRANSMITTED FIELD
34.      KM = 2 ** (NL - 1)
35.      CP = (C(1) + 1.) / 2.
36.      CM = (C(1) - 1.) / 2.
37.      DO 20 K = 1, KM
38.      JJ(K) = JV(1,K) / 2.
39.      JJ(KM + K) = - JW(1,K) / 2.
40.      S(K) = SV(1,K) * CP
41.      S(KM + K) = - SW(1,K) * CM
42.      20 CONTINUE
43.      DO 30 M = 2, NTGP
44.      WW(M) = (WS(M,1) + WS(M,2)) * H
45.      30 CONTINUE
46.      WWC = 2. * H * WS(NTGP,2)
47.      DIV = S(1) + H * WS(1,2)
48.      UT(1) = 0.
49.      UT(2) = UI(2) / DIV
50.      IM = 2 ** NL
51.      DO 40 K = 2, NTGP
52.      KP = K + 1
53.      S1 = 0.00
54.      S2 = 0.00
55.      DO 40 I = 2, K
56.      S1 = S1 + WW(I) * UT(2 + K - I)
57.      40 CONTINUE
58.      DO 50 I = 2, IM
59.      S2 = S2 + S(I) * UT(KP + JJ(I))
60.      50 CONTINUE
61.      UT(KP) = (UI(KP) - S1 - S2) / DIV
62.      60 CONTINUE
63.      KF = NTGP + 1
64.      S2 = 0.00
65.      ITM1 = ITM + 1
66.      DO 90 K = KF, ITM1
67.      KP = K + 1
68.      S1 = 0.00
69.      DO 70 I = 2, NTGP

```

```

61.      S1 = S1 + WW(I) * UT(2 + K - I)
62.      70 CONTINUE
63.      S2 = S2 + UT(KP - NTGP)
64.      S3 = 0.D0
65.      DO 80 I = 2, IM
66.      S3 = S3 + S(I) * UT(KP + JJ(I))
67.      80 CONTINUE
68.      UT(KP) = (UI(KP) - S1 - WWC * S2 - S3) / DIV
69.      90 CONTINUE
70.      C.....CCMPUTE REFLECTED FIELD
71.      DO 91 K = 1, KM
72.      S(K) = - CM * S(K) / CP
73.      S(KM + K) = - CP * S(KM + K) / CM
74.      91 CONTINUE
75.      DO 92 M = 2, NTGP
76.      WW(M) = (VS(M,1) + VS(M,2)) * H
77.      92 CONTINUE
78.      WWC = 2. * H * VS(NTGP,2)
79.      S(1) = S(1) + H * VS(1,2)
80.      UR(1) = 0.
81.      UR(2) = S(1) * UT(2)
82.      DO 95 K = 2, NTGP
83.      KP = K + 1
84.      S1 = 0.D0
85.      S2 = 0.D0
86.      DO 93 I = 2, K
87.      S1 = S1 + WW(I) * UT(2 + K - I)
88.      93 CONTINUE
89.      DO 94 I = 1, IM
90.      S2 = S2 + S(I) * UT(KP + JJ(I))
91.      94 CONTINUE
92.      UR(KP) = S1 + S2
93.      95 CONTINUE
94.      S2 = 0.D0
95.      DO 98 K = KP, ITM
96.      KP = K + 1
97.      S1 = 0.D0
98.      DO 96 I = 2, NTGP
99.      S1 = S1 + WW(I) * UT(2 + K - I)
100.     96 CONTINUE
101.     S2 = S2 + UT(KP - NTGP)
102.     S3 = 0.D0
103.     DO 97 I = 1, IM
104.     S3 = S3 + S(I) * UT(KP + JJ(I))
105.     97 CONTINUE
106.     UR(KP) = S1 + S2 * WWC + S3
107.     98 CONTINUE
108.     DO 100 K = 1, ITM1
109.     V(NL,K,1) = UT(K)
110.     V(NL,K,2) = 0.
111.     100 CONTINUE
112.     CALL PRNT(3)
113.     C.....COMPUTE FIELD AT INTERFACES
114.     IF (NL.EQ.1) GO TO 160
115.     NTD = NL - 1
116.     DO 150 IL = 1, NTD
117.     NCL = NL - IL + 1
118.     NPL = NCL + 1
119.    >NNL = NCL - 1
120.     CV1 = 4*BI(NCL) * (C(NPL) + 1.) / 2.

```

```

121.      CW1 = - APBI(NCL) * (C(NPL) - 1.) / 2.
122.      CV2 = - AMBI(NCL) * (C(NPL) - 1.) / 2.
123.      CW2 =  APBI(NCL) * (C(NPL) + 1.) / 2.
124.      CV3 =  H * AMBI(NCL) / 2.
125.      CW3 =  H * APBI(NCL) / 2.
126.      CV4 =  H * XMC(NCL) * AMBI(NCL) / 4.
127.      CW4 =  H * XPC(NCL) * APBI(NCL) / 4.
128.      CV5 =  H * YMC(NCL) * AMBI(NCL) / 4.
129.      CW5 =  H * YPC(NCL) * APBI(NCL) / 4.
130.      NP1 = NPTL(NCL)
131.      NPL = NPL - 1
132.      V(NNL,1,1) = 0.
133.      V(NNL,1,2) = 0.
134.      DO 120 K = 2, NP1
135.          S1 = - XM(NCL,1) * V(NPL,K,1) / 2.
136.          S2 = - YM(NCL,1) * V(NPL,K,2) / 2.
137.          S3 = - XP(NCL,1) * V(NPL,K,1) / 2.
138.          S4 = - YP(NCL,1) * V(NPL,K,2) / 2.
139.          KM1 = K - 1
140.          DO 110 J = 1, KM1
141.              KP = K + 1 - J
142.              S1 = S1 + XM(NCL,J) * V(NPL,KP,1)
143.              S2 = S2 + YM(NCL,J) * V(NPL,KP,2)
144.              S3 = S3 + XP(NCL,J) * V(NPL,KP,1)
145.              S4 = S4 + YP(NCL,J) * V(NPL,KP,2)
146.      110 CONTINUE
147.          V(NNL,K,1) = CV1 * V(NPL,K,1) + CV2 * V(NPL,K,2) + CV3 *
148.      1          (S1 + S2)
149.          V(NNL,K,2) = CW3 * (S3 + S4)
150.      120 CONTINUE
151.          S5 = 0.00
152.          S6 = 0.00
153.          NP2 = NP1 + 1
154.          NM1 = NP1 - 1
155.          DO 140 K = NP2, ITM1
156.              KM = K - NM1
157.              S1 = (XM(NCL,1) * V(NPL,K,1) + XM(NCL,NP1) * V(NPL,KM,1)) / 2.
158.              S2 = (YM(NCL,1) * V(NPL,K,2) + YM(NCL,NP1) * V(NPL,KM,2)) / 2.
159.              S3 = (XP(NCL,1) * V(NPL,K,1) + XP(NCL,NP1) * V(NPL,KM,1)) / 2.
160.              S4 = (YP(NCL,1) * V(NPL,K,2) + YP(NCL,NP1) * V(NPL,KM,2)) / 2.
161.              DO 130 J = 2, NM1
162.                  KP = K + 1 - J
163.                  S1 = S1 + XM(NCL,J) * V(NPL,KP,1)
164.                  S2 = S2 + YM(NCL,J) * V(NPL,KP,2)
165.                  S3 = S3 + XP(NCL,J) * V(NPL,KP,1)
166.                  S4 = S4 + YP(NCL,J) * V(NPL,KP,2)
167.      130 CONTINUE
168.              S5 = S5 + V(NPL,KM,1) + V(NPL,K-NP1,1)
169.              S6 = S6 + V(NPL,KM,2) + V(NPL,K-NP1,2)
170.              V(NNL,K,1) = CV1 * V(NPL,K,1) + CV2 * V(NPL,K,2) +
171.      1          CV3 * (S1 + S2) + CV4 * S5 + CV5 * S6
172.              V(NNL,K,2) = CW1 * V(NPL,KM,1) + CW2 * V(NPL,KM,2) +
173.      1          CW3 * (S3 + S4) + CW4 * S5 + CW5 * S6
174.      140 CONTINUE
175.      150 CONTINUE
176.      160 CONTINUE
177.          H2 = H * 2.
178.          DO 170 I = 1, ITM1
179.              UT(I) = 1.
180.      170 CONTINUE

```

```

181.      IF (IOPT.EQ.1) GO TO 300
182. C.....CALCULATE FIELD INSIDE MEDIUM AT TIMES T = IT * H * 2
183.      DO 220 IT = IFRQ, ITM, IFRQ
184.      IN = NTGP
185.      IJ = IT + 1 - (NTGP - 1) / 2
186.      U(IN) = V(NL,IJ,1) * UT(IJ)
187.      DO 210 K1 = 1, NL
188.      NCL = NL - K1 + 1
189.      NPL = NCL + 1
190.      NP1 = NPTL(NCL)
191.      NP2 = NP1 + 1
192.      NM1 = NP1 - 1
193.      CM = C(NPL) - 1.
194.      CP = C(NPL) + 1.
195.      NIT = (NP1 - 1) / 2
196.      DO 200 JC = 1, NIT
197.      IN = IN - 2
198.      M = NIT - JC
199.      M2 = M * 2
200.      IJ = IT - M + 1 - X(NCL) / 2
201.      IJM = IJ - NM1 + M2
202.      S1 = (V(NCL,IJ,1) * P(NCL,1,M2 + 1,1) * UT(IJ) +
203. 1      V(NCL,IJM,1) * P(NCL,1,NP1,NP1 - M2) * UT(IJM)) / 2.
204.      S2 = (V(NCL,IJM,2) * P(NCL,2,M2 + 1,1) * UT(IJM) +
205. 1      V(NCL,IJ,2) * P(NCL,2,NP1,NP1 - M2) * UT(IJ)) / 2.
206.      IF (IJ.LE.1) GO TO 190
207.      JM = 2 * JC
208.      IF (IJM.LE.0) JM = IJ - 1
209.      DO 180 J = 2, JM
210.      M2J = M2 + J
211.      S1 = S1 + V(NCL,IJ + 1 - J,1) * P(NCL,1,M2J,J)
212.      S2 = S2 + V(NCL,IJ + 1 - J,2) * P(NCL,2,NP2 - J,NP2 - M2J)
213. 180 CONTINUE
214. 190 CONTINUE
215.      MI2 = M2 + 1
216.      U(IN) = EAMB(NCL,MI2) * (CP * V(NCL,IJ,1) * UT(IJ) -
217. 1      CM * V(NCL,IJ,2) * UT(IJ) + M2 * S1)
218.      U(IN) = U(IN) + EAPB(NCL,MI2) * (CP * V(NCL,IJM,2) * UT(IJM) -
219. 1      CM * V(NCL,IJM,1) * UT(IJM) + M2 * S2)
220. 200 CONTINUE
221. 210 CONTINUE
222.      CALL PRNT(4)
223. 220 CONTINUE
224.      RETURN
225. 300 CONTINUE
226. C.....CALCULATE FIELD INSIDE MEDIUM AT POSITION IX
227.      READ (5,1000) NOBPTS
228.      DO 390 KK = 1, NOBPTS
229.      READ (5,1000) IX
230.      IF (IX - NTGP + 1) 330, 310, 330
231. 310 CONTINUE
232.      JMAX = 1 + (ITM - 1 - (IX / 2)) / IFRQ
233.      DO 320 J = 1, JMAX
234.      U(J) = V(NL,IFRQ * (J - 1) + 1, 1)
235. 320 CONTINUE
236.      GO TO 380
237. 330 CONTINUE
238.      NLP = NL + 1
239.      DO 340 I = 2, NLP
240.      IF (IX.LE.X(1)) GO TO 350

```

```

241.      340 CONTINUE
242.      #WRITE (6,2000) IX
243.      RETURN
244.      350 CONTINUE
245.      NCL = I - 1
246.      NPL = NCL + 1
247.      NP1 = NP1(NCL)
248.      NP2 = NP1 + 1
249.      NM1 = NP1 - 1
250.      CM = C(NPL) - 1.
251.      CP = C(NPL) + 1.
252.      NIT = (NP1 - 1) / 2
253.      M = (IX - X(NCL)) / 2
254.      M2 = M * 2
255.      M12 = M2 + 1
256.      JC = NIT - M
257.      U(1) = 0.
258.      IMAX = 1 + (ITM - (IX / 2)) / IFRQ
259.      DO 370 I = 2, IMAX
260.      IJ = IFRQ * (I - 1) + 1
261.      IJM = IJ - NM1 + M2
262.      S1 = (V(NCL,IJ,1) * P(NCL,1,M2 + 1,1) * UT(IJ) +
263.      1      V(NCL,IJM,1) * P(NCL,1,NP1,NP1 - M2) * UT(IJM)) / 2.
264.      S2 = (V(NCL,IJM,2) * P(NCL,2,M2 + 1,1) * UT(IJM) +
265.      1      V(NCL,IJ,2) * P(NCL,2,NP1,NP1 - M2) * UT(IJ)) / 2.
266.      JM = 2. * JC
267.      IF (IJM.LE.0) JM = IJ - 1
268.      DO 360 J = 2, JM
269.      M2J = M2 + J
270.      S1 = S1 + V(NCL,IJ + 1 - J,1) * P(NCL,1,M2J,J)
271.      S2 = S2 + V(NCL,IJ + 1 - J,2) * P(NCL,2,NP2 - J,NP2 - M2J)
272.      360 CONTINUE
273.      M12 = M2 + 1
274.      U(1) = EAMB(NCL,M12) * (CP * V(NCL,IJ,1) * UT(IJ) -
275.      1      CM * V(NCL,IJ,2) * UT(IJ) + M2 * S1)
276.      U(1) = U(1) + EAPB(NCL,M12) * (CP * V(NCL,IJM,2) * UT(IJM) -
277.      1      CM * V(NCL,IJM,1) * UT(IJM) + M2 * S2)
278.      370 CONTINUE
279.      380 CONTINUE
280.      IT = IX
281.      CALL PRNT(5)
282.      390 CONTINUE
283.      RETURN
284.      1000 FORMAT (I5)
285.      2000 FORMAT (///.1X,9HHEY, IX =.15,14H IS TOO LARGE)
286.      END

```

VIII. SETTING UP SOME PROGRAMS

Consider a three-layer medium of depth 3 cm, with each layer being 1 cm deep. Assume that the permittivity and conductivity are constant on each layer and are given by the values shown in Figure B-3.

N is chosen to be 128. This is not for purposes of accuracy (N=64 yields identical results) but rather for obtaining more grid points, resulting in smoother plots.

Turning to the computation of the $X(i)$'s, note first that

$$x = (0.01)(\sqrt{30} + \sqrt{135} + \sqrt{60})/c$$

where c is the speed of light in free space. Then $X(1)=0$ and

$$x_1 = \sqrt{30}/(\sqrt{30} + \sqrt{135} + \sqrt{60}) = X(2)/128.$$

Thus, $X(2)=28.22 \dots$ and so, upon rounding

$$X(2)=28.$$

This is equivalent to using

$$\epsilon_1(z) = 29.4\epsilon_0 \text{ rather than } \epsilon_1(z) = 30\epsilon_0.$$

Now

$$\begin{aligned} x_2 &= (\sqrt{29.4} + \sqrt{135})/(\sqrt{29.4} + \sqrt{135} + \sqrt{60}) \\ &= X(3)/128 \end{aligned}$$

and so

$$X(3)=88$$

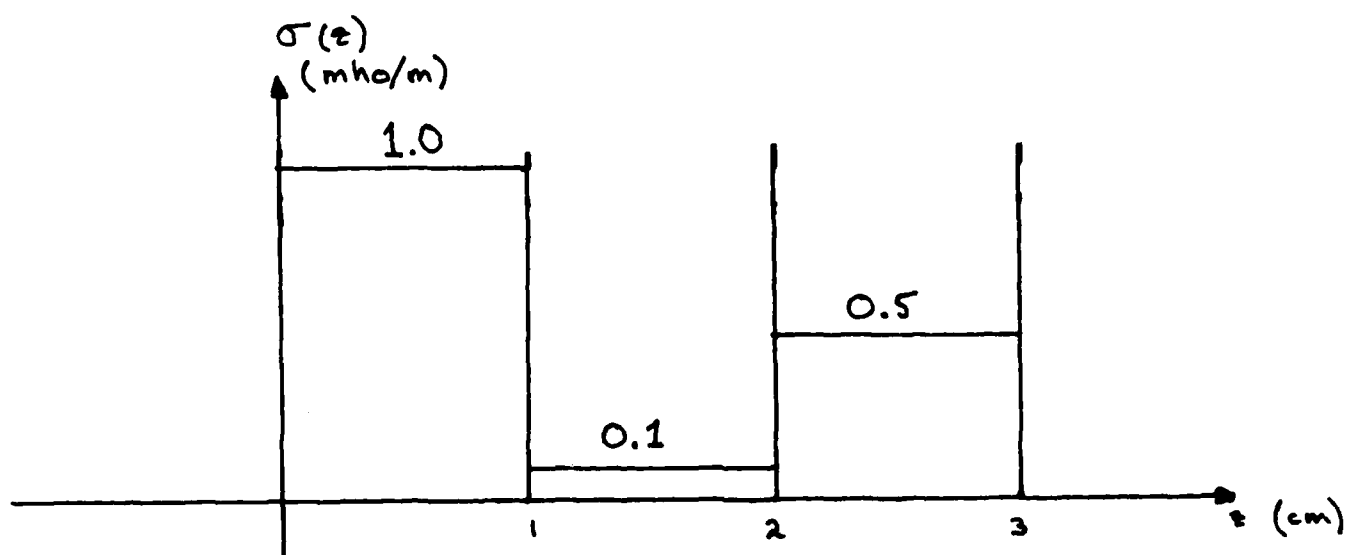
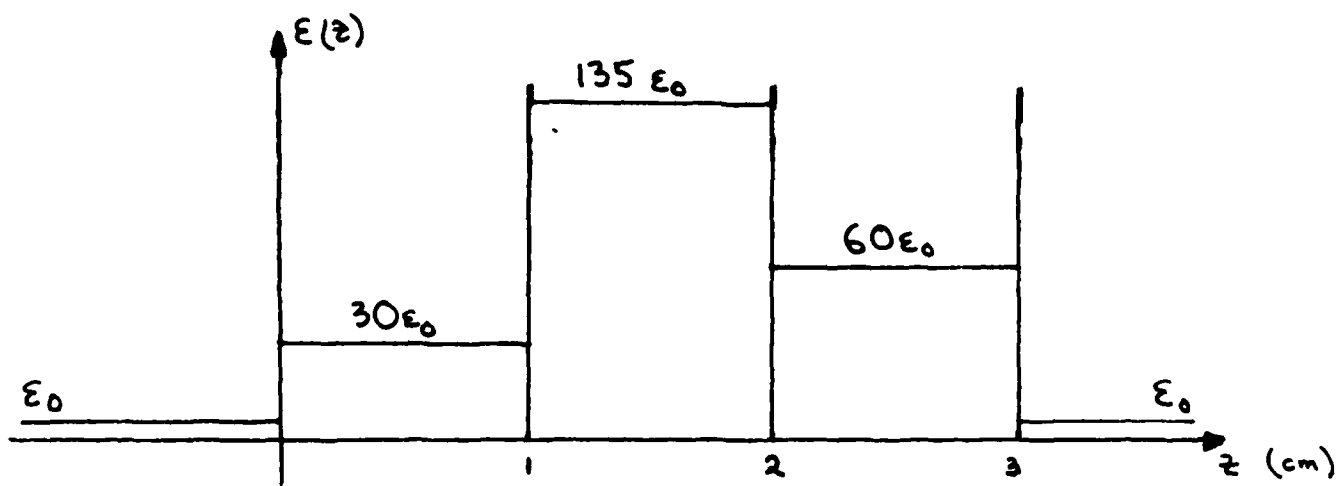


Figure B-3. Permittivity and conductivity profiles for a three-layer medium.

and

$$X(4)=128.$$

Since

$$X(4)-X(3)>32$$

$$X(3)-X(2)>32$$

two additional layers need to be inserted into the medium. (Note that if $N=64$, no additional layers are needed.) Hence, set

$$X(1)=0$$

$$X(2)=28$$

$$X(3)=60$$

$$X(4)=88$$

$$X(5)=108$$

$$X(6)=128.$$

Now

$$C(1)=\sqrt{29.4}$$

$$C(2)=\sqrt{135/29.4}$$

$$C(3)=1.0$$

$$C(4)=\sqrt{60/135}$$

$$C(5)=1.0$$

$$C(6)=\sqrt{1/60}.$$

The only nonzero functions to be entered into subroutine EVAL are

$$B1(x)=-(1.0)l/(29.4\epsilon_0)$$

$$B2(x)=B3(x)=-(0.1)l/(135\epsilon_0)$$

$$B4(x)=B5(x)=-(0.5)l/(60\epsilon_0)$$

A 1000-MHz sinusoidal incident field of duration 20 ns is used. Thus,

$$E^i(-t) = \sin(\omega t) \cdot H_e(20 \cdot 10^{-9} - t)$$

where $\omega = 2\pi \cdot 10^9$ and H_e is the Heaviside function. Since $\tau = t/l$, it follows that the incident field to be used in INCWAV is

$$UINC(\tau) = \sin(-\omega l \tau).$$

Now the τ step size is

$$\Delta\tau = 2H = 1/64$$

so it follows that the t step size is

$$\Delta t = l/64 = .01291 \text{ ns.}$$

The period of the incident signal is 1 ns which corresponds to approximately 77.5 Δt . In light of this the incident signal was modified so that its period was an even multiple of Δt . (This modification is not at all necessary and was performed solely to make numbers turn out "nice.") The period of E^i was lengthened to 78 Δt which corresponds to a 993-MHz signal. Then UINC can be written

$$UINC(\tau) = \sin(-\pi \cdot \tau / (78 \cdot H)).$$

(See Section IX for a source listing of INCWAV for this particular field.)

The duration of this new incident field was lengthened to 20.13951 ns so that 20 cycles would still be considered. Thus, ITM should be at least 1560 so that signal cutoff is observed. It was decided to print out the entire interior field every half period, so IFRO was set equal to 39 and ITM was chosen to be 1950. Setting ISCAT=1 and IOPT=0, the scattered and internal fields were put onto disk storage. These fields were then graphed using the routine given in Section X.

The above results indicate that steady state was essentially attained after six periods of the incident field; i.e., after $468 \Delta t$ intervals. It was then decided to search for the maximum intensity occurring at each grid point during the passing of the transient field (first 6 cycles) and during steady state. Thus, HATS was run a second time, with $IFRQ=1$, $ITM=468$, $ISCAT=0$, and $IOPT=0$ and with the modified subroutine PRNT given in Section XI. To determine the maximum steady state intensities, HATS was run with $IFRQ=780$ and $ITM=858$. Upon the first call to PRNT using PRNT(4), the value of $IFRQ$ was changed to 1 so that the entire period was examined in as much detail as possible. Results from these runs were plotted on the same set of axes.

From these plots it was observed that two points in the medium (one in the first layer, one in the third) had a considerably stronger transient field than steady field. Thus, a fourth run of HATS was made looking at the detailed time behavior at these points as well as at a point in the second layer which seemed to have almost no difference in transient vs. steady state intensity. Thus, three more plots were generated with the parameters used in HATS being $IFRQ=3$; $ITM=1950$; $ISCAT=0$; $IOPT=1$; $NOBPTS=3$; and $IX=20, 32, 92$.

For the sake of interest this entire procedure was pursued a second time for a nonconducting medium. This was achieved by setting $B1(x)=B2(z)=\dots=B5(x)=0$.

Finally, so that the internal reflections in the medium in the nonconducting case could be observed, a triangular incident spike field was used:

$$UI(1)=0$$

$$UI(2)=0.5$$

$$UI(3)=1.0$$

$$UI(4)=0.5$$

$$UI(5)=UI(6)=\dots=0.$$

Plots of all output appear in Section XII.

IX. SOURCE LISTING OF INCWAV FOR FIELD OF FINITE DURATION

```

1.      SUBROUTINE INCWAV
2.      C.....INSERT FUNCTION STATEMENT FOR INCIDENT FIELD
3.      UINC(X) = SIN( - PI * X / (78. * H) )
4.      C.....
5.      COMMON /BLOCK1/H,NL,NTGP,NPTL(5),IFRQ,ITM,IT,ISCAT,IDPT
6.      COMMON /BLOCK5/BUFF(2201),UI(2001),J(2001),JR(2001)
7.      PI = 3.1415927
8.      H2 = H * 2.
9.      ITM1 = ITM + 1
10.     DO 10 J = 1, ITM1
11.     UI(J) = 0.
12.     10 CONTINUE
13.     DO 20 J = 2, 39
14.     X = - (J - 1) * H2
15.     UI(J) = UINC(X)
16.     UI(39 + J) = - UI(J)
17.     20 CONTINUE
18.     DO 40 J = 2, 20
19.     JC = (J - 1) * 78
20.     DO 30 I = 2, 78
21.     UI(JC + I) = UI(I)
22.     30 CONTINUE
23.     40 CONTINUE
24.     RETURN
25.     END

```

X. SOURCE LISTING OF A PLOTTING ROUTINE

```

1.      DIMENSION X(193),A1(2100),A2(65),A3(2200),U(193)
2.      DIMENSION X0(2),X1(2),X2(2),X3(2),X4(2),U0(2),U1(2),J2(2),
3.      1      U3(2),U4(2)
4.      READ (8,2000) N, NL, IFRQ, ITM
5.      N = N / 2
6.      H2 = 1. / N
7.      NP1 = N + 1
8.      N2 = N * 2
9.      N21 = N2 + 1
10.     MN = N * 3 + 1
11.     DO 10 I = 1, MN
12.     X(I) = -1. + (I - 1) * H2
13.     10 CONTINUE
14.     DO 20 I = 1, N
15.     A1(I) = 0.
16.     20 CONTINUE
17.     DO 30 I = 1, N2
18.     A3(I) = 0.
19.     30 CONTINUE
20.     ITM1 = ITM + 1
21.     L = ITM1 + N2
22.     READ (8,1500) (A3(I), I = N21, L)
23.     L = ITM1 + N
24.     READ (8,1500) (A1(I), I = NP1, L)
25.     C.....ESTABLISH ARRAYS TO BE USED IN GRAPHING BOUNDARIES OF
26.     C.....LAYERS AND HORIZONTAL AXIS

```

```

27.      X0(1) = 0.
28.      X0(2) = X0(1)
29.      X1(1) = .21875
30.      X1(2) = X1(1)
31.      X2(1) = .6875
32.      X2(2) = X2(1)
33.      X3(1) = 1.0
34.      X3(2) = X3(1)
35.      X4(1) = -1.
36.      X4(2) = 2.
37.      U0(1) = -1.2
38.      U0(2) = 1.2
39.      U1(1) = -.75
40.      U1(2) = .75
41.      U4(1) = 0.
42.      U4(2) = 0.
43.      DO 100 IT = IFRQ, ITM, IFRQ
44.      READ (8,6000) K
45.      READ (8,1500) (A2(I), I = 1, NP1)
46.      DO 40 I = 1, NP1
47.      U(I) = A1(IT + I)
48.      U(N2 + I) = A3(NP1 + IT - I + 1)
49.      40 CONTINUE
50.      DO 50 I = 1, NP1
51.      U(N + I) = A2(I)
52.      50 CONTINUE
53.      C.....THE ARRAY U NOW CONTAINS THE REFLECTED, INTERNAL AND
54.      C.....TRANSMITTED FIELDS AT TIME TAU = IT * 2 * H
55.      C.....PLOT ARRAY U
56.      CALL GRAPH (MN, X, U, 0, 2, 6.01, 4.01..50,-1...60,-1.2,
57.      1 'NORMALIZED DEPTH:', 'NORMALIZED INTENSITY',
58.      2 ' ', ' ')
59.      C.....PLOT VERTICAL LINES SHOWING POSITION OF LAYERS
60.      CALL GRAPHS (2, X0, U0, 0, 4, ' ')
61.      CALL GRAPHS (2, X1, U1, 0, 4, ' ')
62.      CALL GRAPHS (2, X2, U1, 0, 4, ' ')
63.      CALL GRAPHS (2, X3, U0, 0, 4, ' ')
64.      C.....PLOT HORIZONTAL AXIS AT ZERO INTENSITY
65.      CALL GRAPHS (2, X4, U4, 0, 4, ' ')
66.      100 CONTINUE
67.      STOP
68.      1500 FORMAT (8E10.3)
69.      2000 FORMAT (4I5)
70.      6000 FORMAT (I10)
71.      END

```

XI. SEARCHING FOR MAXIMUM INTENSITY

The following source listing is a modification of subroutine PRNT used to search the program output for maximum intensity.

```

1.      SUBROUTINE PRNT(K)
2.      INTEGER X
3.      COMMON /BLOCK1/H,NL,NTGP,NPTL(5),IFRQ,ITM,IT,ISCAT,IOPT
4.      COMMON /BLOCK2/A(33),B(33),AP(33),BP(33),DT(33),AH(33),BH(33),
5.      1      APH(33),BPH(33),DTH(33),AI(33),BI(33),DTI(33),
6.      2      DP(33),EBI(33),AR(5),AL(5),BR(5),BL(5)
7.      COMMON /BLOCK3/X(6),C(6),DPI(33),DPBI(33),EAMB(5,33),EAPB(5,33),
8.      1      UOLD(33),ONEW(33),XP(5,129),XM(5,129),YP(5,129),
9.      2      YM(5,129),XPC(5),XMC(5),YPC(5),YMC(5),APBI(5),
10.     3      AMBI(5),P(5,2,33,33)
11.     COMMON /BLOCK4/JV(5,16),JW(5,16),SV(5,16),SW(5,16),V(5,2001,2),
12.     1      W(5,129,2),VS(129,2),WS(129,2)
13.     COMMON /BLOCK5/BUFF(2201),UI(2001),U(2001),UR(2001)
14.     DIMENSION EMAX(129),ITIME(129)
15.     IF (K - 2) 1, 2, 50
16.     1 CONTINUE
17.     C.....PRINT OUT INPUT DATA FROM LINES 1, 2, 3
18.     N = NTGP - 1
19.     WRITE (6,2000) N, NL, IFRQ, ITM, ISCAT, IOPT, (X(I), I = 1, NL)
20.     WRITE (6,3000) (C(I), I = 1, NL)
21.     C      WRITE (8,2000) N, NL, IFRQ, ITM
22.     RETURN
23.     2 CONTINUE
24.     RETURN
25.     50 CONTINUE
26.     IF (K - 4) 3, 4, 60
27.     3 CONTINUE
28.     C.....PRINT INCIDENT, REFLECTED AND TRANSMITTED FIELDS IF ISCAT = 1
29.     IF (ISCAT.EQ. 0) RETURN
30.     ITM1 = ITM + 1
31.     WRITE (6,5000)
32.     WRITE (6,1000) (UI(I), I = 1, ITM1)
33.     WRITE (6,5100)
34.     WRITE (6,1000) (UR(I), I = 1, ITM1)
35.     WRITE (6,5200)
36.     WRITE (6,1000) (V(NL,I,1), I = 1, ITM1)
37.     C      WRITE (8,1500) (V(NL,I,1), I = 1, ITM1)
38.     C      WRITE (8,1500) (UR(I), I = 1, ITM1)
39.     RETURN
40.     4 CONTINUE
41.     C.....SEARCH FOR MAXIMUM INTENSITY AT EACH GRID POINT AND RECCRD
42.     C.....THE TIME AT WHICH IT OCCURS
43.     IF (IT.GT.IFRQ) GO TO 110
44.     DO 100 I = 1, NTGP, 2
45.     EMAX(I) = 0.
46.     100 CONTINUE
47.     110 CONTINUE
48.     DO 120 I = 1, NTGP, 2
49.     ABSV = ABS( U(I) )
50.     IF (ABSV.LT.EMAX(I)) GO TO 120
51.     EMAX(I) = ABSV
52.     ITIME(I) = IT
53.     120 CONTINUE
54.     IF (IT.LT.ITM) RETURN
55.     WRITE (6,1500) (EMAX(I), I = 1, NTGP, 2)
56.     WRITE (6,5000)
57.     WRITE (6,9000) (ITIME(I), I = 1, NTGP, 2)
58.     C      WRITE (8,1500) (EMAX(I), I = 1, NTGP, 2)
59.     C      WRITE (8,9000) (ITIME(I), I = 1, NTGP, 2)
60.     RETURN

```

```

61.      60 CONTINUE
62.      IF (K - 6) 5, 6, 6
63.      5 CONTINUE
64.      C.....PRINT FIELD AT POSITION X = IT * H FOR TIMES
65.      C.....TAU = IT * H. (IT + IFRC * 2) * H. ...
66.      IMAX = 1 + (ITM - (IT / 2) ) / IFRC
67.      WRITE (6,8000) IT
68.      WRITE (6,1000) (U(I),I = 1, IMAX)
69.      C      WRITE (8,6000) IT
70.      C      WRITE (8,1500) (U(I),I = 1, IMAX)
71.      RETURN
72.      6 CONTINUE
73.      WRITE (6,4000)
74.      RETURN
75.      1000 FORMAT (1X,9E10.3)
76.      1500 FORMAT (8E10.3)
77.      2000 FORMAT (6I5,/,6I5)
78.      3000 FORMAT (6E12.5)
79.      4000 FORMAT (///,1X,8HFINISHED)
80.      5000 FORMAT (///,1X,14HINCIDENT FIELD,/)
81.      5100 FORMAT (///,1X,15HREFLECTED FIELD,/)
82.      5200 FORMAT (///,1X,17HTRANSMITTED FIELD,/)
83.      6000 FORMAT (I10)
84.      7000 FORMAT (///,1X,29HINTERNAL FIELD AT TIME TAU = .I10.8H * 2 * H,/)
85.      8000 FORMAT (///,1X,21HFIELD AT POSITION X =.I10.4H * H,/)
86.      9000 FORMAT (8I10)
87.      END

```

XII. PLOTS FROM SAMPLE PROGRAMS

Internal, reflected, and transmitted fields for mediums with nonzero and zero conductivity. Incident field is 1000-MHz pulse of duration 20 ns, impinging on the medium from the left at time $t=0$.

	<u>Nonzero conductivity</u>	<u>Zero conductivity</u>
$t=6$ ns	Figure A-4	Figure A-10
$t=21$ ns	" A-5	" A-11

Comparison of transient and steady state intensities in mediums with nonzero and zero conductivity. Incident field is 1000-MHz pulse.

<u>Nonzero conductivity</u>	<u>Zero conductivity</u>
Figure A-6	Figure A-12

Internal fields, nonzero and zero conductivity.

	<u>Nonzero conductivity</u>	<u>Zero conductivity</u>
$x=.15625$	Figure A-7	Figure A-13
$x=.25$	" A-8	" A-14
$x=.71875$	" A-9	" A-15

Reflected and internal fields, zero conductivity, spike incident field.

$t=.181$ ns	Figure A-16
$t=.361$ ns	" A-17
$t=.542$ ns	" A-18
$t=.723$ ns	" A-19

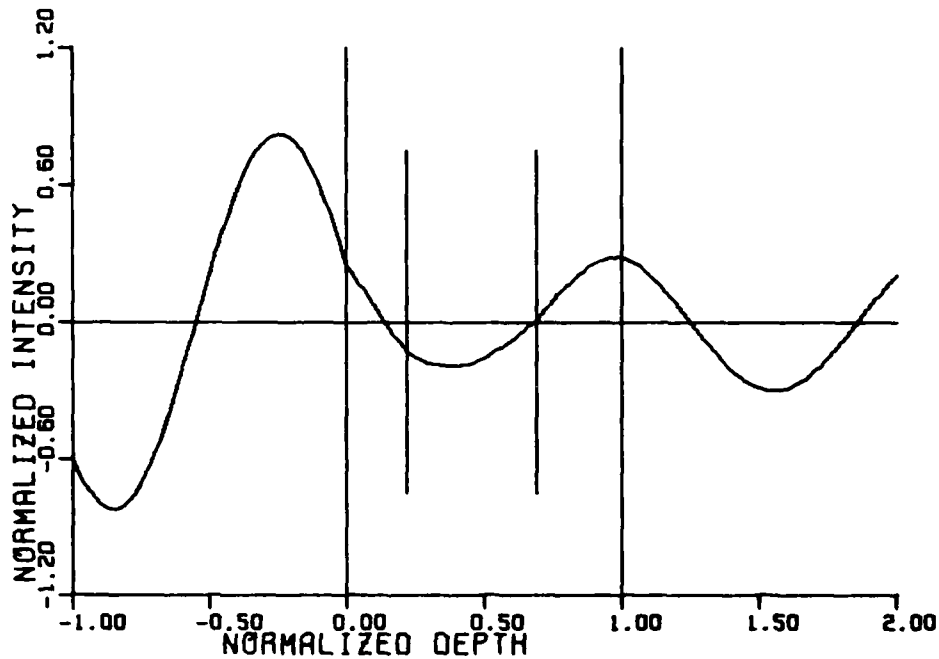


Figure B-4. Internal, reflected, and transmitted fields for a medium with nonzero conductivity, time $t=6$ ns. Incident field is 1000-MHz pulse of 20-ns duration, impinging on the medium from the left at time $t=0$.

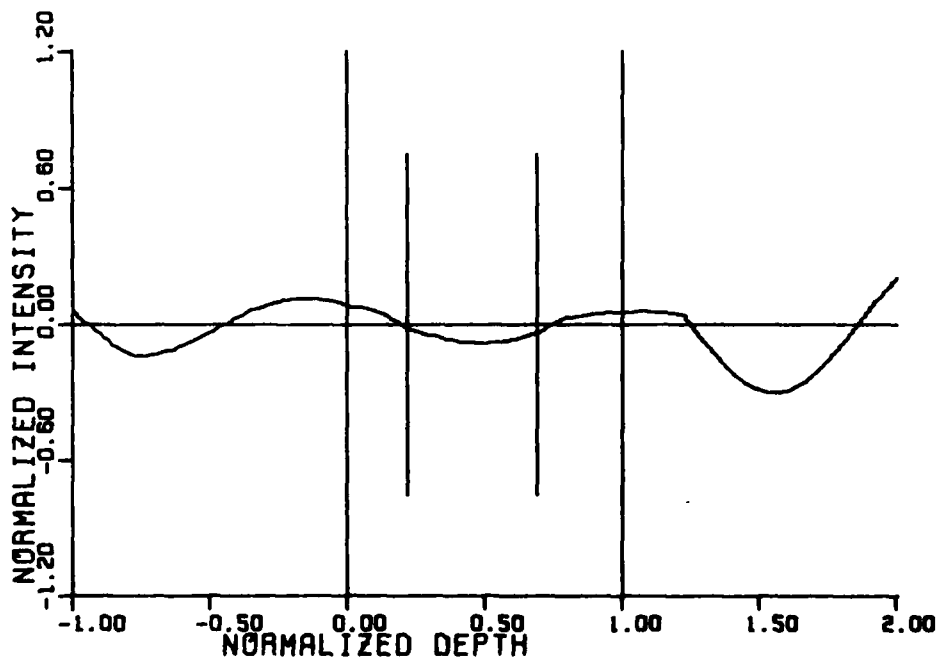


Figure B-5. Internal, reflected, and transmitted fields for a medium with nonzero conductivity, time $t=21$ ns. Incident field is 1000-MHz pulse of 20-ns duration, impinging on the medium from the left at time $t=0$.

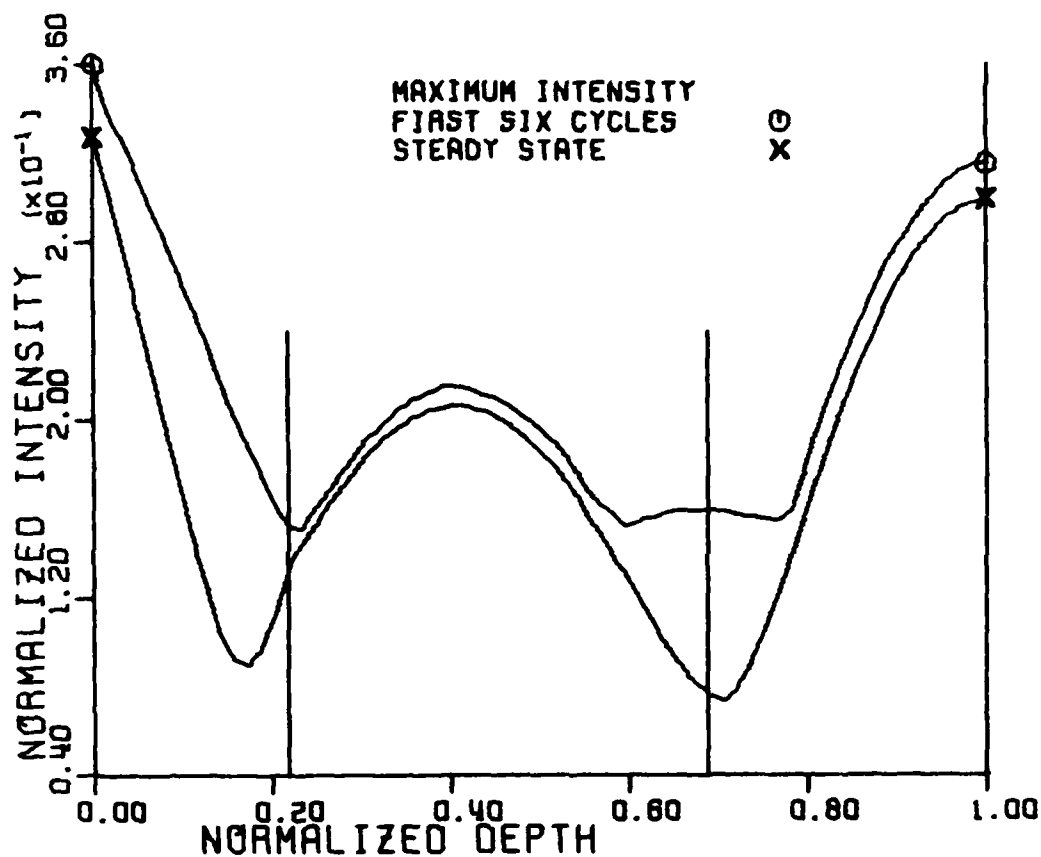


Figure B-6. Comparison of transient and steady state intensities in a medium with nonzero conductivity. Incident field is 1000-MHz pulse.

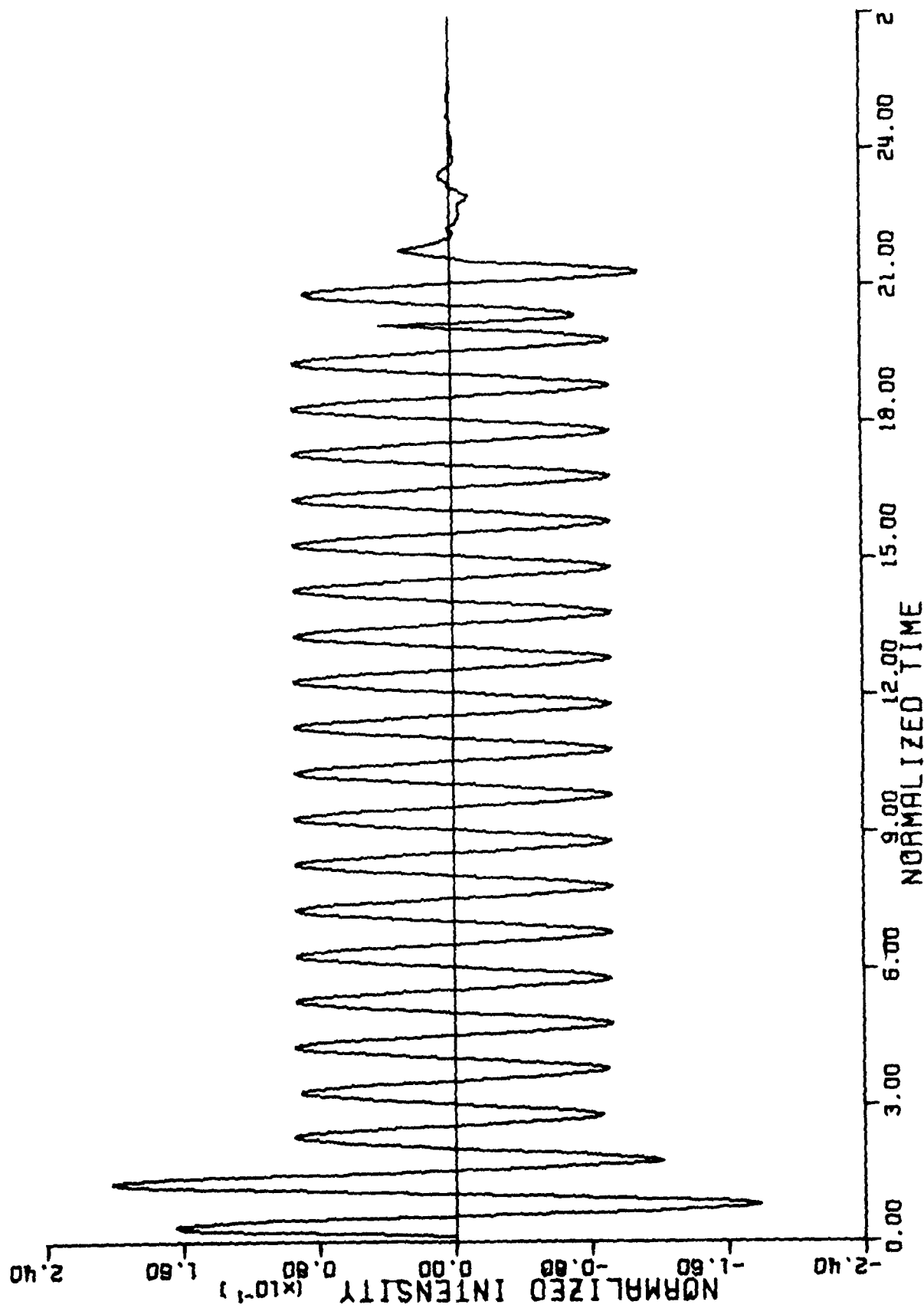


Figure B-7. Internal field ($x=.15625$), nonzero conductivity.

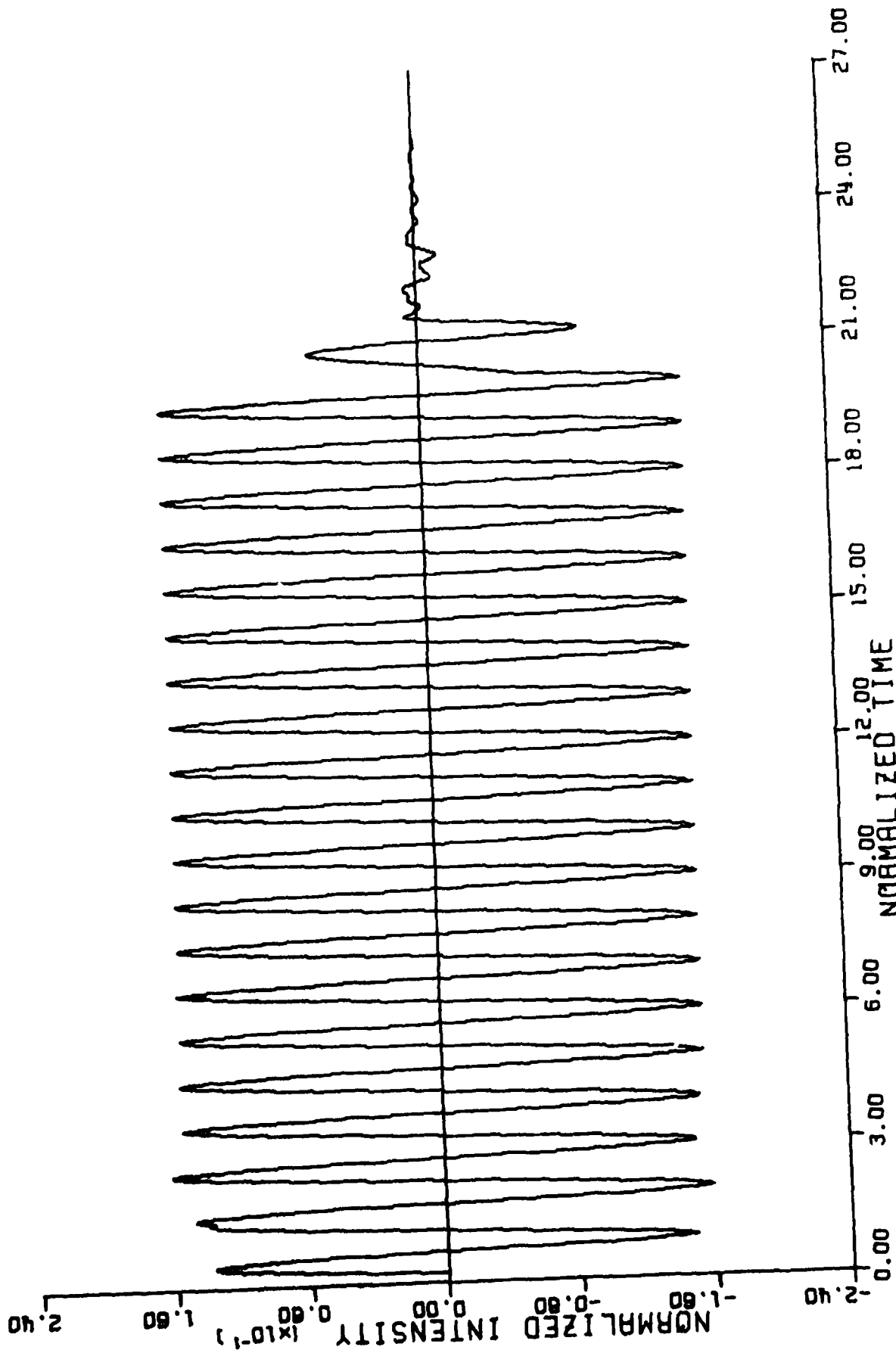


Figure B-8. Internal field ($x=.25$), nonzero conductivity.

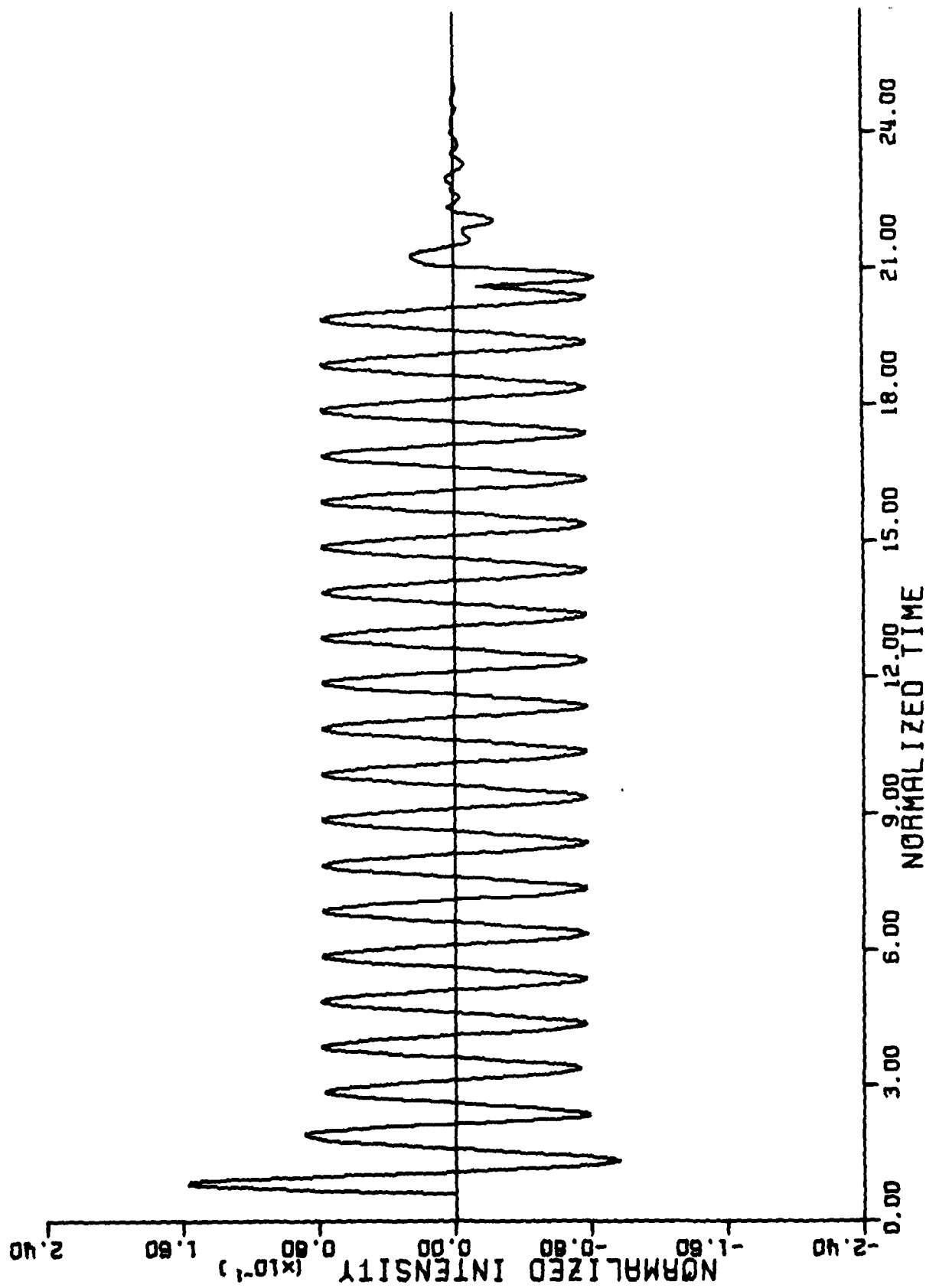


Figure B-9. Internal field ($x=.71875$), nonzero conductivity.

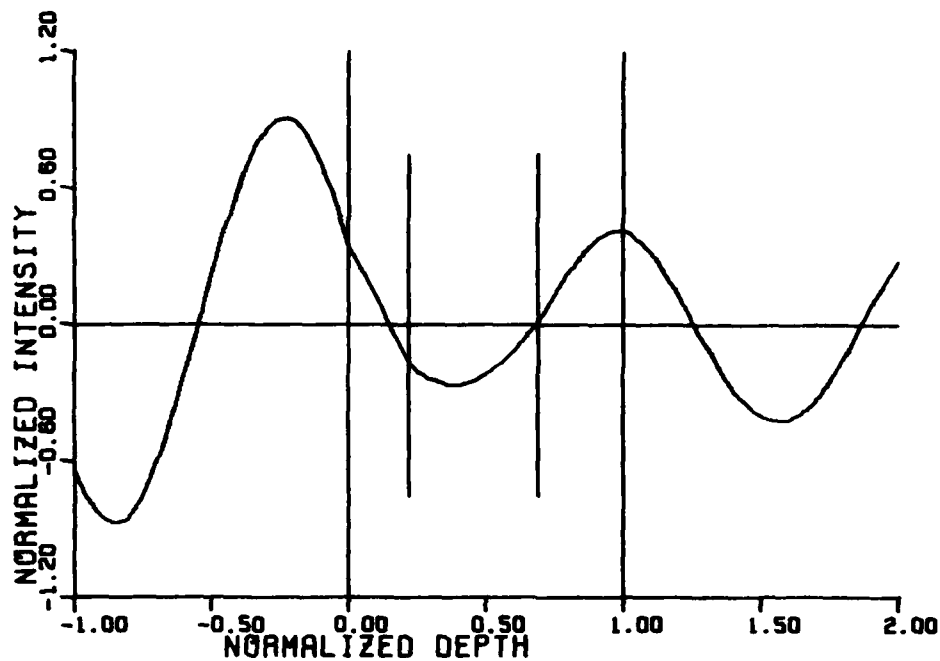


Figure B-10. Internal, reflected, and transmitted fields for a medium with zero conductivity, time $t=6$ ns. Incident field is 1000-MHz pulse of 20-ns duration, impinging on the medium from the left at time $t=0$.

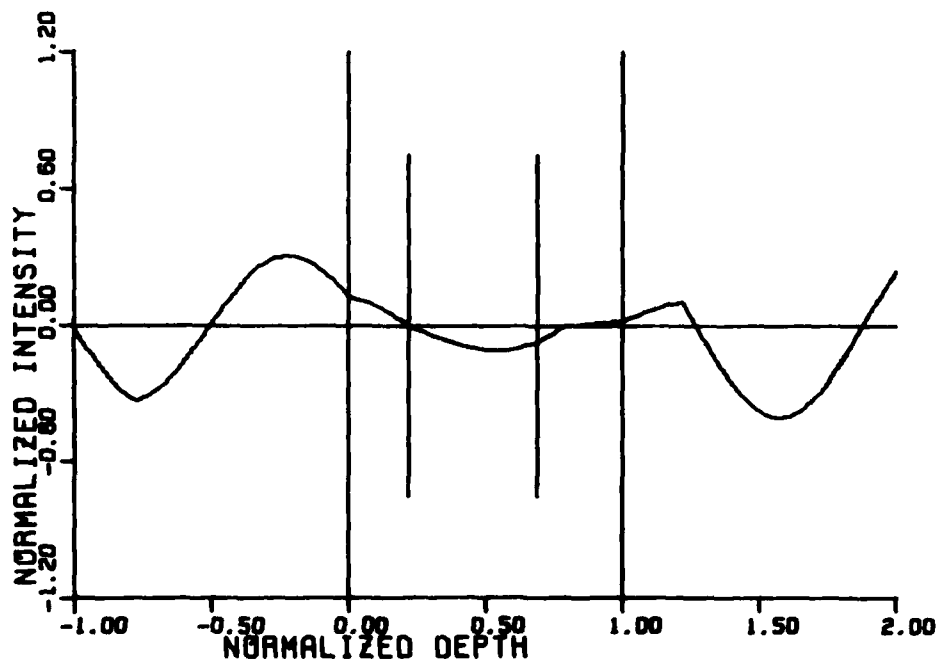


Figure B-11. Internal, reflected, and transmitted fields for a medium with zero conductivity, time $t=21$ ns. Incident field is 1000-MHz pulse of 20-ns duration, impinging on the medium from the left at time $t=0$.

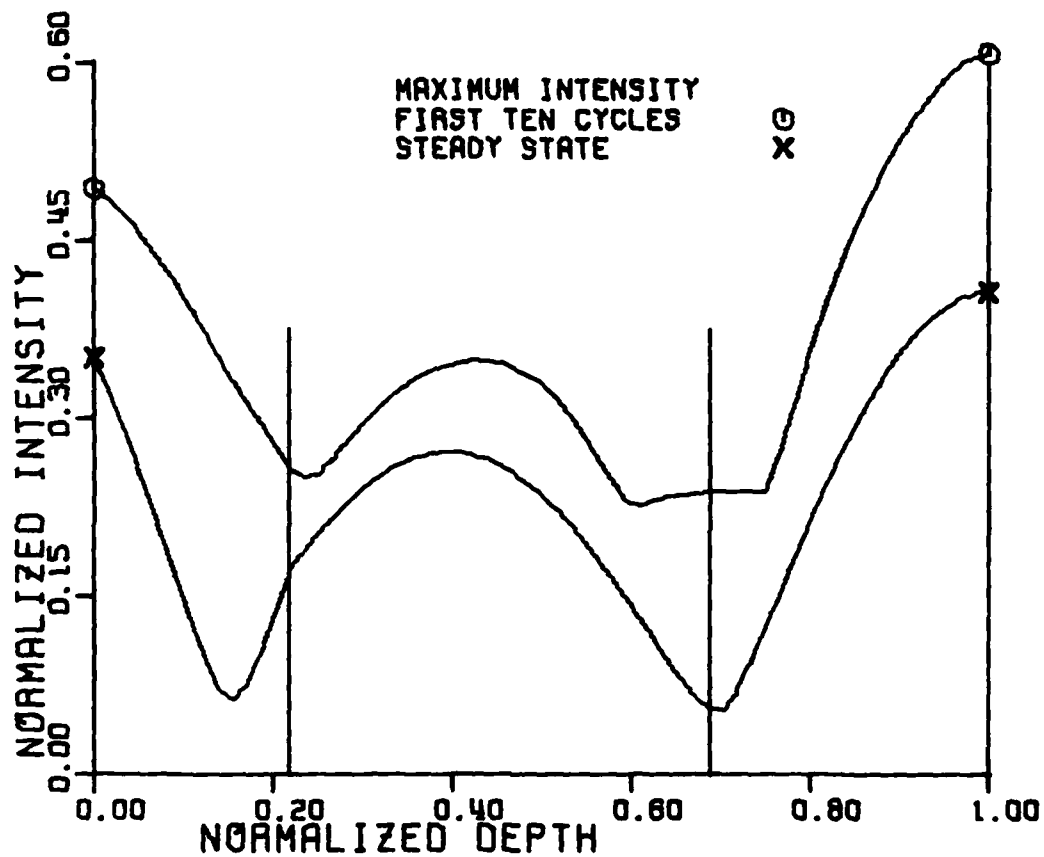


Figure B-12. Comparison of transient and steady state intensities in a medium with zero conductivity. Incident field is 1000-MHz pulse.

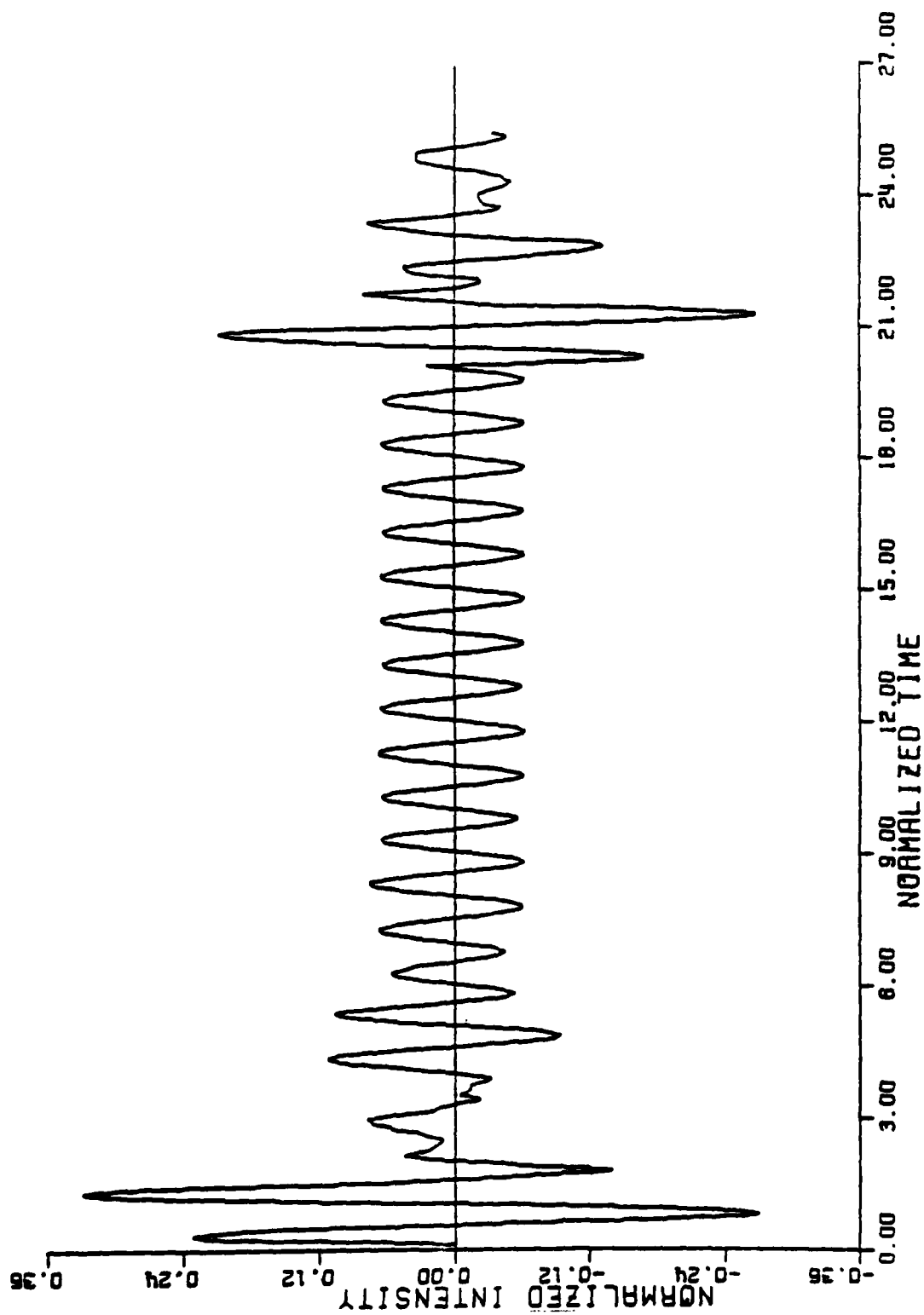


Figure B-13. Internal field ($x=0.15625$), zero conductivity.

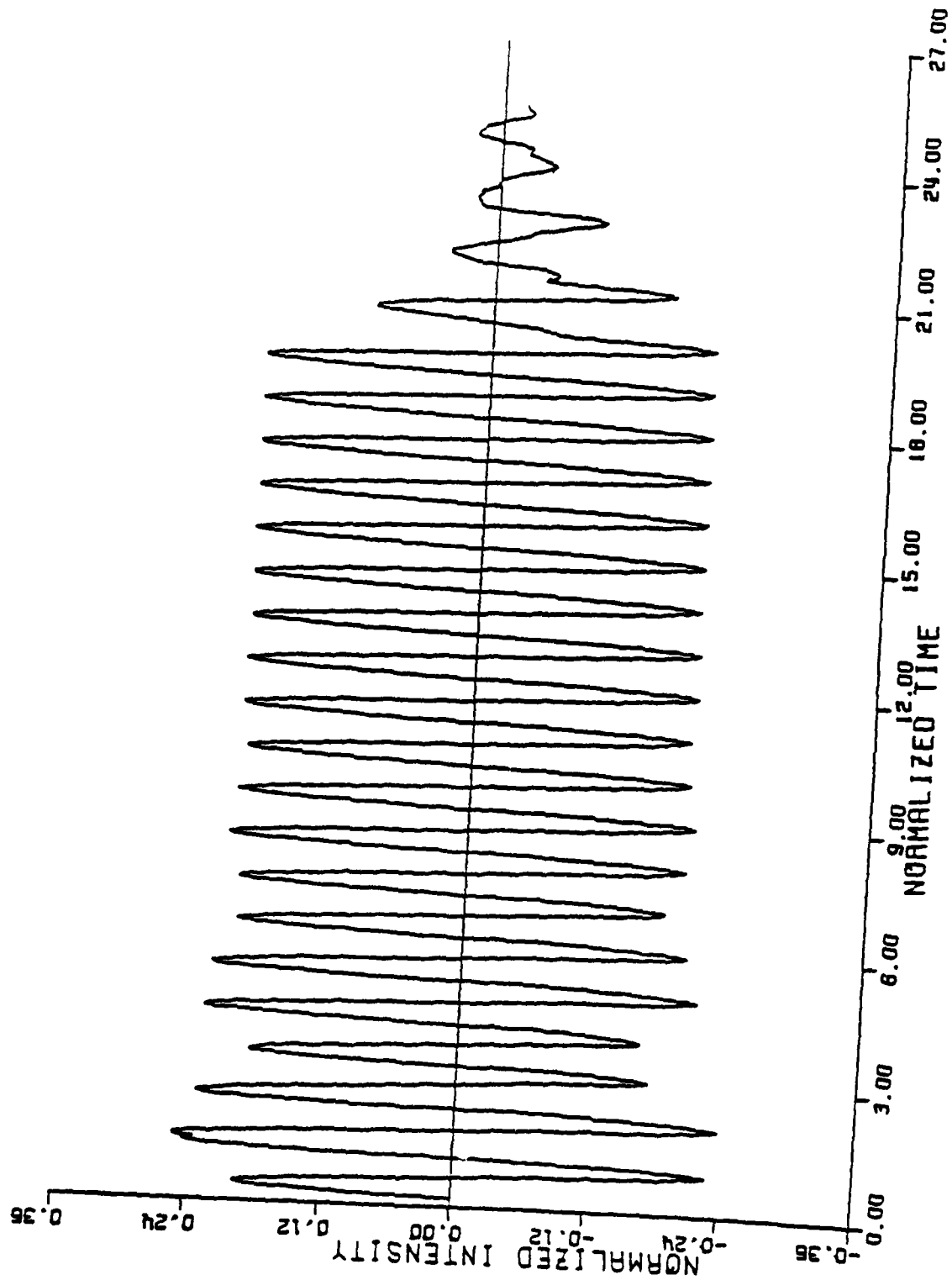


Figure B-14. Internal field ($x=0.25$), zero conductivity.

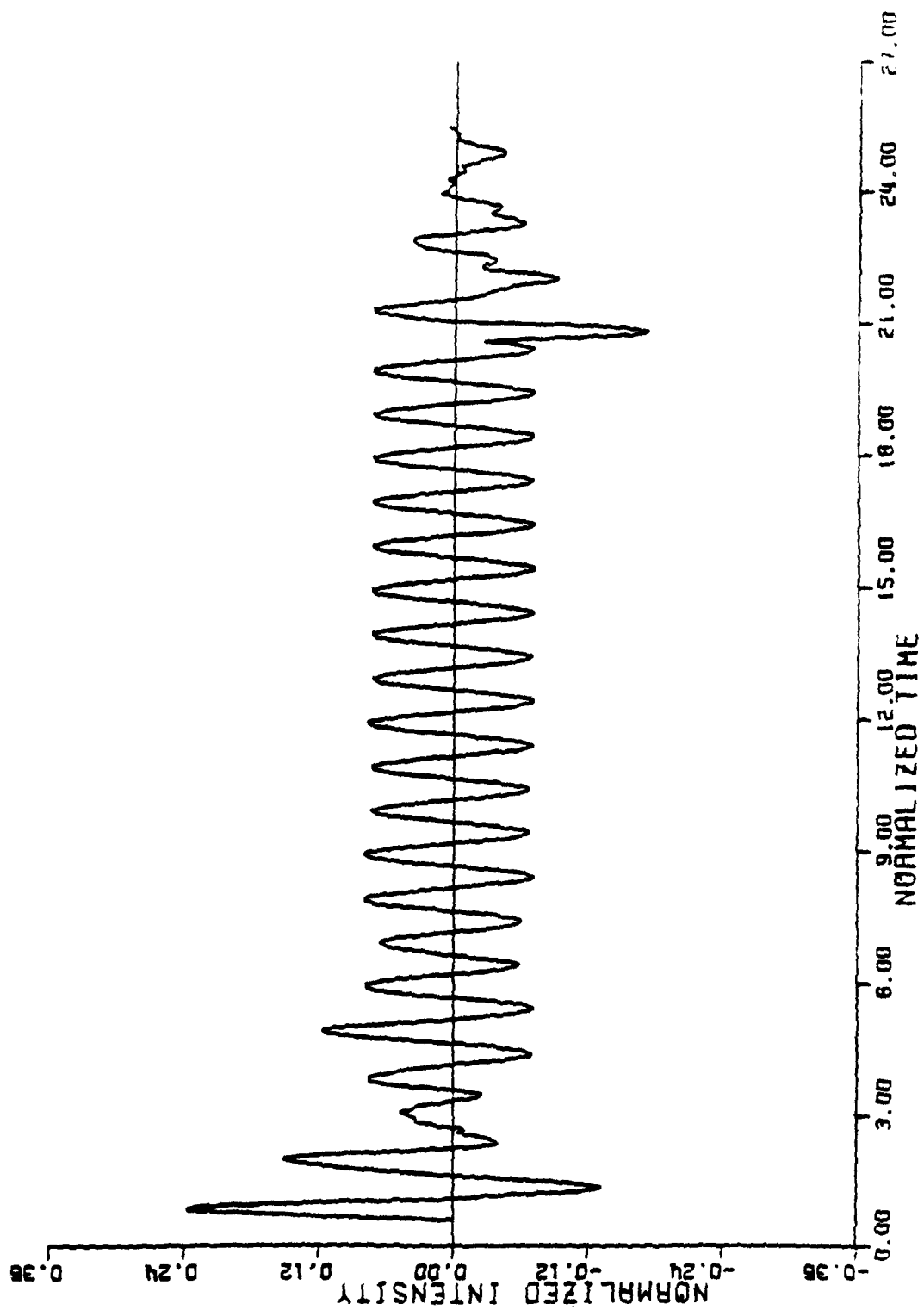


Figure B-15. Internal field ($x=.71875$), zero conductivity.

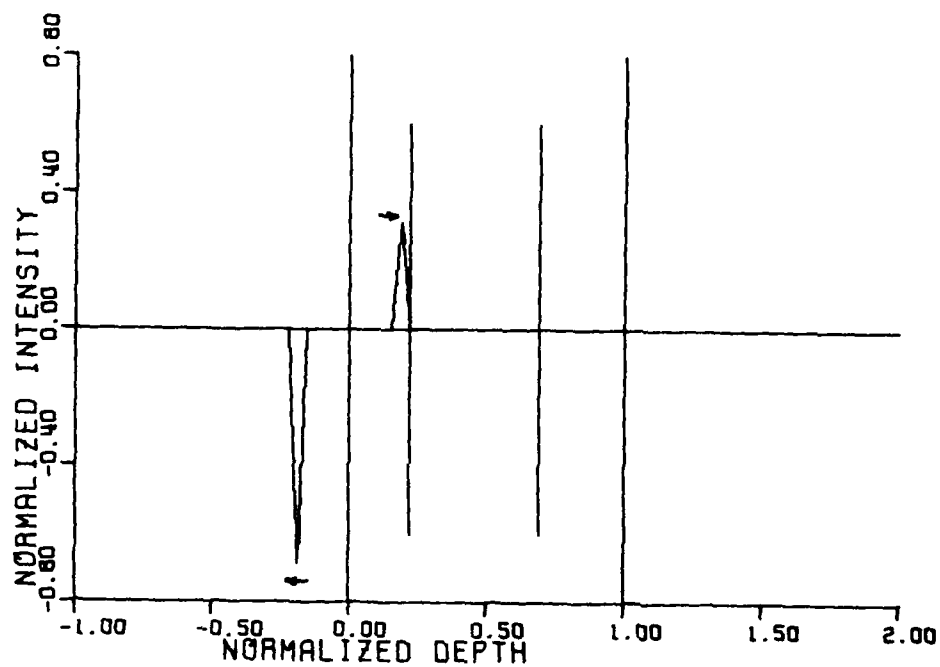


Figure B-16. Reflected and internal fields ($t=0.181$ ns), zero conductivity, spike incident field.

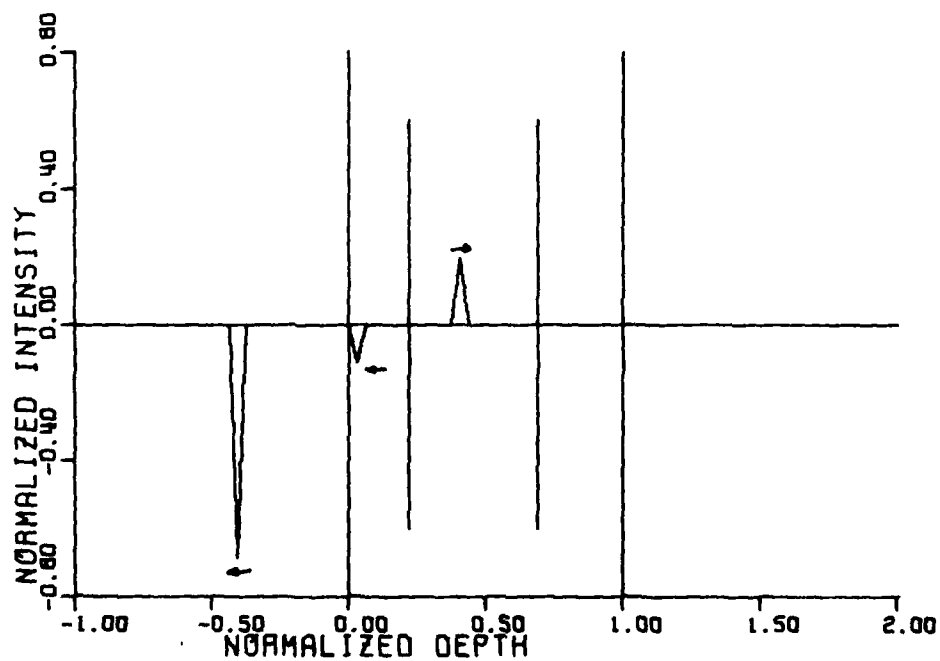


Figure B-17. Reflected and internal fields ($t=0.361$ ns), zero conductivity, spike incident field.

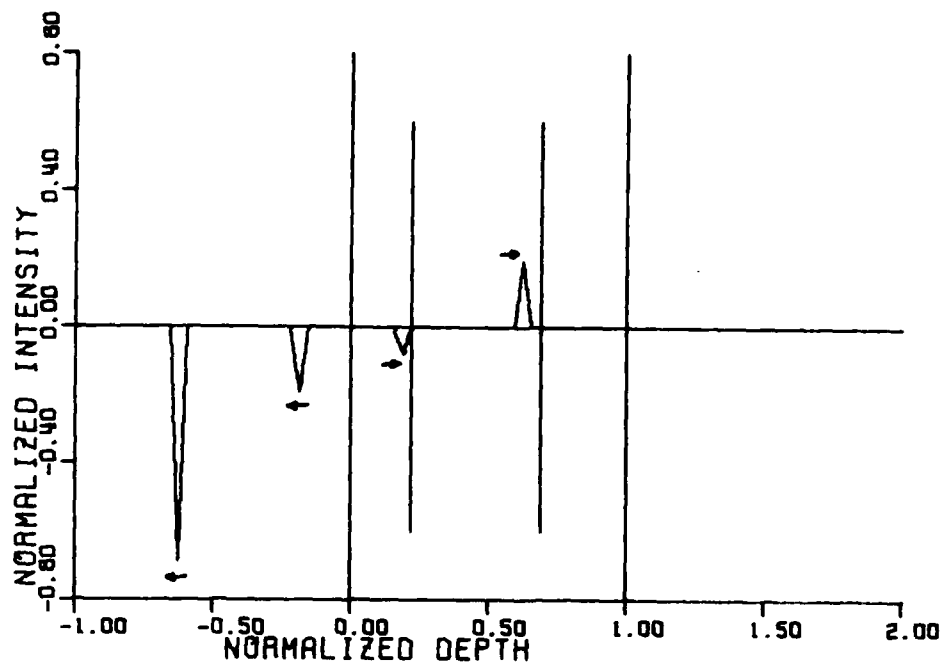


Figure B-18. Reflected and internal fields ($t=0.542$ ns), zero conductivity, spike incident field.

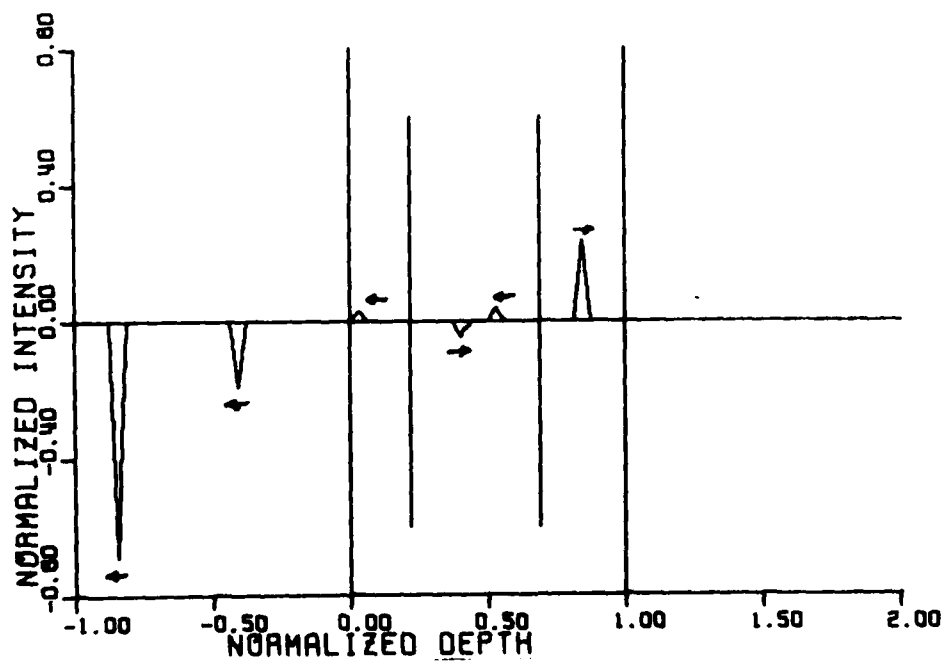


Figure B-19. Reflected and internal fields ($t=0.723$ ns), zero conductivity, spike incident field.

DATE
FILMED
-8