This is to certify that the
thesis entitled

TRANSIENT REFLECTION OF PLANE WAVES FROM A
LORENTZ-MEDIUM HALF-SPACE

presented by

STEVEN MICHAEL COSSMANN

has been accepted towards fulfillment
of the requirements for the

M.S. degree in Electrical Engineering

Major Professor's Signature

08 May 2006

Date

MSU is an Affirmative Action/Equal Opportunity Institution

PLACE IN RETURN BOX to remove this checkout from your record.
TO AVOID FINES return on or before date due.
MAY BE RECALLED with earlier due date if requested.

DATE DUE	DATE DUE	DATE DUE

TRANSIENT REFLECTION OF PLANE WAVES FROM A LORENTZ-MEDIUM HALF-SPACE

By

Steven Michael Cossmann

A THESIS

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

MASTER OF SCIENCE

Electrical and Computer Engineering

2006

ABSTRACT

TRANSIENT REFLECTION OF PLANE WAVES FROM A LORENTZ-MEDIUM HALF-SPACE

By

Steven Michael Cossmann

For materials that exhibit resonances, or have parameters that vary rapidly with frequency, the Lorentz model is becoming increasingly more popular, especially when dealing with signals with content at optical frequencies. The propagation and reflection of transient pulses by these media is of particular interest.

It has been shown previously that the transient plane-wave field reflected from a Lorentz-medium half-space can be represented as an infinite sum of fractional-order Bessel functions. In order to gain more physical insight into the physical behavior of the problem, in this thesis a closed form solution is formulated which has no infinite sums. The solution is obtained by taking an inverse Laplace transform of the frequency-domain reflection coefficient. This is accomplished by rearranging the frequency-domain reflection coefficient into separate terms which have inverse Laplace transforms which are found in standard tables.

The results obtained using the closed-form solutions are verified through comparison with an inverse fast Fourier transform of the frequency-domain reflection coefficient.

For My Parents

ACKNOWLEDGMENTS

There are many people who I would like to thank for all their help and support throughout my six years at Michigan State. First, I would like to thank my parents, Greg and Marianne. You paid my tuition throughout my undergrad career and supported me every step of the way.

I would like to thank Leo Kempel, for making sure I had work and a paycheck and helping to get me interested in electromagnetics, which may not have happened without your guidance. Thanks for convincing me to stay on for my masters degree, it has been worth the time and effort.

I would also definitely like to thank Ed Rothwell for supporting me through my masters degree and helping guide me throughout. The entire basis for this thesis was your idea and our weekly meetings have helped me to no end. It was a great advantage for me to have someone willing to give me advice whenever I needed it and who was always willing to answer questions.

And thanks Shanker B, since almost all of my graduate EM classes were taught by you and I feel like I've definitely learned something during them.

I have to thank all the guys in the office, Andrew, Brad, Dan, Gary, Greg and Jeong. It has been great working at a place with people who are easy to get along with and make work actually fun. Our weekly trips to Oodles (for those that went) kept me happy, fed and fat for 2 years.

And I would like to thank my girlfriend Alexis, for sticking with me through these years when I have had no money to take you anywhere nice.

TABLE OF CONTENTS

LIST OF FIGURES vii

CHAPTER 1
Introduction and Background 1

CHAPTER 2
Frequency Domain Reflection Coefficients 3

2.1 The Frequency Domain Wave Equation 3

2.2 Reflection from a Half-Space 5
2.2.1 TE Polarization 6
2.2.2 TM Polarization 8

CHAPTER 3
TE Plane Wave Reflection for the Optical Case 14

3.1 Laplace Domain Representation 15

3.2 Time-Domain Reflection Coefficient 17
3.2.1 Inversion of the $g_1(s)$ and $g_2(s)$ Terms 18
3.2.2 Final Expression for $\Gamma(t)$ 25

3.3 Numerical Results 26

CHAPTER 4
TM Plane Wave Reflection for the Optical Case 31

4.1 Laplace-Domain Representation 31

4.2 Time-Domain Reflection Coefficient 32
4.2.1 Inversion of the $g_n(s)$ Terms 35
4.2.2 Final expression for $G(t)$ 38
4.2.3 Inversion of the $C(s)$ Term 39

4.3 Numerical Results 41

CHAPTER 5
TE Plane Wave Reflection for the General Case 51

5.1 Laplace Domain Representation 52

5.2 Time-Domain Reflection Coefficient 54
5.2.1 Inversion of the $G(s)$ Term 56
5.2.2 Inversion of the $C(s)$ Term 58

5.3 Numerical Results 59

CHAPTER 6
TM Plane Wave Reflection for the General Case 65

6.1 Laplace Domain Representation 65

6.2 Time-Domain Reflection Coefficient 67

6.2.1 Inversion of the $G(s)$ Term 71

6.2.2 Inversion of the $C(s)$ Term 73

6.3 Numerical Results 75

CHAPTER 7
Conclusions 85

7.1 Suggestions for Future Work 85

APPENDIX A
Fortran Code for Optical TE Case 87

APPENDIX B
Fortran Code for Optical TM Case 103

APPENDIX C
Fortran Code for General TE Case 122

APPENDIX D
Fortran Code for General TM Case 141

BIBLIOGRAPHY 164

LIST OF FIGURES

Figure 2.1 Geometry for an incident wave at a discontinuity between two material regions . . . 11

Figure 2.2 Fields for a TE-polarized plane wave incident on a material half-space. . . . 12

Figure 2.3 Fields for a TM-polarized plane wave incident on a material half-space. . . . 13

Figure 3.1 Time-domain reflection coefficient with incidence angle $\theta = 30^\circ$ and material parameters $\omega_0 = 4.0 \times 10^{16} s^{-1}$, $b^2 = 20.0 \times 10^{32} s^{-2}$, $\delta = 0.28 \times 10^{16} s^{-1}$. This choice of parameters corresponds to case 1, Eq. (3.56). . . . 28

Figure 3.2 Time-domain reflection coefficient with incidence angle $\theta = 30^\circ$ and material parameters $\omega_0 = 2.0 \times 10^{15} s^{-1}$, $b^2 = 20.0 \times 10^{29} s^{-2}$, $\delta = 0.28 \times 10^{16} s^{-1}$. This choice of parameters corresponds to case 2, Eq. (3.53). . . . 29

Figure 3.3 Time-domain reflection coefficient with at an angle of $\theta = 30^\circ$ with parameter choices of $\omega_0 = 2.0 \times 10^{15} s^{-1}$, $b^2 = 20.0 \times 10^{32} s^{-2}$, $\delta = 0.28 \times 10^{16} s^{-1}$. This choice of parameters corresponds to case 3, Eq. (3.58). . . . 30

Figure 4.1 Time-domain reflection coefficient with incidence angle $\theta = 30^\circ$ and material parameters $\omega_0 = 4.0 \times 10^{16} s^{-1}$, $b^2 = 20.0 \times 10^{32} s^{-2}$, $\delta = 0.28 \times 10^{16} s^{-1}$. This choice of parameters corresponds to case 1, Eq. (4.33). . . . 44

Figure 4.2 Time-domain reflection coefficient with incidence angle $\theta = 30^\circ$ and material parameters $\omega_0 = 2.0 \times 10^{15} s^{-1}$, $b^2 = 20.0 \times 10^{29} s^{-2}$, $\delta = 0.28 \times 10^{16} s^{-1}$. This choice of parameters corresponds to case 2, Eq. (4.31). . . . 45

Figure 4.3 Time-domain reflection coefficient with at an angle of $\theta = 30^\circ$ with parameter choices of $\omega_0 = 2.0 \times 10^{15} s^{-1}$, $b^2 = 20.0 \times 10^{32} s^{-2}$, $\delta = 0.28 \times 10^{16} s^{-1}$. This choice of parameters corresponds to case 3, Eq. (4.34). . . . 46

Figure 4.4 Time-domain reflection coefficient with incidence angle $\theta = 50^\circ$ and material parameters $\omega_0 = 4.0 \times 10^{16} s^{-1}$, $b^2 = 20.0 \times 10^{32} s^{-2}$, $\delta = 0.28 \times 10^{16} s^{-1}$. This choice of parameters corresponds to case 1, Eq. (4.33) with non-causal term from Eq. (4.41). . . . 47

Figure 4.5 Non-causal $C(t) - \delta(t)$ for case 1 results. . . . 48

Figure 4.6 Non-causal $[C(t) - \delta(t)] * G(t)$ for case 1 results. . . . 49

Figure 4.7 Close-up of the non-causal $[C(t) - \delta(t)] * G(t)$ for case 1 results. . 50

Figure 5.1 Time-domain reflection coefficient with incidence angle $\theta = 30°$ and material parameters $\omega_0 = 4.0 \times 10^{11} s^{-1}$, $b = 6.24 \times 10^{11} s^{-1}$, $\delta = 2.5 \times 10^{10} s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$. This choice of parameters corresponds to case 1, Eq. (5.34). 62

Figure 5.2 Time-domain reflection coefficient with incidence angle $\theta = 30°$ and material parameters $\omega_0 = 4.0 \times 10^{10} s^{-1}$, $b = 6.24 \times 10^{10} s^{-1}$, $\delta = 2.5 \times 10^{11} s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$. This choice of parameters corresponds to case 2, Eq. (5.32). 63

Figure 5.3 Time-domain reflection coefficient with at an angle of $\theta = 30°$ with parameter choices of $\omega_0 = 4.0 \times 10^{9} s^{-1}$, $b = 6.24 \times 10^{11} s^{-1}$, $\delta = 2.5 \times 10^{10} s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$. This choice of parameters corresponds to case 3, Eq. (5.35). 64

Figure 6.1 Time-domain reflection coefficient with incidence angle $\theta = 30°$ and material parameters $\omega_0 = 4.0 \times 10^{11} s^{-1}$, $b = 6.24 \times 10^{11} s^{-1}$, $\delta = 2.5 \times 10^{10} s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$. This choice of parameters corresponds to case 1, Eq. (6.27). 78

Figure 6.2 Time-domain reflection coefficient with incidence angle $\theta = 30°$ and material parameters $\omega_0 = 4.0 \times 10^{10} s^{-1}$, $b = 6.24 \times 10^{10} s^{-1}$, $\delta = 2.5 \times 10^{11} s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$. This choice of parameters corresponds to case 2, Eq. (6.26). 79

Figure 6.3 Time-domain reflection coefficient with at an angle of $\theta = 30°$ with parameter choices of $\omega_0 = 4.0 \times 10^{9} s^{-1}$, $b = 6.24 \times 10^{11} s^{-1}$, $\delta = 2.5 \times 10^{10} s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$. This choice of parameters corresponds to case 3, Eq. (6.28). 80

Figure 6.4 Time-domain reflection coefficient with at an angle of $\theta = 60°$ with parameter choices of $\omega_0 = 4.0 \times 10^{11} s^{-1}$, $b = 6.24 \times 10^{11} s^{-1}$, $\delta = 2.5 \times 10^{10} s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$. This choice of parameters corresponds to case 1, Eq. (6.27) with a non-causal term from Eq. (6.34). 81

Figure 6.5 Non-causal $C(t)$ term for case 1 results. 82

Figure 6.6 Non-causal $C(t)$ term convolved with $G(t)$ minus the delta function term for case 1 results. 83

Figure 6.7 Non-causal $C(t)$ term convolved with the delta function term for case 1 results. 84

CHAPTER 1

INTRODUCTION AND BACKGROUND

Simple electromagnetic materials are often modeled as having constitutive parameters that are constant with frequency. More complicated materials with frequency-dependent parameters are often represented using the Debye model, which is valid for many liquids at moderate frequency. For materials that exhibit resonances, or have parameters that vary rapidly with frequency, the Lorentz model is becoming increasingly more popular, especially when dealing with signals with content at optical frequencies [1].

The propagation and reflection of short duration transient pulses in these dispersive materials has generated much interest, and the effects of dispersion on the shape of the propagating wave, particularly in the Sommerfeld and Brillouin precursors, has been extensively studied [2]-[5]. Recently, the use of short pulses to probe materials has prompted the investigation of the reflection of transient waves from material half spaces of various types [6]-[9].

The transient field reflection of a plane wave from a Lorentz-medium half-space has been calculated previously for both TE polarization [10] and TM polarization [11] by expanding the frequency-domain reflection coefficient using a simple expansion method and taking the inverse Laplace transform of each term in the infinite series. This produces a result which is given as an infinite sum of Bessel functions of fractional-order. While this method produces a convenient solution for numerical techniques, it provides little insight into the behavior of the reflected field.

This thesis proposes a method for formulating the transient reflection coefficient by rearranging the frequency-domain reflection coefficient into a form which enables the inverse Laplace transform to be taken without making an expansion as done previously. This leads to a solution containing a finite number of convolutions of Bessel functions and exponentials, which gives a clearer picture of the behavior of the reflection coefficient. Material parameter and incidence angle choices determine whether the convolution terms contain ordinary or modified Bessel functions. In order to obtain a complete solution to the problem, different sets of parameters are chosen which are representative of all the possible combinations of standard Bessel functions and modified Bessel functions. The various combinations of standard Bessel functions and modified Bessel functions allow for the general behavior of the reflected wave to be predicted from the material choices. A more oscillatory reflection would be expected from a solution containing only standard Bessel functions, while a less oscillatory reflection would be expected from a solution containing only modified Bessel functions.

With certain material parameters, there are some angles of incidence in which the final transient expression for a TM-polarized wave contains a non-causal term. For a TE-polarized wave this is not the case as long as the material is not highly magnetic. Explanation for these non-causal terms are examined in more detail as they arise in formulation.

Once the result is formulated, the transient reflection coefficient is computed numerically and then compared to the inverse fast Fourier transform (FFT).

CHAPTER 2

FREQUENCY DOMAIN REFLECTION COEFFICIENTS

In order to derive the time-domain reflection coefficient for a Lorentz-medium half-space, the frequency domain reflection coefficient must first be formulated. The most general way to accomplish this is to enforce tangential field continuity across the boundary of the half-space. In this chapter the frequency domain reflection coefficient of an obliquely incident plane wave from a material half-space is derived. This solution is valid for any homogeneous, isotropic material.

2.1 The Frequency Domain Wave Equation

Maxwell's equations for a source-free region of linear, isotropic, homogeneous material for a time-harmonic field are given in point form in terms of $\mathbf{E}(\mathbf{r},\omega)$ and $\mathbf{H}(\mathbf{r},\omega)$ as [12]

$$\nabla \times \mathbf{E} = -j\omega\mu\mathbf{H}, \tag{2.1}$$

$$\nabla \times \mathbf{H} = j\omega\epsilon\mathbf{E}, \tag{2.2}$$

$$\nabla \cdot \mathbf{E} = 0, \tag{2.3}$$

$$\nabla \cdot \mathbf{H} = 0, \tag{2.4}$$

where ϵ is the frequency dependant complex permittivity. Taking the curl of (2.1) produces

$$\begin{aligned} \nabla \times \nabla \times \mathbf{E} &= -j\omega\mu\left(\nabla \times \mathbf{H}\right) \\ &= \omega^2\mu\epsilon\mathbf{E}, \end{aligned} \tag{2.5}$$

where (2.2) has be substituted for $\nabla \times \mathbf{H}$. Utilizing both (2.3) and the vector identity $\nabla \times \nabla \times \mathbf{E} = \nabla(\nabla \cdot \mathbf{E}) - \nabla^2 \mathbf{E}$, allows (2.5) to be rewritten as

$$\nabla^2 \mathbf{E} + k^2 \mathbf{E} = 0, \tag{2.6}$$

where the wave number in the medium is given as $k = \sqrt{\mu\epsilon}$. This is the homogeneous vector Helmholtz equation for the electric field. By a similar process, the homogeneous vector Helmholtz equation for the magnetic field can be written as

$$\nabla^2 \mathbf{H} + k^2 \mathbf{H} = 0. \tag{2.7}$$

In cartesian coordinates, each component of $\mathbf{E}$ and $\mathbf{H}$ must satisfy the scalar Helmholtz equation

$$\nabla^2 \psi + k^2 \psi = 0. \tag{2.8}$$

The expression for the electric field of a plane wave can be written as

$$\mathbf{E} = \mathbf{E}_0 e^{-j\mathbf{k}\cdot\mathbf{r}}, \tag{2.9}$$

where $\mathbf{k}$ is the wave vector defined as

$$\mathbf{k} = \hat{\mathbf{x}} k_x + \hat{\mathbf{y}} k_y + \hat{\mathbf{z}} k_z, \tag{2.10}$$

with $|\mathbf{k}| = k$ and $\mathbf{r}$ is the position vector defined as

$$\mathbf{r} = \hat{\mathbf{x}} x + \hat{\mathbf{y}} y + \hat{\mathbf{z}} z. \tag{2.11}$$

The magnetic field for a uniform plane wave is related to the electric field by

$$\mathbf{H} = \frac{\mathbf{k} \times \mathbf{E}}{\omega\mu}. \tag{2.12}$$

2.2 Reflection from a Half-Space

The problem of interest is a plane wave in free space (region 1) incident on a material half-space. A portion of the wave is reflected and a portion is transmitted into the material half-space (region 2). The problem geometry can be seen in Figure 2.1. Examining the geometry, it can be seen that since the wave is propagating in the x-z plane, the field is invariant in the y-direction which means $k_y = 0$. It can also be seen from examining the geometry that the wavenumber for the incident wave is given as

$$k_{x0} = k_0 \sin\theta_i, \tag{2.13a}$$

$$k_{z0} = k_0 \cos\theta_i, \tag{2.13b}$$

and the wavenumber for the transmitted wave is given as

$$k_x = k \sin\theta_t, \tag{2.14a}$$

$$k_z = k \cos\theta_t. \tag{2.14b}$$

Any uniform plane wave incident on a planar surface may be decomposed into two orthogonal polarization components, one perpendicular and one parallel to the plane of incidence. The perpendicular polarization is also known as TE polarization, and the parallel polarization is also known as TM polarization. Each of these polarizations can be solved for separately and their solutions can be added together to form a complete solution for the total field.

2.2.1 TE Polarization

The fields for a TE-polarized plane wave incident on a material half-space can be seen in Figure 2.2. The fields for the incident plane wave in free space can be given as

$$\mathbf{E}^i_\perp = \hat{\mathbf{y}} E^i_0 e^{-jk_0(x\sin\theta_i + z\cos\theta_i)}, \tag{2.15}$$

$$\mathbf{H}^i_\perp = \frac{E^i_0}{\eta_0}\left(-\hat{\mathbf{x}}\cos\theta_i + \hat{\mathbf{z}}\sin\theta_i\right) e^{-jk_0(x\sin\theta_i + z\cos\theta_i)}. \tag{2.16}$$

were $\eta_0 = \sqrt{\mu_0/\epsilon_0}$ and $k_0 = \omega\sqrt{\mu_0\epsilon_0}$. The total field in region 1 can be expressed as the incident plane wave plus a reflected plane wave. The reflected field is given as

$$\mathbf{E}^r_\perp = \hat{\mathbf{y}} E^r_0 e^{-jk_0(x\sin\theta_r - z\cos\theta_r)}, \tag{2.17}$$

$$\mathbf{H}^r_\perp = \frac{E^r_0}{\eta_0}\left(\hat{\mathbf{x}}\cos\theta_r + \hat{\mathbf{z}}\sin\theta_r\right) e^{-jk_0(x\sin\theta_r - z\cos\theta_r)}. \tag{2.18}$$

In region 2, the area to the right of the interface, the transmitted field can be represented as a plane wave given as

$$\mathbf{E}^t_\perp = \hat{\mathbf{y}} E^t_0 e^{-jk(x\sin\theta_t + z\cos\theta_t)}, \tag{2.19}$$

$$\mathbf{H}^t_\perp = \frac{E^t_0}{\eta}\left(-\hat{\mathbf{x}}\cos\theta_t + \hat{\mathbf{z}}\sin\theta_t\right) e^{-jk(x\sin\theta_t + z\cos\theta_t)}, \tag{2.20}$$

were $\eta = \sqrt{\mu/\epsilon}$. In order to have phase continuity across the interface for all values of x, the exponential terms must be equal at the interface, $z = 0$. This implies that

$$k_0\sin\theta_i = k_0\sin\theta_r = k\sin\theta_t. \tag{2.21}$$

It can be easily deduced from this equation that

$$\theta_i = \theta_r,$$

$$\frac{\sin\theta_t}{\sin\theta_i} = \frac{k_0}{k} = \sqrt{\frac{\epsilon_0\mu_0}{\epsilon\mu}}. \tag{2.22}$$

The boundary conditions at the interface requires the tangential electric fields to be continuous across the interface. This requires

$$\hat{\mathbf{z}} \times \left(\mathbf{E}^i_\perp + \mathbf{E}^r_\perp\right)\Big|_{z=0} = \hat{\mathbf{z}} \times \mathbf{E}^t_\perp\Big|_{z=0}. \tag{2.23}$$

Using (2.23) and the fact that the phase is continuous across the interface, and using the expressions for the electric fields it can be seen that

$$E^i_0 + E^r_0 = E^t_0. \tag{2.24}$$

Dividing this by the incident field amplitude leads to

$$\begin{aligned} 1 + \frac{E^r_0}{E^i_0} &= \frac{E^t_0}{E^i_0}, \\ 1 + \Gamma_\perp &= T_\perp, \end{aligned} \tag{2.25}$$

where $\Gamma_\perp$ is the reflection coefficient and $T_\perp$ is the transmission coefficient.

The boundary conditions also require that the magnetic field be continuous across the interface. This requires

$$\hat{\mathbf{z}} \times \left(\mathbf{H}^i_\perp + \mathbf{H}^r_\perp\right)\Big|_{z=0} = \hat{\mathbf{z}} \times \mathbf{H}^t_\perp\Big|_{z=0}. \tag{2.26}$$

Using (2.26) and the fact that the phase is continuous across the interface, and using the expressions for the magnetic field it can be seen that

$$-\frac{E^i_0}{\eta_0}\cos\theta_i + \frac{E^r_0}{\eta_0}\cos\theta_r = -\frac{E^t_0}{\eta}\cos\theta_t. \tag{2.27}$$

Dividing by the incident wave amplitude again yields

$$
\begin{aligned}
-\frac{1}{\eta_0}\cos\theta_i + \frac{E_0^r}{E_0^i\eta_0}\cos\theta_r &= -\frac{E_0^t}{E_0^i\eta}\cos\theta_t, \\
-\frac{1}{\eta_0}\cos\theta_i + \frac{\Gamma_\perp}{\eta_0}\cos\theta_r &= -\frac{T_\perp}{\eta}\cos\theta_t.
\end{aligned} \tag{2.28}
$$

Combining (2.25) and (2.28) allows both $\Gamma_\perp$ and $T_\perp$ to be solved for as

$$T_\perp = \frac{2Z_\perp}{Z_\perp + Z_0}, \tag{2.29}$$

$$\Gamma_\perp = \frac{Z_\perp - Z_0}{Z_\perp + Z_0}, \tag{2.30}$$

with the definitions

$$Z_0 = \frac{\eta_0}{\cos\theta_i}, \tag{2.31}$$

$$Z_\perp = \frac{\eta}{\cos\theta_t} = \frac{k\eta}{\sqrt{k^2 - k_0^2\sin^2\theta_i}}. \tag{2.32}$$

In the above expression, using (2.22) $\cos\theta_t$ is given as

$$\cos\theta_t = \frac{\sqrt{k^2 - k_0^2\sin^2\theta_i}}{k}. \tag{2.33}$$

2.2.2 TM Polarization

For a TM-polarized plane wave, the method to formulate the reflection and transmission coefficients is similar to the method for the TE case. The fields for a TM-polarized plane wave incident on a material half-space can be seen in Figure 2.3. The fields for the incident plane wave in free-space can be given as

$$\mathbf{E}_\parallel^i = E_0^i\left(\hat{\mathbf{x}}\cos\theta_i - \hat{\mathbf{z}}\sin\theta_i\right)e^{-jk_0(x\sin\theta_i + z\cos\theta_i)}, \tag{2.34}$$

$$\mathbf{H}_{\parallel}^{i} = \hat{\mathbf{y}}\frac{E_0^i}{\eta_0}e^{-jk_0(x\sin\theta_i + z\cos\theta_i)}. \tag{2.35}$$

The reflected and transmitted fields are given as

$$\mathbf{E}_{\parallel}^{r} = E_0^r\left(\hat{\mathbf{x}}\cos\theta_r - \hat{\mathbf{z}}\sin\theta_r\right)e^{-jk_0(x\sin\theta_r - z\cos\theta_r)}, \tag{2.36}$$

$$\mathbf{H}_{\parallel}^{r} = -\hat{\mathbf{y}}\frac{E_0^r}{\eta_0}e^{-jk_0(x\sin\theta_r - z\cos\theta_r)}, \tag{2.37}$$

and

$$\mathbf{E}_{\parallel}^{t} = E_0^t\left(\hat{\mathbf{x}}\cos\theta_t - \hat{\mathbf{z}}\sin\theta_t\right)e^{-jk(x\sin\theta_t + z\cos\theta_t)}, \tag{2.38}$$

$$\mathbf{H}_{\parallel}^{t} = \hat{\mathbf{y}}\frac{E_0^t}{\eta}e^{-jk(x\sin\theta_t + z\cos\theta_t)}, \tag{2.39}$$

respectively. Just as in the TE case, enforcing phase continuity across the boundary leads to

$$\begin{aligned} \theta_i &= \theta_r, \\ \frac{\sin\theta_t}{\sin\theta_i} &= \frac{k_0}{k} = \sqrt{\frac{\epsilon_0\mu_0}{\epsilon^c\mu}}. \end{aligned} \tag{2.40}$$

Using

$$\hat{\mathbf{z}}\times\left(\mathbf{E}_{\parallel}^{i} + \mathbf{E}_{\parallel}^{r}\right)\Big|_{z=0} = \hat{\mathbf{z}}\times\mathbf{E}_{\parallel}^{t}\Big|_{z=0}, \tag{2.41}$$

yields

$$E_0^i\cos\theta_i + E_0^r\cos\theta_r = E_0^t\cos\theta_t. \tag{2.42}$$

Dividing by the incident wave amplitude leads to

$$\begin{aligned} \cos\theta_i + \frac{E_0^r}{E_0^i}\cos\theta_r &= \frac{E_0^t}{E_0^i}\cos\theta_t, \\ \cos\theta_i + \Gamma_{\parallel}\cos\theta_r &= T_{\parallel}\cos\theta_t. \end{aligned} \tag{2.43}$$

Enforcing tangential field continuity for the magnetic field across the interface requires

$$\hat{\mathbf{z}} \times \left(\mathbf{H}_{\|}^{i} + \mathbf{H}_{\|}^{r} \right) \Big|_{z=0} = \hat{\mathbf{z}} \times \mathbf{H}_{\|}^{t} \Big|_{z=0}, \tag{2.44}$$

which leads to

$$\frac{E_0^i}{\eta_0} + \frac{E_0^r}{\eta_0} = \frac{E_0^t}{\eta}. \tag{2.45}$$

Dividing by the incident wave amplitude leads to

$$\begin{aligned} \frac{1}{\eta_0} + \frac{E_0^r}{E_0^i}\frac{1}{\eta_0} &= \frac{E_0^t}{E_0^i}\frac{1}{\eta}, \\ \frac{1}{\eta_0} + \frac{\Gamma_{\|}}{\eta_0} &= \frac{T_{\|}}{\eta}. \end{aligned} \tag{2.46}$$

Combining (2.43) and (2.46) allows $\Gamma_{\|}$ and $T_{\|}$ to be solved for as

$$T_{\|} = \frac{2Z_{\|}}{Z_{\|} + Z_0}, \tag{2.47}$$

$$\Gamma_{\|} = \frac{Z_{\|} - Z_0}{Z_{\|} + Z_0}, \tag{2.48}$$

with the definitions

$$Z_0 = \eta_0 \cos\theta_i, \tag{2.49}$$

$$Z_{\|} = \eta\cos\theta_t = \frac{\eta}{k}\sqrt{k^2 - k_0^2\sin^2\theta_i}. \tag{2.50}$$

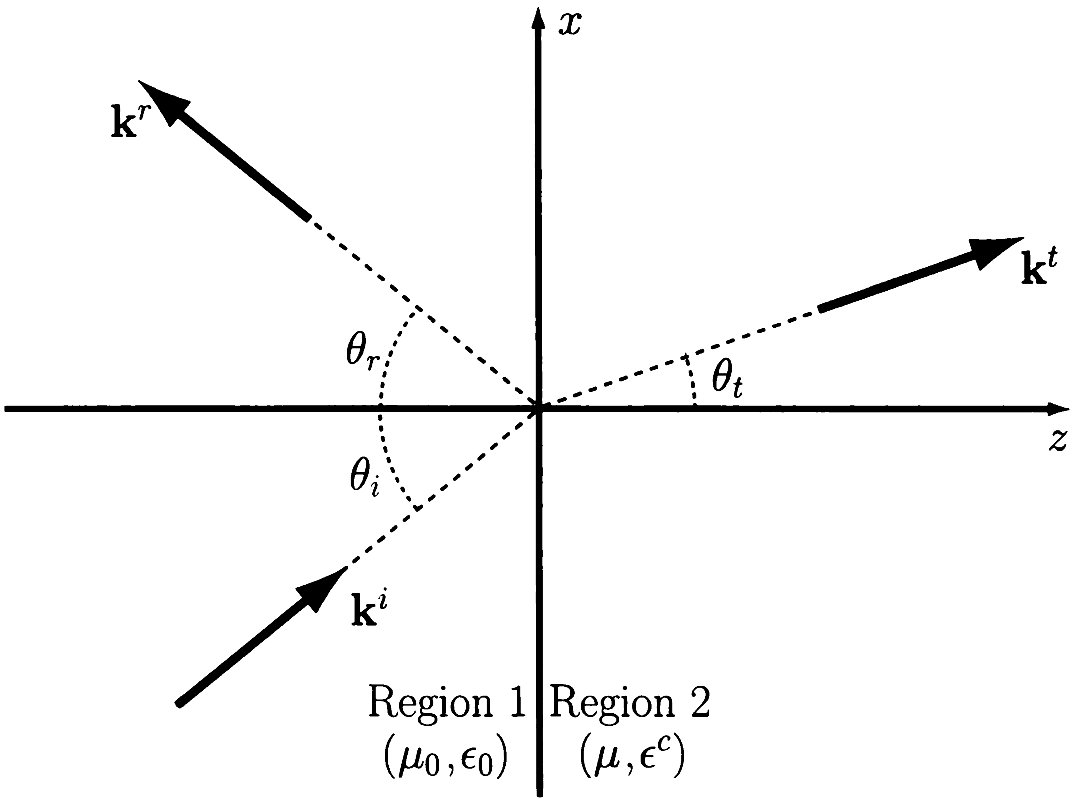

Figure 2.1. Geometry for an incident wave at a discontinuity between two material regions

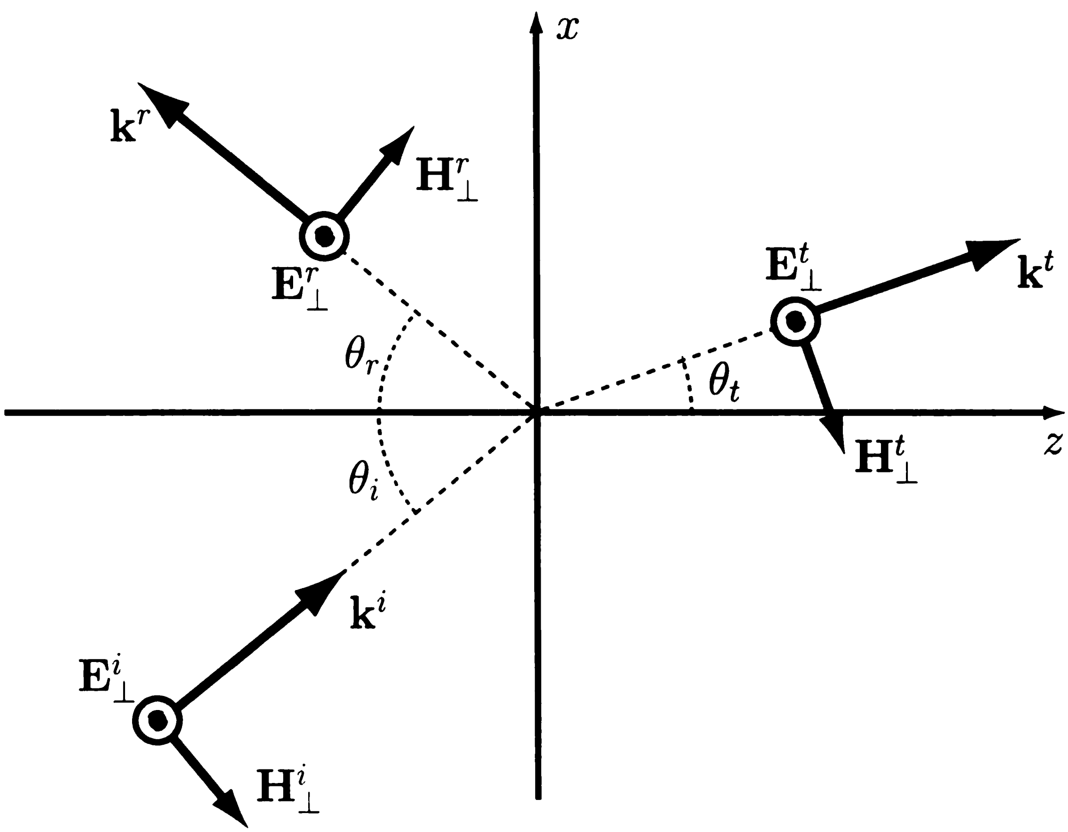

Figure 2.2. Fields for a TE-polarized plane wave incident on a material half-space.

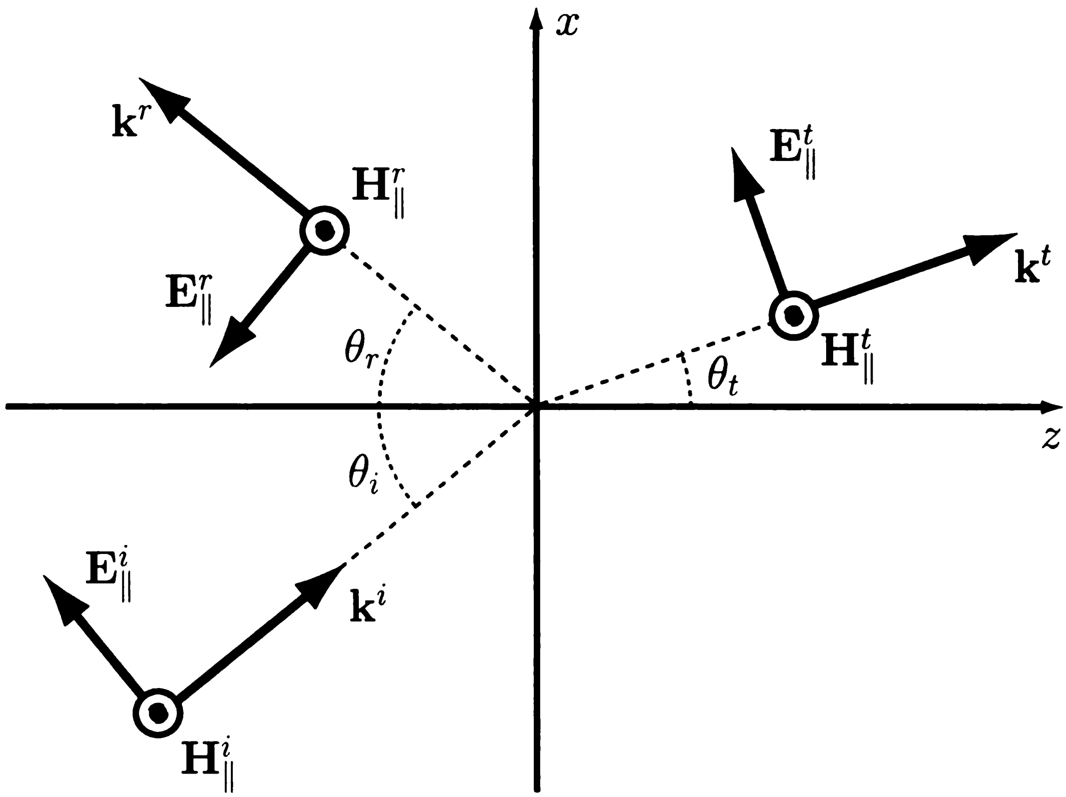

Figure 2.3. Fields for a TM-polarized plane wave incident on a material half-space.

CHAPTER 3

TE PLANE WAVE REFLECTION FOR THE OPTICAL CASE

The transient reflection from a Lorentz-medium half-space for a plane wave can be found analytically using an inverse Laplace transform technique. The simplest case to examine is that of a TE-polarized plane wave. A steady-state TE-polarized plane wave of frequency ω is obliquely incident on an interface separating free space (region 1) from a homogeneous Lorentz medium (region 2). The angle of incidence θ is measured from the normal to the interface. This geometry can be seen in Figure 2.2. As shown in Chapter 2, the reflection coefficient for a plane wave from a material half-space can be expressed as

$$\Gamma_{\perp}(\omega) = \frac{Z_{\perp}(\omega) - Z_0}{Z_{\perp}(\omega) + Z_0}. \tag{3.1}$$

For a TE-polarized plane wave, the impedance of the incident wave is $Z_0 = \eta_0 / \cos\theta$ and the wave impedance of the transmitted wave is given by

$$Z_{\perp}(\omega) = \frac{k(\omega)\eta(\omega)}{k_z(\omega)}. \tag{3.2}$$

where

$$\eta_0 = \sqrt{\mu_0/\epsilon_0}, \tag{3.3a}$$

$$\eta = \sqrt{\mu/\epsilon}, \tag{3.3b}$$

$$k_z = \sqrt{k^2 - k_0^2 \sin^2\theta}, \tag{3.3c}$$

$$k_0 = \omega\sqrt{\mu_0\epsilon_0}, \tag{3.3d}$$

$$k = \omega\sqrt{\mu\epsilon}, \tag{3.3e}$$

$$\epsilon = \epsilon_0\epsilon_r(\omega). \tag{3.3f}$$

The relative permittivity of a single-resonance Lorentz medium is given by [1]

$$\epsilon_r(\omega) = \epsilon_\infty - \frac{\omega_0^2(\epsilon_s - \epsilon_\infty)}{\omega^2 - 2j\omega\delta - \omega_0^2}, \tag{3.4}$$

where ω_0 is the resonance frequency and δ is the damping coefficient. ϵ_s is the static permittivity which is the value of ϵ_r when $\omega = 0$, and ϵ_∞ is the optical permittivity which is the value of ϵ_r as $\omega \to \infty$. In optical problems, the case when $\epsilon_\infty = 1$ and the relative permeability $\mu_r = 1$ is of the most interest. In this case, (3.4) can be rewritten as [4]

$$\epsilon_r(\omega) = 1 + \frac{b^2}{\omega_0^2 - \omega^2 + 2j\omega\delta}, \tag{3.5}$$

where b is the plasma frequency of the medium.

3.1 Laplace Domain Representation

A frequency domain quantity can be generalized to the Laplace domain using $s = j\omega$. Using this, (3.5) can be represented in the Laplace domain as

$$\epsilon_r(s) = 1 + \frac{b^2}{s^2 + 2\delta s + \omega_0^2}. \tag{3.6}$$

The wave number in the Lorentz medium can then be calculated as

$$\begin{aligned} k(s) &= -js\sqrt{\mu\epsilon} \\ &= -js\sqrt{\mu_0\epsilon_0}\sqrt{1 + \frac{b^2}{s^2 + 2\delta s + \omega_0^2}} \\ &= -js\sqrt{\mu_0\epsilon_0}\sqrt{\frac{s^2 + 2\delta s + \omega_0^2 + b^2}{s^2 + 2\delta s + \omega_0^2}}. \end{aligned} \tag{3.7}$$

It is then convenient to make the substitution

$$s^2 + 2\delta s + \omega_0^2 = (s - s_1)(s - s_2), \tag{3.8}$$

where

$$s_{1,2} = -\delta \pm \sqrt{\delta^2 - \omega_0^2}, \tag{3.9}$$

and also the substitution

$$s^2 + 2\delta s + \omega_0^2 + b^2 = (s - s_3)(s - s_4), \tag{3.10}$$

where

$$s_{3,4} = -\delta \pm \sqrt{\delta^2 - \omega_0^2 - b^2}. \tag{3.11}$$

The Laplace domain reflection coefficient can then be written as

$$\begin{aligned}
\Gamma(s) &= \frac{k_{z0} - k_z}{k_{z0} + k_z} \\
&= \frac{\cos\theta - \sqrt{(k/k_0)^2 - \sin^2\theta}}{\cos\theta + \sqrt{(k/k_0)^2 - \sin^2\theta}} \\
&= \frac{\cos\theta - \sqrt{[(s - s_3)(s - s_4)]/[(s - s_1)(s - s_2)] - \sin^2\theta}}{\cos\theta + \sqrt{[(s - s_3)(s - s_4)]/[(s - s_1)(s - s_2)] - \sin^2\theta}} \\
&= \frac{\cos\theta\sqrt{s - s_1}\sqrt{s - s_2} - \sqrt{(s - s_3)(s - s_4) - \sin^2\theta(s - s_1)(s - s_2)}}{\cos\theta\sqrt{s - s_1}\sqrt{s - s_2} + \sqrt{(s - s_3)(s - s_4) - \sin^2\theta(s - s_1)(s - s_2)}}.
\end{aligned} \tag{3.12}$$

Expanding $(s - s_1)(s - s_2)$ and $(s - s_3)(s - s_4)$ using (3.8) and (3.10) respectively allows the reflection coefficient to be written as

$$\begin{aligned}
\Gamma(s) = \Big[&\cos\theta\sqrt{s - s_1}\sqrt{s - s_2} - [s^2(1 - \sin^2\theta) + 2\delta s(1 - \sin^2\theta) + \\
&(\omega_0^2 + b^2 - \omega_0^2\sin^2\theta)]^{1/2}\Big] \Big/ \Big[\cos\theta\sqrt{s - s_1}\sqrt{s - s_2} + [s^2(1 - \sin^2\theta) + \\
&2\delta s(1 - \sin^2\theta) + (\omega_0^2 + b^2 - \omega_0^2\sin^2\theta)]^{1/2}\Big]
\end{aligned}$$

$$= \frac{\sqrt{s-s_1}\sqrt{s-s_2} - \sqrt{s^2+2\delta s+\omega_0^2+b^2/\cos^2\theta}}{\sqrt{s-s_1}\sqrt{s-s_2} + \sqrt{s^2+2\delta s+\omega_0^2+b^2/\cos^2\theta}}. \tag{3.13}$$

It is then convenient to make another substitution with the definition

$$s^2 + 2\delta s + \omega_0^2 + b^2/\cos^2\theta = (s-s_5)(s-s_6), \tag{3.14}$$

where

$$s_{5,6} = -\delta \pm \sqrt{\delta^2 - \omega_0^2 - b^2/\cos^2\theta}. \tag{3.15}$$

Substituting (3.14) into (3.13) allows the reflection coefficient in the Laplace domain to be rewritten as

$$\Gamma(s) = \frac{\sqrt{s-s_1}\sqrt{s-s_2} - \sqrt{s-s_5}\sqrt{s-s_6}}{\sqrt{s-s_1}\sqrt{s-s_2} + \sqrt{s-s_5}\sqrt{s-s_6}}. \tag{3.16}$$

3.2 Time-Domain Reflection Coefficient

Using the frequency-domain reflection coefficient found in Section 3.1, the time-domain reflection coefficient can be found using an inverse Laplace transform [13]. In order to perform the inverse Laplace transform, the frequency-domain reflection coefficient may be rearranged into a more manageable form. Rationalizing the denominator in (3.16) leads to

$$\Gamma(s) = \frac{(\sqrt{s-s_1}\sqrt{s-s_2} - \sqrt{s-s_5}\sqrt{s-s_6})^2}{(s-s_1)(s-s_2) - (s-s_5)(s-s_6)} = \frac{N}{D}. \tag{3.17}$$

Examining the denominator term in more detail it can be seen that

$$\begin{aligned} D &= (s-s_1)(s-s_2) - (s-s_5)(s-s_6) \\ &= (s^2+2\delta s+\omega_0^2) - (s^2+2\delta s+\omega_0^2+\frac{b^2}{\cos^2\theta}) \end{aligned}$$

$$
\begin{aligned}
&= -b^2/\cos^2\theta \\
&= -B^2. \qquad (3.18)
\end{aligned}
$$

The numerator can then be expanded to

$$
\begin{aligned}
N &= (s-s_1)(s-s_2) - 2\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6} + (s-s_5)(s-s_6), \\
&= \left[(s-s_1)(s-s_2) - \sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6}\right] + \\
&\quad \left[(s-s_5)(s-s_6) - \sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6}\right], \\
&= g_1(s) + g_2(s). \qquad (3.19)
\end{aligned}
$$

This last separation is taken in order to make the inverse Laplace transform easier to perform.

The invertible form of the reflection coefficient is then given by

$$
-B^2\Gamma(s) = g_1(s) + g_2(s), \qquad (3.20)
$$

where each term, $g_1(s)$ and $g_2(s)$, can then be inverted separately.

3.2.1 Inversion of the $g_1(s)$ and $g_2(s)$ Terms

When inverting the term $g_1(s)$, it is convenient to make the substitutions

$$
s_{1,2} = -\delta \pm \lambda_1, \qquad (3.21)
$$

$$
s_{5,6} = -\delta \pm \lambda_5, \qquad (3.22)
$$

where

$$
\lambda_1 = \sqrt{\delta^2 - \omega_0^2}, \qquad (3.23a)
$$

$$
\lambda_5 = \sqrt{\delta^2 - \omega_0^2 - B^2}, \qquad (3.23b)
$$

and B^2 is as defined in (3.18). In order to perform the inverse Laplace transform, $g_1(s)$ may be rearranged into

$$g_1(s) = (s-s_1)(s-s_2)(s-s_5)(s-s_6)\times \left[\frac{1}{(s-s_5)(s-s_6)} - \frac{1}{\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6}}\right]. \tag{3.24}$$

Using partial fraction expansion, the first term in the brackets can be rewritten in the form

$$\frac{1}{(s-s_5)(s-s_6)} = \frac{K_1}{s-s_5} + \frac{K_2}{s-s_6}, \tag{3.25}$$

where K_1 and K_2 are given by

$$K_1 = \frac{1}{s_5-s_6} = \frac{1}{2\lambda_5}, \tag{3.26a}$$

$$K_2 = \frac{1}{s_6-s_5} = -\frac{1}{2\lambda_5}. \tag{3.26b}$$

Since ω_0^2 and B^2 are both positive numbers, $Re\{s_{5,6}\} < 0$ always. This means the standard inverse Laplace transform [14]

$$\frac{1}{s+\beta} \longleftrightarrow e^{-\beta t}\mathrm{u}(t), \tag{3.27}$$

can be used, where $\mathrm{u}(t)$ is a unit step function which is zero for $t < 0$ and has unit amplitude for $t \geq 0$. Using this identity, (3.25) can be transformed into

$$\begin{aligned}\frac{1}{(s-s_5)(s-s_6)} \longleftrightarrow& \frac{1}{2\lambda_5}\left[e^{s_5 t} - e^{s_6 t}\right]\mathrm{u}(t)\\ &= \frac{e^{-\delta t}}{2\lambda_5}\left[e^{\lambda_5 t} - e^{-\lambda_5 t}\right]\mathrm{u}(t)\\ &= \frac{e^{-\delta t}}{\lambda_5}\sinh(\lambda_5 t)\mathrm{u}(t).\end{aligned} \tag{3.28}$$

The second term in the bracket in (3.24) can be transformed into a convolution between two terms if the following inverse Laplace transform is used [14]:

$$\frac{1}{\sqrt{s+\rho}\sqrt{s+\sigma}} \longleftrightarrow e^{-\frac{1}{2}(\rho+\sigma)t}\mathrm{I}_0\left[\frac{1}{2}(\rho-\sigma)t\right]\mathrm{u}(t), \tag{3.29}$$

where $\mathrm{I}_n(x)$ is the modified Bessel function of the first kind of order n. Using this, the second term can be transformed into

$$\frac{1}{\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6}} \longleftrightarrow e^{-\delta t}\left[\{\mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\} * \{\mathrm{I}_0(\lambda_5 t)\mathrm{u}(t)\}\right]. \tag{3.30}$$

Using (3.8), (3.14), (3.28) and (3.30), and using the differentiation theorem, it is possible to write $g_1(t)$ as

$$\begin{aligned} g_1(t) = &\left(\frac{\mathrm{d}^2}{\mathrm{d}t^2} + 2\delta\frac{\mathrm{d}}{\mathrm{d}t} + \omega_0^2\right)\left(\frac{\mathrm{d}^2}{\mathrm{d}t^2} + 2\delta\frac{\mathrm{d}}{\mathrm{d}t} + \omega_0^2 + B^2\right) \times \\ &e^{-\delta t}\left[\frac{1}{\lambda_5}\sinh(\lambda_5 t)\mathrm{u}(t) - \{\mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\} * \{\mathrm{I}_0(\lambda_5 t)\mathrm{u}(t)\}\right]. \end{aligned} \tag{3.31}$$

The next step is to carry out the derivatives. It is important to note at this point that when taking a derivative in time of two signals convolved with respect to time, the derivative only needs to be taken over one of the signals in the convolution. This can be expressed mathematically as

$$\frac{\mathrm{d}}{\mathrm{d}t}(f * g) = \frac{\mathrm{d}f}{\mathrm{d}t} * g = f * \frac{\mathrm{d}g}{\mathrm{d}t}. \tag{3.32}$$

Defining

$$\overline{g}_1(t) = e^{-\delta t}\left[\frac{1}{\lambda_5}\sinh(\lambda_5 t)\mathrm{u}(t) - \{\mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\} * \{\mathrm{I}_0(\lambda_5 t)\mathrm{u}(t)\}\right], \tag{3.33}$$

makes the solution easier to compute. The first derivative of $\overline{g}_1(t)$ computed using the product rule is given by

$$\frac{\mathrm{d}}{\mathrm{d}t}\overline{g}_1(t) = -\delta\overline{g}_1(t) + \overline{g}_1'(t), \tag{3.34}$$

where $\overline{g}_1'(t)$ is the exponential term multiplied by the derivative of the term in the brackets in (3.31). Using the bessel function identity [15]

$$\mathrm{I}_0'(x) = \mathrm{I}_1(x), \tag{3.35}$$

and the identity

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{u}(t) = \delta(t), \tag{3.36}$$

allows $\overline{g}_1'(t)$ to be written as

$$\overline{g}_1'(t) = e^{-\delta t}\left[\cosh(\lambda_5 t)\mathrm{u}(t) - \lambda_5\left\{\mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\right\} * \left\{\mathrm{I}_1(\lambda_5 t)\mathrm{u}(t)\right\} - \mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\right]. \tag{3.37}$$

In the convolution term, the derivative was taken on the second modified Bessel function containing the argument $\lambda_5 t$. The modified Bessel function of order zero came from the fact that $\{\mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\} * \{\mathrm{I}_0(\lambda_5 t)\delta(t)\} = \mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)$. The second derivative is given by

$$\begin{aligned}
\frac{\mathrm{d}^2}{\mathrm{d}t^2}\overline{g}_1(t) &= -\delta\left(\frac{\mathrm{d}}{\mathrm{d}t}\overline{g}_1(t)\right) + \frac{\mathrm{d}}{\mathrm{d}t}\overline{g}_1'(t) \\
&= -\delta\left(-\delta\overline{g}_1(t) + \overline{g}_1'(t)\right) - \delta\overline{g}_1'(t) + \overline{g}_1''(t) \\
&= \delta^2\overline{g}_1(t) - 2\delta\overline{g}_1(t)' + \overline{g}_1''(t),
\end{aligned} \tag{3.38}$$

where $\overline{g}_1''(t)$ is defined as the exponential term multiplied by the derivative of the term in the brackets in (3.37). Using the Bessel function identity [15]

$$\mathrm{I}_n'(x) = \frac{1}{2}\left[\mathrm{I}_{n-1}(x) + \mathrm{I}_{n+1}(x)\right], \tag{3.39}$$

and again takin the derivative on the second modified Bessel function in the convolution term allows $\overline{g}_1''(t)$ to be solved for as

$$\begin{aligned}\overline{g}_1''(t) = e^{-\delta t}\Bigg[&\lambda_5 \sinh(\lambda_5 t)\mathrm{u}(t) - \frac{\lambda_5^2}{2}\left\{\mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\right\} * \left\{\mathrm{I}_0(\lambda_5 t)\mathrm{u}(t)\right\} - \\ &\frac{\lambda_5^2}{2}\left\{\mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\right\} * \left\{\mathrm{I}_2(\lambda_5 t)\mathrm{u}(t)\right\} - \lambda_1 \mathrm{I}_1(\lambda_1 t)\mathrm{u}(t)\Bigg].\end{aligned} \tag{3.40}$$

Combining (3.33), (3.34) and (3.38), it can be seen that

$$\begin{aligned}\left(\frac{\mathrm{d}^2}{\mathrm{d}t^2} + 2\delta\frac{\mathrm{d}}{\mathrm{d}t} + \omega_0^2 + B^2\right)\overline{g}_1(t) &= \delta^2\overline{g}_1(t) - 2\delta\overline{g}_1'(t) + \overline{g}_1''(t) - \\ &\quad 2\delta^2\overline{g}_1(t) + 2\delta\overline{g}_1'(t) + (\omega_0^2 + B^2)\overline{g}_1(t) \\ &= \overline{g}_1''(t) - (\delta^2 - \omega_0^2 - B^2)\overline{g}_1(t).\end{aligned} \tag{3.41}$$

Using the definition for λ_5 given in (3.23), (3.41) can be rewritten as

$$\begin{aligned}&\left(\frac{\mathrm{d}^2}{\mathrm{d}t^2} + 2\delta\frac{\mathrm{d}}{\mathrm{d}t} + \omega_0^2 + B^2\right)\overline{g}_1(t) = \overline{g}_1''(t) - \lambda_5^2\overline{g}_1(t) \\ &\quad = e^{-\delta t}\left[\frac{\lambda_5^2}{2}\left\{\mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\right\} * \left\{\mathrm{I}_0(\lambda_5 t)\mathrm{u}(t) - \mathrm{I}_2(\lambda_5 t)\mathrm{u}(t)\right\} - \lambda_1\mathrm{I}_1(\lambda_1 t)\mathrm{u}(t)\right] \\ &\quad = e^{-\delta t}\left[\lambda_5\left\{\mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\right\} * \left\{\frac{\mathrm{I}_1(\lambda_5 t)\mathrm{u}(t)}{t}\right\} - \lambda_1\mathrm{I}_1(\lambda_1 t)\mathrm{u}(t)\right] \\ &\quad = \overline{h}_1(t),\end{aligned} \tag{3.42}$$

which takes advantage of the identity [15]

$$\frac{2n}{x}\mathrm{I}_n(x) = \mathrm{I}_{n-1}(x) - \mathrm{I}_{n+1}(x). \tag{3.43}$$

It is then possible to find

$$\frac{\mathrm{d}}{\mathrm{d}t}\overline{h}_1(t) = -\delta\overline{h}_1(t) + \overline{h}_1'(t), \tag{3.44}$$

where $\overline{h}_1'(t)$ is the exponential term multiplied by the derivative of the term in the brackets in (3.42). Performing the derivative on the first modified Bessel function in the convolution allows $\overline{h}_1'(t)$ to be written as

$$\begin{aligned}\overline{h}_1'(t) = e^{-\delta t}\Bigg[&\lambda_1\lambda_5\left\{\mathrm{I}_1(\lambda_1 t)\mathrm{u}(t)\right\} * \left\{\frac{\mathrm{I}_1(\lambda_5 t)\mathrm{u}(t)}{t}\right\} + \frac{\lambda_5^2}{2}[\mathrm{I}_0(\lambda_5 t) - \mathrm{I}_2(\lambda_5 t)]\mathrm{u}(t) - \\ &\frac{\lambda_1^2}{2}[\mathrm{I}_0(\lambda_1 t) + \mathrm{I}_2(\lambda_1 t)]\mathrm{u}(t)\Bigg].\end{aligned} \tag{3.45}$$

The second derivative is then found to be

$$\begin{aligned}\frac{\mathrm{d}^2}{\mathrm{d}t^2}\overline{h}_1(t) &= -\delta\left[\frac{\mathrm{d}}{\mathrm{d}t}\overline{h}_1(t)\right] + \frac{\mathrm{d}}{\mathrm{d}t}\overline{h}_1'(t) \\ &= -\delta\left[-\delta\overline{h}_1(t) + \overline{h}_1'(t)\right] - \delta\overline{h}_1'(t) + \overline{h}_1''(t) \\ &= \delta^2\overline{h}_1(t) - 2\delta\overline{h}_1'(t) + \overline{h}_1''(t),\end{aligned} \tag{3.46}$$

where $\overline{h}_1''(t)$ is the exponential term multiplied by the derivative of the term in the brackets in (3.45). When taking the derivative of the convolution term, the derivative is taken on the first modified Bessel function. This leads to

$$\overline{h}_1''(t) = e^{-\delta t}\Bigg[\frac{\lambda_1^2\lambda_5}{2}\left\{[\mathrm{I}_0(\lambda_1 t) + \mathrm{I}_2(\lambda_1 t)]\mathrm{u}(t)\right\} * \left\{\frac{\mathrm{I}_1(\lambda_5 t)\mathrm{u}(t)}{t}\right\} +$$

$$\frac{\lambda_5^3}{4}[\mathrm{I}_1(\lambda_5 t) - \mathrm{I}_3(\lambda_5 t)]\mathrm{u}(t) - \frac{\lambda_1^3}{4}[3\mathrm{I}_1(\lambda_1 t) + \mathrm{I}_3(\lambda_1 t)]\mathrm{u}(t)\Bigg] + \frac{\lambda_5^2 - \lambda_1^2}{2}\delta(t). \tag{3.47}$$

Combining (3.42),(3.44) and (3.46) it can be seen that

$$\begin{aligned}\left(\frac{\mathrm{d}^2}{\mathrm{d}t^2} + 2\delta\frac{\mathrm{d}}{\mathrm{d}t} + \omega_0^2\right)\overline{h}_1(t) = {} & \delta^2\overline{h}_1(t) - 2\delta\overline{h}_1'(t) + \overline{h}_1''(t) - \\ & 2\delta^2\overline{h}_1(t) + 2\delta\overline{h}_1'(t) + \omega_0^2\overline{h}_1(t) \\ = {} & \overline{h}_1''(t) - \left(\delta^2 - \omega_0^2\right)\overline{h}_1(t).\end{aligned} \tag{3.48}$$

Using the definition for λ_1 given in (3.23), (3.48) can be rewritten as

$$\begin{aligned}\left(\frac{\mathrm{d}^2}{\mathrm{d}t^2} + 2\delta\frac{\mathrm{d}}{\mathrm{d}t} + \omega_0^2\right)\overline{h}_1(t) &= \overline{h}_1''(t) - \lambda_1^2\overline{h}_1(t) \\ &= g_1(t),\end{aligned} \tag{3.49}$$

which leads to

$$g_1(t) = e^{-\delta t}\left[-\lambda_1^2\lambda_5^2\hat{\mathrm{I}}_1(\lambda_1 t) * \hat{\mathrm{I}}_1(\lambda_5 t) + \lambda_1^3\hat{\mathrm{I}}_2(\lambda_1 t) + \lambda_5^3\hat{\mathrm{I}}_2(\lambda_5 t)\right] + \frac{\lambda_5^2 - \lambda_1^2}{2}\delta(t), \tag{3.50}$$

where

$$\hat{\mathrm{I}}_n(x) = \frac{\mathrm{I}_n(x)}{x}\mathrm{u}(x). \tag{3.51}$$

The same approach can be taken to find

$$g_2(t) = e^{-\delta t}\left[-\lambda_1^2\lambda_5^2\hat{\mathrm{I}}_1(\lambda_1 t) * \hat{\mathrm{I}}_1(\lambda_5 t) + \lambda_1^3\hat{\mathrm{I}}_2(\lambda_1 t) + \lambda_5^3\hat{\mathrm{I}}_2(\lambda_5 t)\right] - \frac{\lambda_5^2 - \lambda_1^2}{2}\delta(t). \tag{3.52}$$

3.2.2 Final Expression for $\Gamma(t)$

Substituting $g_1(t)$ from (3.50) and $g_2(t)$ from (3.52) into (3.20) yields the final expression for the reflection coefficient

$$\Gamma(t) = \frac{2}{B^2} e^{-\delta t} \left[\lambda_1^2 \lambda_5^2 \hat{\mathrm{I}}_1(\lambda_1 t) * \hat{\mathrm{I}}_1(\lambda_5 t) - \lambda_1^3 \hat{\mathrm{I}}_2(\lambda_1 t) - \lambda_5^3 \hat{\mathrm{I}}_2(\lambda_5 t) \right]. \tag{3.53}$$

The solution (3.53) is generally valid for all cases, but when λ_1 or λ_5 is imaginary, the modified Bessel functions can be replaced with ordinary Bessel functions according to three possible cases. Case 1 occurs when $\omega_0^2 > \delta^2$. In this case both λ_1 and λ_5 are purely imaginary and defined as

$$-j\lambda_1 = \overline{\lambda}_1 = \sqrt{\omega_0^2 - \delta^2}, \qquad -j\lambda_5 = \overline{\lambda}_5 = \sqrt{\omega_0^2 + B^2 - \delta^2}. \tag{3.54}$$

Then, using the property [15]

$$\mathrm{I}_n(jx) = j^n \mathrm{J}_n(x), \tag{3.55}$$

allows the reflection coefficient to be rewritten as

$$\Gamma(t) = \frac{2}{B^2} e^{-\delta t} \left[\overline{\lambda}_1^2 \overline{\lambda}_5^2 \hat{\mathrm{J}}_1(\overline{\lambda}_1 t) * \hat{\mathrm{J}}_1(\overline{\lambda}_5 t) - \overline{\lambda}_1^3 \hat{\mathrm{J}}_2(\overline{\lambda}_1 t) - \overline{\lambda}_5^3 \hat{\mathrm{J}}_2(\overline{\lambda}_5 t) \right], \tag{3.56}$$

where, similar to (3.51),

$$\hat{\mathrm{J}}_n(x) = \frac{\mathrm{J}_n(x)}{x} \mathrm{u}(x). \tag{3.57}$$

Case 2 occurs when $\omega_0^2 + B^2 < \delta^2$. In this case, both λ_1 and λ_5 are purely real and the same expression from (3.53) can be used. Case 3 occurs when $-B^2 < \omega_0^2 - \delta^2 < 0$. In this case, λ_1 is purely real and λ_5 is purely imaginary. $\overline{\lambda}_5$ is defined as it was in

(3.54) and the expression for the reflection coefficient can be written as

$$\Gamma(t) = -\frac{2}{B^2}e^{-\delta t}\left[\lambda_1^2\overline{\lambda}_5^2\hat{\mathrm{I}}_1(\lambda_1 t) * \hat{\mathrm{J}}_1(\overline{\lambda}_5 t) + \lambda_1^3\hat{\mathrm{I}}_2(\lambda_1 t) + \overline{\lambda}_5^3\hat{\mathrm{J}}_2(\overline{\lambda}_5 t)\right]. \tag{3.58}$$

3.3 Numerical Results

In order to validate the expressions derived in the previous section, the time-domain reflection coefficient is evaluated numerically in Fortran, and then compared to the inverse FFT of the frequency domain reflection coefficient given in (3.16). The Fortran code is included in Appendix A. The inverse FFT was done using WaveCalc. The frequency-domain data was zero-padded up to the maximum limit allowed by WaveCalc, 32,768, before the inverse FFT was taken. Since the signal had already decayed to zero by this point, no windowing was necessary. A set of parameters corresponding to each of the three possible cases is used. The number of points and step size for the results varied between the different cases.

The first set of parameters is the same as those chosen by Brillouin [16]: $\omega_0 = 4.0 \times 10^{16}\ s^{-1}$, $b^2 = 20.0 \times 10^{32}\ s^{-2}$, $\delta = 0.28 \times 10^{16}\ s^{-1}$. This choice of parameters corresponds to case 1. When computing the numerical results, for the frequency-domain data 16,384 frequency points were calculated with a step size of 8,000 GHz. In the time-domain, 4,096 points were calculated with a step size of 1×10^{-9} ns. Using $\theta = 30°$, (3.56) has been plotted in Figure 3.1 and compared to the inverse FFT. The results show excellent agreement. Since this function includes only standard Bessel functions, which are highly oscillatory, and no modified Bessel functions, which are not oscillatory, the waveform is highly oscillatory and only lightly damped.

The next choice of parameters is: $\omega_0 = 2.0 \times 10^{15}\ s^{-1}$, $b^2 = 20.0 \times 10^{29}\ s^{-2}$, $\delta = 0.28 \times 10^{16}\ s^{-1}$, which corresponds to case 2, (3.53). When computing the numerical results, for the frequency-domain data 4,096 frequency points were calculated with a step size of 400 GHz. In the time-domain, 4,096 points were calculated with a step size of 1×10^{-8} ns. The results are shown in Figure 3.2. Again, the closed-form expression and the inverse FFT compare well. For this choice of parameters $\delta^2 > \omega_0^2 + B^2$, and the resulting waveform is overdamped, showing no oscillatory behavior and only a single negative peak. Since the expression for case 2 only involves modified Bessel functions, which do not have the oscillatory behavior of ordinary Bessel functions, this observed behavior is easily predicted from the mathematical form of the expression.

The final choice of parameters, which corresponds to case 3, is: $\omega_0 = 2.0 \times 10^{15}\ s^{-1}$, $b^2 = 20.0 \times 10^{32}\ s^{-2}$, $\delta = 0.28 \times 10^{16}\ s^{-1}$, which corresponds to case 3, (3.58). When computing the numerical results, for the frequency-domain data 16,384 frequency points were calculated with a step size of 8,000 GHz. In the time-domain, 4,096 points were calculated with a step size of 1×10^{-9} ns. The analytic expression again matches the inverse FFT, as seen in Figure 3.3. As expected, since $\delta > \omega_0$, but $\delta^2 < \omega_0^2 + B^2$, there is more damping and less oscillation than with case 1, but more oscillation than with case 2. Here the expression for the reflection coefficient has a combination of ordinary and modified Bessel functions.

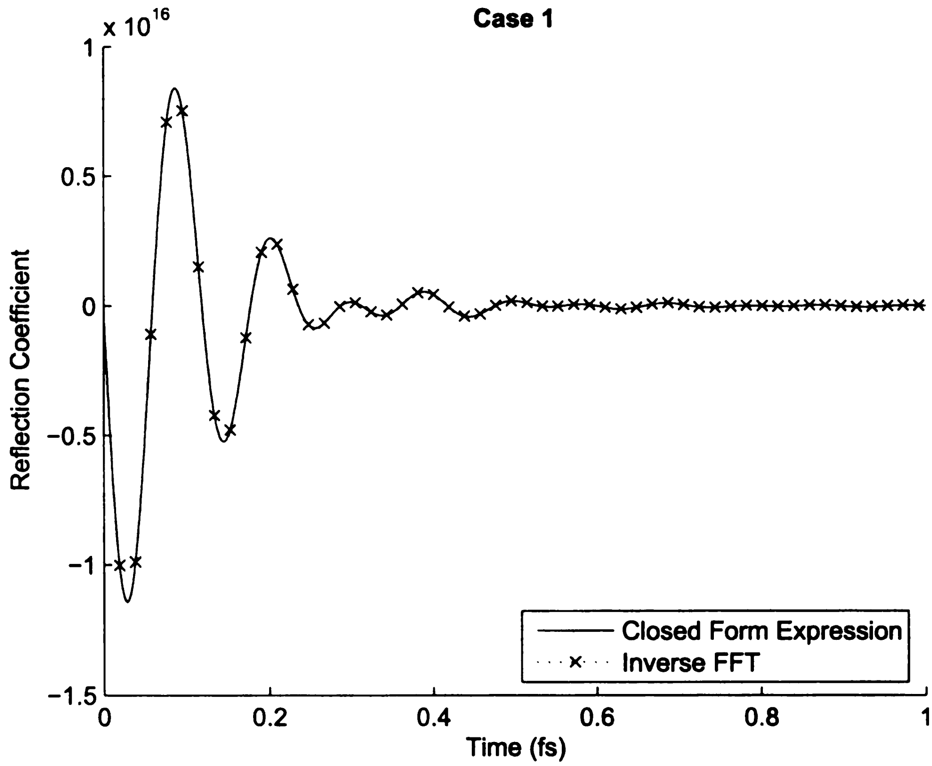

Figure 3.1. Time-domain reflection coefficient with incidence angle $\theta = 30°$ and material parameters $\omega_0 = 4.0 \times 10^{16} s^{-1}$, $b^2 = 20.0 \times 10^{32} s^{-2}$, $\delta = 0.28 \times 10^{16} s^{-1}$. This choice of parameters corresponds to case 1, Eq. (3.56).

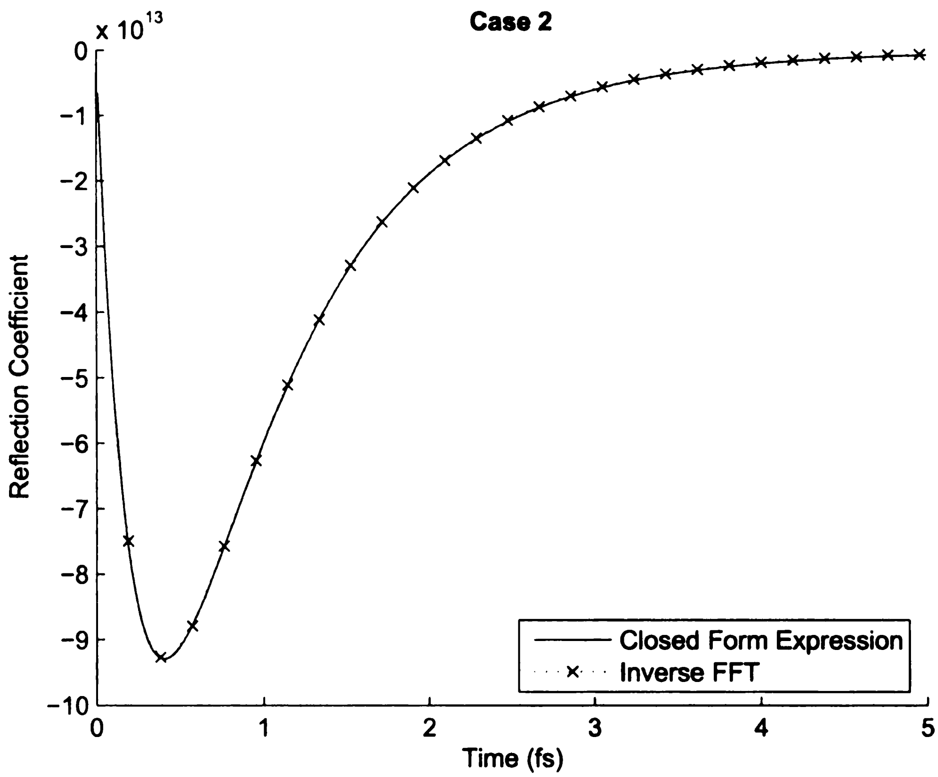

Figure 3.2. Time-domain reflection coefficient with incidence angle $\theta = 30°$ and material parameters $\omega_0 = 2.0 \times 10^{15}s^{-1}$, $b^2 = 20.0 \times 10^{29}s^{-2}$, $\delta = 0.28 \times 10^{16}s^{-1}$. This choice of parameters corresponds to case 2, Eq. (3.53).

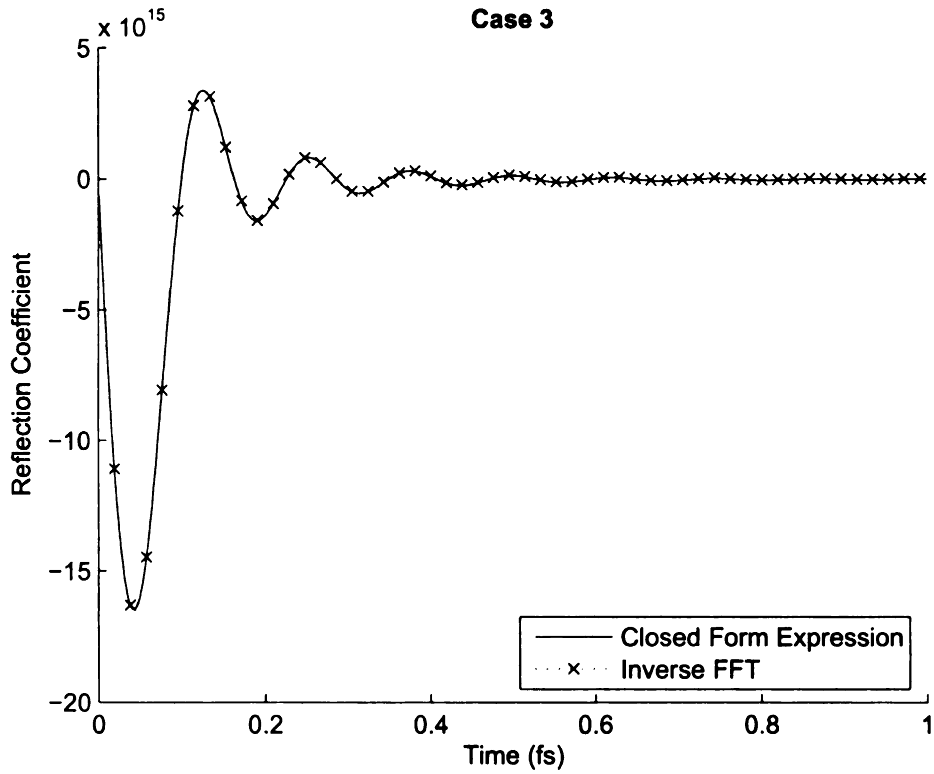

Figure 3.3. Time-domain reflection coefficient with at an angle of $\theta = 30°$ with parameter choices of $\omega_0 = 2.0 \times 10^{15}s^{-1}$, $b^2 = 20.0 \times 10^{32}s^{-2}$, $\delta = 0.28 \times 10^{16}s^{-1}$. This choice of parameters corresponds to case 3, Eq. (3.58).

CHAPTER 4

TM PLANE WAVE REFLECTION FOR THE OPTICAL CASE

The reflection of a TM plane wave from a Lorentz medium half-space can be found with a similar method as for the TE case. The problem is defined as in Chapter 3 with a plane wave from free space obliquely incident on a homogeneous Lorentz medium at an angle θ measured normal to the interface. This geometry can be seen in Figure 2.3. As shown in Chapter 2, the reflection coefficient for a plane wave from a material half-space can be expressed as

$$\Gamma_{\|}(\omega) = \frac{Z_{\|}(\omega) - Z_0}{Z_{\|}(\omega) + Z_0}. \tag{4.1}$$

For a TM-polarized plane wave, the impedance of the incident wave is $Z_0 = \eta_0 \cos\theta$ and the wave impedance of the transmitted wave is given by

$$Z_{\|}(\omega) = \frac{k_z(\omega)\eta(\omega)}{k(\omega)}, \tag{4.2}$$

where the terms k_z, η, and k are defined in (3.3). The relative permittivity is the same as given in (3.5).

4.1 Laplace-Domain Representation

For a TM-polarized plane wave, the Laplace domain reflection coefficient can be written as

$$\begin{aligned}\Gamma(s) &= \frac{k_z - \epsilon_r k_{z0}}{k_z + \epsilon_r k_{z0}} \\ &= \frac{\sqrt{(k/k_0)^2 - \sin^2\theta} - \epsilon_r \cos\theta}{\sqrt{(k/k_0)^2 - \sin^2\theta} + \epsilon_r \cos\theta}\end{aligned}$$

$$= \frac{\sqrt{\epsilon_r - \sin^2\theta} - \epsilon_r \cos\theta}{\sqrt{\epsilon_r - \sin^2\theta} + \epsilon_r \cos\theta}. \tag{4.3}$$

Combining the relative permittivity ϵ_r into one fraction, it can be substituted into (4.3) to give

$$\begin{aligned}
\Gamma(s) &= \Bigg[\sqrt{(s^2 + 2\delta s + \omega_0^2 + b^2)/(s^2 + 2\delta s + \omega_0^2) - \sin^2\theta} - \Big[(s^2 + 2\delta s + \omega_0^2 + b^2)/ \\
&\qquad (s^2 + 2\delta s + \omega_0^2)\Big] \cos\theta\Bigg] \Bigg/ \Bigg[\sqrt{(s^2 + 2\delta s + \omega_0^2 + b^2)/(s^2 + 2\delta s + \omega_0^2) - \sin^2\theta} + \\
&\qquad \Big[(s^2 + 2\delta s + \omega_0^2 + b^2)/(s^2 + 2\delta s + \omega_0^2)\Big] \cos\theta\Bigg] \\
&= \Bigg[\sqrt{s - s_1}\sqrt{s - s_2}\left[s^2 + 2\delta s + \omega_0^2 + b^2 - (s^2 + 2\delta s + \omega_0^2)\sin^2\theta\right]^{1/2} - \\
&\qquad (s - s_3)(s - s_4)\cos\theta\Bigg] \Bigg/ \Bigg[\sqrt{s - s_1}\sqrt{s - s_2}\Big[s^2 + 2\delta s + \omega_0^2 + b^2 - \\
&\qquad (s^2 + 2\delta s + \omega_0^2)\sin^2\theta\Big]^{1/2} + (s - s_3)(s - s_4)\cos\theta\Bigg] \\
&= \frac{\sqrt{s - s_1}\sqrt{s - s_2}\sqrt{s^2 + 2\delta s + \omega_0^2 + b^2/\cos^2\theta} - (s - s_3)(s - s_4)}{\sqrt{s - s_1}\sqrt{s - s_2}\sqrt{s^2 + 2\delta s + \omega_0^2 + b^2/\cos^2\theta} + (s - s_3)(s - s_4)},
\end{aligned} \tag{4.4}$$

where $s_{1,2}$ are defined as in (3.9) and $s_{3,4}$ are defined as in (3.11). Using the substitution defined in (3.14), allows (4.4) to be rewritten into the final frequency-domain reflection coefficient

$$\Gamma(s) = \frac{\sqrt{s - s_1}\sqrt{s - s_2}\sqrt{s - s_5}\sqrt{s - s_6} - (s - s_3)(s - s_4)}{\sqrt{s - s_1}\sqrt{s - s_2}\sqrt{s - s_5}\sqrt{s - s_6} + (s - s_3)(s - s_4)}. \tag{4.5}$$

4.2 Time-Domain Reflection Coefficient

Using the frequency-domain reflection coefficient found in Section 4.1, the time-domain reflection coefficient can be found using an inverse Laplace transform. In order to perform the inverse Laplace transform, the frequency-domain reflection coefficient may be rearranged into a more manageable form. Rationalizing the denominator in

(4.5) leads to

$$\Gamma(s) = \frac{[\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6} - (s-s_3)(s-s_4)]^2}{(s-s_1)(s-s_2)(s-s_5)(s-s_6) - (s-s_3)^2(s-s_4)^2} = \frac{N}{D}. \tag{4.6}$$

Noting that

$$(s-s_3)(s-s_4) = (s-s_1)(s-s_2) + \frac{b^2}{\cos^2\theta}, \tag{4.7}$$

allows the denominator to be expanded into

$$\begin{aligned}
D &= (s-s_1)(s-s_2)\left[(s-s_1)(s-s_2) + \frac{b^2}{\cos^2\theta}\right] - \left[(s-s_1)(s-s_2) + b^2\right]^2 \\
&= b^2\left(\frac{1}{\cos^2\theta} - 2\right)(s-s_1)(s-s_2) - b^4 \\
&= b^2(\tan^2\theta - 1)(s^2 + 2\delta s + \omega_0^2) - b^4 \\
&= b^2(\tan^2\theta - 1)\left(s^2 + 2\delta s + \omega_0^2 - \frac{b^2}{\tan^2\theta - 1}\right) \\
&= b^2(\tan^2\theta - 1)(s-s_A)(s-s_B),
\end{aligned} \tag{4.8}$$

where

$$\begin{aligned}
s_{A,B} &= -\delta \pm \sqrt{\delta^2 - \omega_0^2 + \frac{b^2}{\tan^2\theta - 1}} \\
&= -\delta \pm \lambda_A.
\end{aligned} \tag{4.9}$$

Using this, (4.6) can be rewritten as

$$b^2(\tan^2\theta - 1)\Gamma(s) = \frac{[\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6} - (s-s_3)(s-s_4)]^2}{(s-s_A)(s-s_B)}. \tag{4.10}$$

Using (4.7), the numerator can be expanded into

$$N = (s-s_1)(s-s_2)(s-s_5)(s-s_6) -$$

$$
\begin{aligned}
&2(s-s_3)(s-s_4)\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6}+(s-s_3)^2(s-s_4)^2\\
&=(s-s_1)(s-s_2)(s-s_5)(s-s_6)-\\
&2(s-s_1)(s-s_2)\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6}-\\
&2b^2\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6}+(s-s_1)^2(s-s_2)^2+\\
&2b^2(s-s_1)(s-s_2)+b^4.
\end{aligned} \tag{4.11}
$$

At this point, factoring out $(s-s_1)(s-s_2)$ allows the numerator to be rewritten in a form that includes both $g_1(s)$ and $g_2(s)$ which are defined and inverted previously in Section 3.2.

$$
\begin{aligned}
N=(s-s_1)(s-s_2)\Bigg[&(s-s_1)(s-s_2)+(s-s_5)(s-s_6)-\\
&2\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6}-2b^2\frac{(s-s_5)(s-s_6)}{\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6}}+\\
&\frac{b^2}{(s-s_1)(s-s_2)}+2b^2\Bigg]\\
=(s-s_1)(s-s_2)&\left[g_1(s)+g_2(s)-2b^2g_3(s)+b^4g_4(s)+2b^2\right].
\end{aligned} \tag{4.12}
$$

In the above equation, the $g_n(s)$ terms are defined as

$$g_1(s)=(s-s_1)(s-s_2)-\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6}, \tag{4.13a}$$

$$g_2(s)=(s-s_5)(s-s_6)-\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6}, \tag{4.13b}$$

$$g_3(s)=\frac{(s-s_5)(s-s_6)}{\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6}}, \tag{4.13c}$$

$$g_4(s)=\frac{1}{(s-s_1)(s-s_2)}. \tag{4.13d}$$

By defining

$$C(s)=\frac{(s-s_1)(s-s_2)}{(s-s_A)(s-s_B)}, \tag{4.14}$$

the reflection coefficient can be rewritten substituting (4.8), (4.12) and (4.14) into (4.6) as

$$\begin{aligned} b^2(\tan^2\theta - 1)\Gamma(s) &= C(s)\left[g_1(s) + g_2(s) - 2b^2 g_3(s) + b^4 g_4(s) + 2b^2\right] \\ &= C(s)G(s). \end{aligned} \tag{4.15}$$

Each term in this expression can be inverted separately to give a final transient reflection coefficient.

4.2.1 Inversion of the $g_n(s)$ Terms

Inversion of the $g_1(s)$ and $g_2(s)$ terms are solved for previously in Section 3.2.1 and are given as

$$g_1(t) = e^{-\delta t}\left[-\lambda_1^2\lambda_5^2\hat{\mathrm{I}}_1(\lambda_1 t) * \hat{\mathrm{I}}_1(\lambda_5 t) + \lambda_1^3\hat{\mathrm{I}}_2(\lambda_1 t) + \lambda_5^3\hat{\mathrm{I}}_2(\lambda_5 t)\right] - \frac{\lambda_1^2 - \lambda_5^2}{2}\delta(t), \tag{4.16}$$

$$g_2(t) = e^{-\delta t}\left[-\lambda_1^2\lambda_5^2\hat{\mathrm{I}}_1(\lambda_1 t) * \hat{\mathrm{I}}_1(\lambda_5 t) + \lambda_1^3\hat{\mathrm{I}}_2(\lambda_1 t) + \lambda_5^3\hat{\mathrm{I}}_2(\lambda_5 t)\right] + \frac{\lambda_1^2 - \lambda_5^2}{2}\delta(t). \tag{4.17}$$

In these equations, λ_1 and λ_5 are defined in (3.23) and $\hat{\mathrm{I}}_n(x)$ is defined in (3.51).

The identity from (3.29), redefined below, can be used to perform the inversion on $g_3(s)$ to find $g_3(t)$.

$$\frac{1}{\sqrt{s+\rho}\sqrt{s+\sigma}} \longleftrightarrow e^{-\frac{1}{2}(\rho+\sigma)t}\mathrm{I}_0\left[\frac{1}{2}(\rho-\sigma)t\right]\mathrm{u}(t). \tag{4.18}$$

Using this, it is possible to write

$$g_3(t) = \left(\frac{\mathrm{d}^2}{\mathrm{d}t^2} + 2\delta\frac{\mathrm{d}}{\mathrm{d}t} + \omega_0^2 + \frac{b^2}{\cos^2\theta}\right)e^{-\delta t}\left[\{\mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\} * \{\mathrm{I}_0(\lambda_5 t)\mathrm{u}(t)\}\right]$$

$$= \left(\frac{\mathrm{d}^2}{\mathrm{d}t^2} + 2\delta\frac{\mathrm{d}}{\mathrm{d}t} + \omega_0^2 + \frac{b^2}{\cos^2\theta}\right)\overline{g}_3(t). \tag{4.19}$$

Taking the first derivative of $\overline{g}_3(t)$ using the product rule yields

$$\frac{\mathrm{d}}{\mathrm{d}t}\overline{g}_3(t) = -\delta\overline{g}_3(t) + \overline{g}_3'(t), \tag{4.20}$$

where $\overline{g}_3'(t)$ is the exponential term multiplied by the derivative of the term in the brackets in (4.19). Taking the derivative on the second term in the convolution allows $\overline{g}_3'(t)$ to be written as

$$\overline{g}_3'(t) = e^{-\delta t}\left[\lambda_5\left\{\mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\right\} * \left\{\mathrm{I}_1(\lambda_5 t)\mathrm{u}(t)\right\} + \mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\right]. \tag{4.21}$$

From this, the second derivative can be written as

$$\begin{aligned}\frac{\mathrm{d}^2}{\mathrm{d}t^2}\overline{g}_3(t) &= -\delta\left[\frac{\mathrm{d}}{\mathrm{d}t}\overline{g}_3(t)\right] + \frac{\mathrm{d}}{\mathrm{d}t}\overline{g}_3'(t)\\ &= -\delta\left[-\delta\overline{g}_3(t) + \overline{g}_3'(t)\right] - \delta\overline{g}_3'(t) + \overline{g}_3''(t)\\ &= \delta^2\overline{g}_3(t) - 2\delta\overline{g}_3'(t) + \overline{g}_3''(t),\end{aligned} \tag{4.22}$$

where $\overline{g}_3''(t)$ is the exponential term multiplied by the derivative of the term in the brackets in (4.21). Using the identity given in (3.39), it can then be shown that

$$\overline{g}_3''(t), = e^{-\delta t}\left[\frac{\lambda_5^2}{2}\left\{\mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\right\} * \left\{\left[\mathrm{I}_0(\lambda_5 t) + \mathrm{I}_2(\lambda_5 t)\right]\mathrm{u}(t)\right\} + \lambda_1\mathrm{I}_1(\lambda_1 t)\mathrm{u}(t) + \delta(t)\right], \tag{4.23}$$

where the derivative was again taken on the second term in the convolution. Combining (4.19), (4.20) and (4.22) allows $g_3(t)$ to be written as

$$g_3(t) = \delta^2\overline{g}_3(t) - 2\delta\overline{g}_3'(t) + \overline{g}_3''(t) - 2\delta^2\overline{g}_3(t) + 2\delta\overline{g}_3'(t) + (\omega_0^2 + B^2)\overline{g}_3(t)$$

$$= \overline{g}_3''(t) - (\delta^2 - \omega_0^2 - B^2)\overline{g}_3(t). \tag{4.24}$$

Using the definition for λ_5 given in (3.23), the final expression for $g_3(t)$ is found to be

$$\begin{aligned} g_3(t) &= \overline{g}_3''(t) - \lambda_5^2 \overline{g}_3(t) \\ &= e^{-\delta t}\left[\frac{\lambda_5^2}{2}\{\mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\} * \{[\mathrm{I}_2(\lambda_5 t) - \mathrm{I}_0(\lambda_5 t)]\mathrm{u}(t)\} + \lambda_1 \mathrm{I}_1(\lambda_1 t)\mathrm{u}(t)\right] + \delta(t) \\ &= e^{-\delta t}\left[-\lambda_5^2 \{\mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\} * \hat{\mathrm{I}}_1(\lambda_5 t) + \lambda_1 \mathrm{I}_1(\lambda_1 t)\mathrm{u}(t)\right] + \delta(t). \end{aligned} \tag{4.25}$$

This took advantage of the Bessel identity defined in (3.43).

The final g-term to invert, $g_4(s)$, is given by (4.12) as

$$\frac{1}{(s-s_1)(s-s_2)}. \tag{4.26}$$

Using partial fraction expansion, this term can be rewritten as

$$\begin{aligned} g_4(s) &= \frac{1}{(s-s_1)(s-s_2)} \\ &= \frac{K_1}{s-s_1} + \frac{K_2}{s-s_2}, \end{aligned} \tag{4.27}$$

where

$$K_1 = \frac{1}{2\lambda_1}, \tag{4.28a}$$

$$K_2 = -\frac{1}{2\lambda_1} = -K_1. \tag{4.28b}$$

This allows (4.27) to be rewritten as

$$g_4(s) = \frac{1}{2\lambda_1}\left(\frac{1}{s-s_1} - \frac{1}{s-s_2}\right). \tag{4.29}$$

Using the identity as defined in (3.27), $g_4(t)$ is found to be

$$
\begin{aligned}
g_4(t) \longleftrightarrow g_4(t) &= \frac{1}{2\lambda_1}\left[e^{s_1 t} - e^{s_2 t}\right]\mathrm{u}(t) \\
&= \frac{e^{-\delta t}}{2\lambda_1}\left[e^{\lambda_1 t} - e^{-\lambda_1 t}\right]\mathrm{u}(t) \\
&= \frac{e^{-\delta t}}{\lambda_1}\sinh(\lambda_1 t)\mathrm{u}(t).
\end{aligned} \tag{4.30}
$$

Using (4.16), (4.17), (4.25) and (4.30), $G(t)$ from (4.15) can be written as

$$
\begin{aligned}
G(t) = 2e^{-\delta t}\Big[& -\lambda_1^2\lambda_5^2\hat{\mathrm{I}}_1(\lambda_1 t) * \hat{\mathrm{I}}_1(\lambda_5 t) + \lambda_1^3\hat{\mathrm{I}}_2(\lambda_1 t) + \lambda_5^3\hat{\mathrm{I}}_2(\lambda_5 t) + \\
& b^2\lambda_5^2\left\{\mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\right\} * \hat{\mathrm{I}}_1(\lambda_5 t) - b^2\lambda_1\mathrm{I}_1(\lambda_1 t)\mathrm{u}(t) + \frac{b^4}{2\lambda_1}\sinh(\lambda_1 t)\mathrm{u}(t)\Big].
\end{aligned} \tag{4.31}
$$

4.2.2 Final expression for $G(t)$

Just as in the TE case in Chapter 3, there are cases where λ_1 or λ_5 can be found to be imaginary and the modified Bessel functions in (4.31) can be replaced with ordinary Bessel functions. These cases occur for the same material properties as in the TE case. Case 1 occurs when $\omega_0^2 > \delta^2$. In this case both λ_1 and λ_5 are purely imaginary and defined as in (3.54). Then, using the property defined in (3.55) to replace modified Bessel functions with standard Bessel functions and noting that

$$
\sinh(jx) = j\sin(x), \tag{4.32}
$$

it can be seen that

$$
\begin{aligned}
G(t) = 2e^{-\delta t}\Big[& -\overline{\lambda}_1^2\overline{\lambda}_5^2\hat{\mathrm{J}}_1(\overline{\lambda}_1 t) * \hat{\mathrm{J}}_1(\overline{\lambda}_5 t) + \overline{\lambda}_1^3\hat{\mathrm{J}}_2(\overline{\lambda}_1 t) + \overline{\lambda}_5^3\hat{\mathrm{J}}_2(\overline{\lambda}_5 t) - \\
& b^2\overline{\lambda}_5^2\left\{\mathrm{J}_0(\overline{\lambda}_1 t)\mathrm{u}(t)\right\} * \hat{\mathrm{J}}_1(\overline{\lambda}_5 t) + b^2\overline{\lambda}_1\mathrm{J}_1(\overline{\lambda}_1 t)\mathrm{u}(t) + \frac{b^4}{2\overline{\lambda}_1}\sin(\overline{\lambda}_1 t)\mathrm{u}(t)\Big],
\end{aligned} \tag{4.33}
$$

where $\hat{\mathrm{J}}_n(x)$is defined in (3.57).

Case 2 occurs when $\omega_0^2 + B^2 < \delta^2$. In this case, both λ_1 and λ_5 are purely real and the same expression from (4.31) can be used. Case 3 occurs when $-B^2 < \omega_0^2 - \delta^2 < 0$. In this case, λ_1 is purely real and λ_5 is purely imaginary. $\overline{\lambda}_5$ is defined as it was in (3.54) and the expression for $G(t)$ can be written as

$$G(t) = 2e^{-\delta t}\Big[\lambda_1^2\overline{\lambda}_5^2\hat{\mathrm{I}}_1(\lambda_1 t) * \hat{\mathrm{J}}_1(\overline{\lambda}_5 t) + \lambda_1^3\hat{\mathrm{I}}_2(\lambda_1 t) + \overline{\lambda}_5^3\hat{\mathrm{J}}_2(\overline{\lambda}_5 t) - \\ b^2\overline{\lambda}_5^2\left\{\mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\right\} * \hat{\mathrm{J}}_1(\overline{\lambda}_5 t) - b^2\lambda_1\mathrm{I}_1(\lambda_1 t)\mathrm{u}(t) + \frac{b^4}{2\lambda_1}\sinh(\lambda_1 t)\mathrm{u}(t)\Big]. \quad (4.34)$$

4.2.3 Inversion of the $C(s)$ Term

The $C(s)$ term is defined in (4.14) as

$$C(s) = \frac{(s-s_1)(s-s_2)}{(s-s_A)(s-s_B)}. \quad (4.35)$$

Through algebraic manipulation and partial fraction expansion, it can be rewritten into an invertible form as

$$\begin{aligned} C(s) &= 1 + \frac{(s-s_1)(s-s_2) - (s-s_A)(s-s_B)}{(s-s_A)(s-s_B)} \\ &= 1 + \frac{(s^2+2\delta s+\omega_0^2) - (s^2+2\delta s+\omega_0^2 - \frac{b^2}{\tan^2\theta - 1})}{(s-s_A)(s-s_B)} \\ &= 1 + \frac{b^2}{(\tan^2\theta - 1)(s-s_A)(s-s_B)} \\ &= 1 + \frac{b^2}{2\lambda_A(\tan^2\theta - 1)}\left[\frac{1}{s-s_A} - \frac{1}{s-s_B}\right]. \end{aligned} \quad (4.36)$$

The inversion of $C(s)$ depends on the values of s_A and s_B. If $\omega_0^2 > b^2/(\tan^2\theta - 1)$, $C(s)$ can be inverted using the identity (3.27) into

$$C(s) \longleftrightarrow C(t) = \delta(t) + \frac{b^2}{2\lambda_A(\tan^2\theta - 1)}[e^{s_A t} - e^{s_B t}]\mathrm{u}(t)$$

$$= \delta(t) + \frac{b^2 e^{-\delta t}}{\lambda_A(\tan^2\theta - 1)} \sinh(\lambda_A t)\mathrm{u}(t). \tag{4.37}$$

If $\omega_0^2 > \delta^2 + b^2/(\tan^2\theta - 1)$, λ_A is purely imaginary and defined as

$$-j\lambda_A = \overline{\lambda}_A = \sqrt{\omega_0^2 - \delta^2 - \frac{b^2}{\tan^2\theta - 1}}. \tag{4.38}$$

In this case, using (4.32) $C(t)$ can be simplified into

$$C(t) = \delta(t) + \frac{b^2 e^{-\delta t}}{\overline{\lambda}_A(\tan^2\theta - 1)} \sin(\overline{\lambda}_A t)\mathrm{u}(t). \tag{4.39}$$

If $\omega_0^2 > \delta^2 + b^2/(\tan^2\theta - 1)$, λ_A is real and less than δ. In this case, (4.37) can be used directly. If $\omega_0^2 < b^2/(\tan^2\theta - 1)$, λ_A is real and greater than δ. In this case $Re\{s_A\} > 0$, which requires the following inverse Laplace transform identity to be used [14]:

$$\frac{1}{s - \beta} \longleftrightarrow -e^{\beta t}\mathrm{u}(-t). \tag{4.40}$$

Using this identity and the identity defined previously in (3.27), $C(t)$ can be written as

$$\begin{aligned} C(t) &= \delta(t) + \frac{b^2 e^{-\delta t}}{2\lambda_A(\tan^2\theta - 1)} \left[-e^{\lambda_A t}\mathrm{u}(-t) - e^{-\lambda_A t}\mathrm{u}(t) \right] \\ &= \delta(t) - \frac{b^2 e^{-\delta t}}{2\lambda_A(\tan^2\theta - 1)} e^{-\lambda_A |t|}. \end{aligned} \tag{4.41}$$

The crucial issue to address with this decomposition is the fact that this term is non-causal (ie, has non-zero value for $t < 0$). Further examination of the terms in question show that this case can only occur at angles greater than 45°and between some upper angle limit determined by the material properties. It appears that this non-causality is due to a problem with the model for ϵ_r. A true plane wave is a non-physical construct that is convenient in many cases, but does not exist. When an infinite plane wave is obliquely incident on an infinite half-space, it can be seen that there is no point in time when the wave is not intersecting the half-space. This has the potential to cause problems with causality such as in this case. Ultimately, the non-causal term exists in this case solely due to this fact.

Using the final expressions for $G(t)$ and $C(t)$, the final expression for the reflection coefficient can be written as

$$\Gamma(t) = \frac{1}{b^2(\tan^2\theta - 1)} C(t) * G(t). \tag{4.42}$$

4.3 Numerical Results

In order to validate the expressions derived in the previous section, the time-domain reflection coefficient is evaluated numerically in Fortran, and then compared to the inverse FFT of the frequency domain reflection coefficient given in (4.5). The Fortran code is included in Appendix B. The inverse FFT was done using WaveCalc. The frequency-domain data was zero-padded up to the maximum limit allowed by WaveCalc, 32,768, before the inverse FFT was taken. Since the signal had already decayed to zero by this point, no windowing was necessary. A set of parameters corresponding to each of the three possible cases for $G(t)$ is used. The angle is then adjusted for the case 1 parameters to produce results that contain a non-causal contribution. The number of points and step size for the results varied between the different cases.

The first set of parameters is the same as those chosen by Brillouin: $\omega_0 = 4.0 \times 10^{16}\ s^{-1}$, $b^2 = 20.0 \times 10^{32}\ s^{-2}$, $\delta = 0.28 \times 10^{16}\ s^{-1}$. This choice of parameters corresponds to case 1. When computing the numerical results, for the frequency-domain data 16,384 frequency points were calculated with a step size of 8,000 GHz. In the time-domain, 4,096 points were calculated with a step size of 1×10^{-9} ns. Using $\theta = 30°$, (4.33) has been plotted in Figure 4.1 and compared to the inverse FFT. The results show excellent agreement. Since this function includes only standard Bessel functions, which are highly oscillatory, and no modified Bessel functions, which are not oscillatory, the waveform is highly oscillatory and only lightly damped.

The next choice of parameters is: $\omega_0 = 2.0 \times 10^{15}\ s^{-1}$, $b^2 = 20.0 \times 10^{29}\ s^{-2}$, $\delta = 0.28 \times 10^{16}\ s^{-1}$, which corresponds to case 2, (4.31). When computing the numerical results, for the frequency-domain data 32,768 frequency points were calculated with a step size of 400 GHz. In the time-domain, 4,096 points were calculated with a step size of 1×10^{-9} ns. The results are shown in Figure 4.2. Again, the closed-form expression and the inverse FFT compare well. For this choice of parameters $\delta^2 > \omega_0^2 + B^2$, and the resulting waveform is overdamped, showing no oscillatory behavior and only a single negative peak. Since the expression for case 2 only involves modified Bessel functions, which do not have the oscillatory behavior of ordinary Bessel functions, this observed behavior is easily predicted from the mathematical form of the expression.

The next choice of parameters, which corresponds to case 3, is: $\omega_0 = 2.0 \times 10^{15}\ s^{-1}$, $b^2 = 20.0 \times 10^{32}\ s^{-2}$, $\delta = 0.28 \times 10^{16}\ s^{-1}$, which corresponds to case 3, (4.34). When computing the numerical results, for the frequency-domain data 32,768 frequency points were calculated with a step size of 6,000 GHz. In the time-domain, 16,384 points were calculated with a step size of 1×10^{-10} ns. The analytic expression again matches the inverse FFT, as seen in Figure 4.3. As expected, since $\delta > \omega_0$, but $\delta^2 < \omega_0^2 + B^2$, there is more damping and less oscillation than with case 1, but more oscillation than with case 2. Here the expression for the reflection coefficient has a combination of ordinary and modified Bessel functions.

To examine the non-causal result which is possible for the $C(t)$ term, Brillouin's choice of parameters are used again, but at an angle of 50°. This causes the $C(t)$ term to be non-causal as shown in (4.41). When computing the numerical results, for the frequency-domain data 16,384 frequency points were calculated with a step size of 8,000 GHz. In the time-domain, 4,096 points were calculated with a step size of 1×10^{-9} ns with a starting time value of -4.096×10^{-6} ns. As seen in Figure 4.4, the expression again matches well with the inverse FFT for $t > 0$. For $t < 0$, non-causal term in the analytic expression is extremely small. Unfortunately, the inverse FFT does not have the necessary resolution to capture the small non-causal portion of the reflection coefficient, so it cannot be compared to the analytic expression.

In order to have a better insight into the non-causal term, the delta function term was subtracted from the non-causal C-term and plotted in Figure 4.5. It can immediately be seen that this term is extremely small when compared to the final transient field. In order to examine the non-causality even further, this term is convolved with $G(t)$, as it is in the final expression for the reflection coefficient, and plotted in Figure 4.6. Just as in Figure 4.4, the non-causal term is not even visible in this plot. Taking a closer view of this plot in Figure 4.7 it can be seen that the non-causal portion of the signal is around 3-orders of magnitude less than the causal portion.

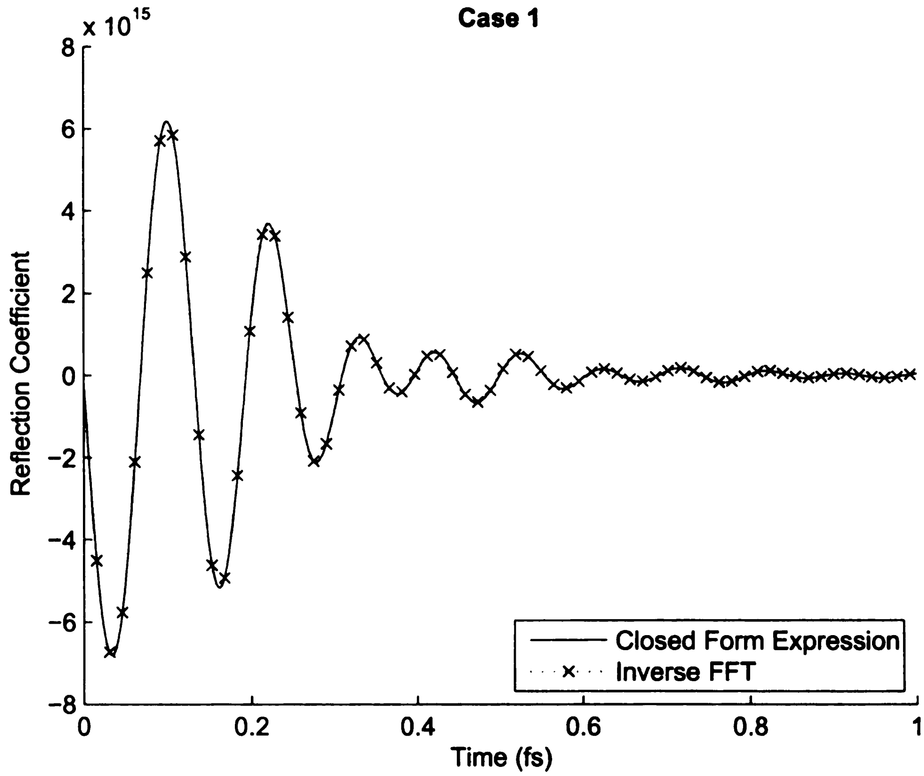

Figure 4.1. Time-domain reflection coefficient with incidence angle $\theta = 30°$ and material parameters $\omega_0 = 4.0 \times 10^{16} s^{-1}$, $b^2 = 20.0 \times 10^{32} s^{-2}$, $\delta = 0.28 \times 10^{16} s^{-1}$. This choice of parameters corresponds to case 1, Eq. (4.33).

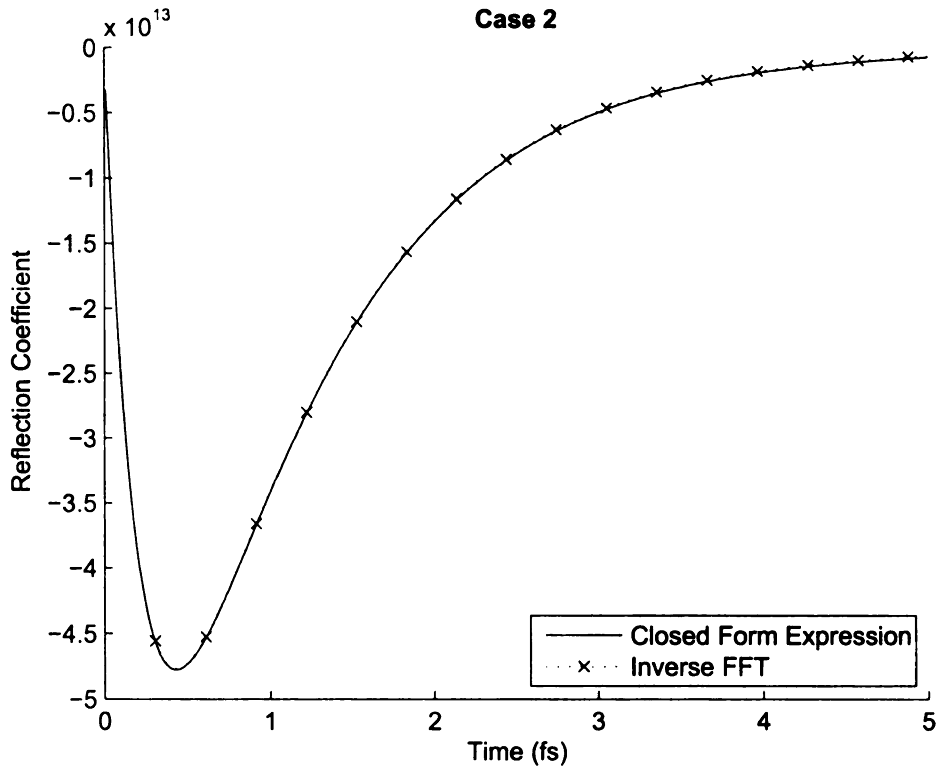

Figure 4.2. Time-domain reflection coefficient with incidence angle $\theta = 30^\circ$ and material parameters $\omega_0 = 2.0 \times 10^{15}s^{-1}$, $b^2 = 20.0 \times 10^{29}s^{-2}$, $\delta = 0.28 \times 10^{16}s^{-1}$. This choice of parameters corresponds to case 2, Eq. (4.31).

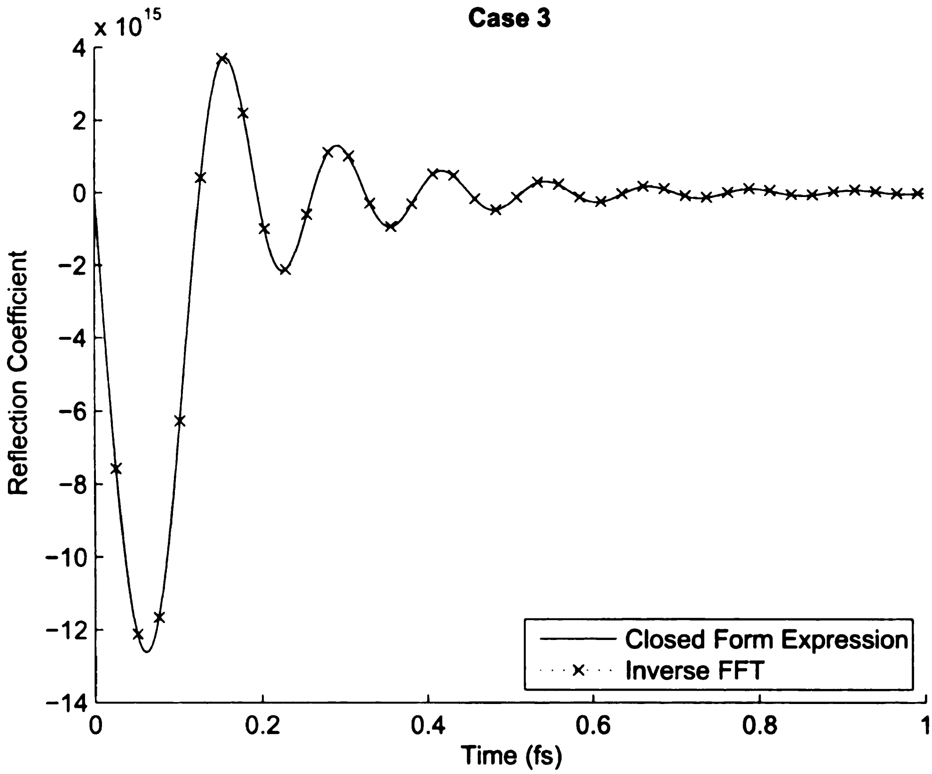

Figure 4.3. Time-domain reflection coefficient with at an angle of $\theta = 30°$ with parameter choices of $\omega_0 = 2.0 \times 10^{15}s^{-1}$, $b^2 = 20.0 \times 10^{32}s^{-2}$, $\delta = 0.28 \times 10^{16}s^{-1}$. This choice of parameters corresponds to case 3, Eq. (4.34).

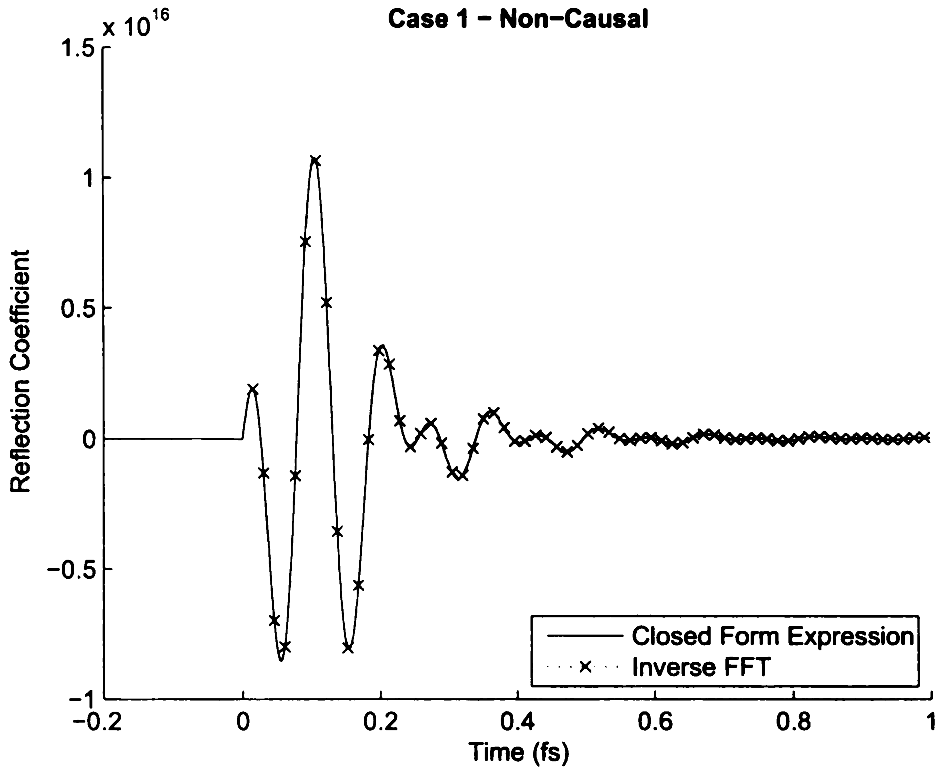

Figure 4.4. Time-domain reflection coefficient with incidence angle $\theta = 50°$ and material parameters $\omega_0 = 4.0 \times 10^{16} s^{-1}$, $b^2 = 20.0 \times 10^{32} s^{-2}$, $\delta = 0.28 \times 10^{16} s^{-1}$. This choice of parameters corresponds to case 1, Eq. (4.33) with non-causal term from Eq. (4.41).

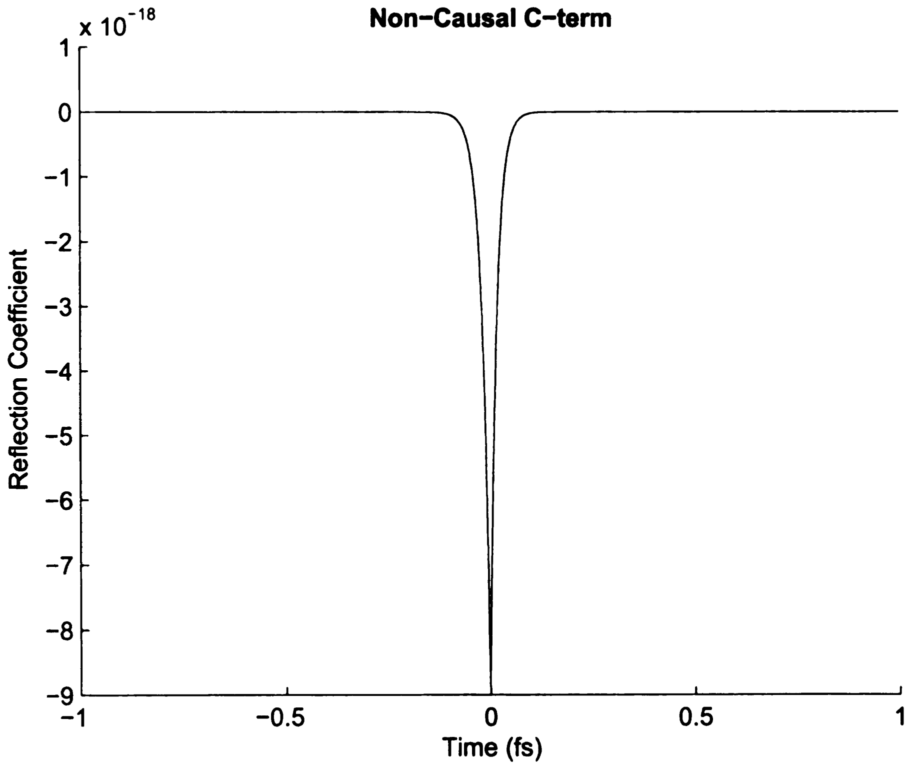

Figure 4.5. Non-causal $C(t) - \delta(t)$ for case 1 results.

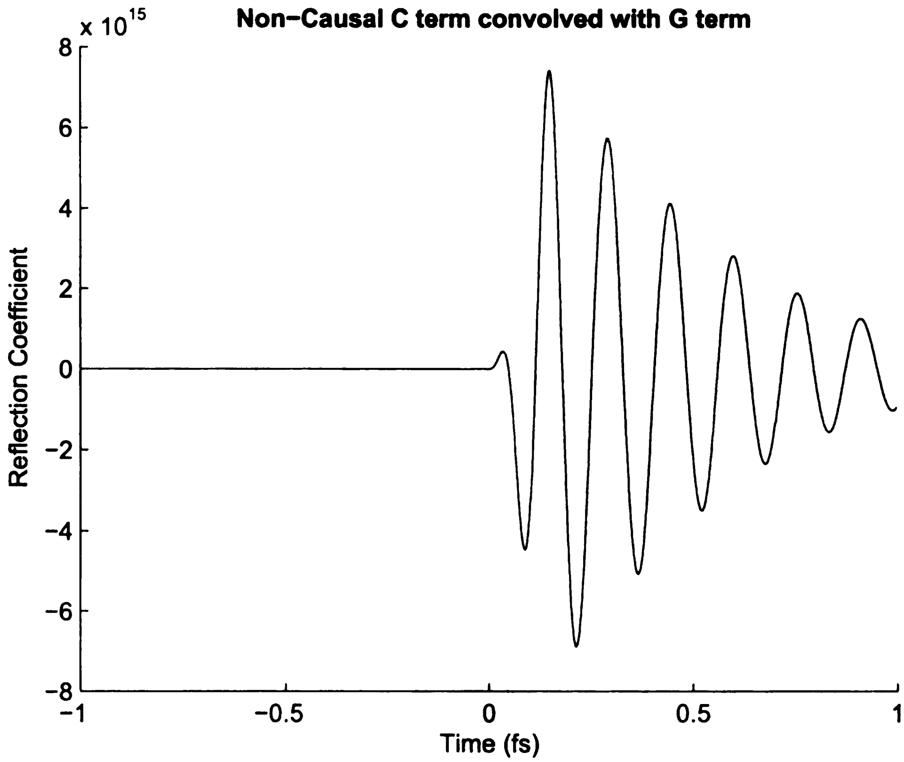

Figure 4.6. Non-causal $[C(t) - \delta(t)] * G(t)$ for case 1 results.

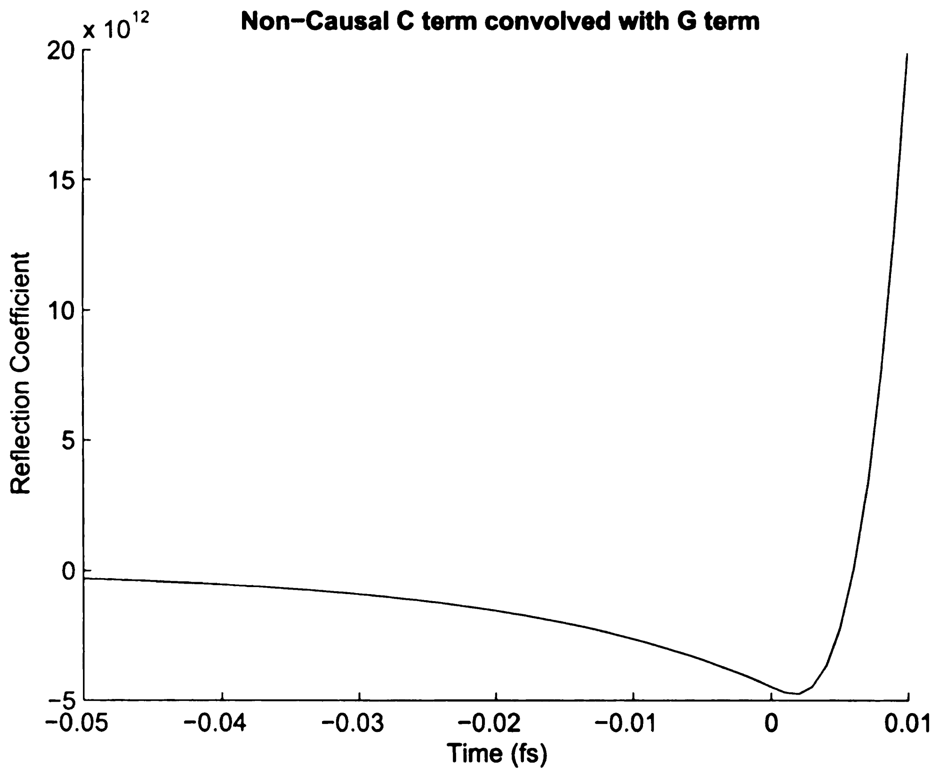

Figure 4.7. Close-up of the non-causal $[C(t) - \delta(t)] * G(t)$ for case 1 results.

CHAPTER 5

TE PLANE WAVE REFLECTION FOR THE GENERAL CASE

In Chapter 3, the transient reflection of a TE plane-wave from the Lorentz-medium half-space is examined for a Lorentz-medium simplified to the optical case. In this chapter, the reflection coefficient will be examined for the most general case. As in the Chapter 3, a steady-state TE-polarized plane wave of frequency ω is obliquely incident on an interface separating free space (region 1) from a homogeneous Lorentz medium (region 2). The angle of incidence θ is measured from the normal to the interface. This geometry can be seen in Figure 2.2. As shown in Chapter 2, the reflection coefficient for a plane wave from a material half-space can be expressed as

$$\Gamma_{\perp}(\omega) = \frac{Z_{\perp}(\omega) - Z_0}{Z_{\perp}(\omega) + Z_0}. \tag{5.1}$$

For a TE-polarized plane wave, the impedance of the incident wave is $Z_0 = \eta_0 / \cos\theta$ and the wave impedance of the transmitted wave is given by

$$Z_{\perp}(\omega) = \frac{k(\omega)\eta(\omega)}{k_z(\omega)}, \tag{5.2}$$

where the terms k_z, η, and k are defined in (3.3). The relative permittivity is the same as given in (3.4). For the general case, the simplifications made in Chapter 3 cannot be made, but the relative permittivity can be rewritten as

$$\epsilon_r(\omega) = \epsilon_\infty + \frac{b^2}{\omega_0^2 - \omega^2 + 2j\omega\delta}, \tag{5.3}$$

where b is the plasma frequency of the medium.

5.1 Laplace Domain Representation

Equation (5.3) can be represented in the Laplace domain as

$$\epsilon_r(s) = \epsilon_\infty + \frac{b^2}{s^2 + 2\delta s + \omega_0^2}. \tag{5.4}$$

Using this, the wave number in the Lorentz medium can then be calculated as

$$\begin{aligned} k(s) &= -js\sqrt{\mu\epsilon} \\ &= -js\sqrt{\mu\epsilon_0}\sqrt{\epsilon_\infty + \frac{b^2}{s^2 + 2\delta s + \omega_0^2}} \\ &= -js\sqrt{\mu\epsilon_0\epsilon_\infty}\sqrt{\frac{s^2 + 2\delta s + \omega_0^2 + b^2/\epsilon_\infty}{s^2 + 2\delta s + \omega_0^2}}. \end{aligned} \tag{5.5}$$

For a TE-polarized plane wave, the Laplace domain reflection coefficient can then be written as

$$\begin{aligned} \Gamma(s) &= \frac{\mu_r k_{z0} - k_z}{\mu_r k_{z0} + k_z} \\ &= \frac{\mu_r \cos\theta - \sqrt{(k/k_0)^2 - \sin^2\theta}}{\mu_r \cos\theta + \sqrt{(k/k_0)^2 - \sin^2\theta}} \\ &= \frac{\mu_r \cos\theta - \sqrt{\mu_r\epsilon_\infty(s - s_3)(s - s_4)/[(s - s_1)(s - s_2)] - \sin^2\theta}}{\mu_r \cos\theta + \sqrt{\mu_r\epsilon_\infty(s - s_3)(s - s_4)[(s - s_1)(s - s_2)] - \sin^2\theta}}, \end{aligned} \tag{5.6}$$

where $s_{1,2}$ are the same as defined in (3.9) and $s_{3,4}$ are defined as

$$s_{3,4} = -\delta \pm \sqrt{\delta^2 - \omega_0^2 - \frac{b^2}{\epsilon_\infty}}, \tag{5.7}$$

which is derived from the fact that

$$s^2 + 2\delta s + \omega_0^2 + \frac{b^2}{\epsilon_\infty} = (s - s_3)(s - s_4). \tag{5.8}$$

Multiplying the numerator and denominator in (5.6) by $\sqrt{(s-s_1)(s-s_2)}$ and expanding $(s-s_1)(s-s_2)$ and $(s-s_3)(s-s_4)$ leads to

$$\Gamma(s) = \frac{\mu_r \cos\theta\sqrt{(s-s_1)(s-s_2)} - \sqrt{\mu_r\epsilon_\infty(s-s_3)(s-s_4) - \sin^2\theta(s-s_1)(s-s_2)}}{\mu_r \cos\theta\sqrt{(s-s_1)(s-s_2)} + \sqrt{\mu_r\epsilon_\infty(s-s_3)(s-s_4) - \sin^2\theta(s-s_1)(s-s_2)}}$$
$$= \Big[\mu_r\cos\theta\sqrt{(s-s_1)(s-s_2)} - \Big(s^2(\mu_r\epsilon_\infty - \sin^2\theta) + 2\delta s(\mu_r\epsilon_\infty - \sin^2\theta) +$$
$$\omega_0^2(\mu_r\epsilon_\infty - \sin^2\theta) + \mu_r b^2\Big)^{1/2}\Big] / \Big[\mu_r\cos\theta\sqrt{(s-s_1)(s-s_2)} +$$
$$\Big(s^2(\mu_r\epsilon_\infty - \sin^2\theta) + 2\delta s(\mu_r\epsilon_\infty - \sin^2\theta) + \omega_0^2(\mu_r\epsilon_\infty - \sin^2\theta) + \mu_r b^2\Big)^{1/2}\Big]. \tag{5.9}$$

For convenience, define

$$K^2 = \mu_r\epsilon_\infty - \sin^2\theta. \tag{5.10}$$

This allows (5.9) to be written as

$$\Gamma(s) = \frac{\mu_r\cos\theta\sqrt{(s-s_1)(s-s_2)} - K\sqrt{s^2 + 2\delta s + \omega_0^2 + \mu_r b^2/K^2}}{\mu_r\cos\theta\sqrt{(s-s_1)(s-s_2)} + K\sqrt{s^2 + 2\delta s + \omega_0^2 + \mu_r b^2/K^2}}. \tag{5.11}$$

It is then convenient to make the substitution

$$s^2 + 2\delta s + \omega_0^2 + \mu_r\frac{b^2}{K^2} = (s-s_5)(s-s_6), \tag{5.12}$$

where

$$s_{5,6} = -\delta \pm \sqrt{\delta^2 - \omega_0^2 - \mu_r\frac{b^2}{K^2}}. \tag{5.13}$$

Substituting (5.12) into (5.11) allows the reflection coefficient in the Laplace domain to be rewritten as

$$\Gamma(s) = \frac{\mu_r\cos\theta\sqrt{s-s_1}\sqrt{s-s_2} - K\sqrt{s-s_5}\sqrt{s-s_6}}{\mu_r\cos\theta\sqrt{s-s_1}\sqrt{s-s_2} + K\sqrt{s-s_5}\sqrt{s-s_6}}. \tag{5.14}$$

5.2 Time-Domain Reflection Coefficient

Using the frequency-domain reflection coefficient found in Section 5.1, the time-domain reflection coefficient can be found using an inverse Laplace transform. In order to perform the inverse Laplace transform, the frequency-domain reflection coefficient may be rearranged into a more manageable form. Before this can be done however, it must be noted that as $s \longrightarrow \infty$, the reflection coefficient does not tend to zero. Instead it can be seen that

$$\text{as} \qquad s \longrightarrow \infty \qquad \Gamma(s) \longrightarrow \frac{\mu_r \cos\theta - K}{\mu_r \cos\theta + K}. \tag{5.15}$$

This term results in a delta function contribution in the final time-domain reflection coefficient. In order to perform the inverse Laplace transform, this term, defined as Γ_∞, needs to be subtracted from the total reflection coefficient. This leads to

$$\begin{aligned}
\widetilde{\Gamma}(s) &= \Gamma(s) - \Gamma_\infty \\
&= \frac{\mu_r \cos\theta\sqrt{s-s_1}\sqrt{s-s_2} - K\sqrt{s-s_5}\sqrt{s-s_6}}{\mu_r \cos\theta\sqrt{s-s_1}\sqrt{s-s_2} + K\sqrt{s-s_5}\sqrt{s-s_6}} - \frac{\mu_r\cos\theta - K}{\mu_r\cos\theta + K} \\
&= \frac{2K\mu_r\cos\theta\left[\sqrt{s-s_1}\sqrt{s-s_2} - \sqrt{s-s_5}\sqrt{s-s_6}\right]}{(\mu_r\cos\theta + K)\left(\mu_r\cos\theta\sqrt{s-s_1}\sqrt{s-s_2} + K\sqrt{s-s_5}\sqrt{s-s_6}\right)}.
\end{aligned} \tag{5.16}$$

Rationalizing the denominator leads to

$$\begin{aligned}
\widetilde{\Gamma}(s) = &\Bigg[2K\mu_r\cos\theta\left(\sqrt{s-s_1}\sqrt{s-s_2} - \sqrt{s-s_5}\sqrt{s-s_6}\right)\left(\mu_r\cos\theta\sqrt{s-s_1}\sqrt{s-s_2} - \right. \\
&\left. K\sqrt{s-s_5}\sqrt{s-s_6}\right)\Bigg] \Big/ \Bigg[(\mu_r\cos\theta + K)\left[\mu_r^2\cos^2\theta(s-s_1)(s-s_2) - \right. \\
&\left. K^2(s-s_5)(s-s_6)\right]\Bigg] \\
= &\, F_1\frac{N}{D},
\end{aligned} \tag{5.17}$$

where

$$F_1 = \frac{2K\mu_r \cos\theta}{\mu_r \cos\theta + K}. \tag{5.18}$$

Expanding the terms in the denominator and examining them in further detail, it can be seen that

$$\begin{aligned}
D &= \mu_r^2 \cos^2\theta(s^2 + 2\delta s + \omega_0^2) - K^2(s^2 + 2\delta s + \omega_0^2 + \frac{\mu_r b^2}{K^2}) \\
&= s^2(\mu_r^2 \cos^2\theta - K^2) + 2\delta s(\mu_r^2 \cos^2\theta - K^2) + \omega_0^2(\mu_r^2 \cos^2\theta - K^2) - \mu_r b^2 \\
&= (\mu_r^2 \cos^2\theta - K^2)\left(s^2 + 2\delta s + \omega_0^2 - \frac{\mu_r b^2}{\mu_r^2 \cos^2\theta - K^2}\right) \\
&= (\mu_r^2 \cos^2\theta - K^2)\left(s^2 + 2\delta s + \gamma^2\right),
\end{aligned} \tag{5.19}$$

with the definition

$$\gamma^2 = \omega_0^2 - \frac{\mu_r b^2}{\mu_r^2 \cos^2\theta - K^2}. \tag{5.20}$$

At this point, it is convenient to define

$$s^2 + 2\delta s + \gamma^2 = (s - s_C)(s - s_D), \tag{5.21}$$

where

$$\begin{aligned}
s_{C,D} &= -\delta \pm \sqrt{\delta^2 - \gamma^2} \\
&= -\delta \pm \lambda_C.
\end{aligned} \tag{5.22}$$

Substituting (5.21) into (5.19) allows the denominator to be written in its final form as

$$D = (\mu_r^2 \cos^2\theta - K^2)(s - s_C)(s - s_D). \tag{5.23}$$

Expanding the numerator and examining it in further detail leads to

$$\begin{aligned} N &= \mu_r \cos\theta \left[(s-s_1)(s-s_2) - \sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6}\right] + \\ &\quad K\left[(s-s_5)(s-s_6) - \sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6}\right] \\ &= \mu_r \cos\theta g_1(s) + K g_2(s). \end{aligned} \tag{5.24}$$

Applying the following definition

$$F = \frac{2K\mu_r \cos\theta}{(\mu_r\cos\theta + K)^2(\mu_r\cos\theta - K)}, \tag{5.25}$$

allows the Laplace domain reflection coefficient to be rewritten by substituting (5.23) and (5.24) into (5.17) as

$$\begin{aligned} \widetilde{\Gamma}(s) &= F\frac{\mu_r\cos\theta g_1(s) + K g_2(s)}{(s-s_C)(s-s_D)} \\ &= FC(s)G(s), \end{aligned} \tag{5.26}$$

where it is defined that

$$C(s) = \frac{1}{(s-s_C)(s-s_D)}, \tag{5.27a}$$

$$G(s) = \mu_r\cos\theta g_1(s) + K g_2(s). \tag{5.27b}$$

5.2.1 Inversion of the $G(s)$ Term

Both terms $g_1(s)$ and $g_2(s)$ in $G(s)$ were inverted previously in Section 3.2.1. The solutions are given as

$$g_1(t) = e^{-\delta t}\left[-\lambda_1^2\lambda_5^2\hat{\mathrm{I}}_1(\lambda_1 t) * \hat{\mathrm{I}}_1(\lambda_5 t) + \lambda_1^3\hat{\mathrm{I}}_2(\lambda_1 t) + \lambda_5^3\hat{\mathrm{I}}_2(\lambda_5 t)\right] - \frac{\lambda_1^2 - \lambda_5^2}{2}\delta(t), \tag{5.28}$$

$$g_2(t) = e^{-\delta t}\left[-\lambda_1^2\lambda_5^2\hat{\mathrm{I}}_1(\lambda_1 t) * \hat{\mathrm{I}}_1(\lambda_5 t) + \lambda_1^3\hat{\mathrm{I}}_2(\lambda_1 t) + \lambda_5^3\hat{\mathrm{I}}_2(\lambda_5 t)\right] + \frac{\lambda_1^2 - \lambda_5^2}{2}\delta(t). \tag{5.29}$$

In these equations, λ_1 and λ_5 are defined from

$$s_{1,2} = -\delta \pm \lambda_1, \tag{5.30}$$

$$s_{5,6} = -\delta \pm \lambda_5, \tag{5.31}$$

and $\hat{\mathrm{I}}_n(x)$ is defined in (3.51). The final expression for $G(t)$ can then be given as

$$\begin{aligned} G(t) = (\mu_r \cos\theta + K)e^{-\delta t}\left[-\lambda_1^2\lambda_5^2\hat{\mathrm{I}}_1(\lambda_1 t) * \hat{\mathrm{I}}_1(\lambda_5 t) + \lambda_1^3\hat{\mathrm{I}}_2(\lambda_1 t) + \lambda_5^3\hat{\mathrm{I}}_2(\lambda_5 t)\right] - \\ \frac{\mu_r b^2}{2K^2}(\mu_r \cos\theta - K)\delta(t). \end{aligned} \tag{5.32}$$

Just as in the optical TE case in Chapter 3, there are cases where λ_1 or λ_5 can be found to be imaginary and the modified Bessel functions in (5.32) can be replaced with ordinary Bessel functions. Case 1 occurs when $\omega_0^2 > \delta^2$. In this case both λ_1 and λ_5 are purely imaginary and defined as

$$-j\lambda_1 = \overline{\lambda}_1 = \sqrt{\omega_0^2 - \delta^2}, \qquad -j\lambda_5 = \overline{\lambda}_5 = \sqrt{\omega_0^2 + \frac{\mu_r b^2}{K^2} - \delta^2}. \tag{5.33}$$

Then, using the identity given in (3.55) to replace modified Bessel functions with ordinary Bessel functions and the identity given in (4.32), (5.32) can be expressed as

$$\begin{aligned} G(t) = (\mu_r \cos\theta + K)e^{-\delta t}\left[-\overline{\lambda}_1^2\overline{\lambda}_5^2\hat{\mathrm{J}}_1(\overline{\lambda}_1 t) * \hat{\mathrm{J}}_1(\overline{\lambda}_5 t) + \overline{\lambda}_1^3\hat{\mathrm{J}}_2(\overline{\lambda}_1 t) + \overline{\lambda}_5^3\hat{\mathrm{J}}_2(\overline{\lambda}_5 t)\right] - \\ \frac{\mu_r b^2}{2K^2}(\mu_r \cos\theta - K)\delta(t), \end{aligned} \tag{5.34}$$

where $\hat{\mathrm{J}}_n(x)$ was defined in (3.57).

Case 2 occurs when $\omega_0^2+\mu_r b^2/K^2 < \delta^2$. In this case, both λ_1 and λ_5 are purely real and the same expression from (5.32) can be used. Case 3 occurs when $-\mu_r b^2/K^2 < \omega_0^2 - \delta^2 < 0$. In this case, λ_1 is purely real and λ_5 is purely imaginary. $\overline{\lambda}_5$ is defined as it was in (5.33) and the expression for $G(t)$ can be written as

$$G(t) = (\mu_r \cos\theta + K)e^{-\delta t}\left[\lambda_1^2\overline{\lambda}_5^2\hat{\mathrm{I}}_1(\lambda_1 t) * \hat{\mathrm{J}}_1(\overline{\lambda}_5 t) + \lambda_1^3\hat{\mathrm{I}}_2(\lambda_1 t) + \overline{\lambda}_5^3\hat{\mathrm{J}}_2(\overline{\lambda}_5 t)\right] - \frac{\mu_r b^2}{2K^2}(\mu_r \cos\theta - K)\delta(t). \tag{5.35}$$

5.2.2 Inversion of the $C(s)$ Term

Using partial fraction expansion, $C(s)$ can be rearranged into

$$\begin{aligned} C(s) &= \frac{1}{(s-s_C)(s-s_D)} \\ &= \frac{1}{2\lambda_C}\left(\frac{1}{s-s_C} - \frac{1}{s-s_D}\right). \end{aligned} \tag{5.36}$$

As in previous cases, the values of s_C and s_D affect the inversion of $C(s)$. If $\gamma^2 > 0$, then $Re\{s_{C,D}\} < 0$ and the inversion identity (3.27) can be used. Using this identity, $C(t)$ is found to be

$$\begin{aligned} C(t) &= \frac{e^{-\delta t}}{2\lambda_C}\left(e^{\lambda_c t} - e^{-\lambda_c t}\right)\mathrm{u}(t) \\ &= \frac{e^{-\delta t}}{\lambda_C}\sinh(\lambda_C t)\mathrm{u}(t). \end{aligned} \tag{5.37}$$

If $\gamma^2 < \delta^2$, λ_C is purely real and (5.37) can be used directly. If $\gamma^2 > \delta^2$, λ_C is purely imaginary and can be defined as

$$-j\lambda_C = \overline{\lambda}_C = \sqrt{\gamma^2 - \delta^2}. \tag{5.38}$$

In this case, using the identity given in (4.32), (5.37) can be rewritten into

$$C(t) = \frac{e^{-\delta t}}{\overline{\lambda}_C} \sin(\overline{\lambda}_C t)\mathrm{u}(t). \tag{5.39}$$

If $\gamma^2 < 0$, λ_C is again purely real, but in this case, $Re\{s_c\} > 0$, which requires the identity given in (4.40) to be used. In this case, $C(t)$ is given as

$$\begin{aligned} C(t) &= -\frac{e^{-\delta t}}{2\lambda_C}\left(e^{\lambda_C t}\mathrm{u}(-t) + e^{-\lambda_C t}\mathrm{u}(t)\right) \\ &= -\frac{e^{-\delta t}}{2\lambda_C} e^{-\lambda_C |t|}. \end{aligned} \tag{5.40}$$

As in Chapter 4 for the TM optical case, there is a problem with causality in this term, however an interesting result can be observed by examining the γ^2 term. In order for $\gamma^2 < 0$ to occur, the following relationship must hold:

$$\omega_0^2 < \frac{\mu_r b^2}{\mu_r^2 \cos^2\theta - K^2}. \tag{5.41}$$

Examining the denominator on the right side of the equation in closer detail reveals that as long as $\mu_r < \epsilon_\infty$, the denominator is negative at any angle. Since this is not the optical case, it holds that $\epsilon_\infty > 1$. The Lorentz model is generally used for a dielectric material, so it is expected that $\mu_r < \epsilon_\infty$ will hold in general. This implies that the causality issue is not a concern in this case.

5.3 Numerical Results

In order to validate the expressions derived in the previous section, the time-domain reflection coefficient is evaluated numerically in Fortran, and then compared to the inverse FFT of the frequency-domain reflection coefficient given in (5.16). The Fortran code is included in Appendix C. The inverse FFT was done using WaveCalc.

The frequency-domain data was zero-padded up to the maximum limit allowed by WaveCalc, 32,768, before the inverse FFT was taken. Since the signal had already decayed to zero by this point, no windowing was necessary. A set of parameters corresponding to each of the three possible cases is used.

The first set of parameters chosen are: $\omega_0 = 4.0 \times 10^{11}\ s^{-1}$, $b = 6.24 \times 10^{11}\ s^{-1}$, $\delta = 2.5 \times 10^{10}\ s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$. This choice of parameters corresponds to case 1. When computing the numerical results, for the frequency-domain data 4,096 frequency points were calculated with a step size of 400 MHz. In the time-domain, 4,096 points were calculated with a step size of 1×10^{-4} ns. Using $\theta = 30°$, (5.34) has been plotted in Figure 5.1 and compared to the inverse FFT. The results show excellent agreement. Since this function includes only standard Bessel functions, which are highly oscillatory, and no modified Bessel functions, which are not oscillatory, the waveform is highly oscillatory and only lightly damped.

The next choice of parameters is: $\omega_0 = 4.0 \times 10^{10}\ s^{-1}$, $b = 6.24 \times 10^{10}\ s^{-1}$, $\delta = 2.5 \times 10^{11}\ s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$, which corresponds to case 2, (5.32). When computing the numerical results, for the frequency-domain data 32,768 frequency points were calculated with a step size of 4 MHz. In the time-domain, 8,192 points were calculated with a step size of 2×10^{-4} ns. The results are shown in Figure 5.2. Again, the closed-form expression and the inverse FFT compare well. For this choice of parameters $\omega_0^2 + \mu_r b^2/K^2 < \delta^2$, and the resulting waveform is overdamped, showing no oscillatory behavior and only a single negative peak. Since the expression for case 2 only involves modified Bessel functions, which do not have the oscillatory behavior of ordinary Bessel functions, this observed behavior is easily predicted from the mathematical form of the expression.

The final choice of parameters, which corresponds to case 3, is: $\omega_0 = 4.0 \times 10^{9}\ s^{-1}$, $b = 6.24 \times 10^{11}\ s^{-1}$, $\delta = 2.5 \times 10^{10}\ s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$, which corresponds to case 3, (5.35). When computing the numerical results, for the frequency-domain data 8,192 frequency points were calculated with a step size of 200 MHz. In the time-domain, 4,096 points were calculated with a step size of 1×10^{-4} ns. The analytic expression again matches the inverse FFT, as seen in Figure 5.3. As expected, since $\delta^2 > \omega_0^2$, but $\delta^2 < \omega_0^2 + \mu_r b^2/K^2$, there is more damping and less oscillation than with case 1, but more oscillation than with case 2. Here the expression for the reflection coefficient has a combination of ordinary and modified Bessel functions.

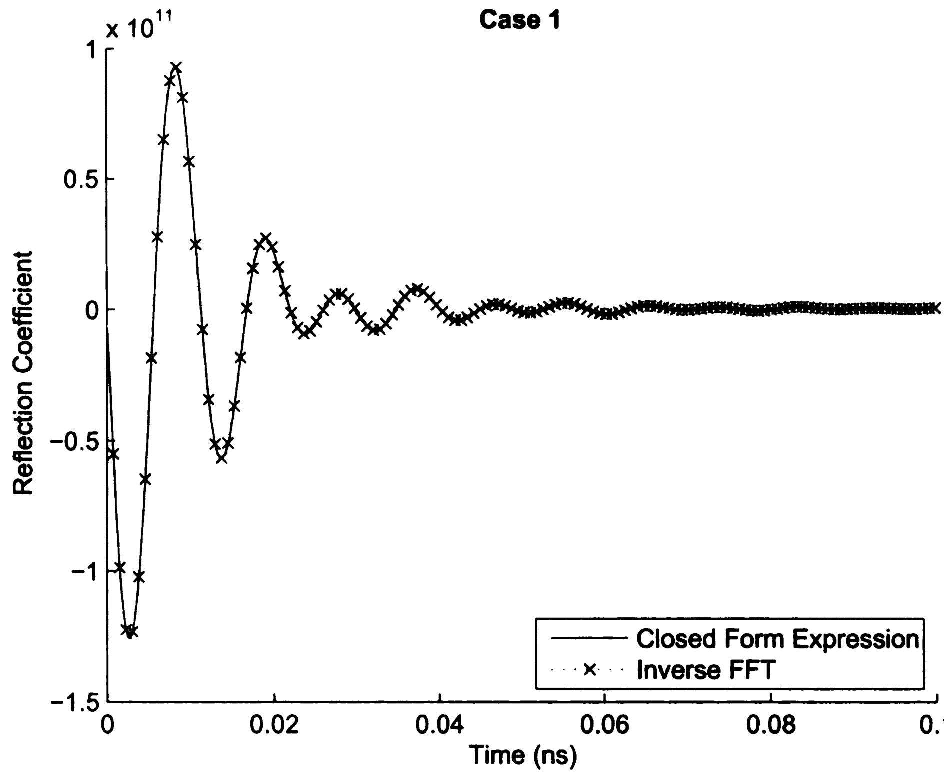

Figure 5.1. Time-domain reflection coefficient with incidence angle $\theta = 30^\circ$ and material parameters $\omega_0 = 4.0 \times 10^{11} s^{-1}$, $b = 6.24 \times 10^{11} s^{-1}$, $\delta = 2.5 \times 10^{10} s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$. This choice of parameters corresponds to case 1, Eq. (5.34).

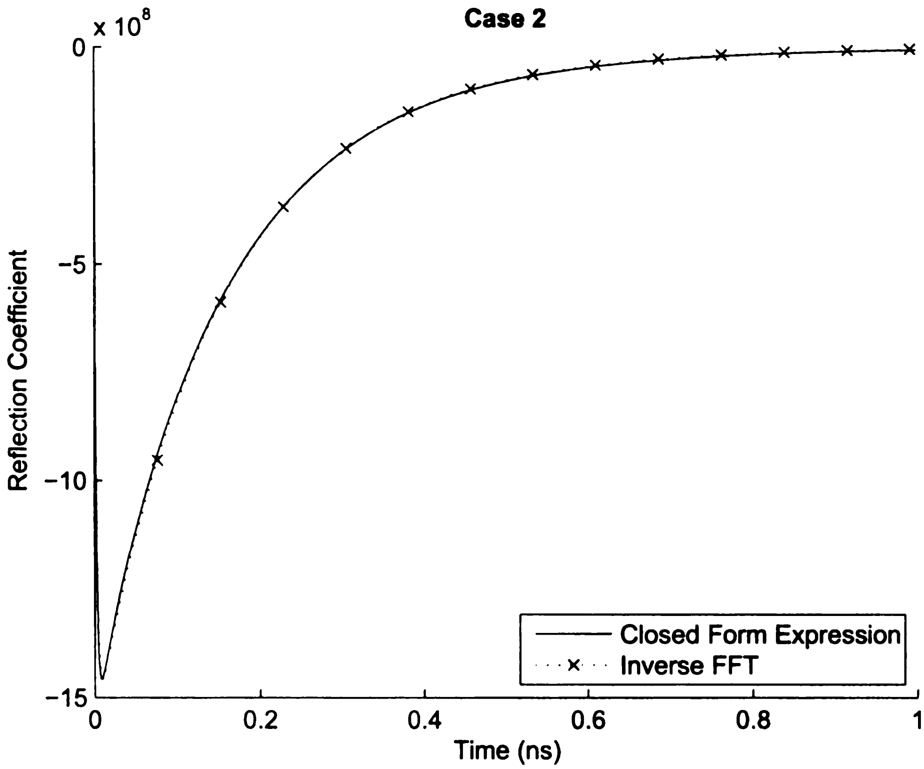

Figure 5.2. Time-domain reflection coefficient with incidence angle $\theta = 30^\circ$ and material parameters $\omega_0 = 4.0 \times 10^{10} s^{-1}$, $b = 6.24 \times 10^{10} s^{-1}$, $\delta = 2.5 \times 10^{11} s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$. This choice of parameters corresponds to case 2, Eq. (5.32).

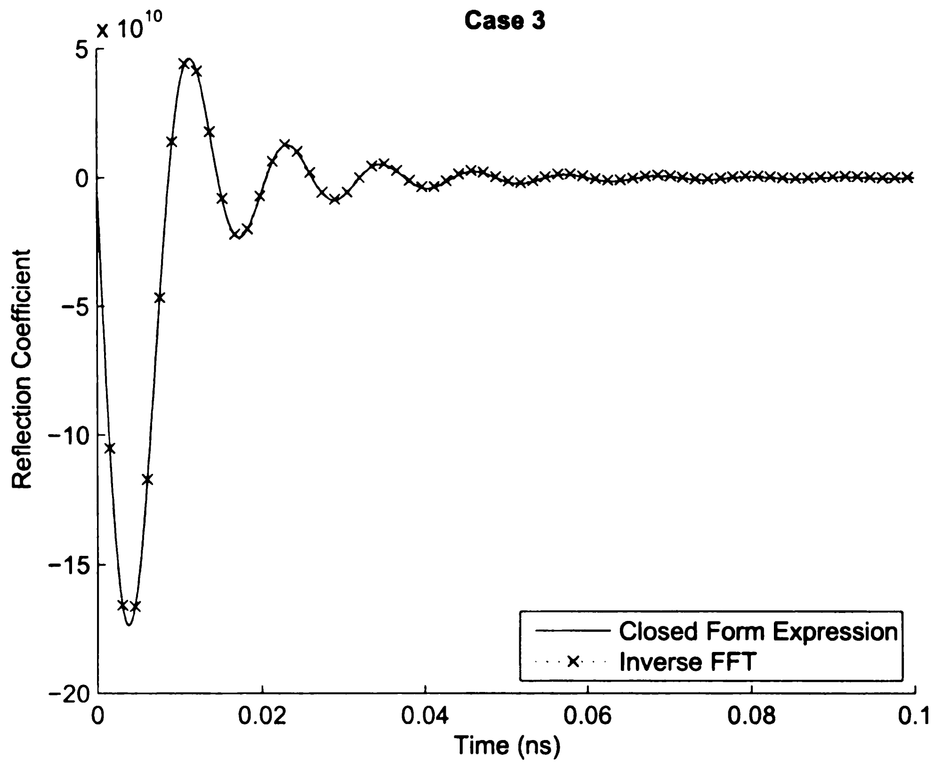

Figure 5.3. Time-domain reflection coefficient with at an angle of $\theta = 30°$ with parameter choices of $\omega_0 = 4.0 \times 10^9 s^{-1}$, $b = 6.24 \times 10^{11} s^{-1}$, $\delta = 2.5 \times 10^{10} s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$. This choice of parameters corresponds to case 3, Eq. (5.35).

CHAPTER 6

TM PLANE WAVE REFLECTION FOR THE GENERAL CASE

In Chapter 4, the transient reflection of a TM plane-wave from a Lorentz-medium half-space is examined for a Lorentz-medium simplified to the optical case. In this chapter, the reflection coefficient will be examined for the most general case. As in the Chapter 4, a steady-state TM-polarized plane wave of frequency ω is obliquely incident on an interface separating free space (region 1) from a homogeneous Lorentz medium (region 2). The angle of incidence θ is measured from the normal to the interface. This geometry can be seen in Figure 2.3. As shown in Chapter 2, the reflection coefficient for a plane wave from a material half-space can be expressed as

$$\Gamma_{\|}(\omega) = \frac{Z_{\|}(\omega) - Z_0}{Z_{\|}(\omega) + Z_0}. \tag{6.1}$$

For a TM-polarized plane wave, the impedance of the incident wave is $\eta_0 \cos\theta$ and the wave impedance of the transmitted wave is given by

$$Z_{\|}(\omega) = \frac{k_z(\omega)\eta(\omega)}{k(\omega)}. \tag{6.2}$$

where the terms k_z, η, and k are defined in (3.3). The relative permittivity is the same as given in (5.3).

6.1 Laplace Domain Representation

The Laplace domain reflection coefficient for a TM-polarized plane wave can be written as

$$\Gamma(s) = \frac{k_z - \epsilon_r k_{z0}}{k_z + \epsilon_r k_{z0}}$$

$$
\begin{aligned}
&= \frac{\sqrt{(k/k_0)^2 - \sin^2\theta} - \epsilon_r \cos\theta}{\sqrt{(k/k_0)^2 - \sin^2\theta} + \epsilon_r \cos\theta} \\
&= \frac{\sqrt{\mu_r\epsilon_r - \sin^2\theta} - \epsilon_r \cos\theta}{\sqrt{\mu_r\epsilon_r - \sin^2\theta} + \epsilon_r \cos\theta} \\
&= \Big[[\mu_r\epsilon_\infty(s-s_3)(s-s_4)/[(s-s_1)(s-s_2)] - \sin^2\theta]^{1/2} - \\
&\quad \epsilon_\infty \cos\theta(s-s_3)(s-s_4)/[(s-s_1)(s-s_2)] \Big] / \\
&\quad \Big[[\mu_r\epsilon_\infty(s-s_3)(s-s_4)/[(s-s_1)(s-s_2)] - \sin^2\theta]^{1/2} + \\
&\quad \epsilon_\infty \cos\theta(s-s_3)(s-s_4)/[(s-s_1)(s-s_2)] \Big],
\end{aligned}
\tag{6.3}
$$

where $s_{1,2}$ are the same as defined in (3.9) and $s_{3,4}$ are the same as defined in (5.7). Multiplying the numerator and denominator by $\sqrt{(s-s_1)(s-s_2)}$ and expanding $(s-s_1)(s-s_2)$ and $(s-s_3)(s-s_4)$ leads to

$$
\begin{aligned}
\Gamma(s) &= \Big[\sqrt{s-s_1}\sqrt{s-s_2}[\mu_r\epsilon_\infty(s-s_3)(s-s_4) - \sin^2\theta(s-s_1)(s-s_2)]^{1/2} - \\
&\quad \epsilon_\infty(s-s_3)(s-s_4)\cos\theta \Big] / \Big[\sqrt{s-s_1}\sqrt{s-s_2}[\mu_r\epsilon_\infty(s-s_3)(s-s_4) - \\
&\quad \sin^2\theta(s-s_1)(s-s_2)]^{1/2} + \epsilon_\infty(s-s_3)(s-s_4)\cos\theta \Big] \\
&= \Big[\sqrt{s-s_1}\sqrt{s-s_2} \Big[(\mu_r\epsilon_\infty - \sin^2\theta)s^2 + 2\delta s(\mu_r\epsilon_\infty - \sin^2\theta) + \\
&\quad \omega_0^2(\mu_r\epsilon_\infty - \sin^2\theta) + \mu_r b^2 \Big]^{1/2} - \epsilon_\infty \cos\theta(s-s_3)(s-s_4) \Big] / \\
&\quad \Big[\sqrt{s-s_1}\sqrt{s-s_2} \Big[(\mu_r\epsilon_\infty - \sin^2\theta)s^2 + 2\delta s(\mu_r\epsilon_\infty - \sin^2\theta) + \\
&\quad \omega_0^2(\mu_r\epsilon_\infty - \sin^2\theta) + \mu_r b^2 \Big]^{1/2} + \epsilon_\infty \cos\theta(s-s_3)(s-s_4) \Big].
\end{aligned}
\tag{6.4}
$$

Substituting K^2 as defined in (5.10) allows (6.4) to be written as

$$\Gamma(s) = \frac{K\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s^2+2\delta s+\omega_0^2+\mu_r b^2/K^2} - \epsilon_\infty \cos\theta(s-s_3)(s-s_4)}{K\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s^2+2\delta s+\omega_0^2+\mu_r b^2/K^2} + \epsilon_\infty \cos\theta(s-s_3)(s-s_4)}. \tag{6.5}$$

Substituting (5.12) into (6.5) allows the final form of the reflection coefficient to be rewritten as

$$\Gamma(s) = \frac{K\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6} - \epsilon_\infty \cos\theta(s-s_3)(s-s_4)}{K\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6} + \epsilon_\infty \cos\theta(s-s_3)(s-s_4)}. \tag{6.6}$$

6.2 Time-Domain Reflection Coefficient

Using the frequency-domain reflection coefficient found in Section 6.1, the time-domain reflection coefficient can be found using an inverse Laplace transform. In order to perform the inverse Laplace transform, the frequency-domain reflection coefficient may be rearranged into a more manageable form. Before this can be done however, it must be noted that as $s \longrightarrow \infty$, the reflection coefficient does not tend to zero. Instead it can be seen that

$$\text{as} \qquad s \longrightarrow \infty \qquad \Gamma(s) \longrightarrow \frac{K - \epsilon_\infty \cos\theta}{K + \epsilon_\infty \cos\theta}, \tag{6.7}$$

which produces a delta function in the final time-domain reflection coefficient. In order to perform the inverse Laplace transform, this term, defined as Γ_∞, needs to be subtracted from the total reflection coefficient. This leads to

$$\begin{aligned}\widetilde{\Gamma}(s) &= \Gamma(s) - \Gamma_\infty \\ &= \frac{K\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_1}\sqrt{s-s_2} - \epsilon_\infty \cos\theta(s-s_3)(s-s_4)}{K\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_1}\sqrt{s-s_2} + \epsilon_\infty \cos\theta(s-s_3)(s-s_4)} - \frac{K - \epsilon_\infty \cos\theta}{K + \epsilon_\infty \cos\theta}\end{aligned}$$

$$= \frac{2K\epsilon_\infty \cos\theta\left[\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6} - (s-s_3)(s-s_4)\right]}{(K+\epsilon_\infty\cos\theta)\left[K\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6} + \epsilon_\infty\cos\theta(s-s_3)(s-s_4)\right]}. \tag{6.8}$$

Rationalizing the denominator allows the denominator to be written as

$$\begin{aligned} D = (K+\epsilon_\infty\cos\theta)\Big[&K^2(s-s_1)(s-s_2)(s-s_5)(s-s_6)- \\ &\epsilon_\infty^2\cos^2\theta(s-s_3)^2(s-s_4)^2\Big], \end{aligned} \tag{6.9}$$

and the numerator to be written as

$$\begin{aligned} N = 2K\epsilon_\infty\cos\theta\big[&K(s-s_1)(s-s_2)(s-s_5)(s-s_6)- \\ &(K+\epsilon_\infty\cos\theta)(s-s_3)(s-s_4)\sqrt{s-s_1}\sqrt{s-s_2}\sqrt{s-s_5}\sqrt{s-s_6}+ \\ &\epsilon_\infty\cos\theta(s-s_3)^2(s-s_4)^2\Big]. \end{aligned} \tag{6.10}$$

Using the definition

$$\gamma^2 = \frac{b^2(\mu_r - 2\epsilon_\infty\cos^2\theta)}{K^2 - \epsilon_\infty^2\cos^2\theta}, \tag{6.11}$$

the denominator term can be expanded and written as

$$\begin{aligned} D = (K+\epsilon_\infty\cos\theta)(K^2-\epsilon_\infty^2\cos\theta)\Bigg[&s^4 + 4\delta s^3 + (2\omega_0^2 + 4\delta^2 + \gamma^2)s^2 + \\ &(4\delta\omega_0^2 + 2\delta\gamma^2)s + \omega_0^4 + \omega_0^2\gamma^2 - \frac{b^4\cos^2\theta}{K^2 - \epsilon_\infty^2\cos^2\theta}\Bigg]. \end{aligned} \tag{6.12}$$

The next step is to factor the denominator term. This is accomplished by dividing the quartic function in the denominator by $s^2 + 2\delta s + \omega_0^2 + \gamma^2/2 + \chi$ using a long division technique. The remainder term is set equal to zero and solved for χ. This allows the denominator to be written as

$$D = (K + \epsilon_\infty \cos\theta)^2 (K - \epsilon_\infty \cos\theta) \left(s^2 + 2\delta s + \omega_0^2 + \frac{\gamma^2}{2} - \chi \right) \times$$
$$\left(s^2 + 2\delta s + \omega_0^2 + \frac{\gamma^2}{2} + \chi \right), \tag{6.13}$$

where χ is defined as

$$\chi = \frac{b^2 \sigma}{2(K^2 - \epsilon_\infty^2 \cos^2\theta)}, \tag{6.14}$$

and σ is defined as

$$\sigma^2 = \mu_r^2 + \sin^2(2\theta). \tag{6.15}$$

It is then convenient to redefine the denominator as

$$D = (K + \epsilon_\infty \cos\theta)^2 (K - \epsilon_\infty \cos\theta)(s - s_E)(s - s_F)(s - s_G)(s - s_H), \tag{6.16}$$

where

$$(s - s_E)(s - s_F) = s^2 + 2\delta s + \omega_0^2 + \frac{\gamma^2}{2} - \chi, \tag{6.17a}$$
$$(s - s_G)(s - s_H) = s^2 + 2\delta s + \omega_0^2 + \frac{\gamma^2}{2} + \chi. \tag{6.17b}$$

The roots can then be found to be

$$\begin{aligned} s_{E,F} &= -\delta \pm \sqrt{\delta^2 - \left(\omega_0^2 + \frac{\gamma^2}{2} - \chi \right)} \\ &= -\delta \pm \lambda_E, \end{aligned} \tag{6.18a}$$
$$\begin{aligned} s_{G,H} &= -\delta \pm \sqrt{\delta^2 - \left(\omega_0^2 + \frac{\gamma^2}{2} + \chi \right)} \\ &= -\delta \pm \lambda_G. \end{aligned} \tag{6.18b}$$

The numerator also needs to be rearranged in order for the inverse Laplace transform to be taken. Noting that

$$(s - s_3)(s - s_4) = (s - s_1)(s - s_2) + \frac{b^2}{\epsilon_\infty}, \tag{6.19}$$

it is possible to rewrite the numerator term as

$$\begin{aligned}
N &= 2K\epsilon_\infty \cos\theta (s - s_1)(s - s_2)\Bigg[\epsilon_\infty \cos\theta[(s - s_1)(s - s_2) - \\
&\quad \sqrt{s - s_1}\sqrt{s - s_2}\sqrt{s - s_5}\sqrt{s - s_6}] + K[(s - s_5)(s - s_6) - \\
&\quad \sqrt{s - s_1}\sqrt{s - s_2}\sqrt{s - s_5}\sqrt{s - s_6}] - \frac{b^2}{\epsilon_\infty}(K + \epsilon_\infty \cos\theta)(s - s_5)(s - s_6) \times \\
&\quad \left[\frac{1}{\sqrt{s - s_1}\sqrt{s - s_2}\sqrt{s - s_5}\sqrt{s - s_6}}\right] + \frac{b^4 \cos\theta}{\epsilon_\infty}\frac{1}{(s - s_1)(s - s_2)} + 2b^2 \cos\theta\Bigg] \\
&= 2K\epsilon_\infty \cos\theta (s - s_1)(s - s_2)\Bigg[\epsilon_\infty \cos\theta g_1(s) + K g_2(s) - \\
&\quad \frac{b^2}{\epsilon_\infty}(K + \epsilon_\infty \cos\theta) g_3(s) + \frac{b^4 \cos\theta}{\epsilon_\infty} g_4(s) + 2b^2 \cos\theta\Bigg] \\
&= 2K\epsilon_\infty \cos\theta (s - s_1)(s - s_2) G(s).
\end{aligned} \tag{6.20}$$

The $g_n(s)$ terms are the same as in the previous cases. The reflection coefficient in the frequency domain can then be written as

$$\widetilde{\Gamma}(s) = \frac{2K\epsilon_\infty \cos\theta (s - s_1)(s - s_2) G(s)}{(K + \epsilon_\infty \cos\theta)^2 (K - \epsilon_\infty \cos\theta)(s - s_E)(s - s_F)(s - s_G)(s - s_H)}. \tag{6.21}$$

Making the convenient substitutions

$$F_2 = \frac{2K\epsilon_\infty \cos\theta}{(K + \epsilon_\infty \cos\theta)^2 (K - \epsilon_\infty \cos\theta)}, \tag{6.22a}$$

$$C(s) = \frac{(s - s_1)(s - s_2)}{(s - s_E)(s - s_F)(s - s_G)(s - s_H)}, \tag{6.22b}$$

allows (6.21) to be rewritten as

$$\widetilde{\Gamma}(s) = F_2C(s)G(s). \tag{6.23}$$

The Laplace transform of both the $C(s)$ and the $G(s)$ term can be taken separately and the time-domain reflection coefficient is simply a convolution of those terms.

6.2.1 Inversion of the $G(s)$ Term

As shown in (6.20), $G(s)$ is given as

$$G(s) = \left[\epsilon_\infty \cos\theta g_1(s) + Kg_2(s) - \frac{b^2}{\epsilon_\infty}(K + \epsilon_\infty \cos\theta)g_3(s) + \frac{b^4\cos\theta}{\epsilon_\infty}g_4(s) + 2b^2\cos\theta\right]. \tag{6.24}$$

All of the terms in $G(s)$ are solved in previous sections. The inversion of the $g_1(s)$ and $g_2(s)$ terms is done in Section 3.2.1 and the inversion of the $g_3(s)$ and $g_4(s)$ is done in Section 4.2. These terms are given as

$$g_1(t) = e^{-\delta t}\left[-\lambda_1^2\lambda_5^2\hat{\mathrm{I}}_1(\lambda_1 t) * \hat{\mathrm{I}}_1(\lambda_5 t) + \lambda_1^3\hat{\mathrm{I}}_2(\lambda_1 t) + \lambda_5^3\hat{\mathrm{I}}_2(\lambda_5 t)\right] - \frac{\lambda_1^2 - \lambda_5^2}{2}\delta(t), \tag{6.25a}$$

$$g_2(t) = e^{-\delta t}\left[-\lambda_1^2\lambda_5^2\hat{\mathrm{I}}_1(\lambda_1 t) * \hat{\mathrm{I}}_1(\lambda_5 t) + \lambda_1^3\hat{\mathrm{I}}_2(\lambda_1 t) + \lambda_5^3\hat{\mathrm{I}}_2(\lambda_5 t)\right] + \frac{\lambda_1^2 - \lambda_5^2}{2}\delta(t), \tag{6.25b}$$

$$g_3(t) = e^{-\delta t}\left[-\lambda_5^2\Big\{ \mathrm{I}_0(\lambda_1 t)\mathrm{u}(t) \Big\} * \hat{\mathrm{I}}_1(\lambda_5 t) + \lambda_1\mathrm{I}_1(\lambda_1 t)\mathrm{u}(t)\right] + \delta(t), \tag{6.25c}$$

$$g_4(t) = \frac{e^{-\delta t}}{\lambda_1}\sinh(\lambda_1 t)\mathrm{u}(t). \tag{6.25d}$$

In these equations, λ_1 and λ_5 are defined as in (5.30) and $\hat{\mathrm{I}}_n(x)$ is defined as in (3.51). The final expression for $G(t)$ can then be given as

$$G(t) = e^{-\delta t}\Bigg\{(K + \epsilon_\infty\cos\theta)\Bigg[-\lambda_1^2\lambda_5^2\hat{\mathrm{I}}_1(\lambda_1 t) * \hat{\mathrm{I}}_1(\lambda_5 t) + \frac{b^2}{\epsilon_\infty}\lambda_5^2\left\{\mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\right\} * \hat{\mathrm{I}}_1(\lambda_5 t) -$$

$$\frac{b^2}{\epsilon_\infty}\lambda_1 \mathrm{I}_1(\lambda_1 t)\mathrm{u}(t) + \lambda_1^3\hat{\mathrm{I}}_2(\lambda_1 t) + \lambda_5^3\hat{\mathrm{I}}_2(\lambda_5 t)\Big] + \frac{b^4\cos\theta}{\epsilon_\infty\lambda_1}\sinh(\lambda_1 t)\mathrm{u}(t)\Big\} +$$
$$b^2(K - \epsilon_\infty\cos\theta)\left(\frac{\mu_r\epsilon_\infty - 2K^2}{2K^2\epsilon_\infty}\right)\delta(t). \tag{6.26}$$

Just as in the optical TM case in Chapter 4, there are cases where λ_1 or λ_5 can be found to be imaginary and the modified Bessel functions in (6.26) can be replaced with ordinary Bessel functions. Case 1 occurs when $\omega_0^2 > \delta^2$. In this case both λ_1 and λ_5 are purely imaginary and defined as in (5.33). Then, using the properties (3.55) and (4.32), it can be seen that (6.26) can be expressed as

$$G(t) = e^{-\delta t}\Big\{(K + \epsilon_\infty\cos\theta)\Big[-\overline{\lambda}_1^2\overline{\lambda}_5^2\hat{\mathrm{J}}_1(\overline{\lambda}_1 t) * \hat{\mathrm{J}}_1(\overline{\lambda}_5 t) - \frac{b^2}{\epsilon_\infty}\overline{\lambda}_5^2\left\{\mathrm{J}_0(\overline{\lambda}_1 t)\mathrm{u}(t)\right\} * \hat{\mathrm{J}}_1(\overline{\lambda}_5 t) +$$
$$\frac{b^2}{\epsilon_\infty}\overline{\lambda}_1 \mathrm{J}_1(\overline{\lambda}_1 t)\mathrm{u}(t) + \overline{\lambda}_1^3\hat{\mathrm{J}}_2(\overline{\lambda}_1 t) + \overline{\lambda}_5^3\hat{\mathrm{J}}_2(\overline{\lambda}_5 t)\Big] + \frac{b^4\cos\theta}{\epsilon_\infty\overline{\lambda}_1}\sin(\overline{\lambda}_1 t)\mathrm{u}(t)\Big\} +$$
$$b^2(K - \epsilon_\infty\cos\theta)\left(\frac{\mu_r\epsilon_\infty - 2K^2}{2K^2\epsilon_\infty}\right)\delta(t), \tag{6.27}$$

where $\hat{\mathrm{J}}_n(x)$ is defined in (3.57).

Case 2 occurs when $\omega_0^2 + \mu_r b^2/K^2 < \delta^2$. In this case, both λ_1 and λ_5 are purely real and the same expression from (6.26) can be used. Case 3 occurs when $-\mu_r b^2/K^2 < \omega_0^2 - \delta^2 < 0$. In this case, λ_1 is purely real and λ_5 is purely imaginary. $\overline{\lambda}_5$ is defined as in (5.30) and the expression for $G(t)$ can be written as

$$G(t) = e^{-\delta t}\Big\{(K + \epsilon_\infty\cos\theta)\Big[\lambda_1^2\overline{\lambda}_5^2\hat{\mathrm{I}}_1(\overline{\lambda}_1 t) * \hat{\mathrm{J}}_1(\overline{\lambda}_5 t) - \frac{b^2}{\epsilon_\infty}\overline{\lambda}_5^2\left\{\mathrm{I}_0(\lambda_1 t)\mathrm{u}(t)\right\} * \hat{\mathrm{J}}_1(\overline{\lambda}_5 t) -$$
$$\frac{b^2}{\epsilon_\infty}\lambda_1 \mathrm{I}_1(\lambda_1 t)\mathrm{u}(t) + \lambda_1^3\hat{\mathrm{I}}_2(\lambda_1 t) + \overline{\lambda}_5^3\hat{\mathrm{J}}_2(\overline{\lambda}_5 t)\Big] + \frac{b^4\cos\theta}{\epsilon_\infty\lambda_1}\sinh(\lambda_1 t)\mathrm{u}(t)\Big\} +$$
$$b^2(K - \epsilon_\infty\cos\theta)\left(\frac{\mu_r\epsilon_\infty - 2K^2}{2K^2\epsilon_\infty}\right)\delta(t). \tag{6.28}$$

6.2.2 Inversion of the $C(s)$ Term

As described in (6.22), $C(s)$ is given in the Laplace domain as

$$C(s) = \frac{(s - s_1)(s - s_2)}{(s - s_E)(s - s_F)(s - s_G)(s - s_H)}. \tag{6.29}$$

Using partial fraction expansion, this term can be rewritten as

$$\begin{aligned} C(s) &= \frac{\sigma - \mu_r + 2\epsilon_\infty \cos^2\theta}{4\sigma\lambda_E}\left[\frac{1}{s - s_E} - \frac{1}{s - s_F}\right] + \\ &\quad \frac{\sigma + \mu_r - 2\epsilon_\infty \cos^2\theta}{4\sigma\lambda_G}\left[\frac{1}{s - s_G} - \frac{1}{s - s_H}\right] \\ &= c_1(s) + c_2(s). \end{aligned} \tag{6.30}$$

An inverse Laplace transform of each term in (6.30) can be taken separately. When inverting $c_1(s)$, if $\omega_0^2 + \gamma^2/2 - \chi > 0$ the standard transform used in (3.27) can be applied directly resulting in the expression

$$\begin{aligned} c_1(s) \longleftrightarrow c_1(t) &= \frac{\sigma - \mu_r + 2\epsilon_\infty \cos^2\theta}{4\sigma\lambda_E} e^{-\delta t}\left[e^{\lambda_E t} - e^{-\lambda_E t}\right] \mathrm{u}(t) \\ &= \frac{\sigma - \mu_r + 2\epsilon_\infty \cos^2\theta}{2\sigma\lambda_E} e^{-\delta t} \sinh(\lambda_E t)\mathrm{u}(t). \end{aligned} \tag{6.31}$$

If $\omega_0^2 + \gamma^2/2 - \chi > \delta^2$, λ_E is purely imaginary and defined as

$$-j\lambda_E = \overline{\lambda}_E = \sqrt{\omega_0^2 + \frac{\gamma^2}{2} - \chi - \delta^2}. \tag{6.32}$$

This allows (6.31) to be rewritten as

$$c_1(t) = \frac{\sigma - \mu_r + 2\epsilon_\infty \cos^2\theta}{2\sigma\overline{\lambda}_E} e^{-\delta t} \sin(\overline{\lambda}_E t)\mathrm{u}(t). \tag{6.33}$$

If $\omega_0^2 + \gamma^2/2 - \chi < 0$ however, $R_e\{s_E\} > 0$. In this case the identity given in (4.40) must be used. This produces the result

$$
\begin{aligned}
c_1(t) &= -\frac{\sigma - \mu_r + 2\epsilon_\infty \cos^2\theta}{4\sigma\lambda_E} e^{-\delta t}\left[e^{\lambda_E t}\mathrm{u}(-t) + e^{-\lambda_E t}\mathrm{u}(t)\right] \\
&= -\frac{\sigma - \mu_r + 2\epsilon_\infty \cos^2\theta}{4\sigma\lambda_E} e^{-\delta t} e^{-\lambda_E |t|},
\end{aligned}
\tag{6.34}
$$

which is non-causal. As argued in Chapter 4, this non-causality is due to the infinite nature of the problem and is not a physical phenomenon.

Taking the Laplace inversion of $c_2(s)$ can be done in much the same manner as for $c_1(s)$. If $\omega_0^2 + \gamma^2/2 + \chi > 0$ the standard transform used in (3.27) can be applied directly resulting in the expression

$$
\begin{aligned}
c_2(s) \longleftrightarrow c_2(t) &= \frac{\sigma + \mu_r - 2\epsilon_\infty \cos^2\theta}{4\sigma\lambda_G} e^{-\delta t}\left[e^{\lambda_G t} - e^{-\lambda_G t}\right]\mathrm{u}(t) \\
&= \frac{\sigma + \mu_r - 2\epsilon_\infty \cos^2\theta}{2\sigma\lambda_G} e^{-\delta t}\sinh(\lambda_G t)\mathrm{u}(t).
\end{aligned}
\tag{6.35}
$$

If $\omega_0^2 + \gamma^2/2 + \chi > \delta^2$, λ_G is purely imaginary and defined as

$$
-j\lambda_G = \overline{\lambda}_G = \sqrt{\omega_0^2 + \frac{\gamma^2}{2} + \chi - \delta^2}. \tag{6.36}
$$

This allows (6.35) to be rewritten as

$$
c_2(t) = \frac{\sigma + \mu_r - 2\epsilon_\infty \cos^2\theta}{2\sigma\overline{\lambda}_G} e^{-\delta t}\sin(\overline{\lambda}_G t)\mathrm{u}(t). \tag{6.37}
$$

If $\omega_0^2 + \gamma^2/2 + \chi < 0$ however, $R_e\{s_G\} > 0$. In this case the identity given in (4.40) must again be used. This produces the result

$$
c_2(t) = -\frac{\sigma + \mu_r - 2\epsilon_\infty \cos^2\theta}{4\sigma\lambda_G} e^{-\delta t}\left[e^{\lambda_G t}\mathrm{u}(-t) + e^{-\lambda_G t}\mathrm{u}(t)\right]
$$

$$= -\frac{\sigma + \mu_r - 2\epsilon_\infty \cos^2\theta}{4\sigma\lambda_G} e^{-\delta t} e^{-\lambda_G |t|}, \tag{6.38}$$

which is again non-causal. The same argument from earlier can be used to explain this non-causality.

Analyzing these two terms, it can be seen that in general, $C(t)$ is causal whenever $\chi < -|\omega_0^2 + \gamma^2/2|$ and non-causal when $\chi > -|\omega_0^2 + \gamma^2/2|$.

6.3 Numerical Results

In order to validate the expressions derived in the previous section, the time-domain reflection coefficient is evaluated numerically in Fortran, and then compared to the inverse FFT of the frequency domain reflection coefficient given in (6.21). The Fortran code is included in Appendix D. The inverse FFT was done using WaveCalc. The frequency-domain data was zero-padded up to the maximum limit allowed by WaveCalc, 32,768, before the inverse FFT was taken. Since the signal had already decayed to zero by this point, no windowing was necessary. A set of parameters corresponding to each of the three possible cases for $G(t)$ is used. The angle is then adjusted for the case 1 parameters to produce results that contain a non-causal contribution. The number of points and step size for the results varied between the different cases.

The first set of parameters chosen are: $\omega_0 = 4.0 \times 10^{11}\ s^{-1}$, $b = 6.24 \times 10^{11}\ s^{-1}$, $\delta = 2.5 \times 10^{10}\ s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$. This choice of parameters corresponds to case 1. When computing the numerical results, for the frequency-domain data 8,192 frequency points were calculated with a step size of 400 MHz. In the time-domain, 8,192 points were calculated with a step size of 1×10^{-4} ns. Using $\theta = 30°$, (6.27) has been plotted in Figure 6.1 and compared to the inverse FFT. The results show excellent agreement. Since this function includes only standard Bessel functions, which are highly oscillatory, and no modified Bessel functions, which are not oscillatory, the waveform is highly oscillatory and only lightly damped.

The next choice of parameters is: $\omega_0 = 4.0 \times 10^{10}\ s^{-1}$, $b = 6.24 \times 10^{10}\ s^{-1}$, $\delta = 2.5 \times 10^{11}\ s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$, which corresponds to case 2, (6.26). When computing the numerical results, for the frequency-domain data 32,768 frequency points were calculated with a step size of 4 MHz. In the time-domain, 8,192 points were calculated with a step size of 1×10^{-4} ns. The results are shown in Figure 6.2. Again, the closed-form expression and the inverse FFT compare well. For this choice of parameters $\omega_0^2 + \mu_r b^2/K^2 < \delta^2$, and the resulting waveform is overdamped, showing no oscillatory behavior and only a single negative peak. Since the expression for case 2 only involves modified Bessel functions, which do not have the oscillatory behavior of ordinary Bessel functions, this observed behavior is easily predicted from the mathematical form of the expression.

The next choice of parameters, which corresponds to case 3, is: $\omega_0 = 4.0 \times 10^9\ s^{-1}$, $b = 6.24 \times 10^{11}\ s^{-1}$, $\delta = 2.5 \times 10^{10}\ s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$, which corresponds to case 3, (6.28). When computing the numerical results, for the frequency-domain data 32,768 frequency points were calculated with a step size of 400 MHz. In the time-domain, 8,192 points were calculated with a step size of 5×10^{-5} ns. The analytic expression again matches the inverse FFT, as seen in Figure 6.3. As expected, since $\delta > \omega_0$, but $\delta^2 < \omega_0^2 + \mu_r b^2/K^2$, there is more damping and less oscillation than with case 1, but more oscillation than with case 2. Here the expression for the reflection coefficient has a combination of ordinary and modified Bessel functions.

To examine the non-causal result which is possible for the $C(t)$ term, the material parameters corresponding to case 1 are used again, but at an angle of 60°. This causes the $c_1(t)$ term in $C(t)$ to be non-causal as shown in (6.34). When computing the numerical results, for the frequency-domain data 32,768 frequency points were calculated with a step size of 400 MHz. In the time-domain, 8,192 points were calculated with a step size of 5×10^{-5} ns with a starting time value of -1.024×10^{-1} ns. As seen in Figure 6.4, the expression again matches well with the inverse FFT for $t > 0$. For $t < 0$, non-causal term in the analytic expression is extremely small. Unfortunately, the inverse FFT does not have the necessary resolution to capture the small non-causal portion of the reflection coefficient, so it cannot be compared to the analytic expression.

In order to gain better insight into the non-causal term, the non-causal $C(t)$ term is plotted in Figure 6.5. The non-causal part is much more significant in this $C(t)$ than in the TM optical case. To investigate further, $C(t)$ was convolved with $G(t)$ minus the delta function term. This can be seen in Figure 6.6. Again, the non-causal portion has a significant contribution to this term. However, when $C(t)$ is convolved with the delta function term, it can be seen in Figure 6.7 that the non-causal portion has the opposite sign of the term in Figure 6.6, and when those terms are added, the non-causal portion becomes extremely small in the final transient reflected field.

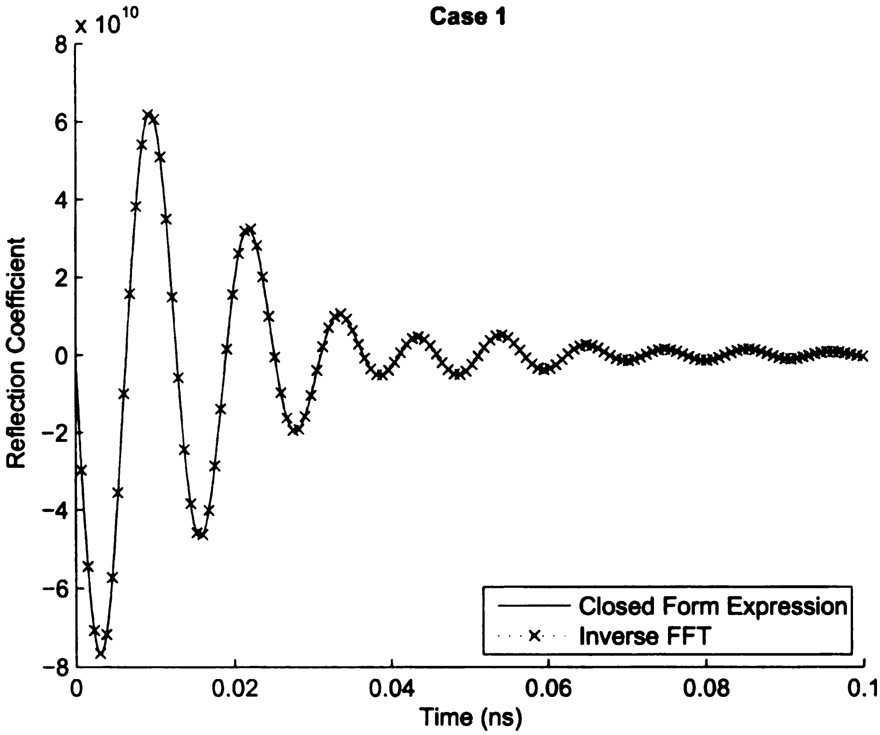

Figure 6.1. Time-domain reflection coefficient with incidence angle $\theta = 30°$ and material parameters $\omega_0 = 4.0 \times 10^{11} s^{-1}$, $b = 6.24 \times 10^{11} s^{-1}$, $\delta = 2.5 \times 10^{10} s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$. This choice of parameters corresponds to case 1, Eq. (6.27).

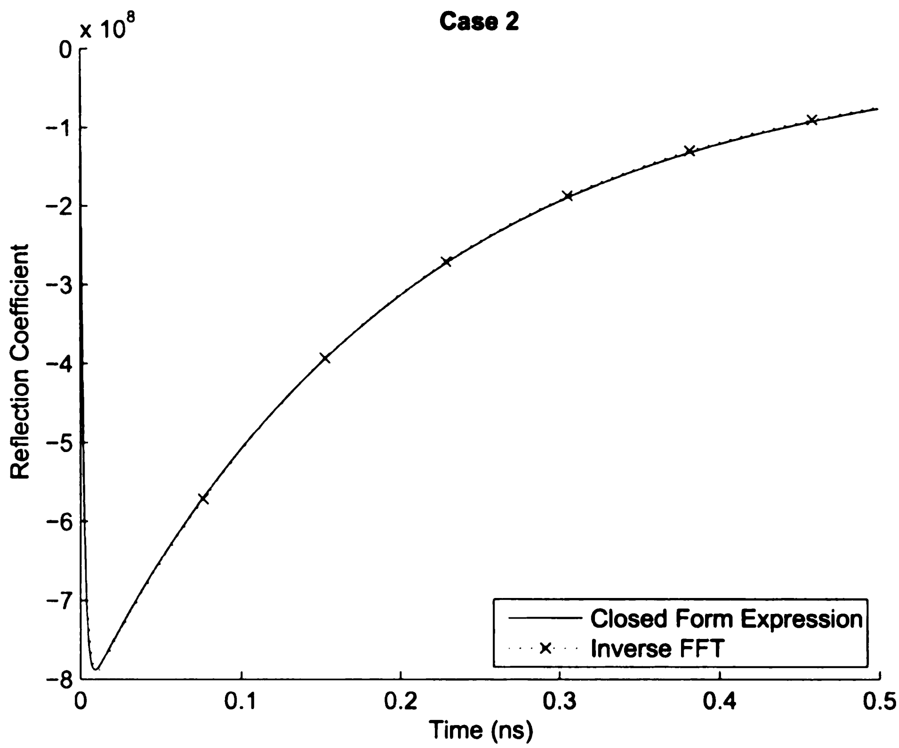

Figure 6.2. Time-domain reflection coefficient with incidence angle $\theta = 30°$ and material parameters $\omega_0 = 4.0 \times 10^{10} s^{-1}$, $b = 6.24 \times 10^{10} s^{-1}$, $\delta = 2.5 \times 10^{11} s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$. This choice of parameters corresponds to case 2, Eq. (6.26).

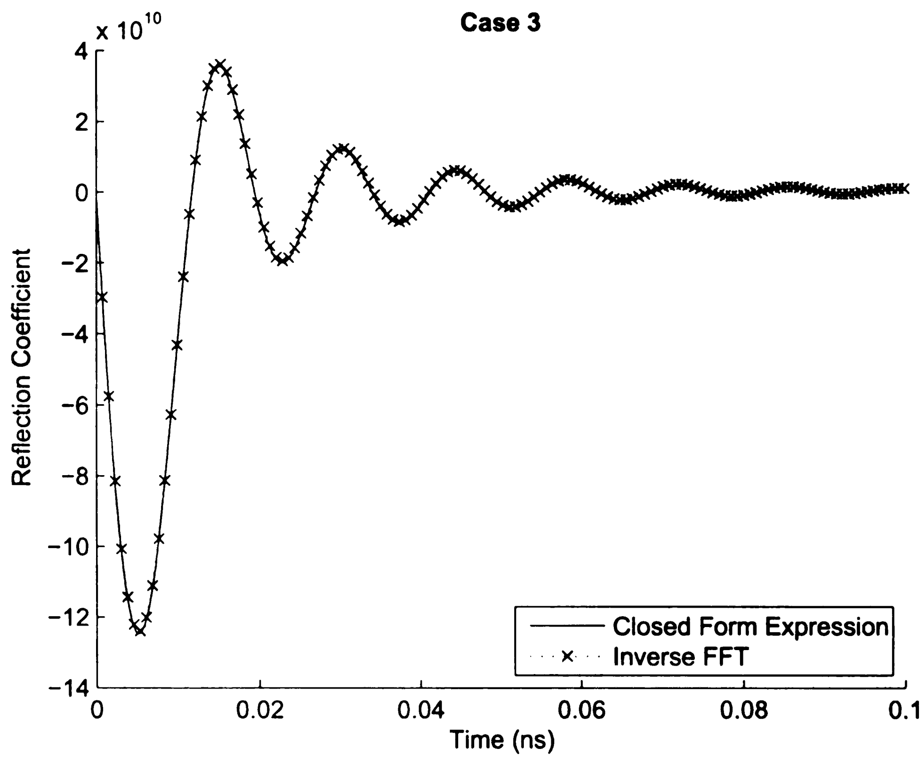

Figure 6.3. Time-domain reflection coefficient with at an angle of $\theta = 30°$ with parameter choices of $\omega_0 = 4.0 \times 10^9 s^{-1}$, $b = 6.24 \times 10^{11} s^{-1}$, $\delta = 2.5 \times 10^{10} s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$. This choice of parameters corresponds to case 3, Eq. (6.28).

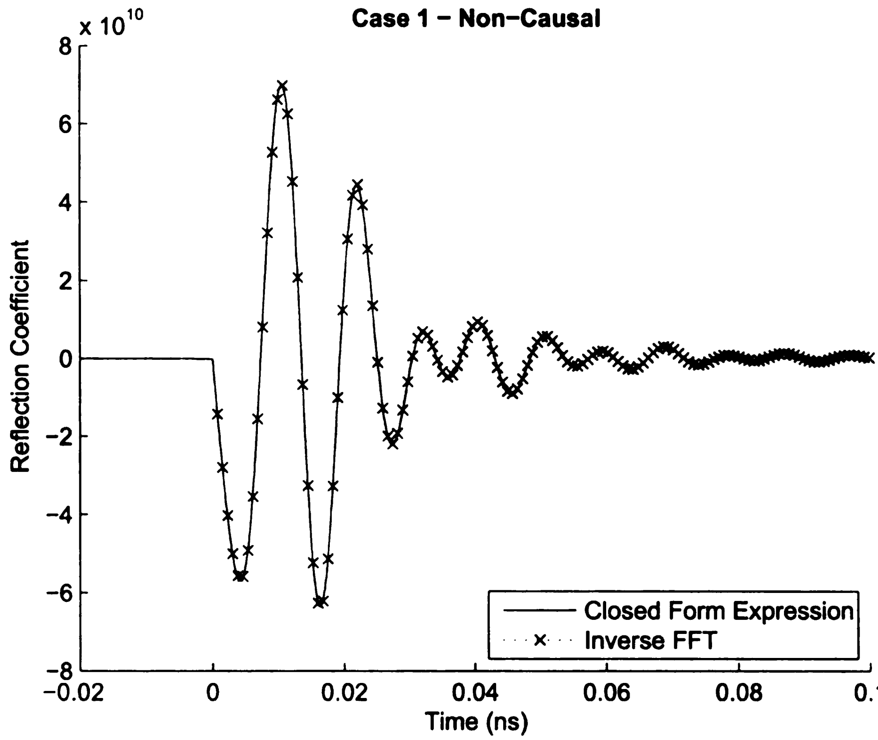

Figure 6.4. Time-domain reflection coefficient with at an angle of $\theta = 60°$ with parameter choices of $\omega_0 = 4.0 \times 10^{11} s^{-1}$, $b = 6.24 \times 10^{11} s^{-1}$, $\delta = 2.5 \times 10^{10} s^{-1}$, $\epsilon_\infty = 2$, $\mu_r = 1$. This choice of parameters corresponds to case 1, Eq. (6.27) with a non-causal term from Eq. (6.34).

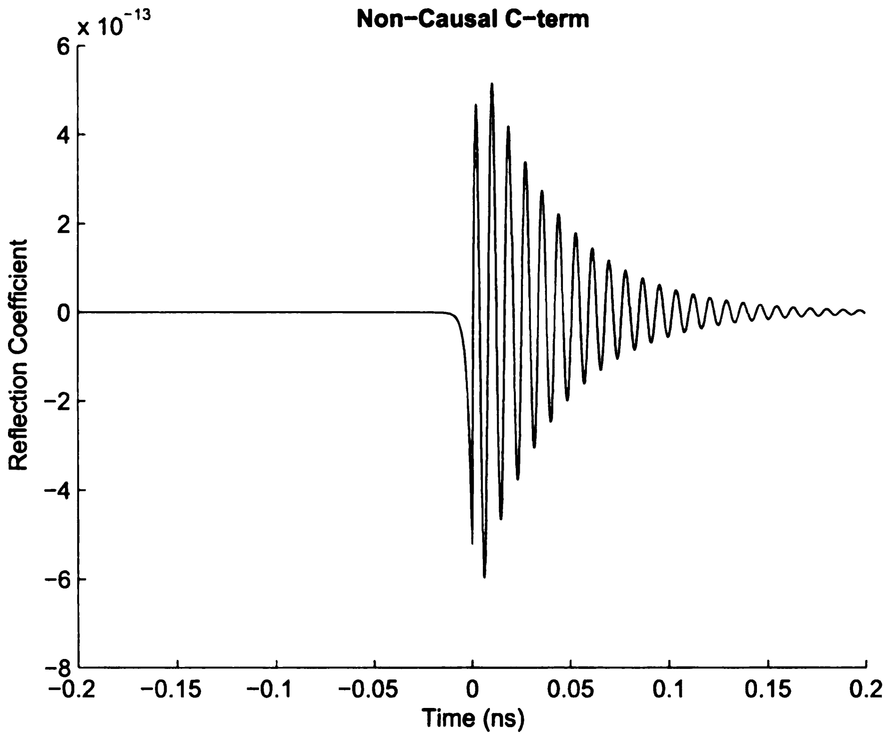

Figure 6.5. Non-causal $C(t)$ term for case 1 results.

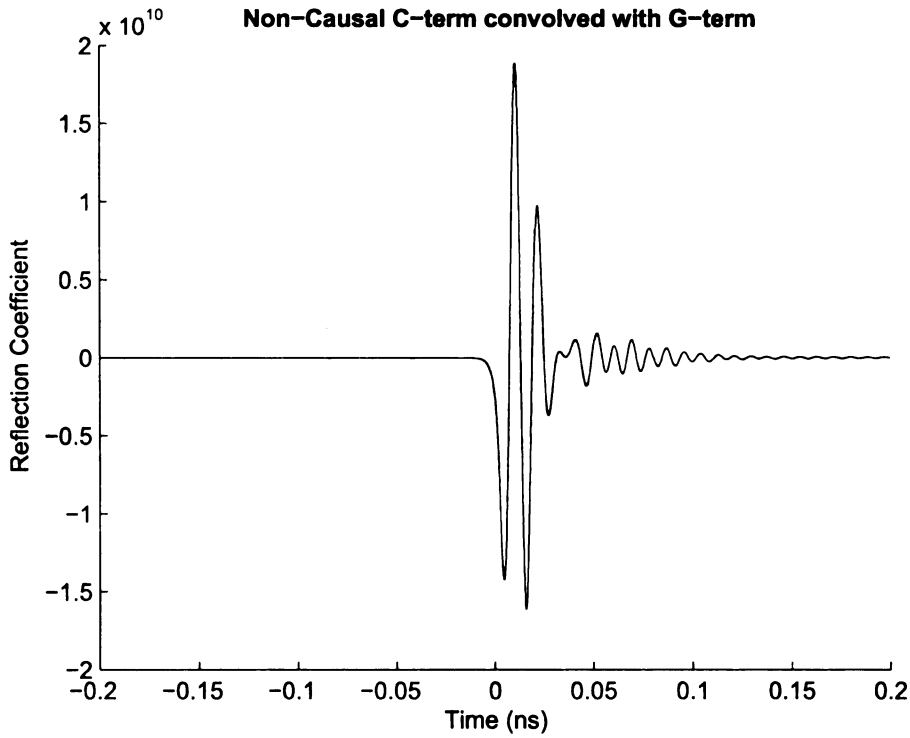

Figure 6.6. Non-causal $C(t)$ term convolved with $G(t)$ minus the delta function term for case 1 results.

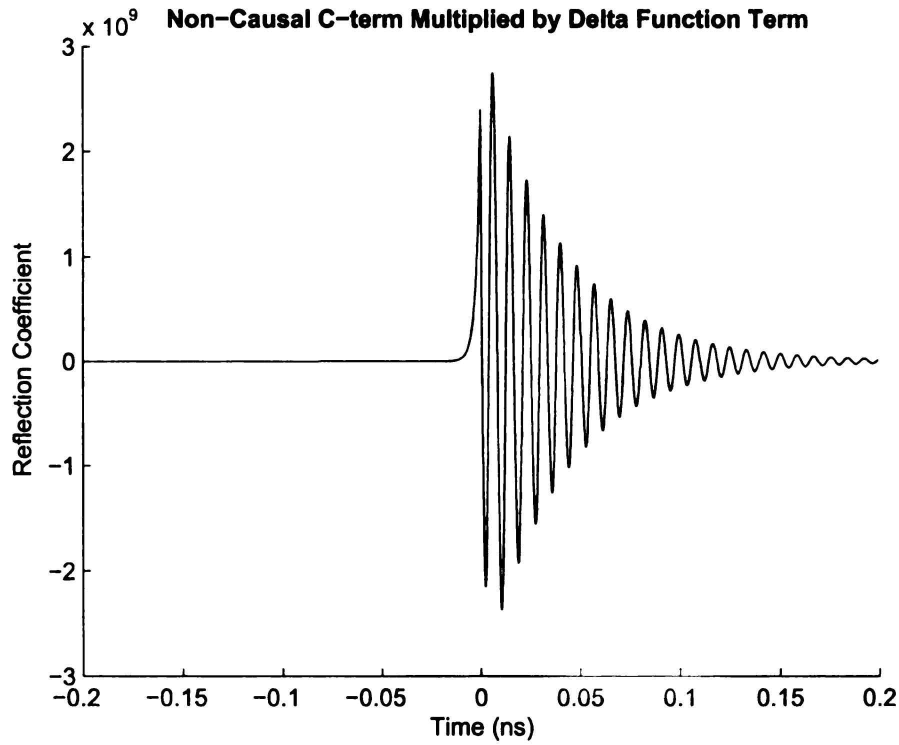

Figure 6.7. Non-causal $C(t)$ term convolved with the delta function term for case 1 results.

CHAPTER 7

CONCLUSIONS

In this thesis an analytic solution for the transient reflection coefficient of plane waves from a Lorentz-medium half-space is formulated analytically. Using an inverse Laplace transform technique, the solution was found to be a combination of Bessel functions and convolutions of Bessel functions with a decaying exponential factor proportional to δ, the damping coefficient of the material.

Unlike previous work [10],[11] which represent the transient field as an infinite sum of fractional-order Bessel functions, the formulation in this thesis has no infinite sums and presents much more physical insight into the behavior of the transient field. Depending on field polarization, material properties, and angle of incidence, the expression for the transient field changes and gives a clearer view of the time-domain behavior. It is also clear that for certain angles there is a problem with causality that occurrs in the TM-polarized case. This non-causal function is due to the infinite nature of the problem and is not a physical phenomenon. It is, however, an interesting discovery.

7.1 Suggestions for Future Work

The Lorentz model is a single resonance model that is closely related to the permittivity of a cold plasma. A very similar reflection coefficient can be developed using plasma equations. Using this same method, developing the transient field reflection coefficient from a cold plasma half-space is a problem that is of some interest.

Transmission into dispersive material is also a topic of interest. It would be interesting to formulate the transient transmitted field using the same basic techniques outlined in this thesis for the reflected field.

APPENDICES

APPENDIX A

FORTRAN CODE FOR OPTICAL TE CASE

```
program special_TE


implicit none


integer nsize
parameter(nsize = 10000)


real*8 BESSI,om0,delta,b,theta,rlam1,rlam5,dt,ct,t,pi
real*8 j1,j2,j1p,j2p,i1,i2,df,f,om,gammat(nsize)
real*8 j11h(nsize),j12h(nsize),j21h(nsize),j22h(nsize)
real*8 i11h(nsize),i12h(nsize),i21h(nsize),i22h(nsize)
real*8 j1j1(nsize),i1i1(nsize),i1j1(nsize),expon(nsize)
integer i,nt,nf
complex*16 z1,z2,s,zj,gamma,zz1,zz2,s1,s2,s5,s6


om0   = 4.d16
b     = dsqrt(20.d32)
delta = 0.28d16
theta = 0.d0
pi = 4.d0*atan(1.d0)
ct = cos(theta*pi/180.d0)
zj=dcmplx(0.d0,1.d0)
```

```
      df = 8d3
      nf = ((4095*2+1)*2+1)
      nt = 4096
      dt = 1.d-18
c
      zz1 = cdsqrt(dcmplx(delta*delta-om0*om0,0.d0))
      s1  = -delta+zz1
      s2  = -delta-zz1
      zz2 = cdsqrt(dcmplx(delta*delta-om0*om0-b*b/(ct*ct),0.d0))
      s5  = -delta+zz2
      s6  = -delta-zz2

      open (10,file='gamma.dat',status='unknown')
      do i=1,nf
          f = df*i
          om = 2.d0*pi*f*1.d9
          s = zj*om
          z1 = cdsqrt(s-s1)*cdsqrt(s-s2)
          z2 = cdsqrt(s-s5)*cdsqrt(s-s6)
          gamma = (z1-z2)/(z1+z2)
          write (10,*) f,real(gamma),aimag(gamma)
      end do
      close (10)

      do i=1,nt
          t = (i-1)*dt+.00001*dt
          expon(i) = exp(-delta*t)
```

```
    end do
    write(*,*) 'om^2      = ',om0*om0
    write(*,*) 'om^2+B^2 = ',om0*om0+b*b/(ct*ct)
    write(*,*) 'delta^2  = ',delta*delta
! ----------------------------------------------------------------------
!    CASE 1
!    om0^2 > delta^2
! ----------------------------------------------------------------------
    if((om0*om0).gt.(delta*delta)) then
        write(*,*) "CASE 1"
        rlam1 = dsqrt(om0*om0-delta*delta)
        rlam5 = dsqrt(om0*om0+(b*b/(ct*ct))-delta*delta)
        do i=1,nt
            t = (i-1)*dt+.00001*dt
            call jbess(rlam1*t,2,j2,j2p)
            j21h(i) = j2/(rlam1*t)
            call jbess(rlam5*t,2,j2,j2p)
            j22h(i) = j2/(rlam5*t)
            call jbess(rlam1*t,1,j1,j1p)
            j11h(i) = j1/(rlam1*t)
            call jbess(rlam5*t,1,j1,j1p)
            j12h(i) = j1/(rlam5*t)
        end do
        call conv(j11h,j12h,nt,dt,j1j1)

        open(10,file='gammat.dat',status='unknown')
        do i=1,nt
```

```
        gammat(i) = expon(i)*(2.d0*(rlam1**3)*j21h(i)+
  2           2.d0*(rlam5**3)*j22h(i)-2.d0*rlam1*rlam1*rlam5*rlam5*
  3           j1j1(i))
             gammat(i) = -gammat(i)*ct*ct/(b*b)
        write(10,*) ((i-1)*dt+.00001*dt)*1.d9,gammat(i)
      end do
      close(10)
! ----------------------------------------------------------------------
!    CASE 2
!    om0^2 + B^2 < delta^2
! ----------------------------------------------------------------------
    elseif((om0*om0+(b*b/(ct*ct))).lt.delta*delta) then

      write(*,*) "CASE 2"
      rlam1 = dsqrt(delta*delta-om0*om0)
      rlam5 = dsqrt(-om0*om0-(b*b/(ct*ct))+delta*delta)
      do i=1,nt
        t = (i-1)*dt+.00001*dt
        i2 = BESSI(2,rlam1*t)
        i21h(i) = i2/(rlam1*t)
        i2 = BESSI(2,rlam5*t)
        i22h(i) = i2/(rlam5*t)
        i1 = BESSI(1,rlam1*t)
        i11h(i) = i1/(rlam1*t)
        i1 = BESSI(1,rlam5*t)
        i12h(i) = i1/(rlam5*t)
      end do
```

```
      call conv(i11h,i12h,nt,dt,i1i1)


      open(10,file='gammat.dat',status='unknown')
      do i=1,nt
          gammat(i) = expon(i)*(2.d0*(rlam1**3)*i21h(i)+
   2            2.d0*(rlam5**3)*i22h(i)-2.d0*rlam1*rlam1*rlam5*rlam5*
   3            i1i1(i))
               gammat(i) = -gammat(i)*ct*ct/(b*b)
          write(10,*) ((i-1)*dt+.00001*dt)*1.d9,gammat(i)
      end do
      close(10)
  else
! ----------------------------------------------------------------------
!   CASE 3
!   om0^2 + B^2 > delta^2 & om0^2 < delta^2
! ----------------------------------------------------------------------
      write(*,*) "CASE 3"
      rlam1 = dsqrt(delta*delta-om0*om0)
      rlam5 = dsqrt(om0*om0+(b*b/(ct*ct))-delta*delta)
      do i=1,nt
          t = (i-1)*dt+.00001*dt
          i2 = BESSI(2,rlam1*t)
          i21h(i) = i2/(rlam1*t)
          call jbess(rlam5*t,2,j2,j2p)
          j22h(i) = j2/(rlam5*t)
          i1 = BESSI(1,rlam1*t)
          i11h(i) = i1/(rlam1*t)
```

```
            call jbess(rlam5*t,1,j1,j1p)
            j12h(i) = j1/(rlam5*t)
         end do
         call conv(i11h,j12h,nt,dt,i1j1)

         open(10,file='gammat.dat',status='unknown')
         do i=1,nt
            gammat(i) = expon(i)*(2.d0*(rlam1**3)*i21h(i)+
     2            2.d0*(rlam5**3)*j22h(i)+2.d0*rlam1*rlam1*rlam5*rlam5*
     3            i1j1(i))
                 gammat(i) = -gammat(i)*ct*ct/(b*b)
            write(10,*) ((i-1)*dt+.00001*dt)*1.d9,gammat(i)
         end do
         close(10)
      endif

      end

! ----------------------------------------------------------------------
      subroutine conv(f,g,n,del,c)
!
!     performs convolution of waveforms f and g using
!      linear interpolation
!
      parameter (nsize=10000)
      parameter (c13=0.33333333333333333333d0,
     2           c16=0.16666666666666666666d0)
```

```
      implicit real*8 (a-h,o-z)
      real*8 f(nsize),g(nsize),c(nsize)
      c(1)=0.d0
      do k=2,n
        sum = 0.d0
        do l=1,k-1
          sum = sum + c13*f(l+1)*g(k-l)
     2               + c13*f(l)*g(k-l+1)
     3               + c16*f(l)*g(k-l)
     4               + c16*f(l+1)*g(k-l+1)
        end do
        c(k) = del*sum
      end do
      return
      end


! ---------------------------------------------------------------------
        FUNCTION BESSI(N,X)
!
!       This subroutine calculates the first kind modified Bessel function
!       of integer order N, for any REAL X. We use here the classical
!       recursion formula, when X > N. For X < N, the Miller's algorithm
!       is used to avoid overflows.
!       REFERENCE:
!       C.W.CLENSHAW, CHEBYSHEV SERIES FOR MATHEMATICAL FUNCTIONS,
!       MATHEMATICAL TABLES, VOL.5, 1962.
```

```
!
      PARAMETER (IACC = 40,BIGNO = 1.D10, BIGNI = 1.D-10)
      REAL *8 X,BESSI,BESSI0,BESSI1,TOX,BIM,BI,BIP
      IF (N.EQ.0) THEN
      BESSI = BESSI0(X)
      RETURN
      ENDIF
      IF (N.EQ.1) THEN
      BESSI = BESSI1(X)
      RETURN
      ENDIF
      IF(X.EQ.0.D0) THEN
      BESSI=0.D0
      RETURN
      ENDIF
      TOX = 2.D0/X
      BIP = 0.D0
      BI  = 1.D0
      BESSI = 0.D0
      M = 2*((N+INT(SQRT(FLOAT(IACC*N)))))
      DO 12 J = M,1,-1
      BIM = BIP+DFLOAT(J)*TOX*BI
      BIP = BI
      BI  = BIM
      IF (ABS(BI).GT.BIGNO) THEN
      BI  = BI*BIGNI
      BIP = BIP*BIGNI
```

```
      BESSI = BESSI*BIGNI
      ENDIF
      IF (J.EQ.N) BESSI = BIP
   12 CONTINUE
      BESSI = BESSI*BESSI0(X)/BI
      RETURN
      END
! ----------------------------------------------------------------------
! Auxiliary Bessel functions for N=0, N=1
      FUNCTION BESSI0(X)
      REAL *8 X,BESSI0,Y,P1,P2,P3,P4,P5,P6,P7,
     *    Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9,AX,BX
      DATA P1,P2,P3,P4,P5,P6,P7/1.D0,3.5156229D0,3.0899424D0,1.2067429D0
     *    ,0.2659732D0,0.360768D-1,0.45813D-2/
      DATA Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9/0.39894228D0,0.1328592D-1,
     *    0.225319D-2,-0.157565D-2,0.916281D-2,-0.2057706D-1,
     *    0.2635537D-1,-0.1647633D-1,0.392377D-2/
      IF(ABS(X).LT.3.75D0) THEN
      Y=(X/3.75D0)**2
      BESSI0=P1+Y*(P2+Y*(P3+Y*(P4+Y*(P5+Y*(P6+Y*P7)))))
      ELSE
      AX=ABS(X)
      Y=3.75D0/AX
      BX=EXP(AX)/SQRT(AX)
      AX=Q1+Y*(Q2+Y*(Q3+Y*(Q4+Y*(Q5+Y*(Q6+Y*(Q7+Y*(Q8+Y*Q9)))))))
      BESSI0=AX*BX
      ENDIF
```

```
      RETURN
      END
! ----------------------------------------------------------------------
      FUNCTION BESSI1(X)
      REAL *8 X,BESSI1,Y,P1,P2,P3,P4,P5,P6,P7,
     *    Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9,AX,BX
      DATA P1,P2,P3,P4,P5,P6,P7/0.5D0,0.87890594D0,0.51498869D0,
     *    0.15084934D0,0.2658733D-1,0.301532D-2,0.32411D-3/
      DATA Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9/0.39894228D0,-0.3988024D-1,
     *    -0.362018D-2,0.163801D-2,-0.1031555D-1,0.2282967D-1,
     *    -0.2895312D-1,0.1787654D-1,-0.420059D-2/
      IF(ABS(X).LT.3.75D0) THEN
      Y=(X/3.75D0)**2
      BESSI1=X*(P1+Y*(P2+Y*(P3+Y*(P4+Y*(P5+Y*(P6+Y*P7))))))
      ELSE
      AX=ABS(X)
      Y=3.75D0/AX
      BX=EXP(AX)/SQRT(AX)
      AX=Q1+Y*(Q2+Y*(Q3+Y*(Q4+Y*(Q5+Y*(Q6+Y*(Q7+Y*(Q8+Y*Q9)))))))
      BESSI1=AX*BX
      ENDIF
      RETURN
      END
! ----------------------------------------------------------------------


    subroutine jbess (x,n,bj,bjp)
```

```
c
c  calculates jn and jn' for n positive or negative
c
      implicit real*8 (a-h,o-z)
c
    if (n .eq. 0) then
        bj  = bessj(0,x)
        bjp = -bessj(1,x)
        return
      endif
c
      if (n .ge. 0) then
        m = n
      else
        m = -n
      endif
c
      if (x .eq. 0.d0) then
        bj  = 0.d0
        bjp1 = bessj(m+1,x)
        bjm1 = bessj(m-1,x)
        bjp  = (bjm1-bjp1)/2.d0
      else
        bj    = bessj(m,x)
        bj1   = bessj(m+1,x)
        bjp   = -bj1 + n*bj/x
      endif
```

```
c
      if (n .lt. 0) then
        bj    = bj*(-1)**n
        bjp   = bjp*(-1)**n
      endif
c
      return
      end
c
c
c
      FUNCTION BESSJ0(X)
      implicit real*8 (a-h,o-z)
      REAL*8 Y,P1,P2,P3,P4,P5,Q1,Q2,Q3,Q4,Q5,R1,R2,R3,R4,R5,R6,
     *    S1,S2,S3,S4,S5,S6
      DATA P1,P2,P3,P4,P5/1.D0,-.1098628627D-2,.2734510407D-4,
     *    -.2073370639D-5,.2093887211D-6/, Q1,Q2,Q3,Q4,Q5/-.1562499995D-
     *1,
     *    .1430488765D-3,-.6911147651D-5,.7621095161D-6,-.934945152D-7/
      DATA R1,R2,R3,R4,R5,R6/57568490574.D0,-13362590354.D0,651619640.7D
     *0,
     *    -11214424.18D0,77392.33017D0,-184.9052456D0/,
     *    S1,S2,S3,S4,S5,S6/57568490411.D0,1029532985.D0,
     *    9494680.718D0,59272.64853D0,267.8532712D0,1.D0/
      IF(ABS(X).LT.8.)THEN
        Y=X**2
        BESSJ0=(R1+Y*(R2+Y*(R3+Y*(R4+Y*(R5+Y*R6)))))
```

```
     *      /(S1+Y*(S2+Y*(S3+Y*(S4+Y*(S5+Y*S6)))))
      ELSE
        AX=ABS(X)
        Z=8./AX
        Y=Z**2
        XX=AX-.785398164
        BESSJ0=SQRT(.636619772/AX)*(COS(XX)*(P1+Y*(P2+Y*(P3+Y*(P4+Y
     *      *P5))))-Z*SIN(XX)*(Q1+Y*(Q2+Y*(Q3+Y*(Q4+Y*Q5)))))
      ENDIF
      RETURN
      END
c
c
c
      FUNCTION BESSJ1(X)
      implicit real*8 (a-h,o-z)
      REAL*8 Y,P1,P2,P3,P4,P5,Q1,Q2,Q3,Q4,Q5,R1,R2,R3,R4,R5,R6,
     *    S1,S2,S3,S4,S5,S6
      DATA R1,R2,R3,R4,R5,R6/72362614232.D0,-7895059235.D0,242396853.1D0
     *,
     *    -2972611.439D0,15704.48260D0,-30.16036606D0/,
     *    S1,S2,S3,S4,S5,S6/144725228442.D0,2300535178.D0,
     *    18583304.74D0,99447.43394D0,376.9991397D0,1.D0/
      DATA P1,P2,P3,P4,P5/1.D0,.183105D-2,-.3516396496D-4,.2457520174D-5
     *,
     *    -.240337019D-6/, Q1,Q2,Q3,Q4,Q5/.04687499995D0,-.2002690873D-3
     *,
```

```
     *    .8449199096D-5,-.88228987D-6,.105787412D-6/
      IF(ABS(X).LT.8.)THEN
        Y=X**2
        BESSJ1=X*(R1+Y*(R2+Y*(R3+Y*(R4+Y*(R5+Y*R6)))))
     *      /(S1+Y*(S2+Y*(S3+Y*(S4+Y*(S5+Y*S6)))))
      ELSE
        AX=ABS(X)
        Z=8./AX
        Y=Z**2
        XX=AX-2.356194491
        BESSJ1=SQRT(.636619772/AX)*(COS(XX)*(P1+Y*(P2+Y*(P3+Y*(P4+Y
     *      *P5))))-Z*SIN(XX)*(Q1+Y*(Q2+Y*(Q3+Y*(Q4+Y*Q5)))))
     *      *SIGN(1.,X)
      ENDIF
      RETURN
      END
c
c
c
      FUNCTION BESSJ(N,X)
      implicit real*8 (a-h,o-z)
      PARAMETER (IACC=40,BIGNO=1.d10,BIGNI=1.d-10)
      if (x .ge. 0.d0) then
      ff = 1.d0
    else
      ff = (-1.d0)**n
    end if
```

```
      xx = abs(x)
      if (n .eq. 0) then
        bessj = ff*bessj0(xx)
        return
       endif
       if (n .eq. 1) then
        bessj = ff*bessj1(xx)
        return
       endif
       TOX=2./Xx
       IF(Xx.GT.FLOAT(N))THEN
        BJM=BESSJ0(Xx)
        BJ=BESSJ1(Xx)
        DO 11 J=1,N-1
          BJP=J*TOX*BJ-BJM
          BJM=BJ
          BJ=BJP
11      CONTINUE
        BESSJ=ff*BJ
       ELSE
        M=2*((N+INT(SQRT(FLOAT(IACC*N))))/2)
        BESSJ=0.
        JSUM=0
        SUM=0.
        BJP=0.
        BJ=1.
        DO 12 J=M,1,-1
```

```
      BJM=J*TOX*BJ-BJP
      BJP=BJ
      BJ=BJM
      IF(ABS(BJ).GT.BIGNO)THEN
        BJ=BJ*BIGNI
        BJP=BJP*BIGNI
        BESSJ=BESSJ*BIGNI
        SUM=SUM*BIGNI
      ENDIF
      IF(JSUM.NE.0)SUM=SUM+BJ
      JSUM=1-JSUM
      IF(J.EQ.N)BESSJ=BJP
12    CONTINUE
      SUM=2.*SUM-BJ
      BESSJ=ff*BESSJ/SUM
    ENDIF
    RETURN
    END
```

APPENDIX B

FORTRAN CODE FOR OPTICAL TM CASE

```
program special_TM


implicit none


integer nsize
parameter(nsize = 10000)


real*8 BESSI,om0,delta,b,theta,dt,ct,tt,t,pi,df,f,om,YY
real*8 rlam1,rlam5,rlame,expon(nsize),gams,temp,term1,term2
real*8 i01(nsize),i11(nsize),i15(nsize),i21(nsize),i25(nsize)
real*8 i11h(nsize),i15h(nsize),i21h(nsize),i25h(nsize),i0i2(nsize)
real*8 i1i1(nsize),i2i2(nsize),i0i1(nsize),i0i0(nsize),i05(nsize)
real*8 c3ft(nsize),c3f_t(nsize),f_t(nsize),temp2(nsize)
real*8 c3(nsize),ft(nsize),gammat(nsize)
real*8 c3shift(nsize),ftshift(nsize),f_tshift(nsize)
integer i,nt,nf,c3case3,shift
complex*16 z1,z2,s,zj,gamma,zz1,zz2,s1,s2,s5,s6,zze,se,sf,num


om0   = 4.d16
b     = dsqrt(20.d32)
delta = 0.28d16
theta = 30.d0
pi = 4.d0*atan(1.d0)
```

```
      ct = cos(theta*pi/180.d0)
      tt = tan(theta*pi/180.d0)
      zj=dcmplx(0.d0,1.d0)
      gams = b*b/(tt*tt-1)
      c3case3 = 0
      shift = 4
      df = 8d3
      nf = 4096*2*2*2
      nt = 4096
      dt = 1.d-18
c
      zz1 = cdsqrt(dcmplx(delta*delta-om0*om0,0.d0))
      s1  = -delta+zz1
        s2  = -delta-zz1
      zz2 = cdsqrt(dcmplx(delta*delta-om0*om0-b*b/(ct*ct),0.d0))
      s5  = -delta+zz2
      s6  = -delta-zz2
      zze = cdsqrt(dcmplx(delta*delta-om0*om0+gams,0.d0))
      se  = -delta+zze
      sf  = -delta-zze
! ----------------------------------------------------------------------
!     Frequency Domain Reflection Coefficient (gamma.dat)
! ----------------------------------------------------------------------
      open (10,file='gamma.dat',status='unknown')
      do i=1,nf
          f = df*i
          om = 2.d0*pi*f*1.d9
```

```
        s = zj*om
        z1 = cdsqrt(s-s1)*cdsqrt(s-s2)*cdsqrt(s-s5)*cdsqrt(s-s6)
        z2 = (s-s1)*(s-s2)+b*b
        gamma = (z1-z2)/(z1+z2)
        write (10,*) f,real(gamma),aimag(gamma)
    end do
    close (10)
    do i=1,nt
        t = (i-1)*dt+.00001*dt
        expon(i) = exp(-delta*t)
    end do

! ----------------------------------------------------------------------
!    Calculate C Term
! ----------------------------------------------------------------------
    if((om0*om0).gt.(delta*delta+gams)) then
        write(*,*) "C term 1"
        rlame = dsqrt(om0*om0-delta*delta-gams)
        do i=1,nt
            t = (i-1)*dt+.00001*dt
            c3(i) = expon(i)*sin(rlame*t)/rlame
        end do
    elseif((om0*om0).gt.(gams)) then
        write(*,*) "C term 2"
        rlame = dsqrt(delta*delta-om0*om0+gams)
        do i=1,nt
            t = (i-1)*dt+.00001*dt
```

```
        c3(i) = expon(i)*sinh(rlame*t)/rlame
      end do
    else
      write(*,*) "C term 3"
      rlame = dsqrt(delta*delta-om0*om0+gams)
      c3case3 = 1
      rlame = dsqrt(delta*delta-om0*om0+gams)
      do i=1,nt/shift
        t = (i-1-nt/shift)*dt+.00001*dt
        c3(i) = -exp((-delta+rlame)*t)/(2.d0*rlame)
      end do
      do i=nt/shift+1,nt
        t = (i-1-nt/shift)*dt+.00001*dt
        c3(i) = -exp(-(delta+rlame)*t)/(2.d0*rlame)
      end do
    endif
! ---------------------------------------------------------------------
!   Calculate F Term
! ---------------------------------------------------------------------
!   Case 1: om0^2>delta^2
! ---------------------------------------------------------------------
    if((om0*om0).gt.(delta*delta)) then
      write(*,*) "CASE 1"
      rlam1 = dsqrt(om0*om0-delta*delta)
      rlam5 = dsqrt(om0*om0+(b*b/(ct*ct))-delta*delta)
      do i=1,nt
        t = (i-1)*dt+.00001*dt
```

```
         call jbess(rlam1*t,0,i01(i),temp)
         call jbess(rlam1*t,1,i11(i),temp)
         call jbess(rlam5*t,1,i15(i),temp)
         call jbess(rlam1*t,2,i21(i),temp)
         call jbess(rlam5*t,2,i25(i),temp)
         i11h(i) = i11(i)/t
         i15h(i) = i15(i)/t
         i21h(i) = i21(i)/t
         i25h(i) = i25(i)/t
      end do
      call conv(i11h,i15h,nt,dt,i1i1)
      call conv(i01,i15h,nt,dt,i0i1)
      do i=1,nt
         t = (i-1)*dt+.00001*dt
         ft(i) = 2.d0*expon(i)*(-rlam1*rlam5*i1i1(i)+rlam1*rlam1*
     2            i21h(i)+rlam5*rlam5*i25h(i)-b*b*rlam5*i0i1(i)+
     3            b*b*rlam1*i11(i)+b*b*b*b*sin(rlam1*t)/(2.d0*rlam1))
      end do
! ---------------------------------------------------------------------
!   Case 2: om0^2<delta^2-b^2/ct^2
! ---------------------------------------------------------------------
    elseif((om0*om0).lt.(delta*delta-(b*b/(ct*ct)))) then
      write(*,*) "CASE 2"
      rlam1 = dsqrt(delta*delta-om0*om0)
      rlam5 = dsqrt(-om0*om0-(b*b/(ct*ct))+delta*delta)
      do i=1,nt
         t = (i-1)*dt+.00001*dt
```

```
        i01(i) = BESSI(0,rlam1*t)
        i11(i) = BESSI(1,rlam1*t)
        i15(i) = BESSI(1,rlam5*t)
        i21(i) = BESSI(2,rlam1*t)
        i25(i) = BESSI(2,rlam5*t)
        i11h(i) = i11(i)/t
        i15h(i) = i15(i)/t
        i21h(i) = i21(i)/t
        i25h(i) = i25(i)/t
    end do
    call conv(i11h,i15h,nt,dt,i1i1)
    call conv(i01,i15h,nt,dt,i0i1)
    do i=1,nt
        t = (i-1)*dt+.00001*dt
        ft(i) = 2.d0*expon(i)*(-rlam1*rlam5*i1i1(i)+rlam1*rlam1*
   2          i21h(i)+rlam5*rlam5*i25h(i)+b*b*rlam5*i0i1(i)-
   3          b*b*rlam1*i11(i)+b*b*b*b*sinh(rlam1*t)/(2.d0*rlam1))
    end do
! ----------------------------------------------------------------------
!   Case 3: om0^2>delta^2-b^2/ct^2 and om0^2<delta^2
! ----------------------------------------------------------------------
  else
    write(*,*) "CASE 3"
    rlam1 = dsqrt(delta*delta-om0*om0)
    rlam5 = dsqrt(om0*om0+(b*b/(ct*ct))-delta*delta)
    do i=1,nt
        t = (i-1)*dt+.00001*dt
```

```
            i01(i) = BESSI(0,rlam1*t)
            i11(i) = BESSI(1,rlam1*t)
            call jbess(rlam5*t,1,i15(i),temp)
            i21(i) = BESSI(2,rlam1*t)
            call jbess(rlam5*t,2,i25(i),temp)
            i11h(i) = i11(i)/t
            i15h(i) = i15(i)/t
            i21h(i) = i21(i)/t
            i25h(i) = i25(i)/t
         end do
         call conv(i11h,i15h,nt,dt,i1i1)
         call conv(i01,i15h,nt,dt,i0i1)
         do i=1,nt
            t = (i-1)*dt+.00001*dt
            ft(i) = 2.d0*expon(i)*(rlam1*rlam5*i1i1(i)+rlam1*rlam1*
     2            i21h(i)+rlam5*rlam5*i25h(i)-b*b*rlam5*i0i1(i)-
     3            b*b*rlam1*i11(i)+b*b*b*b*sinh(rlam1*t)/(2.d0*rlam1))
         end do
      endif
! ----------------------------------------------------------------------
!   Time Domain Reflection Coefficient (gammat.dat)
! ----------------------------------------------------------------------
      term2 = b*b*(tt*tt-1)
      if (theta.eq.45.d0) then
         call conv(ft,ft,nt,dt,f_t)
         do i=1,nt
            gammat(i) = -f_t(i)/(b*b*b*b)
```

```
    end do
elseif (c3case3.eq.0) then
    call conv(c3,ft,nt,dt,c3ft)
    do i=1,nt
        gammat(i) = (ft(i)+gams*c3ft(i))/term2
        temp2(i) = gammat(i)
    end do
else
    do i=1,(nt/shift)
        ftshift(i) = 0
    end do
    do i=(nt/shift+1),nt
        ftshift(i) = ft(i-nt/shift)
    end do
    call conv(c3,ftshift,nt,dt,c3ft)
    do i=1,(nt/shift)
        f_tshift(i) = 0
    end do
    do i=(nt/shift+1),nt
        f_tshift(i) = ftshift(i-nt/shift)
    end do
    do i=1,nt
        gammat(i) = (f_tshift(i)+gams*c3ft(i))/term2
    end do
endif
open(10,file='gammat.dat',status='unknown')
do i=1,nt
```

```
      if (c3case3.eq.0) then
        write(10,*) ((i-1)*dt+.00001*dt)*1.d9,gammat(i)
      else
        write(10,*) ((i-1-2*nt/(shift))*dt+.00001*dt)*1.d9,
   2            gammat(i)
      endif
    end do
    close(10)
    end

! ---------------------------------------------------------------------
    subroutine conv(f,g,n,del,c)
!
!    performs convolution of waveforms f and g using
!     linear interpolation
!
    parameter (nsize=10000)
    parameter (c13=0.33333333333333333333d0,
   2           c16=0.16666666666666666666d0)
    implicit real*8 (a-h,o-z)
    real*8 f(nsize),g(nsize),c(nsize)
    c(1)=0.d0
    do k=2,n
      sum = 0.d0
      do l=1,k-1
        sum = sum + c13*f(l+1)*g(k-l)
   2              + c13*f(l)*g(k-l+1)
```

```
 3              + c16*f(l)*g(k-l)
 4              + c16*f(l+1)*g(k-l+1)
  end do
  c(k) = del*sum
end do
return
end


! ----------------------------------------------------------------------
      FUNCTION BESSI(N,X)
!
!     This subroutine calculates the first kind modified Bessel function
!     of integer order N, for any REAL X. We use here the classical
!     recursion formula, when X > N. For X < N, the Miller's algorithm
!     is used to avoid overflows.
!     REFERENCE:
!     C.W.CLENSHAW, CHEBYSHEV SERIES FOR MATHEMATICAL FUNCTIONS,
!     MATHEMATICAL TABLES, VOL.5, 1962.
!
      PARAMETER (IACC = 40,BIGNO = 1.D10, BIGNI = 1.D-10)
      REAL *8 X,BESSI,BESSI0,BESSI1,TOX,BIM,BI,BIP
      IF (N.EQ.0) THEN
      BESSI = BESSI0(X)
      RETURN
      ENDIF
      IF (N.EQ.1) THEN
```

```
      BESSI = BESSI1(X)
      RETURN
      ENDIF
      IF(X.EQ.0.D0) THEN
      BESSI=0.D0
      RETURN
      ENDIF
      TOX = 2.D0/X
      BIP = 0.D0
      BI  = 1.D0
      BESSI = 0.D0
      M = 2*((N+INT(SQRT(FLOAT(IACC*N)))))
      DO 12 J = M,1,-1
      BIM = BIP+DFLOAT(J)*TOX*BI
      BIP = BI
      BI  = BIM
      IF (ABS(BI).GT.BIGNO) THEN
      BI  = BI*BIGNI
      BIP = BIP*BIGNI
      BESSI = BESSI*BIGNI
      ENDIF
      IF (J.EQ.N) BESSI = BIP
   12 CONTINUE
      BESSI = BESSI*BESSI0(X)/BI
      RETURN
      END
! --------------------------------------------------------------------
```

```
! Auxiliary Bessel functions for N=0, N=1
      FUNCTION BESSI0(X)
      REAL *8 X,BESSI0,Y,P1,P2,P3,P4,P5,P6,P7,
     *    Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9,AX,BX
      DATA P1,P2,P3,P4,P5,P6,P7/1.D0,3.5156229D0,3.0899424D0,1.2067429D0
     *    ,0.2659732D0,0.360768D-1,0.45813D-2/
      DATA Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9/0.39894228D0,0.1328592D-1,
     *    0.225319D-2,-0.157565D-2,0.916281D-2,-0.2057706D-1,
     *    0.2635537D-1,-0.1647633D-1,0.392377D-2/
      IF(ABS(X).LT.3.75D0) THEN
      Y=(X/3.75D0)**2
      BESSI0=P1+Y*(P2+Y*(P3+Y*(P4+Y*(P5+Y*(P6+Y*P7)))))
      ELSE
      AX=ABS(X)
      Y=3.75D0/AX
      BX=EXP(AX)/SQRT(AX)
      AX=Q1+Y*(Q2+Y*(Q3+Y*(Q4+Y*(Q5+Y*(Q6+Y*(Q7+Y*(Q8+Y*Q9)))))))
      BESSI0=AX*BX
      ENDIF
      RETURN
      END
! ----------------------------------------------------------------------
      FUNCTION BESSI1(X)
      REAL *8 X,BESSI1,Y,P1,P2,P3,P4,P5,P6,P7,
     *    Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9,AX,BX
      DATA P1,P2,P3,P4,P5,P6,P7/0.5D0,0.87890594D0,0.51498869D0,
     *    0.15084934D0,0.2658733D-1,0.301532D-2,0.32411D-3/
```

```
      DATA Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9/0.39894228D0,-0.3988024D-1,
     *    -0.362018D-2,0.163801D-2,-0.1031555D-1,0.2282967D-1,
     *    -0.2895312D-1,0.1787654D-1,-0.420059D-2/
      IF(ABS(X).LT.3.75D0) THEN
      Y=(X/3.75D0)**2
      BESSI1=X*(P1+Y*(P2+Y*(P3+Y*(P4+Y*(P5+Y*(P6+Y*P7))))))
      ELSE
      AX=ABS(X)
      Y=3.75D0/AX
      BX=EXP(AX)/SQRT(AX)
      AX=Q1+Y*(Q2+Y*(Q3+Y*(Q4+Y*(Q5+Y*(Q6+Y*(Q7+Y*(Q8+Y*Q9)))))))
      BESSI1=AX*BX
      ENDIF
      RETURN
      END
! ----------------------------------------------------------------------


     subroutine jbess (x,n,bj,bjp)
c
c  calculates jn and jn' for n positive or negative
c
      implicit real*8 (a-h,o-z)
c
     if (n .eq. 0) then
         bj  = bessj(0,x)
         bjp = -bessj(1,x)
```

```
        return
      endif
c
      if (n .ge. 0) then
        m = n
      else
        m = -n
      endif
c
      if (x .eq. 0.d0) then
        bj   = 0.d0
        bjp1 = bessj(m+1,x)
        bjm1 = bessj(m-1,x)
        bjp  = (bjm1-bjp1)/2.d0
      else
        bj     = bessj(m,x)
        bj1    = bessj(m+1,x)
        bjp    = -bj1 + n*bj/x
      endif
c
      if (n .lt. 0) then
        bj     = bj*(-1)**n
        bjp    = bjp*(-1)**n
      endif
c
      return
      end
```

```
c
c
c
      FUNCTION BESSJ0(X)
      implicit real*8 (a-h,o-z)
      REAL*8 Y,P1,P2,P3,P4,P5,Q1,Q2,Q3,Q4,Q5,R1,R2,R3,R4,R5,R6,
     *    S1,S2,S3,S4,S5,S6
      DATA P1,P2,P3,P4,P5/1.D0,-.1098628627D-2,.2734510407D-4,
     *    -.2073370639D-5,.2093887211D-6/, Q1,Q2,Q3,Q4,Q5/-.1562499995D-
     *1,
     *    .1430488765D-3,-.6911147651D-5,.7621095161D-6,-.934945152D-7/
      DATA R1,R2,R3,R4,R5,R6/57568490574.D0,-13362590354.D0,651619640.7D
     *0,
     *    -11214424.18D0,77392.33017D0,-184.9052456D0/,
     *    S1,S2,S3,S4,S5,S6/57568490411.D0,1029532985.D0,
     *    9494680.718D0,59272.64853D0,267.8532712D0,1.D0/
      IF(ABS(X).LT.8.)THEN
        Y=X**2
        BESSJ0=(R1+Y*(R2+Y*(R3+Y*(R4+Y*(R5+Y*R6)))))
     *      /(S1+Y*(S2+Y*(S3+Y*(S4+Y*(S5+Y*S6)))))
      ELSE
        AX=ABS(X)
        Z=8./AX
        Y=Z**2
        XX=AX-.785398164
        BESSJ0=SQRT(.636619772/AX)*(COS(XX)*(P1+Y*(P2+Y*(P3+Y*(P4+Y
     *      *P5))))-Z*SIN(XX)*(Q1+Y*(Q2+Y*(Q3+Y*(Q4+Y*Q5)))))
```

```
      ENDIF
      RETURN
      END
c
c
c
      FUNCTION BESSJ1(X)
      implicit real*8 (a-h,o-z)
      REAL*8 Y,P1,P2,P3,P4,P5,Q1,Q2,Q3,Q4,Q5,R1,R2,R3,R4,R5,R6,
     *    S1,S2,S3,S4,S5,S6
      DATA R1,R2,R3,R4,R5,R6/72362614232.D0,-7895059235.D0,242396853.1D0
     *,
     *    -2972611.439D0,15704.48260D0,-30.16036606D0/,
     *    S1,S2,S3,S4,S5,S6/144725228442.D0,2300535178.D0,
     *    18583304.74D0,99447.43394D0,376.9991397D0,1.D0/
      DATA P1,P2,P3,P4,P5/1.D0,.183105D-2,-.3516396496D-4,.2457520174D-5
     *,
     *    -.240337019D-6/, Q1,Q2,Q3,Q4,Q5/.04687499995D0,-.2002690873D-3
     *,
     *    .8449199096D-5,-.88228987D-6,.105787412D-6/
      IF(ABS(X).LT.8.)THEN
        Y=X**2
        BESSJ1=X*(R1+Y*(R2+Y*(R3+Y*(R4+Y*(R5+Y*R6)))))
     *      /(S1+Y*(S2+Y*(S3+Y*(S4+Y*(S5+Y*S6)))))
      ELSE
        AX=ABS(X)
        Z=8./AX
```

```
        Y=Z**2
        XX=AX-2.356194491
        BESSJ1=SQRT(.636619772/AX)*(COS(XX)*(P1+Y*(P2+Y*(P3+Y*(P4+Y
     *      *P5))))-Z*SIN(XX)*(Q1+Y*(Q2+Y*(Q3+Y*(Q4+Y*Q5)))))
     *      *SIGN(1.,X)
      ENDIF
      RETURN
      END
c
c
c
      FUNCTION BESSJ(N,X)
      implicit real*8 (a-h,o-z)
      PARAMETER (IACC=40,BIGNO=1.d10,BIGNI=1.d-10)
      if (x .ge. 0.d0) then
      ff = 1.d0
     else
      ff = (-1.d0)**n
     end if
     xx = abs(x)
     if (n .eq. 0) then
        bessj = ff*bessj0(xx)
        return
      endif
      if (n .eq. 1) then
        bessj = ff*bessj1(xx)
        return
```

```
      endif
      TOX=2./Xx
      IF(Xx.GT.FLOAT(N))THEN
        BJM=BESSJ0(Xx)
        BJ=BESSJ1(Xx)
        DO 11 J=1,N-1
          BJP=J*TOX*BJ-BJM
          BJM=BJ
          BJ=BJP
11      CONTINUE
        BESSJ=ff*BJ
      ELSE
        M=2*((N+INT(SQRT(FLOAT(IACC*N))))/2)
        BESSJ=0.
        JSUM=0
        SUM=0.
        BJP=0.
        BJ=1.
        DO 12 J=M,1,-1
          BJM=J*TOX*BJ-BJP
          BJP=BJ
          BJ=BJM
          IF(ABS(BJ).GT.BIGNO)THEN
            BJ=BJ*BIGNI
            BJP=BJP*BIGNI
            BESSJ=BESSJ*BIGNI
            SUM=SUM*BIGNI
```

```
        ENDIF
        IF(JSUM.NE.0)SUM=SUM+BJ
        JSUM=1-JSUM
        IF(J.EQ.N)BESSJ=BJP
12    CONTINUE
      SUM=2.*SUM-BJ
      BESSJ=ff*BESSJ/SUM
    ENDIF
    RETURN
    END
```

APPENDIX C

FORTRAN CODE FOR GENERAL TE CASE

```
      program general_TE


      implicit none


      integer nsize
      parameter(nsize = 50000)


      real*8 BESSI,om0,delta,b,theta,dt,ct,tt,t,pi,df,f,om,YY
      real*8 rmur,epsinf,st,rkap,f1,f2
      real*8 rlam1,rlam5,rlama,expon(nsize),gams,temp
      real*8 i11h(nsize),i15h(nsize),i15(nsize),i11(nsize)
      real*8 i21h(nsize),i25h(nsize),i1i1(nsize)
      real*8 c3ft(nsize),c3shift(nsize),ftshift(nsize)
      real*8 c3(nsize),ft(nsize),gammat(nsize)
      integer i,nt,nf,c3case3,shift
      complex*16 z1,z2,s,zj,gamma,zz1,zz2,s1,s2,s5,s6
! ----------------------------------------------------------------------
!    Material Parameters
! ----------------------------------------------------------------------
      rmur = 1.d0
      epsinf = 2.d0
      om0   = 4.d11
      b     = 6.24d11
```

```
      delta = 2.5d10
      theta = 60.d0

      pi = 4.d0*atan(1.d0)
      ct = cos(theta*pi/180.d0)
      st = sin(theta*pi/180.d0)
      tt = tan(theta*pi/180.d0)
      zj=dcmplx(0.d0,1.d0)
      c3case3 = 0
      shift = 8
      rkap = dsqrt(rmur*epsinf-st*st)
      gams = om0*om0-(rmur*b*b)/(rmur*rmur*ct*ct-rkap*rkap)
      f1 = 2.d0*rkap*rmur*ct/(rmur*ct+rkap)
      f2 = f1/(rmur*rmur*ct*ct-rkap*rkap)
      df = 0.4d0
      nf = 4095*2
      nt = 4096*2
      dt = 2.d-13
      write(*,*) om0*om0
      write(*,*) delta*delta-rmur*b*b/(rkap*rkap)
c
      zz1 = cdsqrt(dcmplx(delta*delta-om0*om0,0.d0))
      s1  = -delta+zz1
      s2  = -delta-zz1
      zz2 = cdsqrt(dcmplx(delta*delta-om0*om0-rmur*b*b/(rkap*rkap),0.d0))
      s5  = -delta+zz2
      s6  = -delta-zz2
```

```
! ----------------------------------------------------------------------
!    Frequency Domain Reflection Coefficient (gamma.dat)
! ----------------------------------------------------------------------
   open (10,file='gamma.dat',status='unknown')
   do i=1,nf
       f = df*i
       om = 2.d0*pi*f*1.d9
       s = zj*om
       z1 = rmur*ct*cdsqrt(s-s1)*cdsqrt(s-s2)
       z2 = rkap*cdsqrt(s-s5)*cdsqrt(s-s6)
       gamma = (z1-z2)/(z1+z2)-(rmur*ct-rkap)/(rmur*ct+rkap)
       write (10,*) f,real(gamma),aimag(gamma)
   end do
   close (10)

   do i=1,nt
       t = (i-1)*dt+.00001*dt
       expon(i) = exp(-delta*t)
   end do

! ----------------------------------------------------------------------
!    Calculate C Term
! ----------------------------------------------------------------------
   if(gams.gt.(delta*delta)) then
       write(*,*) "C term 1"
       rlama = dsqrt(gams-delta*delta)
       do i=1,nt
```

```
        t = (i-1)*dt+.00001*dt
        c3(i) = expon(i)*sin(rlama*t)/rlama
    end do
  elseif(0.gt.gams) then
    write(*,*) "C term 3"
    rlama = dsqrt(delta*delta-gams)
    c3case3 = 1
    do i=1,nt/shift
        t = (i-1-nt/shift)*dt+.00001*dt
        c3(i) = -exp((-delta+rlama)*t)/(2.d0*rlama)
    end do
    do i=nt/shift+1,nt
        t = (i-1-nt/shift)*dt+.00001*dt
        c3(i) = -exp(-(delta+rlama)*t)/(2.d0*rlama)
    end do

  else
    write(*,*) "C term 2"
    rlama = dsqrt(delta*delta-gams)
    do i=1,nt
        t = (i-1)*dt+.00001*dt
        c3(i) = expon(i)*sinh(rlama*t)/rlama
    end do

  endif
! ----------------------------------------------------------------------
!   Calculate G Term
```

```
! ----------------------------------------------------------------------
!    Case 1: om0^2>delta^2
! ----------------------------------------------------------------------
   if((om0*om0).gt.(delta*delta)) then
       write(*,*) "CASE 1"
       rlam1 = dsqrt(om0*om0-delta*delta)
       rlam5 = dsqrt(om0*om0+(rmur*b*b/(rkap*rkap))-delta*delta)
       do i=1,nt
           t = (i-1)*dt+.00001*dt
           call jbess(rlam1*t,1,i1lh(i),temp)
           call jbess(rlam5*t,1,i15(i),temp)
           call jbess(rlam1*t,2,i21h(i),temp)
           call jbess(rlam5*t,2,i25h(i),temp)
           i11h(i) = i11h(i)/t
           i15h(i) = i15(i)/t
           i21h(i) = i21h(i)/t
           i25h(i) = i25h(i)/t
       end do
       call conv(i11h,i15h,nt,dt,i1i1)
       do i=1,nt
           ft(i) = expon(i)*(rmur*ct+rkap)*(-rlam1*rlam5*i1i1(i)+
    2            rlam1*rlam1*i21h(i)+rlam5*rlam5*i25h(i))
       end do
! ----------------------------------------------------------------------
!    Case 2: om0^2<delta^2-rmur*b^2/rkap^2
! ----------------------------------------------------------------------
   elseif((om0*om0).lt.(delta*delta-(rmur*b*b/(rkap*rkap)))) then
```

```
      write(*,*) "CASE 2"
      rlam1 = dsqrt(delta*delta-om0*om0)
      rlam5 = dsqrt(delta*delta-om0*om0-rmur*b*b/(rkap*rkap))
      do i=1,nt
          t = (i-1)*dt+.00001*dt
          i11(i) = BESSI(1,rlam1*t)
          i15(i) = BESSI(1,rlam5*t)
          i21h(i) = BESSI(2,rlam1*t)
          i25h(i) = BESSI(2,rlam5*t)
          i11h(i) = i11(i)/t
          i15h(i) = i15(i)/t
          i21h(i) = i21h(i)/t
          i25h(i) = i25h(i)/t
      end do
      call conv(i11h,i15h,nt,dt,i1i1)
      do i=1,nt
          ft(i) = expon(i)*(rmur*ct+rkap)*(-rlam1*rlam5*i1i1(i)+
     2             rlam1*rlam1*i21h(i)+rlam5*rlam5*i25h(i))
      end do
! ---------------------------------------------------------------------
!    Case 3: om0^2>delta^2-rmur*b^2/rkap^2 and om0^2<delta^2
! ---------------------------------------------------------------------
    else
      write(*,*) "CASE 3"
      rlam1 = dsqrt(delta*delta-om0*om0)
      rlam5 = dsqrt(om0*om0+(rmur*b*b/(rkap*rkap))-delta*delta)
      do i=1,nt
```

```
        t = (i-1)*dt+.00001*dt
        i11h(i) = BESSI(1,rlam1*t)
        call jbess(rlam5*t,1,i15(i),temp)
        i21h(i) = BESSI(2,rlam1*t)
        call jbess(rlam5*t,2,i25h(i),temp)
        i11h(i) = i11h(i)/t
        i15h(i) = i15(i)/t
        i21h(i) = i21h(i)/t
        i25h(i) = i25h(i)/t
    end do
    call conv(i11h,i15h,nt,dt,i1i1)
    do i=1,nt
        ft(i) = expon(i)*(rmur*ct+rkap)*(rlam1*rlam5*i1i1(i)+
   2            rlam1*rlam1*i21h(i)+rlam5*rlam5*i25h(i))
    end do
  endif
! ----------------------------------------------------------------------
!   Time Domain Reflection Coefficient (gammat.dat)
! ----------------------------------------------------------------------
  YY = (rmur*b*b)*(rkap-rmur*ct)/(2.d0*rkap*rkap)
  if (ct.eq.rkap/rmur) then
    write(*,*) 'Special Denom'
    do i=1,nt
        gammat(i) = -f1*ft(i)/(rmur*b*b)
    end do
  elseif (c3case3.eq.0) then
    call conv(c3,ft,nt,dt,c3ft)
```

```
    do i=1,nt
        gammat(i) = f2*(c3ft(i)+YY*c3(i))
    end do
else
    write(*,*) 'non-causal'
    do i=1,(nt/shift)
        ftshift(i) = 0
    end do
    do i=(nt/shift+1),nt
        ftshift(i) = ft(i-nt/shift)
    end do
    call conv(c3,ftshift,nt,dt,c3ft)
    do i=1,(nt/shift+1)
        c3shift(i) = 0
    end do
    do i=(nt/shift+2),nt
        c3shift(i) = c3(i-nt/shift-1)
    end do
    do i=1,nt
        gammat(i) = f2*(c3ft(i)+YY*c3shift(i))
    end do
endif

open(10,file='gammat.dat',status='unknown')
do i=1,nt
    if (c3case3.eq.0) then
        write(10,*) ((i-1)*dt+.00001*dt)*1.d9,gammat(i)
```

```
      else
          write(10,*) ((i-1-2*nt/(shift))*dt+.00001*dt)*1.d9,gammat(i)
      endif
    end do
    close(10)


    end

! ---------------------------------------------------------------------
    subroutine conv(f,g,n,del,c)
!
!    performs convolution of waveforms f and g using
!     linear interpolation
!
    parameter (nsize=10000)
    parameter (c13=0.33333333333333333333d0,
     2            c16=0.16666666666666666666d0)
    implicit real*8 (a-h,o-z)
    real*8 f(nsize),g(nsize),c(nsize)
    c(1)=0.d0
    do k=2,n
      sum = 0.d0
      do l=1,k-1
        sum = sum + c13*f(l+1)*g(k-l)
     2              + c13*f(l)*g(k-l+1)
     3              + c16*f(l)*g(k-l)
     4              + c16*f(l+1)*g(k-l+1)
```

```
      end do
      c(k) = del*sum
    end do
    return
    end


! ----------------------------------------------------------------------
      FUNCTION BESSI(N,X)
!
!     This subroutine calculates the first kind modified Bessel function
!     of integer order N, for any REAL X. We use here the classical
!     recursion formula, when X > N. For X < N, the Miller's algorithm
!     is used to avoid overflows.
!     REFERENCE:
!     C.W.CLENSHAW, CHEBYSHEV SERIES FOR MATHEMATICAL FUNCTIONS,
!     MATHEMATICAL TABLES, VOL.5, 1962.
!
      PARAMETER (IACC = 40,BIGNO = 1.D10, BIGNI = 1.D-10)
      REAL *8 X,BESSI,BESSI0,BESSI1,TOX,BIM,BI,BIP
      IF (N.EQ.0) THEN
      BESSI = BESSI0(X)
      RETURN
      ENDIF
      IF (N.EQ.1) THEN
      BESSI = BESSI1(X)
      RETURN
```

```
      ENDIF
      IF(X.EQ.0.D0) THEN
      BESSI=0.D0
      RETURN
      ENDIF
      TOX = 2.D0/X
      BIP = 0.D0
      BI  = 1.D0
      BESSI = 0.D0
      M = 2*((N+INT(SQRT(FLOAT(IACC*N)))))
      DO 12 J = M,1,-1
      BIM = BIP+DFLOAT(J)*TOX*BI
      BIP = BI
      BI  = BIM
      IF (ABS(BI).GT.BIGNO) THEN
      BI  = BI*BIGNI
      BIP = BIP*BIGNI
      BESSI = BESSI*BIGNI
      ENDIF
      IF (J.EQ.N) BESSI = BIP
   12 CONTINUE
      BESSI = BESSI*BESSI0(X)/BI
      RETURN
      END
! ----------------------------------------------------------------------
! Auxiliary Bessel functions for N=0, N=1
      FUNCTION BESSI0(X)
```

```
      REAL *8 X,BESSI0,Y,P1,P2,P3,P4,P5,P6,P7,
     *    Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9,AX,BX
      DATA P1,P2,P3,P4,P5,P6,P7/1.D0,3.5156229D0,3.0899424D0,1.2067429D0
     *    ,0.2659732D0,0.360768D-1,0.45813D-2/
      DATA Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9/0.39894228D0,0.1328592D-1,
     *    0.225319D-2,-0.157565D-2,0.916281D-2,-0.2057706D-1,
     *    0.2635537D-1,-0.1647633D-1,0.392377D-2/
      IF(ABS(X).LT.3.75D0) THEN
      Y=(X/3.75D0)**2
      BESSI0=P1+Y*(P2+Y*(P3+Y*(P4+Y*(P5+Y*(P6+Y*P7)))))
      ELSE
      AX=ABS(X)
      Y=3.75D0/AX
      BX=EXP(AX)/SQRT(AX)
      AX=Q1+Y*(Q2+Y*(Q3+Y*(Q4+Y*(Q5+Y*(Q6+Y*(Q7+Y*(Q8+Y*Q9)))))))
      BESSI0=AX*BX
      ENDIF
      RETURN
      END
! -----------------------------------------------------------------
      FUNCTION BESSI1(X)
      REAL *8 X,BESSI1,Y,P1,P2,P3,P4,P5,P6,P7,
     *    Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9,AX,BX
      DATA P1,P2,P3,P4,P5,P6,P7/0.5D0,0.87890594D0,0.51498869D0,
     *    0.15084934D0,0.2658733D-1,0.301532D-2,0.32411D-3/
      DATA Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9/0.39894228D0,-0.3988024D-1,
     *    -0.362018D-2,0.163801D-2,-0.1031555D-1,0.2282967D-1,
```

```
     *    -0.2895312D-1,0.1787654D-1,-0.420059D-2/
      IF(ABS(X).LT.3.75D0) THEN
      Y=(X/3.75D0)**2
      BESSI1=X*(P1+Y*(P2+Y*(P3+Y*(P4+Y*(P5+Y*(P6+Y*P7))))))
      ELSE
      AX=ABS(X)
      Y=3.75D0/AX
      BX=EXP(AX)/SQRT(AX)
      AX=Q1+Y*(Q2+Y*(Q3+Y*(Q4+Y*(Q5+Y*(Q6+Y*(Q7+Y*(Q8+Y*Q9)))))))
      BESSI1=AX*BX
      ENDIF
      RETURN
      END
! ----------------------------------------------------------------------


    subroutine jbess (x,n,bj,bjp)
c
c  calculates jn and jn' for n positive or negative
c
      implicit real*8 (a-h,o-z)
c
    if (n .eq. 0) then
        bj  = bessj(0,x)
        bjp = -bessj(1,x)
        return
      endif
```

```
c
      if (n .ge. 0) then
        m = n
      else
        m = -n
      endif
c
      if (x .eq. 0.d0) then
        bj   = 0.d0
        bjp1 = bessj(m+1,x)
        bjm1 = bessj(m-1,x)
        bjp  = (bjm1-bjp1)/2.d0
      else
        bj     = bessj(m,x)
        bj1    = bessj(m+1,x)
        bjp    = -bj1 + n*bj/x
      endif
c
      if (n .lt. 0) then
        bj     = bj*(-1)**n
        bjp    = bjp*(-1)**n
      endif
c
      return
      end
c
c
```

```
c
      FUNCTION BESSJ0(X)
      implicit real*8 (a-h,o-z)
      REAL*8 Y,P1,P2,P3,P4,P5,Q1,Q2,Q3,Q4,Q5,R1,R2,R3,R4,R5,R6,
     *    S1,S2,S3,S4,S5,S6
      DATA P1,P2,P3,P4,P5/1.D0,-.1098628627D-2,.2734510407D-4,
     *    -.2073370639D-5,.2093887211D-6/, Q1,Q2,Q3,Q4,Q5/-.1562499995D-
     *1,
     *    .1430488765D-3,-.6911147651D-5,.7621095161D-6,-.934945152D-7/
      DATA R1,R2,R3,R4,R5,R6/57568490574.D0,-13362590354.D0,651619640.7D
     *0,
     *    -11214424.18D0,77392.33017D0,-184.9052456D0/,
     *    S1,S2,S3,S4,S5,S6/57568490411.D0,1029532985.D0,
     *    9494680.718D0,59272.64853D0,267.8532712D0,1.D0/
      IF(ABS(X).LT.8.)THEN
        Y=X**2
        BESSJ0=(R1+Y*(R2+Y*(R3+Y*(R4+Y*(R5+Y*R6)))))
     *      /(S1+Y*(S2+Y*(S3+Y*(S4+Y*(S5+Y*S6)))))
      ELSE
        AX=ABS(X)
        Z=8./AX
        Y=Z**2
        XX=AX-.785398164
        BESSJ0=SQRT(.636619772/AX)*(COS(XX)*(P1+Y*(P2+Y*(P3+Y*(P4+Y
     *      *P5))))-Z*SIN(XX)*(Q1+Y*(Q2+Y*(Q3+Y*(Q4+Y*Q5)))))
      ENDIF
      RETURN
```

```
      END
c
c
c
      FUNCTION BESSJ1(X)
      implicit real*8 (a-h,o-z)
      REAL*8 Y,P1,P2,P3,P4,P5,Q1,Q2,Q3,Q4,Q5,R1,R2,R3,R4,R5,R6,
     *    S1,S2,S3,S4,S5,S6
      DATA R1,R2,R3,R4,R5,R6/72362614232.D0,-7895059235.D0,242396853.1D0
     *,
     *    -2972611.439D0,15704.48260D0,-30.16036606D0/,
     *    S1,S2,S3,S4,S5,S6/144725228442.D0,2300535178.D0,
     *    18583304.74D0,99447.43394D0,376.9991397D0,1.D0/
      DATA P1,P2,P3,P4,P5/1.D0,.183105D-2,-.3516396496D-4,.2457520174D-5
     *,
     *    -.240337019D-6/, Q1,Q2,Q3,Q4,Q5/.04687499995D0,-.2002690873D-3
     *,
     *    .8449199096D-5,-.88228987D-6,.105787412D-6/
      IF(ABS(X).LT.8.)THEN
        Y=X**2
        BESSJ1=X*(R1+Y*(R2+Y*(R3+Y*(R4+Y*(R5+Y*R6)))))
     *      /(S1+Y*(S2+Y*(S3+Y*(S4+Y*(S5+Y*S6)))))
      ELSE
        AX=ABS(X)
        Z=8./AX
        Y=Z**2
        XX=AX-2.356194491
```

```
        BESSJ1=SQRT(.636619772/AX)*(COS(XX)*(P1+Y*(P2+Y*(P3+Y*(P4+Y
     *      *P5))))-Z*SIN(XX)*(Q1+Y*(Q2+Y*(Q3+Y*(Q4+Y*Q5)))))
     *      *SIGN(1.,X)
      ENDIF
      RETURN
      END
c
c
c
      FUNCTION BESSJ(N,X)
      implicit real*8 (a-h,o-z)
      PARAMETER (IACC=40,BIGNO=1.d10,BIGNI=1.d-10)
      if (x .ge. 0.d0) then
      ff = 1.d0
    else
      ff = (-1.d0)**n
    end if
    xx = abs(x)
    if (n .eq. 0) then
        bessj = ff*bessj0(xx)
        return
      endif
      if (n .eq. 1) then
        bessj = ff*bessj1(xx)
        return
      endif
      TOX=2./Xx
```

```
      IF(Xx.GT.FLOAT(N))THEN
        BJM=BESSJ0(Xx)
        BJ=BESSJ1(Xx)
        DO 11 J=1,N-1
          BJP=J*TOX*BJ-BJM
          BJM=BJ
          BJ=BJP
11      CONTINUE
        BESSJ=ff*BJ
      ELSE
        M=2*((N+INT(SQRT(FLOAT(IACC*N))))/2)
        BESSJ=0.
        JSUM=0
        SUM=0.
        BJP=0.
        BJ=1.
        DO 12 J=M,1,-1
          BJM=J*TOX*BJ-BJP
          BJP=BJ
          BJ=BJM
          IF(ABS(BJ).GT.BIGNO)THEN
            BJ=BJ*BIGNI
            BJP=BJP*BIGNI
            BESSJ=BESSJ*BIGNI
            SUM=SUM*BIGNI
          ENDIF
          IF(JSUM.NE.0)SUM=SUM+BJ
```

```
        JSUM=1-JSUM
        IF(J.EQ.N)BESSJ=BJP
12    CONTINUE
      SUM=2.*SUM-BJ
      BESSJ=ff*BESSJ/SUM
    ENDIF
    RETURN
    END
```

APPENDIX D

FORTRAN CODE FOR GENERAL TM CASE

```
      program general_TM


      implicit none


      integer nsize
      parameter(nsize = 50000)


      real*8 BESSI,om0,delta,b,theta,dt,ct,st,tt,t,pi,df,f,om,YY
      real*8 rmur,epsinf,rkap,f1,sigma,chi
      real*8 rlam1,rlam5,rlamc,rlame,expon(nsize),gams,temp
      real*8 i11h(nsize),i15h(nsize),i15(nsize),i11(nsize)
      real*8 i21h(nsize),i25h(nsize),i1i1(nsize),i01(nsize)
      real*8 c3ft(nsize),c3shift(nsize),ftshift(nsize)
      real*8 c3(nsize),ft(nsize),gammat(nsize),i0i1(nsize)
      real*8 c1(nsize),c2(nsize),ta(nsize),scale
      integer i,nt,nf,c1nc,c2nc,shift,c3case3
      complex*16 z1,z2,s,zj,gamma,zz1,zz2,s1,s2,s5,s6
! ----------------------------------------------------------------------
!    Material Parameters
! ----------------------------------------------------------------------
      rmur = 1.d0
      epsinf = 2.d0
      om0   = 4.d11
```

```
b     = 6.24d11
delta = 2.5d10
theta = 60.d0

pi = 4.d0*atan(1.d0)
ct = cos(theta*pi/180.d0)
st = sin(theta*pi/180.d0)
tt = tan(theta*pi/180.d0)
zj = dcmplx(0.d0,1.d0)
c1nc = 0
c2nc = 0
c3case3 = 0
scale = 1.d-20

rkap = dsqrt(rmur*epsinf-st*st)
gams = b*b*(rmur-2.d0*epsinf*ct*ct)/(rkap*rkap-epsinf*epsinf*ct*ct)
sigma = dsqrt(rmur*rmur-sin(2.d0*theta*pi/180.d0)*
 2    sin(2.d0*theta*pi/180.d0))
chi = (b*b*sigma)/(2.d0*(rkap*rkap-epsinf*epsinf*ct*ct))
shift = 8
YY = b*b*(rkap-epsinf*ct)*(rmur*epsinf-2.d0*rkap*rkap)/
 2    (2.d0*rkap*rkap*epsinf)
f1 = (2.d0*rkap*epsinf*ct)/((rkap+epsinf*ct)*(rkap+epsinf*ct)*
 2    (rkap-epsinf*ct))

df = 0.4d0
nf = 4095*8
```

```
      nt = 4096*2
      dt = 1d-13
c
      zz1 = cdsqrt(dcmplx(delta*delta-om0*om0,0.d0))
      s1  = -delta+zz1
      s2  = -delta-zz1
      zz2 = cdsqrt(dcmplx(delta*delta-om0*om0-rmur*b*b/(rkap*rkap),0.d0))
      s5  = -delta+zz2
      s6  = -delta-zz2
! ---------------------------------------------------------------------
!     Frequency Domain Reflection Coefficient (gamma.dat)
! ---------------------------------------------------------------------
      open (10,file='gamma.dat',status='unknown')
      do i=1,nf
          f = df*i
          om = 2.d0*pi*f*1.d9
          s = zj*om
          z1 = cdsqrt(s-s1)*cdsqrt(s-s2)*cdsqrt(s-s5)*cdsqrt(s-s6)
          z2 = epsinf*ct*((s-s1)*(s-s2)+b*b/epsinf)
          gamma = (rkap*z1-z2)/(rkap*z1+z2)-(rkap-epsinf*ct)/(rkap+epsinf*ct)
          write (10,*) f,real(gamma),aimag(gamma)
      end do
      close (10)


      do i=1,nt
          t = (i-1)*dt+.00001*dt
```

```
      expon(i) = exp(-delta*t)
   end do

! -----------------------------------------------------------------------
!    Calculate C1 Term
! -----------------------------------------------------------------------
   if(om0*om0+(gams/2.d0)-chi.gt.(delta*delta)) then
      write(*,*) "C1 term 1"
      rlamc = dsqrt(om0*om0+(gams/2.d0)-chi-delta*delta)
      do i=1,nt
         t = (i-1)*dt+.00001*dt
         c1(i) = (sigma-rmur+2.d0*epsinf*ct*ct)*expon(i)*
    2            sin(rlamc*t)/(2.d0*sigma*rlamc)
      end do
   elseif(om0*om0+(gams/2.d0)-chi.lt.0) then
      write(*,*) "C1 term 3"
      rlamc = dsqrt(-om0*om0-(gams/2.d0)+chi+delta*delta)
      c1nc = 1
      c3case3 = 1
      do i=1,nt/shift
         t = (i-1-nt/shift)*dt+.00001*dt
         c1(i) = -(sigma-rmur+2.d0*epsinf*ct*ct)*
    2            exp((-delta+rlamc)*t)/(4.d0*sigma*rlamc)
      end do
      do i=nt/shift+1,nt
         t = (i-1-nt/shift)*dt+.00001*dt
         c1(i) = -(sigma-rmur+2.d0*epsinf*ct*ct)*
```

```
 2            exp(-(delta+rlamc)*t)/(4.d0*sigma*rlamc)
    end do


  else
    write(*,*) "C1 term 2"
    rlamc = dsqrt(-om0*om0-(gams/2.d0)+chi+delta*delta)
    do i=1,nt
        t = (i-1)*dt+.00001*dt
        c1(i) = (sigma-rmur+2.d0*epsinf*ct*ct)*expon(i)*
 2            sinh(rlamc*t)/(2.d0*sigma*rlamc)
    end do


  endif
! ----------------------------------------------------------------------
!   Calculate C2 Term
! ----------------------------------------------------------------------
  if(om0*om0+(gams/2.d0)+chi.gt.(delta*delta)) then
    write(*,*) "C2 term 1"
    rlame = dsqrt(om0*om0+(gams/2.d0)+chi-delta*delta)
    do i=1,nt
        t = (i-1)*dt+.00001*dt
        c2(i) = (sigma+rmur-2.d0*epsinf*ct*ct)*expon(i)*
 2            sin(rlame*t)/(2.d0*sigma*rlame)
    end do
  elseif(om0*om0+(gams/2.d0)+chi.lt.0) then
    write(*,*) "C2 term 3"
    rlame = dsqrt(-om0*om0-(gams/2.d0)-chi+delta*delta)
```

```
      c2nc = 1
      c3case3 = 1
      do i=1,nt/shift
          t = (i-1-nt/shift)*dt+.00001*dt
          c2(i) = -(sigma+rmur-2.d0*epsinf*ct*ct)*
     2            exp((-delta+rlame)*t)/(4.d0*sigma*rlame)
      end do
      do i=nt/shift+1,nt
          t = (i-1-nt/shift)*dt+.00001*dt
          c2(i) = -(sigma+rmur-2.d0*epsinf*ct*ct)*
     2            exp(-(delta+rlame)*t)/(4.d0*sigma*rlame)
      end do

  else
      write(*,*) "C2 term 2"
      rlame = dsqrt(-om0*om0-(gams/2.d0)-chi+delta*delta)
      do i=1,nt
          t = (i-1)*dt+.00001*dt
          c2(i) = (sigma+rmur-2.d0*epsinf*ct*ct)*expon(i)*
     2            sinh(rlame*t)/(2.d0*sigma*rlame)
      end do

  endif
! ---------------------------------------------------------------------
!   Shift and add C terms
! ---------------------------------------------------------------------
  if (c3case3.ne.0) then
```

```
    if (c1nc.eq.0) then
        write(*,*) 'shift c1'
        ta = c1;
        do i=1,(nt/shift)
            c1(i) = 0
        end do
        do i=(nt/shift+1),nt
            c1(i) = ta(i-nt/shift)
        end do
    endif
    if (c2nc.eq.0) then
        write(*,*) 'shift c2'
        ta = c2;
        do i=1,(nt/shift)
            c2(i) = 0
        end do
        do i=(nt/shift+1),nt
            c2(i) = ta(i-nt/shift)
        end do
    endif
endif

do i=1,nt
    c3(i) = c1(i)+c2(i)
end do
```

```
! ---------------------------------------------------------------------
!    Calculate F Term
! ---------------------------------------------------------------------
!    Case 1: om0^2>delta^2
! ---------------------------------------------------------------------
    if((om0*om0).gt.(delta*delta)) then
        write(*,*) "CASE 1"
        rlam1 = dsqrt(om0*om0-delta*delta)
        rlam5 = dsqrt(om0*om0+(rmur*b*b/(rkap*rkap))-delta*delta)
        do i=1,nt
            t = (i-1)*dt+.00001*dt
            call jbess(rlam1*t,0,i01(i),temp)
            call jbess(rlam1*t,1,i11(i),temp)
            call jbess(rlam5*t,1,i15h(i),temp)
            call jbess(rlam1*t,2,i21h(i),temp)
            call jbess(rlam5*t,2,i25h(i),temp)
            i11h(i) = i11(i)/t
            i15h(i) = i15h(i)/t
            i21h(i) = i21h(i)/t
            i25h(i) = i25h(i)/t
        end do
        call conv(i01,i15h,nt,dt,i0i1)
        call conv(i11h,i15h,nt,dt,i1i1)
        do i=1,nt
            t = (i-1)*dt+.00001*dt
            ft(i) = expon(i)*((epsinf*ct+rkap)*(-rlam1*rlam5*i1i1(i)-
     2            (b*b/epsinf)*rlam5*i0i1(i)+(b*b/epsinf)*rlam1*i11(i)+
```

```
     3           rlam1*rlam1*i21h(i)+rlam5*rlam5*i25h(i))+
     4           (b*b*b*b*ct/(epsinf*rlam1))*sin(rlam1*t))
        end do
! -----------------------------------------------------------------------
!    Case 2: om0^2<delta^2-rmur*b^2/rkap^2
! -----------------------------------------------------------------------
      elseif((om0*om0).lt.(delta*delta-(rmur*b*b/(rkap*rkap)))) then
        write(*,*) "CASE 2"
        rlam1 = dsqrt(delta*delta-om0*om0)
        rlam5 = dsqrt(delta*delta-om0*om0-rmur*b*b/(rkap*rkap))
        do i=1,nt
          t = (i-1)*dt+.00001*dt
          i01(i) = BESSI(0,rlam1*t)
          i11(i) = BESSI(1,rlam1*t)
          i15h(i) = BESSI(1,rlam5*t)
          i21h(i) = BESSI(2,rlam1*t)
          i25h(i) = BESSI(2,rlam5*t)
          i11h(i) = i11(i)/t
          i15h(i) = i15h(i)/t
          i21h(i) = i21h(i)/t
          i25h(i) = i25h(i)/t
        end do
        call conv(i01,i15h,nt,dt,i0i1)
        call conv(i11h,i15h,nt,dt,i1i1)
        do i=1,nt
          t = (i-1)*dt+.00001*dt
          ft(i) = expon(i)*((epsinf*ct+rkap)*(-rlam1*rlam5*i1i1(i)+
```

```
    2            (b*b/epsinf)*rlam5*i0i1(i)-(b*b/epsinf)*rlam1*i11(i)+
    3            rlam1*rlam1*i21h(i)+rlam5*rlam5*i25h(i))+
    4            (b*b*b*b*ct/(epsinf*rlam1))*sinh(rlam1*t))
      end do
! ----------------------------------------------------------------------
!    Case 3: om0^2>delta^2-rmur*b^2/rkap^2 and om0^2<delta^2
! ----------------------------------------------------------------------
    else
      write(*,*) "CASE 3"
      rlam1 = dsqrt(delta*delta-om0*om0)
      rlam5 = dsqrt(om0*om0+(rmur*b*b/(rkap*rkap))-delta*delta)
      do i=1,nt
          t = (i-1)*dt+.00001*dt
          i01(i) = BESSI(0,rlam1*t)
          i11(i) = BESSI(1,rlam1*t)
          call jbess(rlam5*t,1,i15h(i),temp)
          i21h(i) = BESSI(2,rlam1*t)
          call jbess(rlam5*t,2,i25h(i),temp)
          i11h(i) = i11(i)/t
          i15h(i) = i15h(i)/t
          i21h(i) = i21h(i)/t
          i25h(i) = i25h(i)/t
      end do
      call conv(i01,i15h,nt,dt,i0i1)
      call conv(i11h,i15h,nt,dt,i1i1)
      do i=1,nt
          t = (i-1)*dt+.00001*dt
```

```
            ft(i) = expon(i)*((epsinf*ct+rkap)*(rlam1*rlam5*i1i1(i)-
     2              (b*b/epsinf)*rlam5*i0i1(i)-(b*b/epsinf)*rlam1*i11(i)+
     3              rlam1*rlam1*i21h(i)+rlam5*rlam5*i25h(i))+
     4              (b*b*b*b*ct/(epsinf*rlam1))*sinh(rlam1*t))
         end do
      endif
! ----------------------------------------------------------------------
!     Time Domain Reflection Coefficient (gammat.dat)
! ----------------------------------------------------------------------
      if (c3case3.eq.0) then
         call conv(c3,ft,nt,dt,c3ft)
         do i=1,nt
            gammat(i) = f1*(c3ft(i)+YY*c3(i))
         end do
      else
         write(*,*) 'non-causal'
         do i=1,(nt/shift)
            ftshift(i) = 0
         end do
         do i=(nt/shift+1),nt
            ftshift(i) = ft(i-nt/shift)
         end do
         call conv(c3,ftshift,nt,dt,c3ft)
         do i=1,(nt/shift+1)
            c3shift(i) = 0
         end do
         do i=(nt/shift+2),nt
```

```
            c3shift(i) = c3(i-nt/shift-1)
        end do
        do i=1,nt
            gammat(i) = f1*(c3ft(i)+YY*c3shift(i))
        end do
    endif

    open(10,file='gammat.dat',status='unknown')
    do i=1,nt
        if (c3case3.eq.0) then
            write(10,*) ((i-1)*dt+.00001*dt)*1.d9,gammat(i)
        else
            write(10,*) ((i-1-2*nt/(shift))*dt+.00001*dt)*1.d9,gammat(i)
        endif
    end do
    close(10)

    end

! ----------------------------------------------------------------------
    subroutine conv(f,g,n,del,c)
!
!    performs convolution of waveforms f and g using
!     linear interpolation
!
    parameter (nsize=10000)
    parameter (c13=0.33333333333333333333d0,
```

```
     2             c16=0.16666666666666666666d0)
      implicit real*8 (a-h,o-z)
      real*8 f(nsize),g(nsize),c(nsize)
      c(1)=0.d0
      do k=2,n
        sum = 0.d0
        do l=1,k-1
          sum = sum + c13*f(l+1)*g(k-l)
     2                + c13*f(l)*g(k-l+1)
     3                + c16*f(l)*g(k-l)
     4                + c16*f(l+1)*g(k-l+1)
        end do
        c(k) = del*sum
      end do
      return
      end


! ----------------------------------------------------------------------
        FUNCTION BESSI(N,X)
!
!     This subroutine calculates the first kind modified Bessel function
!     of integer order N, for any REAL X. We use here the classical
!     recursion formula, when X > N. For X < N, the Miller's algorithm
!     is used to avoid overflows.
!     REFERENCE:
!     C.W.CLENSHAW, CHEBYSHEV SERIES FOR MATHEMATICAL FUNCTIONS,
```

```
!     MATHEMATICAL TABLES, VOL.5, 1962.
!
      PARAMETER (IACC = 40,BIGNO = 1.D10, BIGNI = 1.D-10)
      REAL *8 X,BESSI,BESSI0,BESSI1,TOX,BIM,BI,BIP
      IF (N.EQ.0) THEN
      BESSI = BESSI0(X)
      RETURN
      ENDIF
      IF (N.EQ.1) THEN
      BESSI = BESSI1(X)
      RETURN
      ENDIF
      IF(X.EQ.0.D0) THEN
      BESSI=0.D0
      RETURN
      ENDIF
      TOX = 2.D0/X
      BIP = 0.D0
      BI  = 1.D0
      BESSI = 0.D0
      M = 2*((N+INT(SQRT(FLOAT(IACC*N)))))
      DO 12 J = M,1,-1
      BIM = BIP+DFLOAT(J)*TOX*BI
      BIP = BI
      BI  = BIM
      IF (ABS(BI).GT.BIGNO) THEN
      BI  = BI*BIGNI
```

```
      BIP = BIP*BIGNI
      BESSI = BESSI*BIGNI
      ENDIF
      IF (J.EQ.N) BESSI = BIP
   12 CONTINUE
      BESSI = BESSI*BESSI0(X)/BI
      RETURN
      END
! ---------------------------------------------------------------------
! Auxiliary Bessel functions for N=0, N=1
      FUNCTION BESSI0(X)
      REAL *8 X,BESSI0,Y,P1,P2,P3,P4,P5,P6,P7,
     *    Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9,AX,BX
      DATA P1,P2,P3,P4,P5,P6,P7/1.D0,3.5156229D0,3.0899424D0,1.2067429D0
     *    ,0.2659732D0,0.360768D-1,0.45813D-2/
      DATA Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9/0.39894228D0,0.1328592D-1,
     *    0.225319D-2,-0.157565D-2,0.916281D-2,-0.2057706D-1,
     *    0.2635537D-1,-0.1647633D-1,0.392377D-2/
      IF(ABS(X).LT.3.75D0) THEN
      Y=(X/3.75D0)**2
      BESSI0=P1+Y*(P2+Y*(P3+Y*(P4+Y*(P5+Y*(P6+Y*P7)))))
      ELSE
      AX=ABS(X)
      Y=3.75D0/AX
      BX=EXP(AX)/SQRT(AX)
      AX=Q1+Y*(Q2+Y*(Q3+Y*(Q4+Y*(Q5+Y*(Q6+Y*(Q7+Y*(Q8+Y*Q9)))))))
      BESSI0=AX*BX
```

```
      ENDIF
      RETURN
      END
! -----------------------------------------------------------------------
      FUNCTION BESSI1(X)
      REAL *8 X,BESSI1,Y,P1,P2,P3,P4,P5,P6,P7,
     *    Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9,AX,BX
      DATA P1,P2,P3,P4,P5,P6,P7/0.5D0,0.87890594D0,0.51498869D0,
     *    0.15084934D0,0.2658733D-1,0.301532D-2,0.32411D-3/
      DATA Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9/0.39894228D0,-0.3988024D-1,
     *    -0.362018D-2,0.163801D-2,-0.1031555D-1,0.2282967D-1,
     *    -0.2895312D-1,0.1787654D-1,-0.420059D-2/
      IF(ABS(X).LT.3.75D0) THEN
      Y=(X/3.75D0)**2
      BESSI1=X*(P1+Y*(P2+Y*(P3+Y*(P4+Y*(P5+Y*(P6+Y*P7))))))
      ELSE
      AX=ABS(X)
      Y=3.75D0/AX
      BX=EXP(AX)/SQRT(AX)
      AX=Q1+Y*(Q2+Y*(Q3+Y*(Q4+Y*(Q5+Y*(Q6+Y*(Q7+Y*(Q8+Y*Q9)))))))
      BESSI1=AX*BX
      ENDIF
      RETURN
      END
! -----------------------------------------------------------------------
```

```
      subroutine jbess (x,n,bj,bjp)
c
c  calculates jn and jn' for n positive or negative
c
        implicit real*8 (a-h,o-z)
c
      if (n .eq. 0) then
          bj  = bessj(0,x)
          bjp = -bessj(1,x)
          return
        endif
c
        if (n .ge. 0) then
          m = n
        else
          m = -n
        endif
c
        if (x .eq. 0.d0) then
          bj  = 0.d0
          bjp1 = bessj(m+1,x)
          bjm1 = bessj(m-1,x)
          bjp  = (bjm1-bjp1)/2.d0
        else
          bj    = bessj(m,x)
          bj1   = bessj(m+1,x)
          bjp   = -bj1 + n*bj/x
```

```
      endif
c
      if (n .lt. 0) then
        bj    = bj*(-1)**n
        bjp   = bjp*(-1)**n
      endif
c
      return
      end
c
c
c
      FUNCTION BESSJ0(X)
      implicit real*8 (a-h,o-z)
      REAL*8 Y,P1,P2,P3,P4,P5,Q1,Q2,Q3,Q4,Q5,R1,R2,R3,R4,R5,R6,
     *    S1,S2,S3,S4,S5,S6
      DATA P1,P2,P3,P4,P5/1.D0,-.1098628627D-2,.2734510407D-4,
     *    -.2073370639D-5,.2093887211D-6/, Q1,Q2,Q3,Q4,Q5/-.1562499995D-
     *1,
     *    .1430488765D-3,-.6911147651D-5,.7621095161D-6,-.934945152D-7/
      DATA R1,R2,R3,R4,R5,R6/57568490574.D0,-13362590354.D0,651619640.7D
     *0,
     *    -11214424.18D0,77392.33017D0,-184.9052456D0/,
     *    S1,S2,S3,S4,S5,S6/57568490411.D0,1029532985.D0,
     *    9494680.718D0,59272.64853D0,267.8532712D0,1.D0/
      IF(ABS(X).LT.8.)THEN
        Y=X**2
```

```
        BESSJ0=(R1+Y*(R2+Y*(R3+Y*(R4+Y*(R5+Y*R6)))))
     *        /(S1+Y*(S2+Y*(S3+Y*(S4+Y*(S5+Y*S6)))))
      ELSE
        AX=ABS(X)
        Z=8./AX
        Y=Z**2
        XX=AX-.785398164
        BESSJ0=SQRT(.636619772/AX)*(COS(XX)*(P1+Y*(P2+Y*(P3+Y*(P4+Y
     *        *P5))))-Z*SIN(XX)*(Q1+Y*(Q2+Y*(Q3+Y*(Q4+Y*Q5)))))
      ENDIF
      RETURN
      END
c
c
c
      FUNCTION BESSJ1(X)
      implicit real*8 (a-h,o-z)
      REAL*8 Y,P1,P2,P3,P4,P5,Q1,Q2,Q3,Q4,Q5,R1,R2,R3,R4,R5,R6,
     *    S1,S2,S3,S4,S5,S6
      DATA R1,R2,R3,R4,R5,R6/72362614232.D0,-7895059235.D0,242396853.1D0
     *,
     *    -2972611.439D0,15704.48260D0,-30.16036606D0/,
     *    S1,S2,S3,S4,S5,S6/144725228442.D0,2300535178.D0,
     *    18583304.74D0,99447.43394D0,376.9991397D0,1.D0/
      DATA P1,P2,P3,P4,P5/1.D0,.183105D-2,-.3516396496D-4,.2457520174D-5
     *,
     *    -.240337019D-6/, Q1,Q2,Q3,Q4,Q5/.04687499995D0,-.2002690873D-3
```

```
     *,
     *    .8449199096D-5,-.88228987D-6,.105787412D-6/
      IF(ABS(X).LT.8.)THEN
        Y=X**2
        BESSJ1=X*(R1+Y*(R2+Y*(R3+Y*(R4+Y*(R5+Y*R6)))))
     *      /(S1+Y*(S2+Y*(S3+Y*(S4+Y*(S5+Y*S6)))))
      ELSE
        AX=ABS(X)
        Z=8./AX
        Y=Z**2
        XX=AX-2.356194491
        BESSJ1=SQRT(.636619772/AX)*(COS(XX)*(P1+Y*(P2+Y*(P3+Y*(P4+Y
     *      *P5))))-Z*SIN(XX)*(Q1+Y*(Q2+Y*(Q3+Y*(Q4+Y*Q5)))))
     *      *SIGN(1.,X)
      ENDIF
      RETURN
      END
c
c
c
      FUNCTION BESSJ(N,X)
      implicit real*8 (a-h,o-z)
      PARAMETER (IACC=40,BIGNO=1.d10,BIGNI=1.d-10)
      if (x .ge. 0.d0) then
      ff = 1.d0
    else
      ff = (-1.d0)**n
```

```
      end if
      xx = abs(x)
      if (n .eq. 0) then
          bessj = ff*bessj0(xx)
          return
        endif
        if (n .eq. 1) then
          bessj = ff*bessj1(xx)
          return
        endif
        TOX=2./Xx
        IF(Xx.GT.FLOAT(N))THEN
          BJM=BESSJ0(Xx)
          BJ=BESSJ1(Xx)
          DO 11 J=1,N-1
            BJP=J*TOX*BJ-BJM
            BJM=BJ
            BJ=BJP
11        CONTINUE
          BESSJ=ff*BJ
        ELSE
          M=2*((N+INT(SQRT(FLOAT(IACC*N))))/2)
          BESSJ=0.
          JSUM=0
          SUM=0.
          BJP=0.
          BJ=1.
```

```
        DO 12 J=M,1,-1
          BJM=J*TOX*BJ-BJP
          BJP=BJ
          BJ=BJM
          IF(ABS(BJ).GT.BIGNO)THEN
            BJ=BJ*BIGNI
            BJP=BJP*BIGNI
            BESSJ=BESSJ*BIGNI
            SUM=SUM*BIGNI
          ENDIF
          IF(JSUM.NE.0)SUM=SUM+BJ
          JSUM=1-JSUM
          IF(J.EQ.N)BESSJ=BJP
12      CONTINUE
        SUM=2.*SUM-BJ
        BESSJ=ff*BESSJ/SUM
      ENDIF
      RETURN
      END
```

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] R.M. Joseph, S.C. Hagness, and A. Taflove, "Direct time integration of Maxwell's equations in linear dispersive media with absorption for scattering and propagation of femtosecond electromagnetic pulses," Opt. Lett. **16**, 1412-1414 (1991).

[2] K.E. Oughstun and G.C. Sherman, "Propagation of electromagnetic pulses in a linear dispersive medium (the Lorentz medium)," J. Opt. Soc. Am. B **5**, 817-849 (1988).

[3] K.E. Oughstun and G.C. Sherman, "Uniform asymptotic description of electromagnetic pulse propagation in a linear dispersive medium with absorption (the Lorentz medium)," J. Opt. Soc. Am. A **6**, 1394-1420 (1989).

[4] E.L. Mokole and S.N. Samaddar, "Transmission and reflection of normally incident, pulsed electromagnetic plane waves upon a Lorentz half-space," J. Opt. Soc. Am. B **16**, 812-831 (1999).

[5] J.G. Blaschak and J. Franzen, "Precursor propagation in dispersive media from short-rise-time pulses at oblique incidence," J. Opt. Soc. Am. B **12**, 1501-1512 (1995).

[6] G.J. Stenholm, E.J. Rothwell, D.P. Nyquist, L.C. Kempel, and L.L. Frasch, "E-Pulse diagnostics of simpled layered materials," IEEE Trans. Antennas Propag. **51**, 3221-3227 (2003).

[7] H.-Y. Pao, S.L. Dvorak, and D.G. Dudley, "An accurate and efficient analysis for transient plane waves obliquely incident on a conductive half space (TM case)," IEEE Trans. Antennas Propag. **44**, 925-932 (1996).

[8] J.C. Oh, E. Rothwell, B.T. Perry, and M.J. Havrilla, "Natural Resonance Representation of the Transient Field Reflected by a Conductor-Backed Layer of Debye Material, J. Electromagnetic Waves and Applications **18**, 571-589 (2004).

[9] J.W. Suk and E.J. Rothwell, "Transient Analysis of TM-Plane Wave Reflection from a Layered Medium," J. Electromagnetic Waves and Applications **16**, 1195-1208 (2002).

[10] K.G. Gray, "The reflected impulse response of a Lorentz medium," Proc. IEEE **68**, 408-409 (1980).

[11] B.V. Stanic, D.R. Milanovic, and J.M. Cvetic, "Pulse reflection from a lossy Lorentz medium half-space (TM polarization)," J. Phys. D: Appl. Phys. **24**, 1245-1249 (1991).

[12] E.J. Rothwell and M.J. Cloud, *Electromagnetics* (CRC Press, Boca Raton, 2001).

[13] W.R. LePage, *Complex Variables and the Laplace Transform for Engineers* (Dover Press, New York, 1980).

[14] G.A. Campbell and R.M. Foster, *Fourier Integrals for Practical Applications, 2nd ed.* (D. Van Nostrand Co., New York, 1951).

[15] M. Abramowitz and I.S. Stegun, *Handbook of Mathematical Functions* (Dover, New York, 1965).

[16] L. Brillouin, *Wave Propagation and Group Velocity* (Academic Press, New York, 1960).