Brigham Young University
BYU ScholarsArchive

Theses and Dissertations

2006-12-01

Computationally Modeling the Effects of Surface Roughness on Soft X-Ray Multilayer Reflectors

Jedediah Edward Jensen Johnson
Brigham Young University - Provo

Follow this and additional works at: https://scholarsarchive.byu.edu/etd

 Part of the Astrophysics and Astronomy Commons, and the Physics Commons

BYU ScholarsArchive Citation
Johnson, Jedediah Edward Jensen, "Computationally Modeling the Effects of Surface Roughness on Soft X-Ray Multilayer Reflectors" (2006). *Theses and Dissertations*. 1075.
https://scholarsarchive.byu.edu/etd/1075

This Thesis is brought to you for free and open access by BYU ScholarsArchive. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of BYU ScholarsArchive. For more information, please contact scholarsarchive@byu.edu, ellen_amatangelo@byu.edu.

www.manaraa.com

COMPUTATIONALLY MODELING THE EFFECTS OF SURFACE ROUGHNESS ON SOFT X-RAY MULTILAYER REFLECTORS

by

Jedediah Edward Jensen Johnson

A thesis submitted to the faculty of

Brigham Young University

in partial fulfillment of the requirements for the degree of

Master of Science

Department of Physics and Astronomy

Brigham Young University

December 2006

www.manaraa.com

Copyright © 2006 Jedediah Edward Jensen Johnson

All Rights Reserved

www.manaraa.com

BRIGHAM YOUNG UNIVERSITY

GRADUATE COMMITTEE APPROVAL

of a thesis submitted by

Jedediah Edward Jensen Johnson

This thesis has been read by each member of the following graduate committee and by majority vote has been found to be satisfactory.

Date	R. Steven Turley, Chair
Date	David D. Allred
Date	Ross L. Spencer

www.manaraa.com

BRIGHAM YOUNG UNIVERSITY

As chair of the candidate's graduate committee, I have read the dissertation of Jedediah Edward Jensen Johnson in its final form and have found that (1) its format, citations, and bibliographical style are consistent and acceptable and fulfill university and department style requirements; (2) its illustrative materials including figures, tables, and charts are in place; and (3) the final manuscript is satisfactory to the graduate committee and is ready for submission to the university library.

Date

R. Steven Turley
Chair, Graduate Committee

Accepted for the Department

Scott Sommerfeldt, Chair
Department of Physics and Astronomy

Accepted for the College

Thomas W. Sederberg, Associate Dean
College of Mathematics and Physical Sciences

www.manaraa.com

ABSTRACT

COMPUTATIONALLY MODELING THE EFFECTS OF SURFACE ROUGHNESS ON SOFT X-RAY MULTILAYER REFLECTORS

Jedediah Edward Jensen Johnson

Department of Physics and Astronomy

Master of Science

Electromagnetic scattering from a rough two dimensional homogeneous scatterer was computationally modeled. The scatterer is intended to simulate reflection from a two interface multilayer. The rough scatterer was created from Gaussian random points centered about an ideal interface. The points were connected with a third order spline interpolant which accounts for correlation between neighboring surface atoms. The scalar electric field integral equation (EFIE) and magnetic field integral equation (MFIE) were solved using the Nystrom method to obtain the reflected intensity as a function of observation angle. Verification of the accuracy of the code was obtained by means of comparison with well-known analytic solutions and approximations. The predicted Nevot-Croce factor drop in reflectance was found to be in general agreement with the computed decrease in reflectance due to surface roughness. However, an angle dependent difference was also noticed, indicating the

www.manaraa.com

Nevot-Croce factor might need revision. The code is being modified to run on a supercomputing cluster where longer, more realistic surfaces can be analyzed to determine whether an improved roughness correction factor is needed.

www.manaraa.com

ACKNOWLEDGMENTS

This thesis could not have been a success without the help and support of many individuals. I would like to thank my advisor Dr. Turley for introducing me to the topic of computational electrodynamics and providing me with the tools necessary to complete this project. Without the countless hours of attention he directed towards my research problems and issues, I would have been lost from the outset. I also appreciate the support and availability of the other members of my graduate committee, Dr. Allred and Dr. Spencer. I am grateful to the BYU Department of Physics and Astronomy for the opportunity to represent the department and for funding my research. Last but not least, I thank my wife Marie, who gave me the flexibility to pursue my academic interests while gracefully serving as a wonderful wife, friend, and parent.

www.manaraa.com

المنارة للاستشارات
www.manaraa.com

Contents

www.manaraa.com

www.manaraa.com

www.manaraa.com

المنارة للاستشارات
www.manaraa.com

List of Figures

www.manaraa.com

www.manaraa.com

www.manaraa.com

www.manaraa.com

Chapter 1

Introduction

1.1 EUV Mirrors

The EUV (extreme ultraviolet) is a portion of the electromagnetic spectrum ranging from about 50-500 Å. While the science of designing mirrors has been well defined for decades in the visible light range, technological limitations and lack of interest have restricted progress in the EUV. Recently, however, potential applications of EUV mirrors have stimulated new research.

Computer chips are commonly fabricated through a process known as optical lithography. Even though improvements in technique have allowed chips to become smaller, optical lithography is limited by the resolution of large optical wavelengths. Shorter EUV wavelengths are capable of finer etching, leading to much faster chips. One of the main problems in harnessing this technology is developing optics with sufficiently high reflectance [1].

The biological community is also beginning to recognize applications for EUV radiation. Living material is generally more fragile and sensitive to high energy photons. The soft x-ray/extreme ultraviolet range seems to be promising in that

www.manaraa.com

it balances higher resolution with the delicateness of organic structures. Sample preparation is another advantage of soft x-ray imaging. Because carbon is opaque and water is relatively transparent from 24 to 44 Å, cells can be imaged in their native environment without dehydration or staining [2].

Astronomers have found uses for EUV optics. Members of the BYU XUV research group have participated in designing optics for the EUV instrument on the IMAGE (Imager for Magnetopause-to-Aurora Global Exploration) satellite [3]. It was designed to provide continuous images of the magnetosphere of the earth by viewing 304 Å light from singly ionized He atoms. Other distant astronomical phenomena could also be more effectively studied with better optics [4].

1.2 Multilayer Reflection Theory

1.2.1 Index of Refraction

Designing reflective surfaces in the EUV requires an understanding of the interaction between light and the material. When electromagnetic radiation is incident on a surface, it can either be reflected, transmitted, or absorbed. This is mathematically expressed as:

$$R + T + A = 1 \tag{1.1}$$

where R is the reflectance, T is the transmittance, and A is the absorption by the sample.

The complex index of refraction, a frequency-dependent property, can be used to predict how light will behave in a material. It is given by

$$\mathcal{N} = n + ik \tag{1.2}$$

where n is the real part of the index of refraction and k is the imaginary part, or

www.manaraa.com

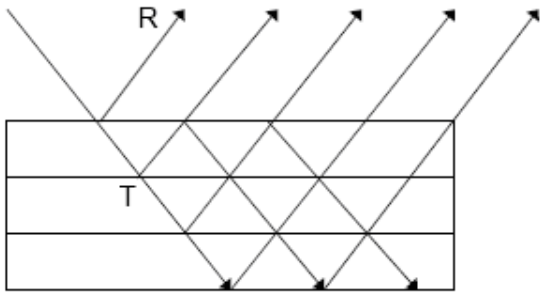

Figure 1.1 Reflection and transmission in a multilayer stack [5].

absorption coefficient. Both n and k alone are real numbers. The permittivity of the medium ϵ is related to the complex index of refraction by

$$\epsilon = \epsilon_0 \mathcal{N}^2 \tag{1.3}$$

where ϵ_0 is the familiar permittivity of free space [6]. Because values of n in the EUV are so close to unity, it is traditional to express them in terms of a related quantity as shown

$$\delta = 1 - n \tag{1.4}$$

The imaginary part of $\mathcal{N}$ is also renamed [7].

$$\beta = k \tag{1.5}$$

1.2.2 Polarization and Fresnel Coefficients

Incident radiation can be decomposed into two polarizations, known to physicists as s and p. In engineering literature, these are referred to as TM and TE, respectively. p polarized fields are parallel to the plane of incidence, while s fields are perpendicular. The plane of incidence is shown schematically in Figure 1.2.

These two polarizations are linearly independent, serving as the basis by which arbitrary incident fields are constructed by linear superposition. The convenient thing

www.manaraa.com

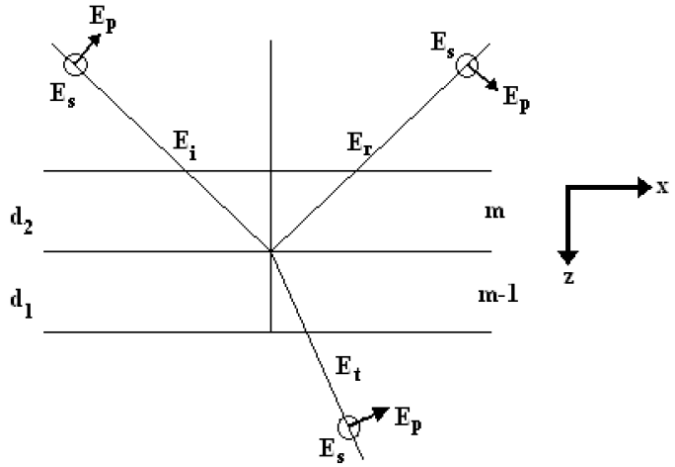

Figure 1.2 The geometry of s and p polarization on a multilayer stack. The plane of incidence coincides with the plane of the paper [8].

about working with s and p is that they can be treated as physically non-interacting entities. Although this breaks down at very high intensities as nonlinear effects begin to dominate, for our purposes this can be neglected. This allows us to solve each problem separately and then combine the solutions as we see fit.

Because of their different orientations, s polarization behaves differently than p at an interface. Maxwell's equations lead to basic EM boundary conditions, which can be applied to an infinite interface and used to derive the Fresnel coefficients:

$$f_{p,m} = \frac{\mathcal{N}_{m-1}^2 q_m - \mathcal{N}_m^2 q_{m-1}}{\mathcal{N}_{m-1}^2 q_m + \mathcal{N}_m^2 q_{m-1}} \tag{1.6}$$

$$f_{s,m} = \frac{q_m - q_{m-1}}{q_m + q_{m-1}} \tag{1.7}$$

where m is the mth interface in the stack, q is given by

$$q_m = \sqrt{\mathcal{N}^2 - \cos^2\theta_m} \tag{1.8}$$

and θ is the incident angle referenced from grazing as shown in figure 1.3 [9].

Once the Fresnel coefficients have been determined, the recursive Parratt formula is applied to calculate the overall reflection coefficient of the stack:

www.manaraa.com

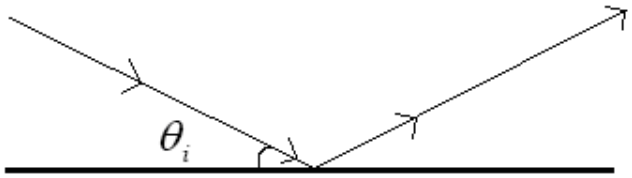

Figure 1.3 Grazing incidence reflection.

$$r_{p,m} = C_m^4 \frac{f_{p,m} + r_{p,m-1}}{1 + f_{p,m} r_{p,m-1}} \tag{1.9}$$

$$r_{s,m} = C_m^4 \frac{f_{s,m} + r_{s,m-1}}{1 + f_{s,m} r_{s,m-1}} \tag{1.10}$$

where

$$C_m = \exp\left(-\frac{i\pi d_m p_m}{\lambda_m}\right), \tag{1.11}$$

d_m is the layer thickness, and λ_m is the wavelength of light in the medium [10]. The actual reflectance of the stack is then found from

$$R_p = |r_p|^2 \tag{1.12}$$

$$R_s = |r_s|^2 \tag{1.13}$$

where r_p and r_s are the final coefficients after all the successive applications of the recursive Parratt formula.

1.3 Roughness

1.3.1 Types of Roughness

The Fresnel coefficients assume an infinitely long surface with a perfectly smooth, abrupt interface. Outside of textbooks, such surfaces do not exist. Even though

www.manaraa.com

they might appear smooth to the naked eye, there are significant imperfections at the microscopic level. This includes surface roughness, among other non-idealities such as density/material gradients, vacancies, and grain boundaries [11]. Both the wavelength of soft x-ray/extreme ultraviolet radiation and the magnitude of thin film surface roughness are on the order of nanometers. Consequently, roughness can have a considerable effect on thin film reflectance.

Surface roughness can be classified in a number of general ways. The RMS amplitude of a rough surface refers to root mean square average of the height, or elevation, of the surface. Jagged surfaces with a large RMS roughness compared with the wavelength of incident light will diffusely scatter the beam. This spreads the narrow specular reflection into a broader range of nonspecular angles, resulting in a diminished reflected intensity at the specular angle.

Surfaces characterized by RMS roughess on the order of, or smaller than a wavelength, can still have an appreciable effect on measured reflectance. Understanding this phenomenon requires looking at the problem in terms of wave mechanics. Perfectly specular reflection from an infinite smooth surface occurs because the reflected wave interferes with itself destructively at all other angles. If the surface deviates slightly from an otherwise smooth surface, there will be slight changes in the amplitude and phase of the reflected wave along the surface. This leads to some of the same effects observed from surfaces with a larger RMS roughness [8].

The spatial frequency of surface roughness can also potentially influence reflectance. Low frequency roughness can be compared to spread out, rolling hills. Intuitively, it seems like this type of roughness would tend to reflect different sections of the incident beam in different directions. Conversely, high frequency roughness is akin to a craggy, jagged ridge. Diffraction will dominate more as the period of the roughness becomes comparable with the wavelength.

www.manaraa.com

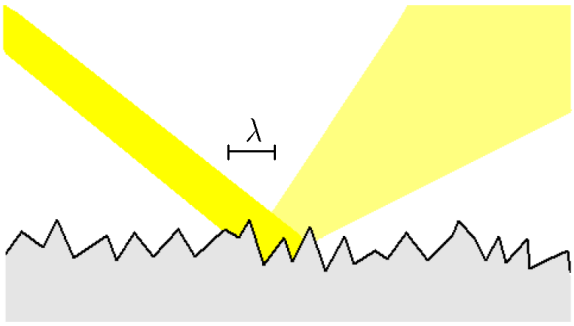

Figure 1.4 Diffuse scattering from a rough surface [8].

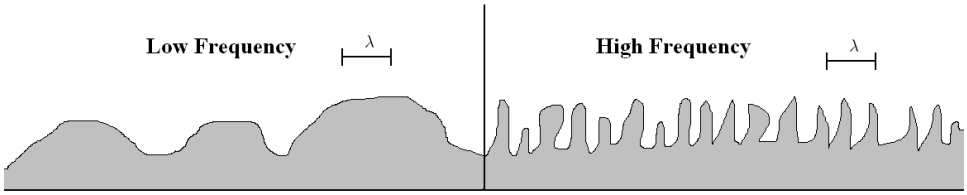

Figure 1.5 Low vs. high frequency roughness.

1.3.2 Correcting for Roughness

The prevailing method used to correct for roughness in thin film reflection modeling is the implementation of scalar correction factors [8] (primary source [12, 13]). The two most well-known of these are the Debye-Waller and Nevot-Croce factors [8] (primary source [14, 15]).

The Debye-Waller factor is derived by first assuming that the total reflectivity of an interface can be broken up into a Gaussian distribution centered on the ideal interface. This distribution's width is parameterized by σ, which is the RMS roughness of the surface. In the context in which it was first derived, this Gaussian comes from the superposition of lattice vibrations with varying amplitudes and phases. In this case, however, it is generalized to the reflectance at a rough interface. When a Fourier transform is applied to this distribution, we obtain an expression for the amplitude

www.manaraa.com

of the reflected field

$$r(q) = r_0 \exp(-\frac{q^2\sigma^2}{2}) \tag{1.14}$$

where r_0 is the reflectance off a smooth surface, and

$$q = \frac{4\pi n}{\lambda} \sin\theta \tag{1.15}$$

λ is the wavelength in vacuum, n is the index of refraction, and θ is the angle from grazing. Squaring (1.14) gives us the reflected intensity

$$R(q) = R_0 \exp(-q^2\sigma^2) \tag{1.16}$$

where R_0 is the smooth surface reflected intensity. The exponential is referred to as the Debye-Waller factor.

The Nevot-Croce factor is a modification of the Debye-Waller factor. The quantity q can be defined on either side of the interface by q_1 or q_2. Because the index of refraction and propagation angle change as light is refracted, the geometric average $q_1 q_2$ replaces the q^2 term in the Debye-Waller factor as shown.

$$R(q) = R_0 \exp(-q_1 q_2 \sigma^2) \tag{1.17}$$

Another method sometimes employed to adjust for roughness involves modeling the rough interface as a series of smooth, thin layers. The indices of refraction of these layers vary linearly between the two extremes given by the actual indices of the two materials. This is a valid approach because the indices of refraction are expected to scale as the density of the material [7]. The peaks and valleys of the rough surface correspond to an overall change in density as a function of depth of an effective layer. For computational purposes, this transition layer thus be broken up into a stack of small layers each with a slightly different index of refraction [8].

www.manaraa.com

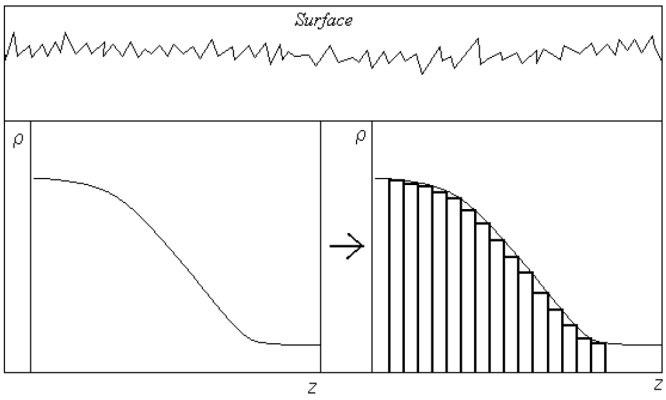

Figure 1.6 Approximating a rough surface as a series of thin homogeneous layers characterized by a varying density [8].

1.4 Previous Research

The BYU XUV optics research group has already made a substantial effort to experimentally determine the optical constants of several materials which show promise in the EUV. The art of designing precision optical equipment involves fabricating mirrors which reflect and transmit light within strict tolerances. To theoretically model such a mirror requires reliable, accurate optical constant data.

A fitting program was developed by Nicole Brimhall of the BYU group which takes measured reflectance and/or transmission data as inputs [8]. The program can then fit for the optical constants of the thin film. It displays the curve of best fit alongside the original data. Initially, the reflectance at 150 Å of a single thorium layer was fit assuming a perfect interface with no roughness. This fit was clearly deficient throughout large angular regions as shown in Figure 1.7.

Subsequently, the correction methods described in section 1.3.2 were incorporated into the fits with varying degrees of success. The Debye-Waller factor improved the fit at low angles, the Nevot-Croce factor improved the fit at both low and high angles, while the use of a 50 Å transition layer appeared to help at low and middle range angles. It was tentatively concluded that the best way to account for roughness was

www.manaraa.com

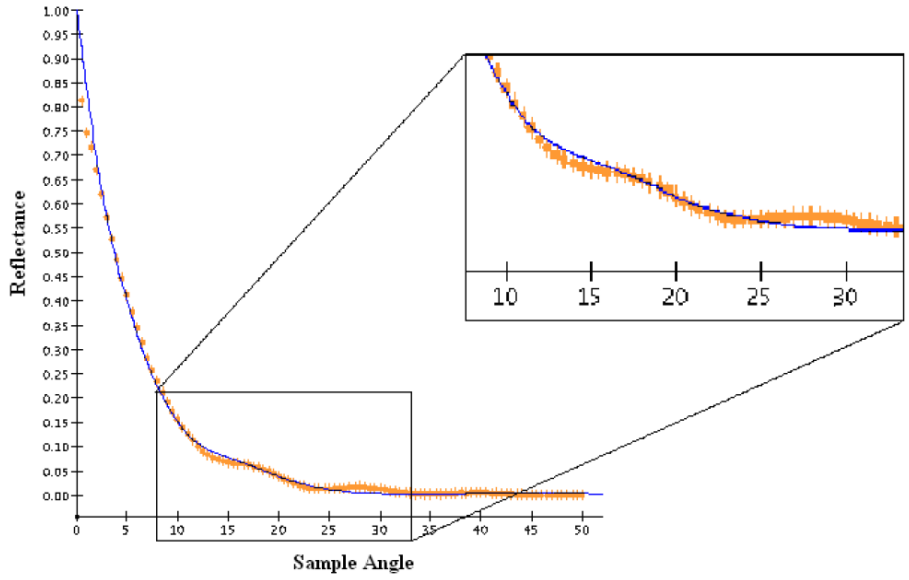

Figure 1.7 Reflectance vs. angle data along with fit of a single thorium layer on a silicon substrate at 150 Å. The fit is particularly bad through the middle angles $\theta = 10° - 30°$ [8].

a combination of the Nevot-Croce factor and the 50 Å transition layer [8].

Another study documented in Brimhall's thesis calls into question the validity of roughness measurements taken with the Atomic Force Microscope (AFM). The AFM works by essentially dragging a small tip over the sample surface, which maps out a profile of the three dimensional surface contour. Because the AFM tip has a finite width, it will not be able to accurately trace out narrow sections of the surface. Figure 1.12 compares a hypothetical surface generated by a Gaussian random distribution with the measured surface detected by a tip of non-zero width. In this example, the AFM measurement of 2.3 Å RMS roughness was less than 25% of the actual 10.3 Å value [8]. The extent to which our AFM tips distort the actual surface is unknown.

D. G. Stearns of Lawrence Livermore National Laboratory has written a number of papers looking into the effects of roughness on x-ray reflectance [16]. Similar to the Nevot-Croce and Debye-Waller factors, Stearns derives analytic expressions to predict the decrease in reflectance due to roughness. The rough or interdiffuse surface

www.manaraa.com

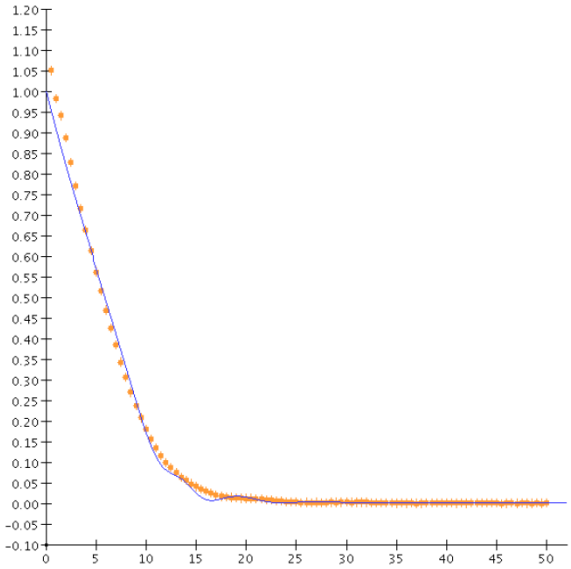

Figure 1.8 Corrected fit of the reflection of the single thorium layer using the Debye-Waller factor. The fit was improved only at the low angles [8].

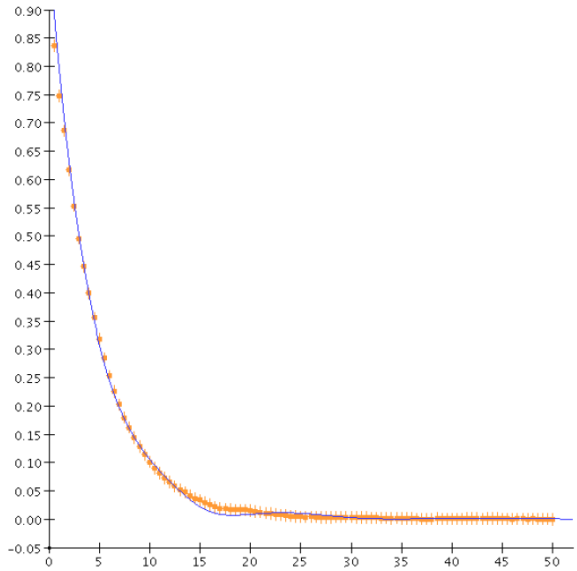

Figure 1.9 Corrected fit of the reflection of the single thorium layer using the Nevot-Croce factor. The fit was improved throughout both the low and high angles [8].

www.manaraa.com

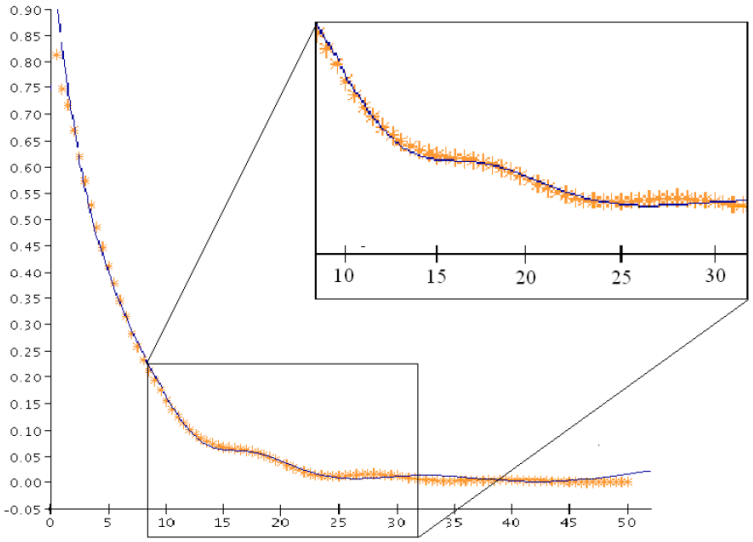

Figure 1.10 Fit of the reflection of the single thorium layer corrected with a 50 Å transition layer. This layer was approximated as 5 10 Å thick layers whose indices of refraction varied linearly. The fit was improved throughout both the low and middle angles [8].

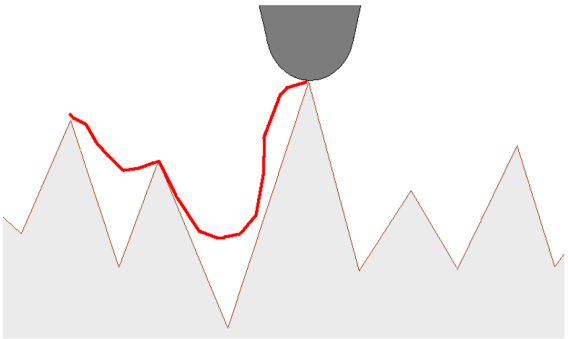

Figure 1.11 AFM tip dragged across a rough surface. The finite thickness of the AFM tip can produce an inaccurate profile of the surface [8].

www.manaraa.com

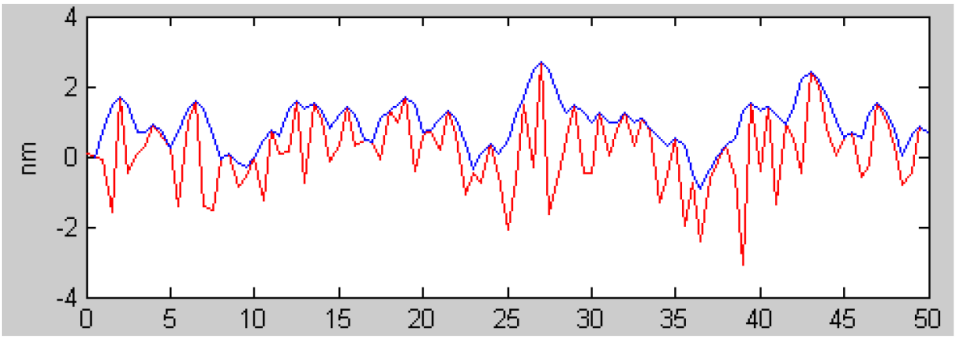

Figure 1.12 A modeled surface (red line) and the surface profile detected by the AFM with a tip of finite width (blue line). The actual RMS roughness of the surface is 10.3 Å , but the AFM reports an RMS roughness of 2.3 Å [8].

is expressed using an average density which is a function of surface depth. This is reminiscent of the last roughness model mentioned in Section 1.3.2.

There are a number of reasons why we wish to go beyond the research of Stearns. He relies on the approximations that the surface is only 'slightly rough,' that the scattering is 'weak,' and that no energy is coupled from nonspecular scattering back into specular scattering. This allows certain terms to be expanded and truncated to first order. A notable consequence of these assumptions is that the amount of power scattered into nonspecular angles is directly proportional to the power spectrum of the surface. These approximations do not always hold, however, in the experiments that the BYU group performs. Namely, we work with small x-ray wavelengths which can be on the order of the surface roughness. Reflectance measurements are taken at angles close to grazing where the scattering is strong. Additionally, in a rough multilayer stack, interference effects between rough layers could potentially return nonspecular radiation back into the specular peak.

www.manaraa.com

1.5 Project Scope and Applications

The purpose of this research was to computationally model the scattering of electromagnetic radiation from rough multilayer surfaces. This involved formulating and solving a series of integral equations in `Matlab` to determine the angular spread and intensity of the reflected wave. Rough surfaces were generated using a Gaussian random distribution centered on the ideal interface. Points were generated at specified intervals and were connected using third order spline interpolation. The problem was restricted to two dimensions, meaning there was translational symmetry along the invariant axis. Also, for computational efficiency, the dielectric multilayer was limited to two interfaces.

This numerical simulation was created to address the issues raised in Section 1.4. Because of the limitations of the AFM, we seek a better understanding of the parameters of our surfaces. These include, but are not limited to: spatial frequency, magnitude of RMS roughness, correlation between adjacent interfaces, and correlation between neighboring surface atoms (or regions). Moreover, we would like to determine how the AFM tip's finite width affects the quality of its measurement. By varying the physical characteristics of a theoretical rough surface until its reflectance signature mirrors that of the actual sample, we could recalibrate the AFM to give us correct data.

Another major goal of this research is to develop a new roughness correction method which more accurately fits our measured data. Through analysis of comprehensive data sets acquired from this simulation, we hope to come up with an improved empirical expression which can be easily applied. This could either be a modification of the existing standard or an entirely new approach.

www.manaraa.com

Chapter 2

Derivations

2.1 Helmholtz Equation

The well-known Maxwell's Equations in a material in the absence of sources are given by

$$\nabla \cdot \mathbf{D} = 0 \tag{2.1}$$

$$\nabla \cdot \mathbf{B} = 0 \tag{2.2}$$

$$\nabla \times \mathbf{H} = \frac{\partial \mathbf{D}}{\partial t} \tag{2.3}$$

$$\nabla \times \mathbf{E} + \frac{\partial \mathbf{B}}{\partial t} = 0 \tag{2.4}$$

If solutions with harmonic time dependence $e^{-i\omega t}$ are assumed, from which we can build arbitrary solutions by Fourier superposition, equations (2.3) and (2.4) can be written as

$$\nabla \times \mathbf{H} = -i\omega\epsilon\mathbf{E} \tag{2.5}$$

$$\nabla \times \mathbf{E} = i\omega\mu\mathbf{H} \tag{2.6}$$

www.manaraa.com

where we have noted that $\mathbf{D} = \epsilon\mathbf{E}$ and $\mathbf{H} = \frac{\mathbf{B}}{\mu}$. Taking the curl of both sides of these equations and applying a vector identity yields

$$\nabla \times (\nabla \times \mathbf{H}) = \nabla(\nabla \cdot \mathbf{H}) - \nabla^2\mathbf{H} = -i\omega\epsilon(\nabla \times \mathbf{E}) \tag{2.7}$$

$$\nabla \times (\nabla \times \mathbf{E}) = \nabla(\nabla \cdot \mathbf{E}) - \nabla^2\mathbf{E} = i\omega\mu(\nabla \times \mathbf{H}) \tag{2.8}$$

In general ϵ can be a tensor, but locally assuming an isotropic, homogeneous medium, it is constant and commutes with the derivative operators. At this point, we note that the divergence of both fields is zero, and substitute equations (2.5) and (2.6) into (2.7) and (2.8) respectively. With some rearranging, the result is the Helmholtz equation, shown only for the electric field:

$$(\nabla^2 + k^2)\mathbf{E} = 0 \tag{2.9}$$

where $k^2 = \mu\epsilon\omega^2$.

2.2 Green's Function

We now establish that the two dimensional free space Green's Function of the Helmholtz operator can be found by solving

$$(\nabla^2 + k^2)G(\mathbf{x}, \mathbf{x}') = -\delta^{(2)}(\mathbf{x} - \mathbf{x}') \tag{2.10}$$

For future reference, it is important to establish that the primed coordinates refer to the sources, while the unprimed coordinates refer to the point of observation. The polar coordinate equivalent of (2.10) is

$$\frac{d^2G}{dr^2} + \frac{1}{r}\frac{dG}{dr} + k^2G = -\frac{\delta(r)}{\pi r} \tag{2.11}$$

where $r = \sqrt{(x - x')^2 + (y - y')^2}$, $\mathbf{x} = (x, y)$, and $\mathbf{x}' = (x', y')$. The origin has been shifted to the singular point, and the angular derivative has been dropped because

www.manaraa.com

of symmetry. If $r \neq 0$, then equation (2.11) can be identified as Bessel's Equation of order zero, with solutions given by

$$G(x, y, x', y') = C_1 J_0(kr) + C_2 N_0(kr) \tag{2.12}$$

where $J_0(kr)$ and $N_0(kr)$ are the Bessel functions of the first and second kind, respectively. Alternatively, the solution can be represented as Hankel functions, which are linear combinations of the aforementioned Bessel functions.

$$G(\mathbf{x}, \mathbf{x}') = C_1 H_0^{(1)}(kr) + C_2 H_0^{(2)}(kr) \tag{2.13}$$

The outgoing scattered fields are required to satisfy the Sommerfeld radiation condition. Thus, the Green function is also under the same restriction. If the incident wave is an incoming plane wave, this requirement can be expressed as [17]

$$\lim_{r\to\infty} r^{\frac{1}{2}} \left(\frac{\partial}{\partial r} - ik \right) G(\mathbf{x}, \mathbf{x}') = 0 \tag{2.14}$$

From this condition, we recognize that $C_2 = 0$ in order to guarantee an outgoing scattered wave. The remaining expression is

$$G(\mathbf{x}, \mathbf{x}') = C_1 H_0^{(1)}(kr) \tag{2.15}$$

To find the normalization constant C_1, equation (2.15) is substituted back into (2.10). This is then integrated over a circle centered at the origin of radius ϵ as shown

$$\int_S (\nabla^2 + k^2) C_1 H_0^{(1)}(kr)\, d^2\mathbf{x} = -1 \tag{2.16}$$

$$\int_S C_1 \nabla \cdot \left(\nabla H_0^{(1)}(kr) \right) d^2\mathbf{x} + C_1 k^2 \int_S H_0^{(1)}(kr)\, d^2\mathbf{x} = -1 \tag{2.17}$$

At this point, we invoke the divergence theorem. Otherwise known as Gauss's Theorem, this allows us to write

$$\int_S \nabla \cdot \mathbf{u}\, d^2\mathbf{x} = \int_C \mathbf{u} \cdot \hat{\mathbf{n}}\, dl, \tag{2.18}$$

www.manaraa.com

where $\hat{\mathbf{n}}$ is the outward normal unit vector. Applying this to the first integral in equation (2.17) and continuing leaves us with

$$C_1 \int_C \nabla H_0^{(1)}(kr) \cdot \hat{\mathbf{r}}\, dl + C_1 k^2 \int_S H_0^{(1)}(kr)\, d^2\mathbf{x} = -1 \tag{2.19}$$

$$C_1 \int_0^{2\pi} \frac{d}{dr}\left(H_0^{(1)}(kr)\right)\Big|_{r=\epsilon} \epsilon\, d\theta + C_1 k^2 \int_0^{2\pi}\int_0^{\epsilon} H_0^{(1)}(kr) r\, dr\, d\theta = -1 \tag{2.20}$$

$$-2\pi\epsilon k C_1 H_1^{(1)}(k\epsilon) + 2\pi C_1 k^2 \int_0^{\epsilon} H_0^{(1)}(kr) r\, dr = -1 \tag{2.21}$$

Now, let's shrink the circle so that its area goes to zero.

$$\lim_{\epsilon\to 0}\left\{-2\pi\epsilon k C_1 H_1^{(1)}(k\epsilon) + 2\pi C_1 k^2 \int_0^{\epsilon} H_0^{(1)}(kr) r\, dr\right\} = -1 \tag{2.22}$$

If we look at the behavior of $H_1^{(1)}(k\epsilon)$ as $\epsilon \to 0$, we see that the leading order term goes like

$$H_1^{(1)}(k\epsilon) \sim -\frac{2i}{k\pi\epsilon} \tag{2.23}$$

Additionally, if we look at the behavior of the integrand in the second integral as $\epsilon \to 0$, we see that

$$\lim_{\epsilon\to 0}\left\{\epsilon H_0^{(1)}(k\epsilon)\right\} = 0 \tag{2.24}$$

The integral of a term which is going to zero evaluated from zero to something approaching zero will equal zero. Therefore, we can remove the second term and solve for C_1 [18].

$$4iC_1 = -1 \tag{2.25}$$

$$C_1 = \frac{i}{4} \tag{2.26}$$

The final Green function is

$$G(\mathbf{x}, \mathbf{x}') = \frac{i}{4} H_0^{(1)}(k|\mathbf{x} - \mathbf{x}'|) \tag{2.27}$$

www.manaraa.com

2.3 Source Field Relations

Maxwell's Equations can be re-written in the suggestive form

$$\nabla \cdot (\epsilon_0 \mathbf{E}) = \rho_e \tag{2.28}$$

$$\nabla \cdot (\mu_0 \mathbf{H}) = \rho_m \tag{2.29}$$

$$\nabla \times \mathbf{E} = i\omega\mu_0 \mathbf{H} - \mathbf{K} \tag{2.30}$$

$$\nabla \times \mathbf{H} = -i\omega\epsilon_0 \mathbf{E} + \mathbf{J} \tag{2.31}$$

where

$$\rho_e = \epsilon_0 \epsilon_r \mathbf{E} \cdot \nabla \left(\frac{1}{\epsilon_r} \right) \tag{2.32}$$

$$\rho_m = \mu_0 \mu_r \mathbf{H} \cdot \nabla \left(\frac{1}{\mu_r} \right) \tag{2.33}$$

$$\mathbf{K} = -i\omega\mu_0(\mu_r - 1)\mathbf{H} \tag{2.34}$$

$$\mathbf{J} = -i\omega\epsilon_0(\epsilon_r - 1)\mathbf{E} \tag{2.35}$$

and

$$\epsilon_r = \frac{\epsilon}{\epsilon_0} \tag{2.36}$$

$$\mu_r = \frac{\mu}{\mu_0} \tag{2.37}$$

It now appears as though the equations describe homogeneous vacuum, except for the addition of source terms ρ_e, ρ_m, K, and J. These new terms carry the information about the particular material we are working in. The ρ terms containing the gradient operator may seem rather mysterious, but they have been constructed to accurately account for both volume polarization currents and surface currents. The remainder of this chapter is based on derivations presented in reference [19].

The scattering problem can be attacked by breaking up the total field present into the incident and scattered field. Working only with the electric field, this is expressed

www.manaraa.com

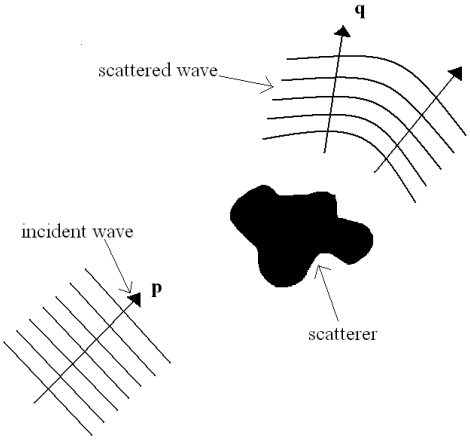

Figure 2.1 A depiction of an incident and scattered wave. **p** is the incident wavevector, and **q** is the scattered wavevector.

as

$$\mathbf{E} = \mathbf{E}^{inc} + \mathbf{E}^{s} \tag{2.38}$$

where it is assumed that the fields satisfy the Helmholtz equation. The equation for the scattered field is derived using the following identity for the vector Laplacian:

$$\nabla^2 \mathbf{E}^s = \nabla(\nabla \cdot \mathbf{E}^s) - \nabla \times \nabla \times \mathbf{E}^s \tag{2.39}$$

Taking the divergence of (2.31), and noting that the divergence of a curl is always zero, we have the continuity equation

$$\cancelto{0}{\nabla \cdot (\nabla \times \mathbf{H}^s)} = \nabla \cdot (-i\omega\epsilon_0 \mathbf{E}^s) + \nabla \cdot \mathbf{J} \tag{2.40}$$

$$\nabla \cdot \mathbf{E}^s = \frac{\nabla \cdot \mathbf{J}}{i\omega\epsilon_0} \tag{2.41}$$

Substituting both (2.41) and (2.30) into (2.39) gives

$$\nabla^2 \mathbf{E}^s = \nabla\left(\frac{\nabla \cdot \mathbf{J}}{i\omega\epsilon_0}\right) - \nabla \times (i\omega\mu_0 \mathbf{H}^s - \mathbf{K}) \tag{2.42}$$

$$= \nabla\left(\frac{\nabla \cdot \mathbf{J}}{i\omega\epsilon_0}\right) - i\omega\mu_0 (\nabla \times \mathbf{H}^s) + \nabla \times \mathbf{K} \tag{2.43}$$

www.manaraa.com

Now, we substitute (2.31) into (2.43) to produce

$$\nabla^2\mathbf{E}^s = \nabla\left(\frac{\nabla\cdot\mathbf{J}}{i\omega\epsilon_0}\right) - i\omega\mu_0\left(-i\omega\epsilon_0\mathbf{E}^s + \mathbf{J}\right) + \nabla\times\mathbf{K} \tag{2.44}$$

$$\nabla^2\mathbf{E}^s = \nabla\left(\frac{\nabla\cdot\mathbf{J}}{i\omega\epsilon_0}\right) - k^2\mathbf{E}^s - i\omega\mu_0\mathbf{J} + \nabla\times\mathbf{K} \tag{2.45}$$

$$(\nabla^2 + k^2)\mathbf{E}^s = \nabla\left(\frac{\nabla\cdot\mathbf{J}}{i\omega\epsilon_0}\right) - i\omega\mu_0\mathbf{J} + \nabla\times\mathbf{K} \tag{2.46}$$

This equation looks like the familiar Helmholtz equation with the addition of several source terms on the right hand side. In order to solve this equation for E^s, we first define the new quantities

$$\mathbf{A}(\mathbf{x}) = \int_S \mathbf{J}(\mathbf{x}')G(\mathbf{x},\mathbf{x}')\,d^2\mathbf{x}' \tag{2.47}$$

$$\mathbf{F}(\mathbf{x}) = \int_S \mathbf{K}(\mathbf{x}')G(\mathbf{x},\mathbf{x}')\,d^2\mathbf{x}' \tag{2.48}$$

The Green's function is the solution to (2.10). Right multiplying by $\mathbf{J}(\mathbf{x}')$ renders

$$(\nabla^2 + k^2)G(\mathbf{x},\mathbf{x}')\mathbf{J}(\mathbf{x}') = -\delta^{(2)}(\mathbf{x}-\mathbf{x}')\mathbf{J}(\mathbf{x}') \tag{2.49}$$

We then integrate with respect to the primed coordinates and are left with

$$\int_S (\nabla^2 + k^2)G(\mathbf{x},\mathbf{x}')\mathbf{J}(\mathbf{x}')\,d^2\mathbf{x}' = \int_S -\delta^{(2)}(\mathbf{x}-\mathbf{x}')\mathbf{J}(\mathbf{x}')\,d^2\mathbf{x}' \tag{2.50}$$

$$(\nabla^2 + k^2)\int_S G(\mathbf{x},\mathbf{x}')\mathbf{J}(\mathbf{x}')\,d^2\mathbf{x}' = -\mathbf{J}(\mathbf{x}) \tag{2.51}$$

$$(\nabla^2 + k^2)\mathbf{A}(\mathbf{x}) = -\mathbf{J}(\mathbf{x}) \tag{2.52}$$

Similarly, it can be shown that

$$(\nabla^2 + k^2)\mathbf{F}(\mathbf{x}) = -\mathbf{K}(\mathbf{x}) \tag{2.53}$$

Using these relationships, and the fact that unprimed operators commute with integrals over primed coordinates, it is clear that

$$\mathbf{E}^s = -\frac{\nabla\left(\nabla\cdot\mathbf{A}\right) + k^2\mathbf{A}}{i\omega\epsilon_0} - \nabla\times\mathbf{F} \tag{2.54}$$

www.manaraa.com

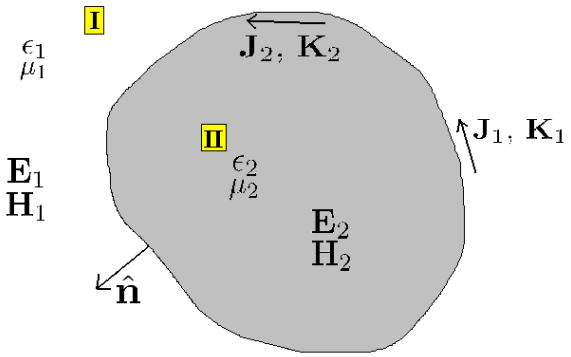

Figure 2.2 An illustration of the parameters of the two geometries used to derive the surface equivalence principle.

Carrying out a similar procedure for $\mathbf{H}$ yields

$$\mathbf{H}^s = -\frac{\nabla\left(\nabla\cdot\mathbf{F}\right)+k^2\mathbf{F}}{i\omega\mu_0}+\nabla\times\mathbf{A} \tag{2.55}$$

2.4 Surface Equivalence Principle

The surface equivalence principle allows us to define the mathematical sources J and K on the boundaries between regions of space. If the composition of these regions is homogeneous, then these surface sources are sufficient to describe the scattering problem. Volume sources are only necessary in the presence of inhomogeneities contained within the boundaries.

Although there is a lengthy derivation associated with the surface equivalence principle, its results are very powerful. Figure 2.2 will help illustrate the concepts. There are two regions of space, I and II, both homogenous materials characterized by ϵ_1, μ_1 and ϵ_2, μ_2 respectively. The total fields in each domain are $\mathbf{E}_1$,$\mathbf{H}_1$ and $\mathbf{E}_2$,$\mathbf{H}_2$.

First, we consider the exterior region I. According to the surface equivalence prin-

www.manaraa.com

ciple, we are able to define two mathematical surface currents $\mathbf{J}_1$ and $\mathbf{K}_1$ as

$$\mathbf{J}_1 = \hat{\mathbf{n}} \times \mathbf{H}_1 \tag{2.56}$$

$$\mathbf{K}_1 = \mathbf{E}_1 \times \hat{\mathbf{n}} \tag{2.57}$$

where $\hat{\mathbf{n}}$ is the outward normal from the surface. It points from region II into region I. These mathematical sources radiate to produce the scattered fields $\mathbf{E}^s$ and $\mathbf{H}^s$. In region I, they combine with the incident fields $\mathbf{E}^{inc}$ and $\mathbf{H}^{inc}$ to reproduce the total fields $\mathbf{E}$ and $\mathbf{H}$ as shown in (2.38). In region II however, it turns out that the fields always combine in such a way to produce null fields. This allows us to replace the material in region II with that of region I without changing the result, effectively transforming the problem into one infinite region of material ϵ_1 and μ_1. Obviously, the fields in region II will not always have zero magnitude. This process only leads us to the correct fields in region I.

To find the correct fields in region II, the same surface equivalence technique is applied. Sources $\mathbf{J}_2$ and $\mathbf{K}_2$ are defined on the boundary as

$$\mathbf{J}_2 = (-\hat{\mathbf{n}}) \times \mathbf{H}_2 \tag{2.58}$$

$$\mathbf{K}_2 = \mathbf{E}_2 \times (-\hat{\mathbf{n}}) \tag{2.59}$$

where $\hat{\mathbf{n}}$ still points from region II into region I. This time, the scattered fields produced by the radiating sources $\mathbf{J}_2$ and $\mathbf{K}_2$ are the total fields. The incident field is defined only in region I and is left out of region II. These sources will produce the correct fields in region II with null fields in region I, allowing us to replace the entire problem with a material characterized by ϵ_2 and μ_2.

Because $\mathbf{J}$ and $\mathbf{K}$ have been defined as surface currents, $\mathbf{A}$ and $\mathbf{F}$ become

$$\mathbf{A}(\mathbf{x}) = \int_C \mathbf{J}(\mathbf{x}')G(\mathbf{x}, \mathbf{x}')\, d\mathbf{x}' \tag{2.60}$$

$$\mathbf{F}(\mathbf{x}) = \int_C \mathbf{K}(\mathbf{x}')G(\mathbf{x}, \mathbf{x}')\, d\mathbf{x}' \tag{2.61}$$

www.manaraa.com

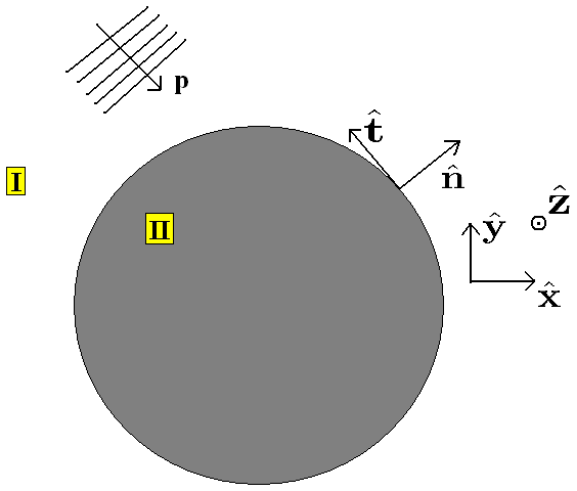

Figure 2.3 An illustration of the orientation of the scattering surfaces relative to the coordinate axes.

where C denotes the integral over the surface contour.

2.5 Electric Field Integral Equation (EFIE)

The overall scattering problem is now ready to be setup and solved. Substituting (2.38) into (2.54) leaves us with

$$\mathbf{E}^{inc} = \mathbf{E} + \frac{\nabla\left(\nabla \cdot \mathbf{A}\right) + k^2\mathbf{A}}{i\omega\epsilon_0} + \nabla \times \mathbf{F} \tag{2.62}$$

The electric field integral equation is most conducive to s polarized waves. This means that the fields are polarized along the z-axis in our geometry. With this in mind, (2.62) becomes

$$E_z^{inc} = E_z + \frac{\nabla\underbrace{\left(\nabla \cdot \mathbf{A}\right)}_{0} + k^2 A_z}{i\omega\epsilon_0} + \left[\nabla \times \mathbf{F}\right]_z \tag{2.63}$$

The divergence term goes to zero because the sources are invariant in the z direction. Looking at (2.57), we see that if $\mathbf{E}$ only has a z component, then $\mathbf{K}$ will only have a component tangent to the surface. $\hat{\mathbf{n}}$, $\mathbf{K}_t$, and $\mathbf{E}_z$ (subscript t indicating tangential

www.manaraa.com

component) are then all mutually perpendicular, and we can note for the outside region

$$E_z = K_t \tag{2.64}$$

The inside region will pick up a minus sign. Using this result and continuing,

$$E_z^{inc} = K_t - ik\eta A_z + \left[\frac{\partial F_y}{\partial x} - \frac{\partial F_x}{\partial y}\right] \tag{2.65}$$

where $\eta = \sqrt{\frac{\mu}{\epsilon}}$. This equation is known as the electric field integral equation, specialized to our two dimensional geometry with s polarized light.

To solve the scattering problem of a closed surface surrounded by another material, we first invoke the surface equivalence principle. Using the electric field integral equation (2.65), we write down the two equations representing both the outside and the inside problem.

$$E_z^{inc} = K_{1t} - ik_1\eta_1 A_{1z} + \left[\frac{\partial F_{1y}}{\partial x} - \frac{\partial F_{1x}}{\partial y}\right] \tag{2.66}$$

$$0 = -K_{2t} - ik_2\eta_2 A_{2z} + \left[\frac{\partial F_{2y}}{\partial x} - \frac{\partial F_{2x}}{\partial y}\right] \tag{2.67}$$

The subscripts 1 and 2 represent that the terms contain either surface currents or other parameters as defined in regions I and II.

At this point, we have two equations and four unknowns. In order to solve the problem, we must have equal numbers of equations and unknowns. Applying boundary conditions addresses this issue. The electromagnetic boundary conditions on the tangential components of **E** and **H** are

$$\hat{\mathbf{n}} \times (\mathbf{E}_1 - \mathbf{E}_2) = 0 \tag{2.68}$$

$$\hat{\mathbf{n}} \times (\mathbf{H}_1 - \mathbf{H}_2) = 0 \tag{2.69}$$

Looking only at the E field, equations (2.57) and (2.59) can be added together to

www.manaraa.com

produce

$$\mathbf{K}_1 + \mathbf{K}_2 = \mathbf{E}_1 \times \hat{\mathbf{n}} + \mathbf{E}_2 \times (-\hat{\mathbf{n}}) \tag{2.70}$$

$$\mathbf{K}_1 + \mathbf{K}_2 = (-\hat{\mathbf{n}}) \times \mathbf{E}_1 + \hat{\mathbf{n}} \times \mathbf{E}_2 \tag{2.71}$$

$$\mathbf{K}_1 + \mathbf{K}_2 = \hat{\mathbf{n}} \times (\mathbf{E}_1 - \mathbf{E}_2) \quad \cancelto{0}{} \tag{2.72}$$

$$\mathbf{K}_1 = -\mathbf{K}_2 \tag{2.73}$$

Similarly, it can be shown that

$$\mathbf{J}_1 = -\mathbf{J}_2 \tag{2.74}$$

Using this to eliminate the variables from regions 2, (2.66) and (2.67) become

$$E_z^{inc} = K_{1t} - ik_1\eta_1 A_z^{(1)} + \left[\frac{\partial F_y^{(1)}}{\partial x} - \frac{\partial F_x^{(1)}}{\partial y}\right] \tag{2.75}$$

$$0 = -K_{1t} - ik_2\eta_2 A_z^{(2)} + \left[\frac{\partial F_y^{(2)}}{\partial x} - \frac{\partial F_x^{(2)}}{\partial y}\right] \tag{2.76}$$

where now the notation has been changed so that everything is in terms of $\mathbf{K}_1$ and $\mathbf{J}_1$:

$$\mathbf{A}^{(w)}(\mathbf{x}) = \int_C \mathbf{J}_1(\mathbf{x}')\frac{i}{4}H_0^{(1)}(k_w|\mathbf{x} - \mathbf{x}'|)\,d\mathbf{x}' \tag{2.77}$$

$$\mathbf{F}^{(w)}(\mathbf{x}) = \int_C \mathbf{K}_1(\mathbf{x}')\frac{i}{4}H_0^{(1)}(k_w|\mathbf{x} - \mathbf{x}'|)\,d\mathbf{x}' \tag{2.78}$$

2.6 Magnetic Field Integral Equation (MFIE)

The fields describing the behavior of the magnetic field are similar to those of the electric field. In fact, a similar procedure to that outlined in the previous sections can be used to derive an equation for magnetic field. Instead of reworking all these

www.manaraa.com

details, however, the so-called duality relationships can be applied.

$$\mathbf{E} \rightarrow \mathbf{H} \tag{2.79}$$

$$\mathbf{H} \rightarrow -\mathbf{E} \tag{2.80}$$

$$\mathbf{J} \rightarrow \mathbf{K} \tag{2.81}$$

$$\mathbf{K} \rightarrow -\mathbf{J} \tag{2.82}$$

$$\rho_e \rightarrow \rho_m \tag{2.83}$$

$$\rho_m \rightarrow -\rho_e \tag{2.84}$$

$$\epsilon \rightarrow \mu \tag{2.85}$$

$$\mu \rightarrow \epsilon \tag{2.86}$$

$$\mathbf{A} \rightarrow \mathbf{F} \tag{2.87}$$

$$\mathbf{F} \rightarrow -\mathbf{A} \tag{2.88}$$

If the quantities on the left are substituted for those on the right, the resulting relationships are valid. Performing these substitutions on (2.75) and (2.76) gives us the desired version of the magnetic field integral equation.

$$H_z^{inc} = -J_{1t} - i\frac{k_1}{\eta_1}F_z^{(1)} - \left[\frac{\partial A_y^{(1)}}{\partial x} - \frac{\partial A_x^{(1)}}{\partial y}\right] \tag{2.89}$$

$$0 = J_{1t} - i\frac{k_2}{\eta_2}F_z^{(2)} - \left[\frac{\partial A_y^{(2)}}{\partial x} - \frac{\partial A_x^{(2)}}{\partial y}\right] \tag{2.90}$$

The MFIE relates the incident magnetic field to the induced sources. Since the magnetic field is polarized along the z-axis, the accompanying electric field in the propagating wave will be polarized perpendicular to the z-axis. This corresponds to p polarization. So, using (2.89) and (2.90) to solve for the scattered magnetic field is equivalent to solving for the scattered p polarized electric field, up to a proportionally factor relating the fields.

www.manaraa.com

www.manaraa.com

Chapter 3

Problem Setup and Solution Techniques

3.1 Numerical Quadrature

3.1.1 Basics

Definite integrals can be approximated using a discretization scheme known as numerical quadrature. The basic idea is that the integral can be represented as

$$\int_a^b f(x)dx = \sum_i c_i f(x_i) \tag{3.1}$$

where x_i are a series of x values at which the function is to be evaluated. Almost always, x_i are contained within the limits of integration. c_i are the coefficients of the series, which depend on the type and order of the particular quadrature rule.

The simplest rule is the so called rectangle or midpoint rule. The integral is essentially approximated as the sum of rectangularly shaped boxes which are arranged to fit under the curve of the function. In mathematical language, this can be expressed

www.manaraa.com

as

$$\int_a^b f(x)dx = \sum_i hf(x_i) \tag{3.2}$$

where h is the distance between the evenly spaced x values.

The rectangle rule uses a zero order quadrature rule. In other words, the function is approximated as a constant over the range of each cell. In order to improve accuracy without adding more discretization points, it is necessary to use a higher order rule which more closely mirrors the function across each cell. The trapezoid rule and Simpson's rule are two well known higher order rules.

3.1.2 Regular Integrals

In our case, solving for the mathematical surface currents poses a challenge. We would prefer to employ a quadrature rule of higher order than the simple zero order midpoint rule. The reason for this is higher order rules will converge more quickly to the exact solution for a given number of quadrature points. However, high order quadrature rules necessitate the use of a greater number of quadrature points. Also, solving for and implementing such a complex rule is not ideal. This aside, because there will be many current oscillations over the surface, the advantages of a high order rule are attractive.

The solution to this issue comes in the use of patches. The surface is divided up into sections which are separately integrated using a third order quadrature rule. This rule is exact for all polynomials third order and smaller. A third order rule is still relatively easy to work with, yet guarantees better convergence.

To integrate one of these patches, four equally spaced points are identified at $\frac{l}{8}$, $\frac{3l}{8}$, $\frac{5l}{8}$, and $\frac{7l}{8}$, where l is the length of the patch. The distance between the points is then $h = \frac{l}{4}$. In general, it is not necessary for the patch to be perfectly flat, as l could be a parametric parameter. If even spaced values in a given coordinate are used

www.manaraa.com

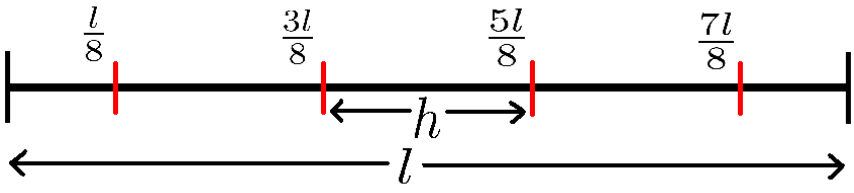

Figure 3.1 The layout of a typical patch.

as the quadrature points, a Jacobian factor can be introduced to compensate for the particular contour of the surface. This idea will be explicitly related to our particular problem later in the chapter. We must first find the values of the four quadrature weights c_i by solving a system of four equations. The rule must be able to integrate a function of each order exactly, and all the equations will have the form

$$c_1 f\left(\frac{l}{8}\right) + c_2 f\left(\frac{3l}{8}\right) + c_3 f\left(\frac{5l}{8}\right) + c_4 f\left(\frac{7l}{8}\right) = A \tag{3.3}$$

where A is the area of the given order's function over the range of the patch. The zero order equation will represent a line at $f(x) = 1$:

$$c_1 + c_2 + c_3 + c_4 = l \tag{3.4}$$

Similarly, the other three equations will look like:

$$c_1\frac{l}{8} + c_2\frac{3l}{8} + c_3\frac{5l}{8} + c_4\frac{7l}{8} = \frac{l^2}{2} \quad ; \quad f(x) = x \tag{3.5}$$

$$c_1\frac{l^2}{64} + c_2\frac{9l^2}{64} + c_3\frac{25l^2}{64} + c_4\frac{49l^2}{64} = \frac{l^3}{3} \quad ; \quad f(x) = x^2 \tag{3.6}$$

$$c_1\frac{l^3}{512} + c_2\frac{27l^3}{512} + c_3\frac{125l^3}{512} + c_4\frac{343l^3}{512} = \frac{l^4}{4} \quad ; \quad f(x) = x^3 \tag{3.7}$$

www.manaraa.com

When this linear system is solved, the quadrature weights turn out to be:

$$c_1 = \frac{13l}{48} = \frac{13h}{12} \tag{3.8}$$

$$c_2 = \frac{11l}{48} = \frac{11h}{12} \tag{3.9}$$

$$c_3 = \frac{11l}{48} = \frac{11h}{12} \tag{3.10}$$

$$c_4 = \frac{13l}{48} = \frac{13h}{12} \tag{3.11}$$

Now that the quadrature rule has been developed, an arbitrary function can be approximately integrated if the value of the function is known at the quadrature points. This will work for a general class of functions, provided the function is well defined at the quadrature point.

3.1.3 Singular Integrals

If the patch contains a quadrature point which is singular, the quadrature rule breaks down. There is no way to evaluate the function at the singular point and we have to develop another technique. This depends on what type of singularity we are dealing with. Weakly singular kernels contain a singularity but can be integrated. More strongly singular kernels (i.e. hypersingular) are not integrable and require some more care and finesse [20]. Luckily, all the terms found in our integral equations can be integrated, which allows us to use the following quadrature rule.

If a function is of the form

$$\int_a^b \zeta(x) f(x)\, dx \tag{3.12}$$

where $\zeta(x)$ contains an integrable singularity, then it is possible to find an n-point quadrature rule which approximates the integral. Again, we choose 4 evenly spaced points per patch at $\frac{l}{8}$, $\frac{3l}{8}$, $\frac{5l}{8}$, and $\frac{7l}{8}$, where l is the length of the patch. The distance

www.manaraa.com

between the points is again $h = \frac{l}{4}$. The rule will be exact up to third order polynomials. Solving the four independent equations for the four quadrature weights as performed in section 3.1.2, we find that [21]

$$\int_a^b \zeta(x) f(x)\, dx = c_1 f\left(\frac{l}{8}\right) + c_2 f\left(\frac{3}{8}\right) + c_3 f\left(\frac{5}{8}\right) + c_4 f\left(\frac{7}{8}\right) \tag{3.13}$$

where

$$c_1 = \frac{1}{6}\Big[13.125W_0 - 17.75W_1 + 7.5W_2 - W_3\Big] \tag{3.14}$$

$$c_2 = \frac{1}{2}\Big[-4.375W_0 + 11.75W_1 - 6.5W_2 + W_3\Big] \tag{3.15}$$

$$c_3 = \frac{1}{2}\Big[2.625W_0 - 7.75W_1 + 5.5W_2 - W_3\Big] \tag{3.16}$$

$$c_4 = \frac{1}{6}\Big[-1.875W_0 + 5.75W_1 - 4.5W_2 + W_3\Big] \tag{3.17}$$

and

$$W_n \equiv \frac{1}{h^n}\int_a^b x^n \zeta(x)\, dx \tag{3.18}$$

3.2 Path Integrals

The integral equations we need to solve contain integrals of the form

$$\int_C f(\mathbf{x})\, d\mathbf{x} \tag{3.19}$$

where the C denotes that the integral is performed around a specific contour. These are different from the standard integrals we are used to such as

$$\int_a^b f(x)\, dx \tag{3.20}$$

In this integral, the function $f(x)$ is integrated from one x value to another. In the path integral of equation (3.19), the function $f(\mathbf{x})$ is integrated along the path that is traced out by C. It is important to realize that $f(\mathbf{x})$ is NOT the path of integration.

www.manaraa.com

In the two dimensional case, $f(\mathbf{x})$ can be a function of both x and y depending on the integration path.

Equation 3.19 can be expressed in two dimensions as

$$\int_C f(x,y)\,ds \tag{3.21}$$

where ds indicates the path of integration. If we use the Pythagorean Theorem to break up the differential ds, we can write

$$\int_C f(x,y)\sqrt{dx^2+dy^2} \tag{3.22}$$

Factoring, we can pull out a dx and re-express the limits of the integral in terms of x.

$$\int_{x1}^{x2} f(x,y)\sqrt{1+\left(\frac{dy}{dx}\right)^2}\,dx \tag{3.23}$$

The integral can now be evaluated using standard methods. The square root term is simply tacked on to the rest of the integrand, which is evaluated as a whole. As a last clarification, the derivative term $\left(\frac{dy}{dx}\right)^2$ pertains to the surface and not the integrated function $f(x,y)$ [22].

3.3 Nystrom Method

3.3.1 General Technique

We have now built up the mathematical machinery to begin solving (2.75) and (2.76). Replacing A and F with (2.77) and (2.78), (2.75) becomes

$$\begin{aligned} E_z^{inc}(x,y) = K_t(x,y) - ik_1\eta_1 \int_C J_z(x',y')\frac{i}{4}H_0^{(1)}(k_1 r)\,ds' \\ + \frac{\partial}{\partial x}\left[\hat{y}\cdot\int_C \hat{t}(x',y')K_t(x',y')\frac{i}{4}H_0^{(1)}(k_1 r)\,ds'\right] \\ - \frac{\partial}{\partial y}\left[\hat{x}\cdot\int_C \hat{t}(x',y')K_t(x',y')\frac{i}{4}H_0^{(1)}(k_1 r)\,ds'\right] \end{aligned} \tag{3.24}$$

www.manaraa.com

where ds' indicates integration over the surface, and r is given by

$$r = \sqrt{(x - x')^2 + (y - y')^2} \tag{3.25}$$

The dot products can be taken care of by noting

$$\hat{y} \cdot \hat{t} = \sin(\theta') \tag{3.26}$$

$$\hat{x} \cdot \hat{t} = \cos(\theta') \tag{3.27}$$

and identifying θ' as the angle tangent to the surface. Also, we can make use of the fact that

$$\frac{\partial}{\partial z} H_0^{(1)}(z) = -H_1^{(1)}(z) \tag{3.28}$$

Using these, (3.24) becomes

$$\begin{aligned} E_z^{inc} = K_t + \frac{k_1 \eta_1}{4} \int_C J_z(x', y') H_0^{(1)}(k_1 r)\, ds' \\ - \frac{i}{4} \int_C \sin(\theta') K_t(x', y') H_1^{(1)}(k_1 r) k_1 \frac{1}{2r} 2(x - x')\, ds' \\ + \frac{i}{4} \int_C \cos(\theta') K_t(x', y') H_1^{(1)}(k_1 R) k_1 \frac{1}{2r} 2(y - y')\, ds' \end{aligned} \tag{3.29}$$

which simplifies to

$$\begin{aligned} E_z^{inc} = K_t + \frac{k_1 \eta_1}{4} \int_C J_z(x', y') H_0^{(1)}(k_1 r)\, ds' \\ + \frac{i k_1}{4} \int_C K_t(x', y') \frac{H_1^{(1)}(k_1 r)}{r} \Big[\cos(\theta')(y - y') - \sin(\theta')(x - x') \Big] ds' \end{aligned} \tag{3.30}$$

Now, we can use the discretization schemes of section 3.1 to solve this integral equation (along with (2.76)). We discretize the integral by writing

$$\begin{aligned} E_z^{inc} = K_t + \frac{k_1 \eta_1}{4} \sum_i c_i S_i (J_z)_i H_0^{(1)}(k_1 r_i) \\ + \frac{i k_1}{4} \sum_i c_i S_i (K_t)_i \frac{H_1^{(1)}(k_1 r_i)}{r_i} \Big[\cos(\theta_i')(y - y_i') - \sin(\theta_i')(x - x_i') \Big] \end{aligned} \tag{3.31}$$

www.manaraa.com

In this equation, we are summing over i, which are the discretization points. The values c_i are the quadrature weights, and S_i is the Jacobian factor which takes into account the path integration over the surface. From section 3.2, we saw that this can be defined as

$$S_i \equiv \sqrt{1+\left(\frac{dy_i'}{dx_i'}\right)^2} \tag{3.32}$$

when the integration is performed over dx'. If the surface configuration is circular, a change to polar coordinates is optimal. S_i takes the form

$$S_i \equiv R \tag{3.33}$$

when parameterized by $d\theta'$, where R is the radius of the circle.

The next step is the key in the magic behind the Nystrom Method [23]. If we also evaluate the equations at the exact same points as the ones we discretized the integrals at, then we have j equations as follows

$$\begin{aligned}(E_z^{inc})_j = (K_t)_j &+ \frac{k_1\eta_1}{4}\sum_{i,j} c_i S_i (J_z)_i H_0^{(1)}(k_1 r_{ij}) \\ &+ \frac{ik_1}{4}\sum_{i,j} c_i S_i (K_t)_i \frac{H_1^{(1)}(k_1 r_{ij})}{r_{ij}}\left[\cos(\theta_i')(y_j - y_i') - \sin(\theta_i')(x_j - x_i')\right]\end{aligned} \tag{3.34}$$

As a side note, using basic trigonometry to recognize the following relationships,

$$\cos(\theta_i') = \frac{1}{\sqrt{1+\left(\frac{dy_i'}{dx_i'}\right)^2}} = \frac{1}{S_i} \tag{3.35}$$

$$\sin(\theta_i') = \frac{1}{\sqrt{1+\left(\frac{dy_i'}{dx_i'}\right)^2}}\left(\frac{dy_i'}{dx_i'}\right) = \frac{1}{S_i}\left(\frac{dy_i'}{dx_i'}\right) \tag{3.36}$$

it is possible to simplify the second term of (3.34):

$$\begin{aligned}(E_z^{inc})_j = (K_t)_j &+ \frac{k_1\eta_1}{4}\sum_{i,j} c_i S_i (J_z)_i H_0^{(1)}(k_1 r_{ij}) \\ &+ \frac{ik_1}{4}\sum_{i,j} c_i (K_t)_i \frac{H_1^{(1)}(k_1 r_{ij})}{r_{ij}}\left[(y_j - y_i') - \left(\frac{dy_i'}{dx_i'}\right)(x_j - x_i')\right]\end{aligned} \tag{3.37}$$

www.manaraa.com

This form of the equation allows us to easily see that the second term goes to zero if the surface of integration is flat and the evaluation point has the same y value as the integration point.

Returning to the derivation, there is a crucial subtlety in performing the step of (3.34) which slightly alters the equations. If not respected, it introduces a catastrophic error into the simulation. It can be shown that the second integral in (3.30) can be expressed in the alternate form:

$$\frac{ik_1}{4}\int_C K_t(x',y')\frac{H_1^{(1)}(k_1 r)}{r}\Big[\cos(\theta')(y-y')-\sin(\theta')(x-x')\Big]ds' = \int_C K_t(x',y')\hat{\mathbf{n}}(\mathbf{x}')\cdot\nabla G(\mathbf{x},\mathbf{x}') \quad (3.38)$$

It turns out that if the observation point $\mathbf{x}$ coincides with the boundary $\mathbf{x}'$, which we just established is the case with (3.34), then a careful limit needs to be taken. The details of this limit can be found in a number of references including [24], and the result is

$$\lim_{x\to x'}\hat{\mathbf{n}}(\mathbf{x}')\cdot\nabla G(\mathbf{x},\mathbf{x}') = -\frac{1}{2}\delta(\mathbf{x}-\mathbf{x}') + \hat{\mathbf{n}}(\mathbf{x}')\cdot\nabla G(\mathbf{x},\mathbf{x}') \quad (3.39)$$

The second term remains the same as it would if we had naively missed performing this limit. The first term, however, is a delta function which picks a $-\frac{1}{2}K_t$ term out of the integral. If the limit is performed from the other side of the boundary in region II, a $\frac{1}{2}K_t$ term is picked out. These terms will be included in the equations to follow.

The index notation shown in (3.34) can be worked with to produce a matrix equation. First, we define some quantities:

$$M_{ji}^{(w)} \equiv \frac{k_w\eta_w}{4}c_i S_i H_0^{(1)}(k_w r_{ij}) \quad (3.40)$$

$$N_{ji}^{(w)} \equiv \frac{ik_w}{4}c_i S_i \frac{H_1^{(1)}(k_w r_{ij})}{r_{ij}}\Big[\cos(\theta_i')(y_j-y_i')-\sin(\theta_i')(x_j-x_i')\Big] \quad (3.41)$$

Dropping the subscripts and superscripts on $\mathbf{E}$, $\mathbf{J}$, and $\mathbf{K}$, this allows us to write

www.manaraa.com

(3.34) as

$$E_j = \frac{1}{2}K_j + M^{(1)}_{ji}J_i + N^{(1)}_{ji}K_i \tag{3.42}$$

where there is an implicit summation over i. Re-writing this in vector notation,

$$\mathbf{E} = \frac{1}{2}\mathbf{K} + \mathbf{M}^{(1)}\mathbf{J} + \mathbf{N}^{(1)}\mathbf{K} \tag{3.43}$$

or

$$\mathbf{E} = \left(\frac{\mathbf{1}}{\mathbf{2}} + \mathbf{N}^{(1)}\right)\mathbf{K} + \mathbf{M}^{(1)}\mathbf{J} \tag{3.44}$$

where $\frac{\mathbf{1}}{\mathbf{2}}$ is half the identity matrix. Along with equation (3.44), the second equation (2.76) in the coupled pair can be expressed as

$$\mathbf{0} = \left(-\frac{\mathbf{1}}{\mathbf{2}} + \mathbf{N}^{(2)}\right)\mathbf{K} + \mathbf{M}^{(2)}\mathbf{J} \tag{3.45}$$

Combining these into one larger block matrix equation, we have

$$\begin{bmatrix} \mathbf{E} \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \left(\frac{\mathbf{1}}{\mathbf{2}} + \mathbf{N}^{(1)}\right) & \mathbf{M}^{(1)} \\ \left(-\frac{\mathbf{1}}{\mathbf{2}} + \mathbf{N}^{(2)}\right) & \mathbf{M}^{(2)} \end{bmatrix} \begin{bmatrix} \mathbf{K} \\ \mathbf{J} \end{bmatrix} \tag{3.46}$$

The currents $\mathbf{J}$ and $\mathbf{K}$ can be solved for by inverting the $\mathbf{2} \times \mathbf{2}$ block matrix.

3.3.2 Singular Patches

There is an important correction which needs to be noticed in (3.40) and (3.41). If $i = j$, then the matrix elements are singular and the formalism breaks down. The ripple effect is that the 4 quadrature points i of the entire patch containing the singular index j are unusable. The quadrature rule of section 3.1.3 must be used for these patches, which form a diagonal strip through each block. The form of the weight

www.manaraa.com

integrals of (3.18), neglecting the constant in front of the integral, will thus be

$$\mathbf{M}^{(w)} \rightarrow W_n^{(w)} = \frac{k_w \eta_w}{4} \int_{patch} x'^n S H_0^{(1)}(k_w r_j)\, dx' \tag{3.47}$$

$$\mathbf{N}^{(w)} \rightarrow W_n^{(w)} = \frac{ik_w}{4} \int_{patch} x'^n S \frac{H_1^{(1)}(k_w r_j)}{r_j} \Big[\cos(\theta')(y_j - y') - \sin(\theta')(x_j - x') \Big] dx' \tag{3.48}$$

where

$$r_j = \sqrt{(x_j - x')^2 + (y_j - y')^2} \tag{3.49}$$

M and **N** do not transform exactly into the W_n's, but they are loosely related. Studying the details of section 3.1.3 should elucidate the differences. If the integral is performed in polar coordinates, x and y transform in the usual manner, S takes on the form of (3.33), and $x'^n \rightarrow \theta'^n$. More on the implementation of this rule will be discussed in section 3.6.3.

3.3.3 Incident Field

For the incident field, the simplest wave is an infinitely extending uniform plane wave. Mathematically, this looks like

$$E_z^{inc} = E_0 e^{i\mathbf{k}\cdot\mathbf{x}} \tag{3.50}$$

$$= E_0 e^{i(k_x x + k_y y)} \tag{3.51}$$

$$= E_0 e^{ik[x\cos(\theta) + y\sin(\theta)]} \tag{3.52}$$

where θ is the incident angle and E_0 can be set to 1 as a convenient normalization. Also, because the convention is for the incident beam to impinge from the top, a minus sign is added in front of the $\sin(\theta)$ term. In index form, E_z^{inc} then takes the final form

$$\left(E_z^{inc}\right)_i = e^{ik[x_i\cos(\theta) - y_i\sin(\theta)]} \tag{3.53}$$

www.manaraa.com

3.4 Far Field Scattered Intensity

3.4.1 Green's Function Expansion

Once the currents have been solved for, we can use them with source-field relations to find the scattered fields in the far field limit. First, let's expand the Green's function in the far field limit $|\mathbf{x}| >> |\mathbf{x}'|$

$$G(\mathbf{x},\mathbf{x}') = \frac{i}{4}H_0^{(1)}(k|\mathbf{x}-\mathbf{x}'|) \tag{3.54}$$

$$= \frac{i}{4}H_0^{(1)}\left(k\sqrt{\rho^2+\rho'^2-2\mathbf{x}\cdot\mathbf{x}'}\right) \tag{3.55}$$

$$= \frac{i}{4}H_0^{(1)}\left(k\rho\sqrt{1+\frac{\cancel{\rho'^2}}{\cancel{\rho^2}}-\frac{2\hat{\mathbf{x}}\cdot\mathbf{x}'}{x}}\right) \tag{3.56}$$

where the magnitudes of $\mathbf{x}$ and $\mathbf{x}'$ have been expressed in polar coordinates [18]. The second term under the radical is dropped because it is very small, and the radical is Taylor expanded to first order about $\frac{2\hat{\mathbf{x}}\cdot\mathbf{x}'}{\rho}=0$

$$G(\mathbf{x},\mathbf{x}') \approx \frac{i}{4}H_0^{(1)}\left(k\rho\left(1-\frac{\hat{\mathbf{x}}\cdot\mathbf{x}'}{\rho}\right)\right) \tag{3.57}$$

$$\approx \frac{i}{4}H_0^{(1)}\left(k\left(\rho-\hat{\mathbf{x}}\cdot\mathbf{x}'\right)\right) \tag{3.58}$$

Now, we can expand the Hankel function for very large arguments to see that the dominant far field term is

$$H_0^{(1)}(z) \sim \sqrt{\frac{2}{\pi z}}e^{i(z-\frac{\pi}{4})} \quad ; \quad z\to\infty \tag{3.59}$$

Therefore,

$$G(\mathbf{x},\mathbf{x}') \approx \frac{i}{4}\sqrt{\frac{2}{\pi k\rho}}e^{i\left(k(\rho-\hat{\mathbf{x}}\cdot\mathbf{x}')-\frac{\pi}{4}\right)} \tag{3.60}$$

$$\approx \frac{1}{4}\sqrt{\frac{2}{\pi k\rho}}e^{i\frac{\pi}{2}}e^{-i\frac{\pi}{4}}e^{ik\rho}e^{-ik\hat{\mathbf{x}}\cdot\mathbf{x}'} \tag{3.61}$$

$$\approx \frac{e^{i\left(k\rho+\frac{\pi}{4}\right)}}{2\sqrt{2\pi k\rho}}e^{-ik\hat{\mathbf{x}}\cdot\mathbf{x}'} \tag{3.62}$$

www.manaraa.com

The first fraction term only contains the normalization constant and the outgoing spherical wave information. It shows the $\frac{e^{ik\rho}}{\sqrt{\rho}}$ behavior that we expect for an outgoing spherical wave in two dimensions. The second term holds all the angular information as a result of the dot product in the exponent.

3.4.2 Scattered Wave

Using the techniques of section 2.5, (2.54) can be written as

$$\begin{aligned} E_z^s(x,y) = ik_1\eta_1 \int_C J_z(x',y')\frac{i}{4}H_0^{(1)}(k_1 r)\,ds' \\ - \frac{\partial}{\partial x}\left[\hat{y}\cdot\int_C \hat{t}(x',y')K_t(x',y')\frac{i}{4}H_0^{(1)}(k_1 r)\,ds'\right] \\ + \frac{\partial}{\partial y}\left[\hat{x}\cdot\int_C \hat{t}(x',y')K_t(x',y')\frac{i}{4}H_0^{(1)}(k_1 r)\,ds'\right] \end{aligned} \tag{3.63}$$

Putting in the far field Green function expansion (3.62) and continuing,

$$\begin{aligned} E_z^s(\rho,\phi) = ik_1\eta_1\frac{e^{i\left(k_1\rho+\frac{\pi}{4}\right)}}{2\sqrt{2\pi k_1\rho}}\int_C J_z(x',y')e^{-ik_1(\cos(\phi)x'+\sin(\phi)y')}\,ds' \\ - \frac{\partial}{\partial x}\int_C \sin(\theta')K_t(x',y')\frac{e^{i\left(k_1\rho+\frac{\pi}{4}\right)}}{2\sqrt{2\pi k_1\rho}}e^{-ik_1(\cos(\phi)x'+\sin(\phi)y')}\,ds' \\ + \frac{\partial}{\partial y}\int_C \cos(\theta')K_t(x',y')\frac{e^{i\left(k_1\rho+\frac{\pi}{4}\right)}}{2\sqrt{2\pi k_1\rho}}e^{-ik_1(\cos(\phi)x'+\sin(\phi)y')}\,ds' \end{aligned} \tag{3.64}$$

The partial derivatives with respect to x and y will act on all the terms containing unprimed coordinates. Explicitly in terms of x and y,

$$\rho = \sqrt{x^2+y^2} \tag{3.65}$$

$$\cos(\phi) = \frac{x}{\sqrt{x^2+y^2}} \tag{3.66}$$

$$\sin(\phi) = \frac{y}{\sqrt{x^2+y^2}} \tag{3.67}$$

Performing these derivatives using the product rule produces several terms which are added together. Because we only consider the dominant far field term as $\rho \to \infty$, all

www.manaraa.com

terms of order greater than $\frac{1}{\sqrt{\rho}}$ are dropped. The only term which survives effectively brings an extra ϕ-dependent factor down from the exponent.

$$E_z^s(\rho,\phi) = \frac{e^{i\left(k_1\rho+\frac{\pi}{4}\right)}}{2\sqrt{2\pi k_1\rho}}\Bigg\{ik_1\eta_1\int_C J_z(x',y')e^{-ik_1(\cos(\phi)x'+\sin(\phi)y')}\,ds' + ik_1\int_C\left[\cos(\theta')\sin(\phi)-\sin(\theta')\cos(\phi)\right]K_t(x',y')e^{-ik_1(\cos(\phi)x'+\sin(\phi)y')}\,ds'\Bigg\} \quad (3.68)$$

The same quadrature rule of section 3.1 can be used to numerical integrate this expression. First, we define

$$P_i \equiv c_i S_i k_1\eta_1 e^{i\frac{3\pi}{4}}e^{-ik_1\left(\cos(\phi)x_i'+\sin(\phi)y_i'\right)} \quad (3.69)$$

$$Q_i \equiv c_i S_i k_1\left[\cos(\theta_i')\sin(\phi)-\sin(\theta_i')\cos(\phi)\right]e^{i\frac{3\pi}{4}}e^{-ik_1\left(\cos(\phi)x_i'+\sin(\phi)y_i'\right)} \quad (3.70)$$

Then, the scattered field is simply given by

$$E_z^s(\rho,\phi) = \frac{e^{ik_1\rho}}{2\sqrt{2\pi k_1\rho}}\left(P_iJ_i+Q_iK_i\right) = g(\rho)f(\phi) \quad (3.71)$$

where

$$g(\rho) \equiv \frac{e^{ik_1\rho}}{2\sqrt{2\pi k_1\rho}} \quad (3.72)$$

$$f(\phi) \equiv \left(P_iJ_i+Q_iK_i\right) \quad (3.73)$$

An implicit sum over i (Einstein notation) is assumed in (3.73). $f(\phi)$ will prove to be the more useful term because it contains all the angular information. $g(\rho)$ adjusts the phase and normalization of the wave at an arbitrary distance ρ.

3.5 Geometry of the Scatterer

3.5.1 General Description

The scatterer used in this numerical simulation is designed to probe how roughness affects reflection from deposited multilayer surfaces. Its general shape is that of a

www.manaraa.com

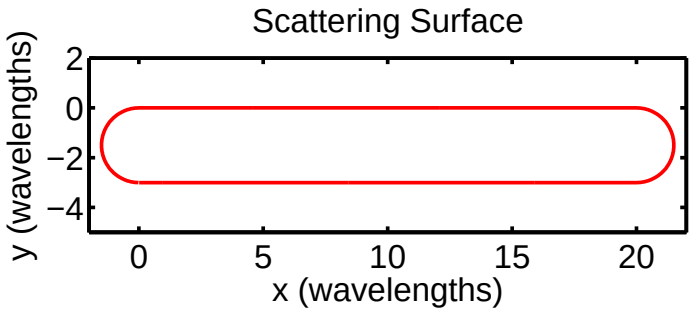

Figure 3.2 A typical scattering surface with a length of 20 wavelengths and a thickness of 3 wavelengths.

deformed athletic jogging track. There are two flat strips (actually rough depending on the surfaces) which are connected by semicircle regions on either end. For the integral equations to represent a well-posed problem, the scatterer must be a closed surface [19]. This excludes the possibility of constructing the problem as two parallel rough surfaces which either extend outward indefinitely or abruptly end, leaving the sides open.

The interior of the scatterer is composed of dielectric material. The incident wave impinges on the scatterer from above and passes through two interfaces. Because there are only two materials present, both interfaces involve the same two materials, albeit in opposite alignments. This limits the number of materials we can model, but still allows us to study the effects of multiple surfaces on reflection. Additionally, it greatly assuages the computational burden of solving for many more surface unknowns.

Circular caps were chosen to close off the sides because they are smooth and don't create abrupt edges. Such jagged, pointy regions cause major spikes and even singularities in surface currents [6], making them computationally difficult to handle.

www.manaraa.com

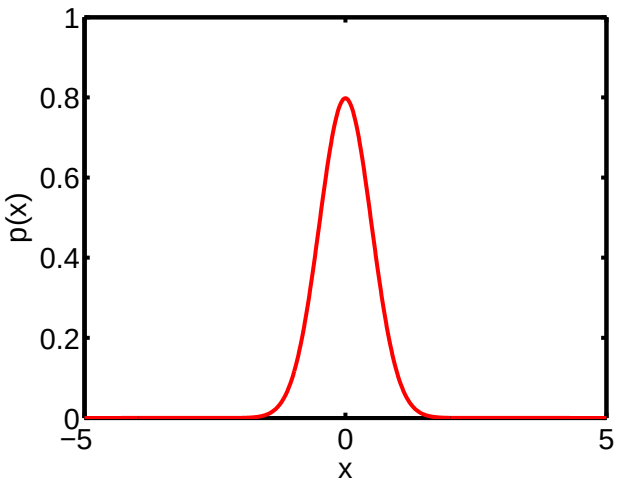

Figure 3.3 A normal distribution with $\mu = 0$ and $\sigma = 0.5$.

3.5.2 Modeling the Rough Sections

Roughness is added to the scatterer using a random normal (Gaussian) distribution. The probability density of this distribution is defined as

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{3.74}$$

where μ in this case is the mean and σ is the standard deviation. The probability density has the property

$$\int_{-\infty}^{\infty} p(x)\,dx = 1 \tag{3.75}$$

so that the absolute probability of obtaining a value in the domain $a < x < b$ is

$$P(x) = \int_{a}^{b} p(x)\,dx \tag{3.76}$$

In our case, μ is the height of the ideal interface, while σ is the targeted RMS roughness. Points are first identified along the surface at equally spaced intervals. Then, the distribution is applied to each of these points to create an array of Gaussian random points which will define the surface. A smooth connection is drawn between

www.manaraa.com

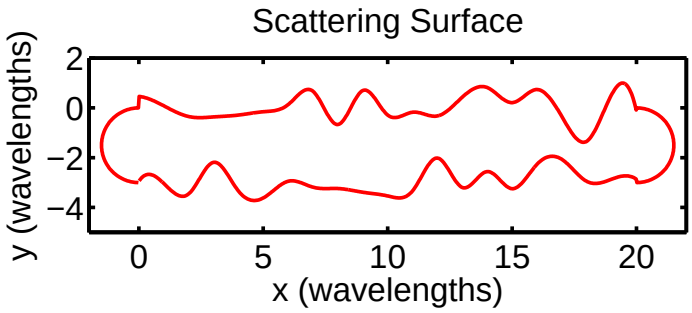

Figure 3.4 An example of uncorrelated roughness. The top and bottom surfaces are generated independent of one another so that there is no visible relation between the two.

these points using a piecewise continuous third order spline interpolant. (see `spline` in `Matlab`)

This is an important step because in using this interpolating function we are implicitly introducing information about the correlation between neighboring atoms. This opposes the notion of the surface being a totally random, uncorrelated collection of atoms. This type of surface would be present if there were no energetic or physical interactions between neighboring atoms. During film deposition, atoms would simply stack in a random manner. In reality, adjacent atoms must have some type of effect on each other. This leads to the expectation that there is a degree of local continuity on the surface. The spline in effect forces the locations of adjacent atoms to be related. The third order character of the spline ensures that there will not be an excessive number of wild oscillations between the chosen surface points.

Uncorrelated roughness means that the top and bottom surfaces are generated independent of each other. The word 'correlated' is now being used in a different context than the previous paragraph. A random distribution is separately applied to each so that the bottom surface does not influence the top.

Correlated roughness takes the bottom surface into account when modeling the top. This is potentially seen a more realistic model because thin films are deposited

www.manaraa.com

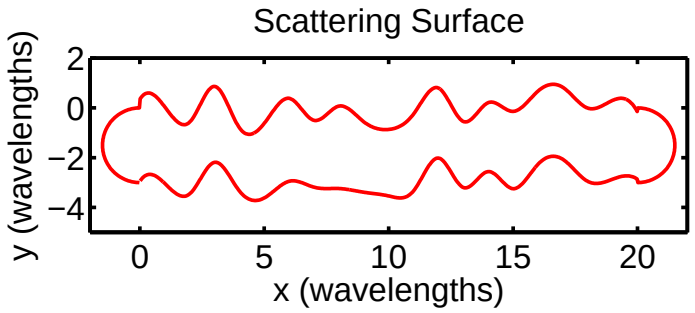

Figure 3.5 An example of correlated roughness. The top surface is generated based on the parameters of the bottom surface. Although they are not identical, the top surface shows traits and the general shape of the bottom surface.

on substrates which are already rough. First, the rough bottom surface is specified. The top surface is then generated using the normal distribution, except that μ at each point now deviates from the ideal interface by its corresponding random deviation from the bottom surface. This creates a scatterer where similarities are seen between the bottom and top surfaces.

3.6 Program Methods and Structure

3.6.1 Overview

Because of its familiarity to myself and other members of the BYU XUV optics group, the scattering problem was solved using `Matlab`. One of the strengths of `Matlab` is its ability to manipulate large vectors and matrices with relative ease. The code was broken up into sections known as `.m` files, which were then also broken up into subsections and so on. This allows for hierarchal organization intended to avoid an excessively lengthy, unwieldy program. Additionally, it prepares the program to run as a parallel process. The specifics of this issue will be discussed in section 4.2.

For reference, the entire program is cataloged in appendix A, with the `.m` files

www.manaraa.com

listed in alphabetical order. The primary `.m` files which are run by the user from the `Matlab` interface are `TM.m` (pg. 120) and `TE.m` (pg. 97). These master files contain a sequence of `.m` files which the program will execute in solving the scattering problem. Some of these subroutines are necessary, while others are options which can be commented out if not needed. All the uncommented `.m` files in the versions of `TM.m` and `TE.m` shown in Appendix A are necessary.

The other file which is important for routine calculations is `userinputs.m` (pg. 144). This file contains all the parameters which are routinely toggled by the user. At this stage, both `TM.m` and `TE.m` run off this same input file. Comments are present next to the variables which explain their purpose. If the user desires to execute a series of scattering problems as a function of one of the parameters in the `userinputs.m` file, this parameter can be commented out of `userinputs.m` and added as an input of `TM.m` or `TE.m`. Note that if this action is performed an output must also be specified.

The length scale of surface features and incident x-ray radiation is on the order of nanometers. It can become confusing and cumbersome to always work with such small numbers. To make things easier, the unit of length has been defined as the wavelength of the incident light. This allows us to easily compare relative lengths. Also, parallels can easily be seen between combinations of larger scatterers and incident wavelengths.

3.6.2 The Scatterer

The top and bottom portions of the scattering surface are each divided into an equal number of patches given by the parameter `patches`. Likewise, the number of patches on each side section is `tpatches`. It is important to understand that these quadrature points are different than the points which are used to define the surface. The quadrature points are computed after the surface has been generated by the procedure detailed in section 3.5.2. The spacing of the array of surface points is governed

www.manaraa.com

by `rfreq`. This parameter specifies the number of random surfaces points to place per wavelength. For example, if the scattering surface has a length 10λ, and `rfreq` is assigned the value 0.5, then the program generates 20 random surface points. Therefore, as `rfreq` decreases, the frequency of surface roughness increases.

There are some other subtleties with the rough surfaces which are important to recognize. In `userinputs.m`, there are terms which deal with RMS roughness heights and correlation. The surfaces can be forced to repeatedly generate the same random surfaces by fixing the seed of the random number generator. Conversely, if it is allowed to float, new surfaces are created. The `fixsurface` terms control whether the seed is allowed to float, and if they are not, the `inpstate` terms specify the seed.

The bottom surface is always generated independent of the top and its RMS height is expected to correlate with the standard deviation σ of its normal distribution. The same cannot always be said of the top surface. If the surfaces are uncorrelated, the top surface behaves as we expect from its specified σ. The presence of correlation, however, effectively creates a surface with the combination of two standard deviations. An overall RMS height can be predicted in the standard method by adding the two σ values in quadrature. Correlation is toggled on and off using the `correlatedroughness` variable in `userinputs.m`.

3.6.3 Nystrom Matrix Fill

The `.m` files `TEnystromfill.m` (pg. 101) and `TMnystromfill.m` (pg. 125) fill the entire Nystrom matrix for each polarization. The structure of these files is such that the fill is broken up into different sections of the matrix. This is done because the matrix element calculations depend on whether they correspond with J or K elements, are located on the inner or outer region, or are top/bottom or side sections of the scatterer. The scatterer has been broken up into four sections: the bottom,

www.manaraa.com

the right side, the top, and the left side. For each of these sections, there is a J and a K element, which each also have an outer and an inner region element. Therefore, the Nystrom fill is composed of 16 block submatrix fills.

Each block submatrix is filled using a double loop. The outer loop moves through all the possible observation points, while the inner one goes through the integration quadrature points of the particular block submatrix. The shape of these block submatrices is thus a long, skinny rectangle. Each iteration of the inner loop contains a check to establish whether the quadrature points and observation point lie on the same patch or not. This determines whether the matrix fill is done using the standard quadrature rule of section 3.1.2 or the singularity-containing rule of section 3.1.3.

While the off-diagonal patch fill is relatively straightforward, the singular patches require more care. The W integrals are performed by calling either `cartK.m` (pg. 88), `cartJ.m` (pg. 88), `polK.m` (pg. 94), or `polJ.m` (pg. 94). The names of these files are constructed so that the J and K refer to the mathematical surface currents J and K, and 'cart' and 'pol' indicate either cartesian or polar coordinates. This notation is only relevant in the TM case. Because the side sections are perfectly circular, the path integrals over their surfaces are ideally performed in polar coordinates with the origin shifted to the center of the circle.

There are several arguments which are passed into these four `.m` files. The first two numbers are always the limits of integration, the third identifies which point on the patch is singular, and the fourth sets the order of the monomial weight function (either x'^n or θ'^x) in the integrand of W. We want the monomial weight function in each W integral to range from 0 to the end of the patch, but the rest of the integrand to be evaluated at its specific location on the scatterer. The variable `offset` lets the function know which part of the scatterer to integrate since the limits of integration which are passed into the function are always the same. The other input terms send

www.manaraa.com

information about the surfaces and parameters of the problem into the function.

The four `.m` files each call the subfunction `llquadr.m` (pg. 92) twice. This routine numerically integrates the function from the singular point (where it may contain a logarithmic singularity such as the zeroth order Hankel function) to one end of the patch. `llquadr.m`'s four inputs are the symbolic integrand, the order, and the limits of integration. It is called twice because the contributions from the singular point to both ends of the patch are added together. The integrands are given by `cartKfunc.m` (pg. 89), `cartJfunc.m` (pg. 88), `polKfunc.m` (pg. 94), and `polJfunc.m` (pg. 94). Even though the polar integrals are performed over θ', the symbolic variable used in these functions is still x.

The order input of `llquadr.m` lets the function know how many points to use in its quadrature rule. The location of the quadrature points and the weights of this rule are contained in `llquadzw.txt` (pg. 144) [25]. The higher the order of the monomial weight function in W and the longer the patch, the more quadrature points are necessary for a given accuracy. `llquadr.m` uses `linlogOrder.m` (pg. 92) to predict the necessary order for about 7 digits of accuracy using the monomial weight order and the length of the patch as inputs. This algorithm was developed by comparing 'exact' integrals against the quadrature result and looking at the error for different lengths and monomial orders. Using this input, `llquadr.m` calls `linlogweights.m` (pg. 90), which loads the quadrature points and weights of the particular order. Finally, `llquadr.m` computes the value of the integral.

The matrix elements corresponding to the top section of the scatterer all have an overall minus sign with respect to the bottom section. This is because the path integral around the surface has been defined in the counterclockwise direction. The bottom section is integrated to the right (positive x direction), while the top is integrated to the left (negative x direction). The limits of integration are essentially

www.manaraa.com

reversed between the top and bottom surfaces, introducing the minus sign.

3.6.4 Far Field Calculation

After the files `TEsolvematrix.m` (pg. 119) and `TMsolvematrix.m` (pg. 143) solve the matrix system for **J** and **K**, `TEfarfield` (pg. 98) and `TMfarfield` (pg. 122) perform the far field calculation. This is much simpler than the matrix fill because the observation point not located on the surface and the quadrature rule of section 3.1.2 is sufficient. The dominant far field term of the reflected electric field is given by (3.71). The path integrations are again broken up into both the side and top/bottom regions of the scatterer. Also, the minus sign associated with the top surface integration is present.

In general, $f(\phi)$ as defined in (3.73) is a complex quantity associated with the angular distribution of the scattered field. When a reflectance measurement is taken, it is not the ratio of the fields which is acquired, but rather the intensities. The intensity is given by the absolute square of the field. Therefore, the quantity of interest which will be plotted and compared is actually

$$F(\phi) = |f(\phi)|^2 \tag{3.77}$$

3.6.5 Extras

The other `.m` files contained in `TE.m` and `TM.m` are extra options which are not essential for basic program operation. `TEcylcon.m` (pg. 97) and `TMcylcon.m` (pg. 121) plot the numerical solution against that of a cylindrical conductor of radius $\frac{1}{2}t$. The origin and applications of this analytical conductor solution will be addressed later. For the sake of comparison, it is desirable to change the limits of the far field angle evaluation to include the entire 2π angular range. This is automatically built into the program

www.manaraa.com

if `len` and `patches` are both set to zero. Shrinking the length of the top and bottom sections to zero leaves only the two adjoining side sections, which forms a perfect circle of radius $\frac{1}{2}t$.

`TMphysoptcompare.m` (pg. 142) compares the the numerical solution against the TM physical optics approximate solution of a perfectly flat conductor. The theory and implications of this will also be thoroughly examined in subsequent chapters.

www.manaraa.com

Chapter 4

Numerical Issues

4.1 Convergence

The accuracy of the scattering problem solution depends on how finely the scatterer is discretized. This corresponds to the choice of `patches` and `tpatches` for a given geometry. In general, there are two physical factors which influence how these parameters are chosen.

The first deals with the number of quadrature points per wavelength of incident light. At the absolute minimum, at least two quadrature points are necessary per wavelength. If fewer are taken, we don't obtain enough information about the wave for the solution to converge to the correct value. The literature suggests using an absolute minimum of 10 points per wavelength for low order quadrature rules to obtain reasonable accuracies. For most applications, many more points are desired [19].

The second consideration is the physical characteristics of the scatterer. Figure 4.1 depicts a rough scatterer which is discretized using two different sets of quadrature points. For the set of quadrature points to be a good approximation of the surface, they must trace out the details of the surface. The red dots are spaced so that they

www.manaraa.com

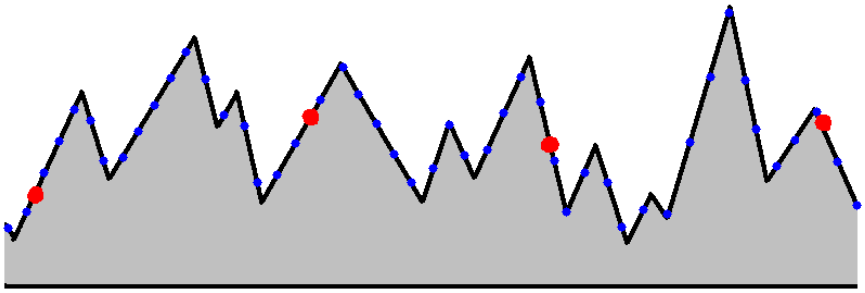

Figure 4.1 A comparison of two quadrature point schemes. The closely spaced blue dots preserve more information about the surface than the red dots.

completely leave out several peaks. If only these points were referenced without any prior knowledge of the actual surface, we would be unable to recreate it, leaving its structure lost. Using only the blue points, however, we would be able to reproduce a surface much like the actual one. The number of quadrature points must be chosen so as to contain a high level of information about the structure of the surface.

Figure 4.2 shows the computed reflectance at 20 degrees from grazing of a 50 wavelength long, 0.5 wavelength thick flat scatterer as a function of the number of patches per straight section. The number of patches per curved side section was held fixed at 10, and the index of refraction was $n = 1.05$, $k = 0.05$. The method of computing reflectance is unimportant at this point and will be explained in subsequent chapters. The same is true for the ideal blue reflectance, which is predicted using the Fresnel coefficients. The point relevant to this discussion is that the reflectance appears to be asymptotically converging to a value which coincides with the blue line. It is reasonable to infer that as the number of patches is increased, the accuracy of computed reflectance improves.

Roughness has been added to the specifications of Figure 4.2 to produce Figures 4.3 and 4.4. Figure 4.3 has been changed so that `rheighttop` $= 0.1$ and `rfreq` $= 0.5$. In Figure 4.4, `rheighttop` $= 0.05$ and `rfreq` $= 3$, so that both the frequency and

www.manaraa.com

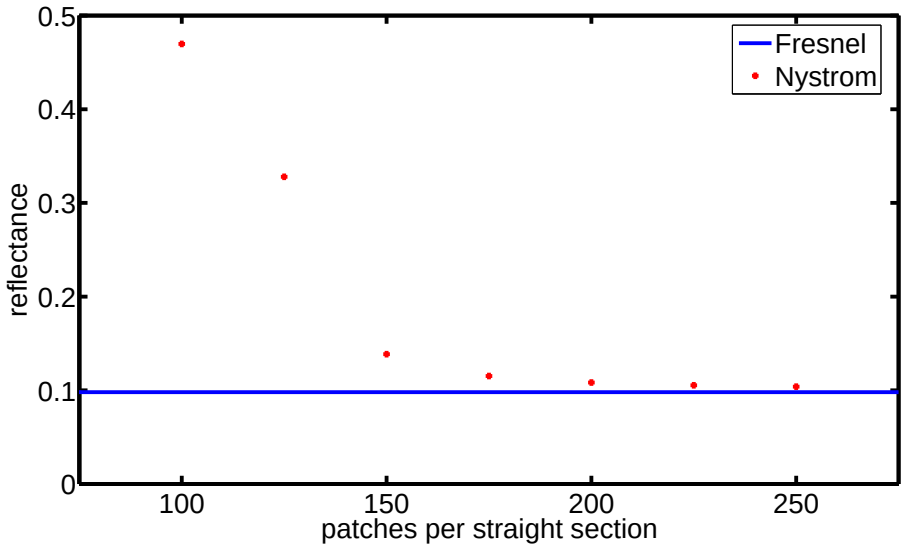

Figure 4.2 Computed reflectance at $\theta = 20°$ as a function of `patches`, where `len` $= 50$, `t` $= 0.5$, `tpatches` $= 10$, $n = 1.05$, and $k = 0.05$. The is no roughness present and exponential convergence is observed.

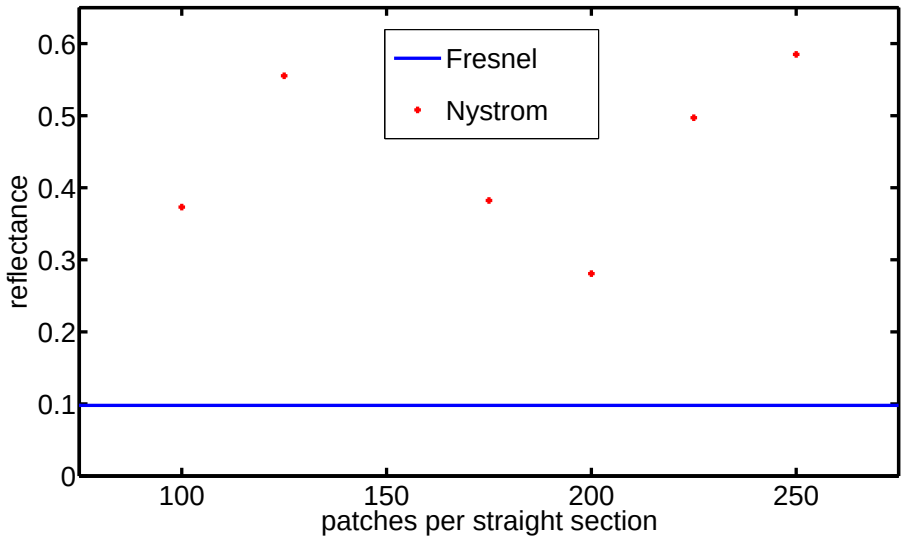

Figure 4.3 Computed reflectance at $\theta = 20°$ as a function of `patches`, where `len` $= 50$, `t` $= 0.5$, `tpatches` $= 10$, $n = 1.05$, $k = 0.05$, `rheighttop` $= 0.1$, and `rfreq` $= 0.5$. No convergence is observed due to the significant amount of roughness.

www.manaraa.com

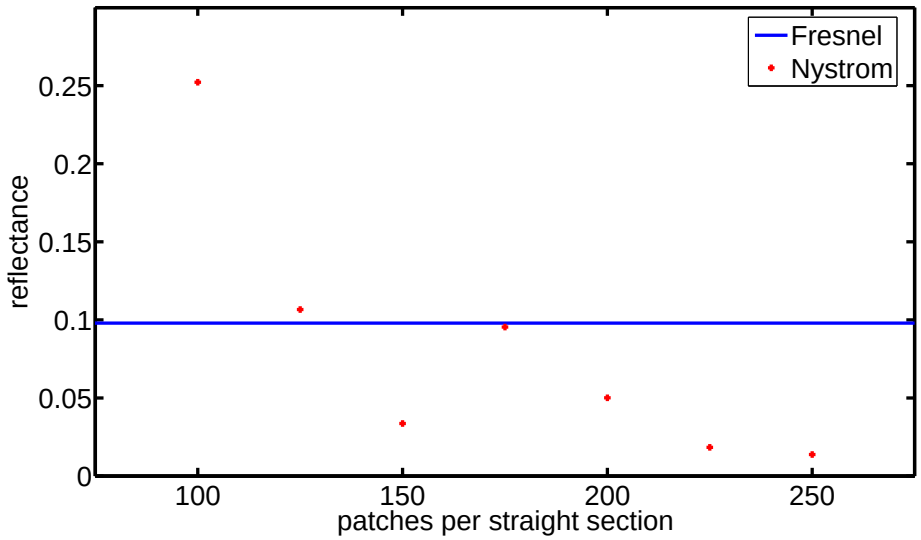

Figure 4.4 Computed reflectance at $\theta = 20°$ as a function of `patches`, where `len` $= 50$, `t` $= 0.5$, `tpatches` $= 10$, $n = 1.05$, $k = 0.05$, `rheighttop` $= 0.05$, and `rfreq` $= 3$. A pattern consistent with convergence appears at 175 patches.

RMS height of the roughness are smaller. The erratic pattern of the red dots in Figure 4.3 indicates that after 250 patches the program has still not begun to converge. We do not see the signatures of the exponential convergence apparent in Figure 4.2. We conclude that enough roughness has been introduced so that 250 patches is now inadequate.

Figure 4.4 shows a combination of Figures 4.2 and 4.3. The red dots at first seem to jump around chaotically, which is typical before convergence. At about 175 patches, however, the pattern shifts so that we see the exponential convergence pattern of Figure 4.2. This seems to indicate that the addition of roughness does indeed increase the number of patches needed for a desired level of accuracy. At this point, no method has been developed for predicting the number of required patches. A few calibration runs to check for convergence is necessary.

www.manaraa.com

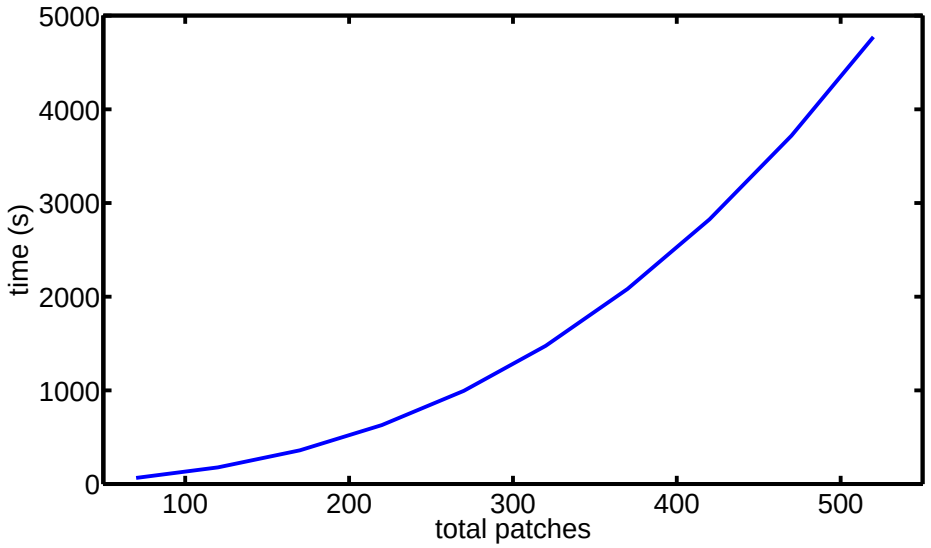

Figure 4.5 Run time vs. total patches.

4.2 Run Time

The practicality of using the program depends on the amount of time it takes to run. As more unknowns are added to the mix, the computational price is increased. Figure 4.5 shows the time of calculation as a function of the total number of patches. This relationship is clearly nonlinear, and is expected to be second order in character. This is mainly because the number of Nystrom matrix entries scales as the number of quadrature points squared. The diagonal patch blocks of the Nystrom matrix employ the quadrature rule of section 3.1.3, where the integration and observation points are on the same patch. These matrix elements require a numerical integration subroutine, which is expected to increase their proportional runtime significantly. The time expenditure of these regions, along with other code sections such as the far field intensity, should scale linearly with the number of patches.

If Figure 4.5 is fit with a second order polynomial, the result is qualitatively poor. When a third order term is included, however, the result is much much more accurate.

www.manaraa.com

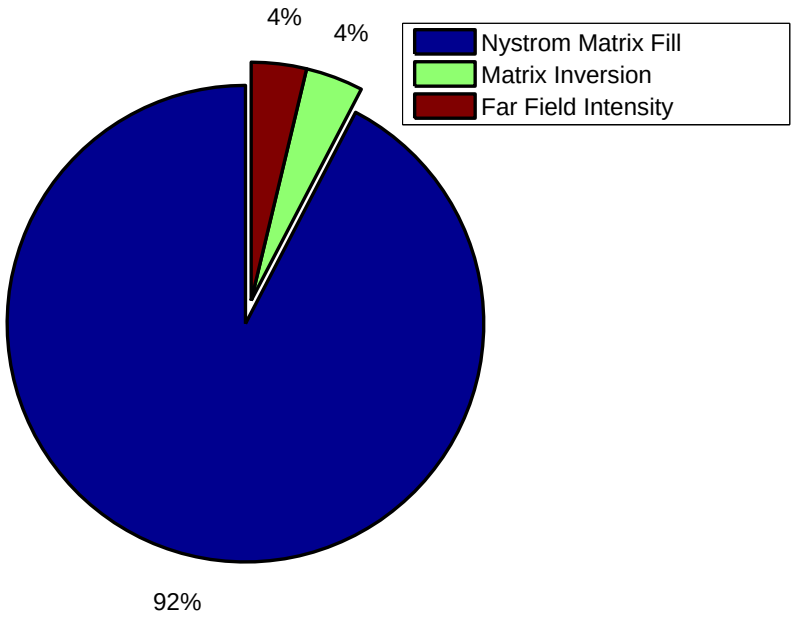

Figure 4.6 A breakdown of run time into individual program sections of a long run.

The approximate overall run time as a function of the total number of patches is

$$t = 1.708 \cdot 10^{-5} p^3 + 8.542 \cdot 10^{-3} p^2 + 9.482 \cdot 10^{-2} p + 12.33 \tag{4.1}$$

where t is the time in seconds and p is the total number of patches. It is not intuitively clear where the third order term originates. This formula will of course depend on the processing speed and memory of the computer. (4.1) was generated using a 1.4 GHz Pentium M processor with 512 MB of RAM.

A realistic model of x-ray reflectance from a rough surface requires a plate which is long with respect to the wavelength. As the length of the plate increases, more quadrature points are necessary to preserve accuracy, especially as the surfaces become rough. Figure 4.6 shows the time breakdown of a 520 patch (2080 quadrature point) run which had flat section lengths of 50 wavelengths. The Nystrom fill overwhelmingly dominates for this specific number of quadrature points. Eventually, the matrix inversion will begin to catch up and overtake the fill. At this point, how-

www.manaraa.com

ever, the Nystrom fill is a perfect candidate to be broken up in a massively parallel process. The elements of the matrix are calculated independent of one another, suggesting they could be parceled out to several CPU's for calculation. These individual sections could then be recombined in preparation for matrix inversion and subsequent subroutines. The Marylou supercomputing clusters at BYU are ideal for such a serial process.

www.manaraa.com

www.manaraa.com

Chapter 5

Validation

5.1 Physical Optics Flat Plate (TM)

5.1.1 Derivation

The TM scattered wave is given by (3.64). Let's consider a flat perfect conductor of length L. The plate will lie along the x axis and be centered at the origin. The unit normal from the surface is $\hat{\mathbf{y}}$. The tangential component of the total electric field on the boundary is zero for a perfect conductor, so $K_t = 0$. (3.64) becomes

$$E_z^s(\rho,\phi) = ik_1\eta_1 \frac{e^{i\left(k_1\rho+\frac{\pi}{4}\right)}}{2\sqrt{2\pi k_1\rho}} \int_C J_z(x',y') e^{-ik_1(\cos(\phi)x'+\sin(\phi)y')}\, ds' \tag{5.1}$$

www.manaraa.com

From the definition of **J** in (2.56),

$$\mathbf{J} = \hat{\mathbf{y}} \times \mathbf{H} \tag{5.2}$$

$$= \hat{\mathbf{y}} \times \left(\mathbf{H}^{inc} + \mathbf{H}^{s}\right) \tag{5.3}$$

$$= \left(H^{inc} + H^{s}\right) \sin(\theta)\hat{\mathbf{z}} \tag{5.4}$$

$$= \frac{1}{\mu_1}\left(B^{inc} + B^{s}\right) \sin(\theta)\hat{\mathbf{z}} \tag{5.5}$$

$$= \frac{1}{\mu_1 c}\left(E^{inc} + E^{s}\right) \sin(\theta)\hat{\mathbf{z}} \tag{5.6}$$

$$= \frac{1}{\mu_1 c}\left(e^{i\mathbf{p}\cdot\mathbf{x}} + e^{i\mathbf{q}\cdot\mathbf{x}}\right) \sin(\theta)\hat{\mathbf{z}} \tag{5.7}$$

$$= \frac{1}{\mu_1 c}\left[e^{i(p_x x + p_y y)} + e^{i(q_x x + q_y y)}\right] \sin(\theta)\hat{\mathbf{z}} \tag{5.8}$$

where θ is the incident angle and $\mathbf{p}$ and $\mathbf{q}$ represent the wavevectors of the incoming and outgoing waves. Note there is no phase shift in H after reflection from a perfect conductor. This is the reason the incident and scattered field magnitudes add together. Only the E field undergoes a π radian phase change. The approximation is made as we invoke the law of reflection:

$$p_x = k_1 \cos(\theta) = q_x \tag{5.9}$$

$$p_y = -k_1 \sin(\theta) = -q_y \tag{5.10}$$

Putting this all together into (5.1) gives

$$E_z^s(\rho,\phi) = ik_1\eta_1 \frac{e^{i\left(k_1\rho + \frac{\pi}{4}\right)}}{2\sqrt{2\pi k_1 \rho}} \int_C \left\{ \frac{e^{ik_1(\cos(\theta)x' - \sin(\theta)y')} + e^{ik_1(\cos(\theta)x' + \sin(\theta)y')}}{\mu_1 c} \cdot \sin(\theta) e^{-ik_1(\cos(\phi)x' + \sin(\phi)y')}\, ds' \right\} \tag{5.11}$$

www.manaraa.com

In the limit as $y' \to 0$, and if region I is vacuum, then E_z^s becomes [18, 26]

$$E_z^s(\rho,\phi) = \frac{ik_1\eta_1}{\mu_1 c}\frac{e^{i\left(k_1\rho+\frac{\pi}{4}\right)}}{2\sqrt{2\pi k_1\rho}}\sin(\theta)\int_{-L/2}^{L/2} 2e^{ik_1\cos(\theta)x'}\cdot e^{-ik_1\cos(\phi)x'}\,dx' \tag{5.12}$$

$$= ik_1\frac{e^{i\left(k_1\rho+\frac{\pi}{4}\right)}}{\sqrt{2\pi k_1\rho}}\sin(\theta)\int_{-L/2}^{L/2} e^{ik_1x'[\cos(\theta)-\cos(\phi)]}\,dx' \tag{5.13}$$

$$= \frac{2e^{i\left(k_1\rho+\frac{3\pi}{4}\right)}}{\sqrt{2\pi k_1\rho}}\frac{\sin(\theta)}{\delta}\sin\left(\frac{\delta k_1 L}{2}\right) \tag{5.14}$$

$$= \frac{e^{ik_1\rho}}{2\sqrt{2\pi k_1\rho}}e^{i\frac{3\pi}{4}}\frac{4\sin(\theta)}{\delta}\sin\left(\frac{\delta k_1 L}{2}\right) \tag{5.15}$$

where

$$\delta = \cos(\theta) - \cos(\phi) \tag{5.16}$$

Therefore,

$$f(\phi) = 4e^{i\frac{3\pi}{4}}\frac{\sin(\theta)}{\delta}\sin\left(\frac{\delta k_1 L}{2}\right) \tag{5.17}$$

5.1.2 Comparison

The computed far field intensity was compared against the physical optics solution of a flat conductor excited by a TM wave. The thickness of the plate is mostly unimportant because a prefect conductor reflects 100% of the incident radiation without any transmission. This negates any possible interference effects from the back surface. The only possible effect thickness could have is from the circular side sections of the scatterer, which change dimensions as the thickness is increased. The so called edge effects would then also be altered. The physical optics, or Kirchoff approximation, does not take these edge effects into account. Rather, it assumes an induced current based on an infinite conducting plate tangent to each surface point. This suggests the approximation gets better as the length of the plate increases.

In order to simulate a perfect conductor, the real part of the index of refraction is set to the very large value of $n = 10^{10}$. Figures 5.1, 5.2, and 5.3 show comparison

www.manaraa.com

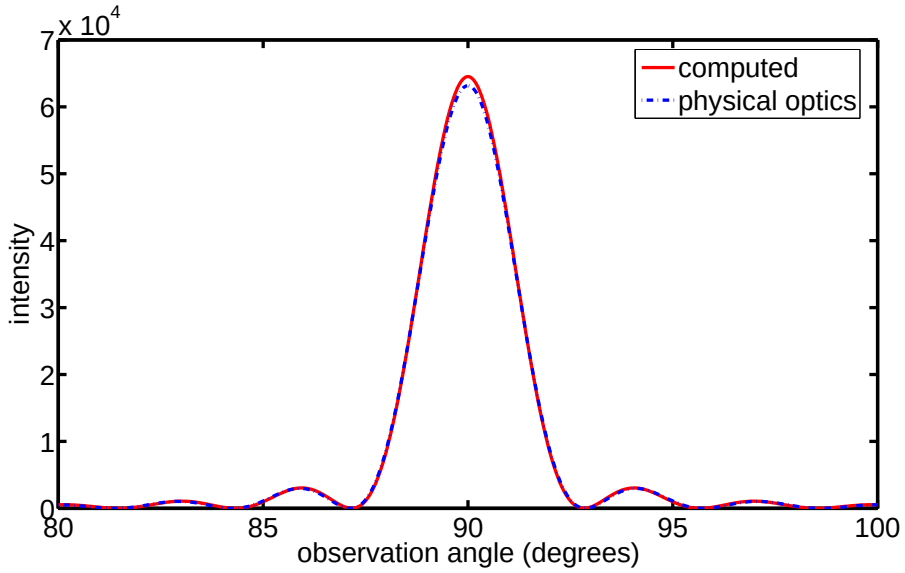

Figure 5.1 Comparison of computed solution with physical optics approximation for a flat perfect conductor. `len` = 20, `patches` = 50, `t` = 0.5, `tpatches` = 10, `n` = 10^{10}, `k` = 0, and `thetadeg` = 90. The polarization is TM.

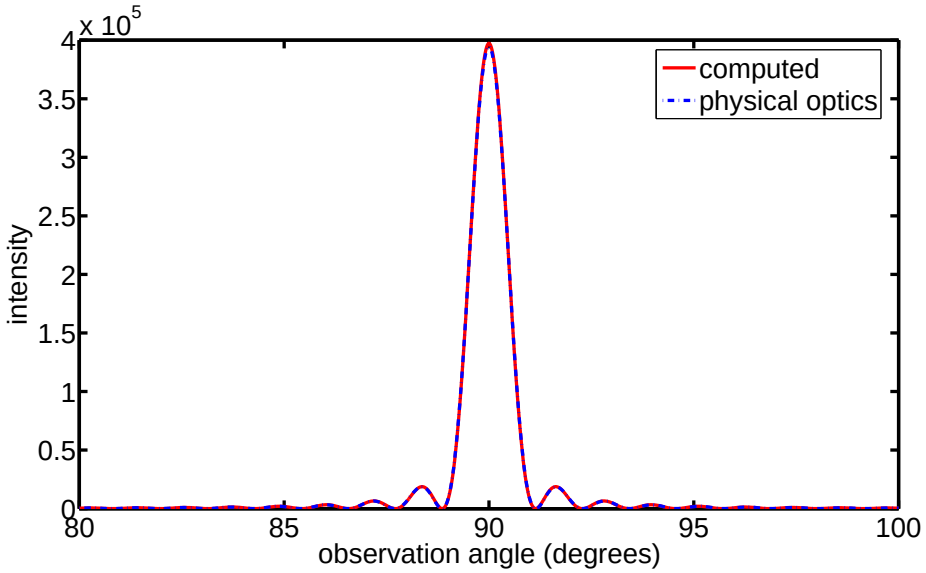

Figure 5.2 Comparison of computed solution with physical optics approximation for a flat perfect conductor. `len` = 50, `patches` = 100, `t` = 0.5, `tpatches` = 10, `n` = 10^{10}, `k` = 0, and `thetadeg` = 90. The polarization is TM.

www.manaraa.com

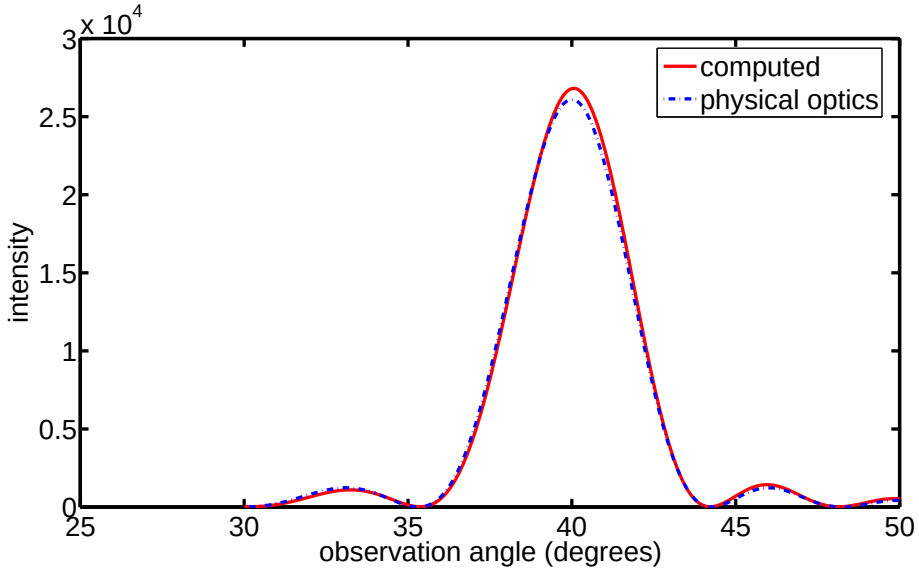

Figure 5.3 Comparison of computed solution with physical optics approximation for a flat perfect conductor. `len` = 20, `patches` = 100, `t` = 0.5, `tpatches` = 10, `n` = 10^{10}, `k` = 0, and `thetadeg` = 40. The polarization is TM.

plots between the computed solution and the physical optics solution for different plate lengths and incident angles. The reflected diffraction patterns match almost perfectly in each case. Slight discrepancies can be noticed in the height of the central specular peaks of the shorter plate graphs. This is presumably due to the increasing contribution of the edge effects.

The height of the specular peak increases and its width decreases as the plate length increases. This can be seen in Figures 5.1 and 5.2. These effects occur because the phase of the reflected wave at each point on the surface is such that more constructive interference is produced at the specular angle as length is increased. In the limit of an infinite plate, the angular intensity distribution will be a delta function at the specular angle of with a strength of the incident intensity.

www.manaraa.com

5.2 Perfectly Conducting Cylinder

The perfectly conducting cylinder provides another useful validation of the code, particularly the side sections which are integrated using polar coordinates. The two dimensional circular shape is constructed by setting both `len` and `patches` to zero so that the two hemispherical side sections are in contact. Twice the `tpatches` parameter is then the total number of patches of the circle. The index of refraction is again set to a very large value to mimic a perfect conductor. The angle of incidence `thetadeg` is set to zero and the angular distribution of the scattered intensity is plotted over the entire 2π range.

The analytic Mie series solutions of the two dimensional perfectly conducting cylinder are well known. The solution using our notation and normalization convention for both polarizations is

$$f(\phi) = -4\sum_{n=0}^{\infty} \epsilon_n (-1)^n \frac{J_n(ka)}{H_n^{(1)}(ka)} \cos(n\phi) \qquad \text{(TM)} \tag{5.18}$$

$$f(\phi) = -4\sum_{n=0}^{\infty} \epsilon_n (-1)^n \frac{J_n'(ka)}{H_n^{(1)'}(ka)} \cos(n\phi) \qquad \text{(TE)} \tag{5.19}$$

where $\epsilon_n = 1$ if $n = 0$ and 2 otherwise [27]. Figures 5.4 and 5.5 show TM comparisons, while Figures 5.6 and 5.7 show TE comparisons. There is excellent quantitative agreement between the analytic predictions and the computed Nystrom solutions.

5.3 Dielectric Cylinder

Up to this point, only the extreme limit of a perfect conductor has been verified. The index of refraction for a thin film in the x-ray region however is typically much closer to that of vacuum. It becomes necessary to check the code against known solutions

www.manaraa.com

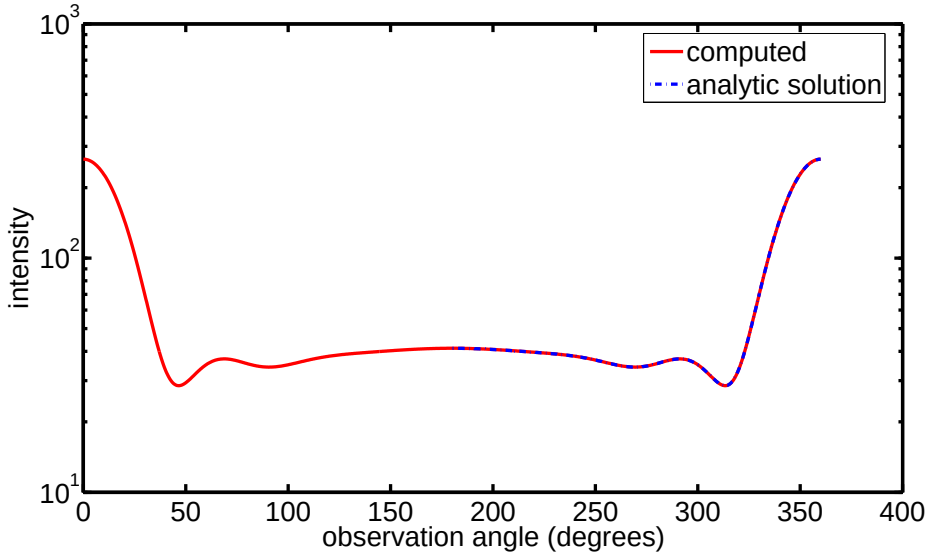

Figure 5.4 Comparison of computed solution with analytic solution for a cylindrical perfect conductor. `len` = 0, `patches` = 0, `t` = 1, `tpatches` = 10, `n` = 10^{10}, `k` = 0, and `thetadeg` = 0. The polarization is TM. Note the logarithmic y-axis scale.

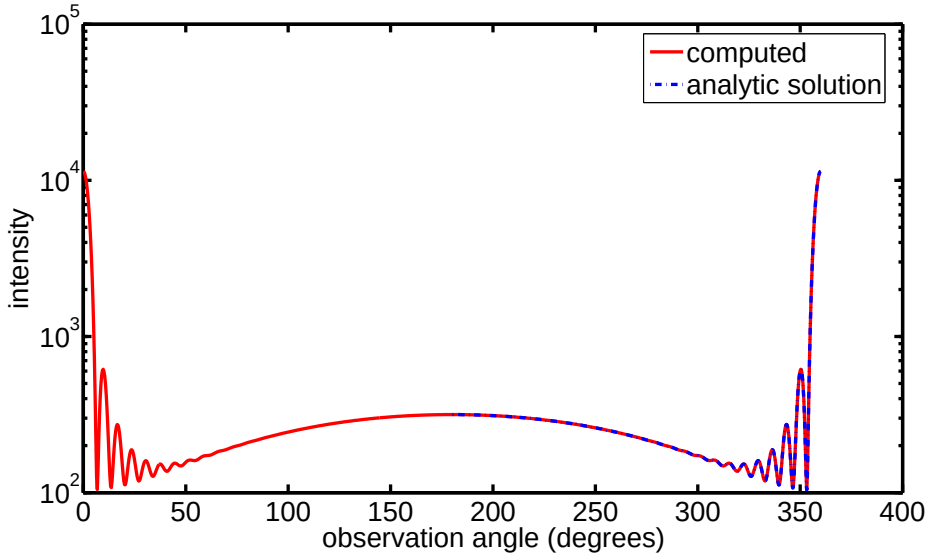

Figure 5.5 Comparison of computed solution with analytic solution for a cylindrical perfect conductor. `len` = 0, `patches` = 0, `t` = 8, `tpatches` = 50, `n` = 10^{10}, `k` = 0, and `thetadeg` = 0. The polarization is TM. Note the logarithmic y-axis scale.

www.manaraa.com

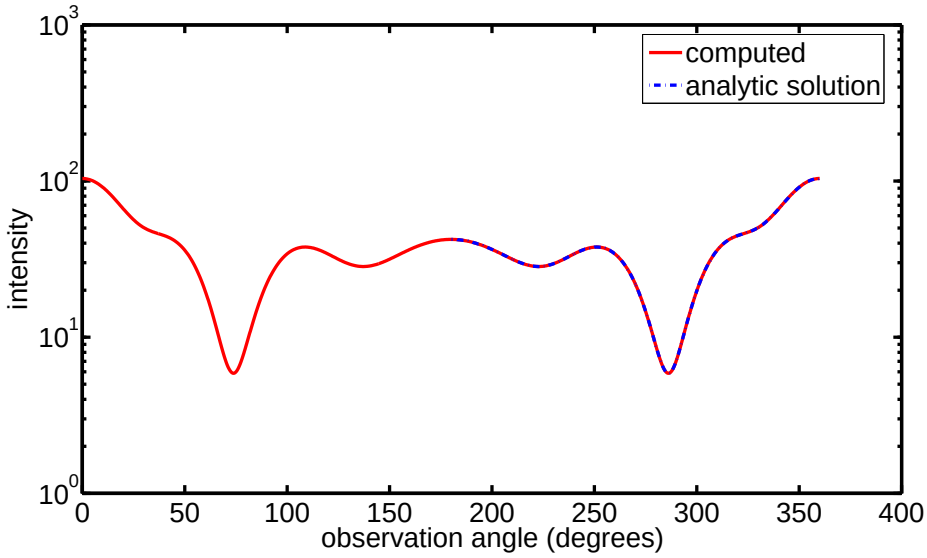

Figure 5.6 Comparison of computed solution with analytic solution for a cylindrical perfect conductor. `len` = 0, `patches` = 0, `t` = 1, `tpatches` = 10, `n` = 10^{10}, `k` = 0, and `thetadeg` = 0. The polarization is TE. Note the logarithmic y-axis scale.

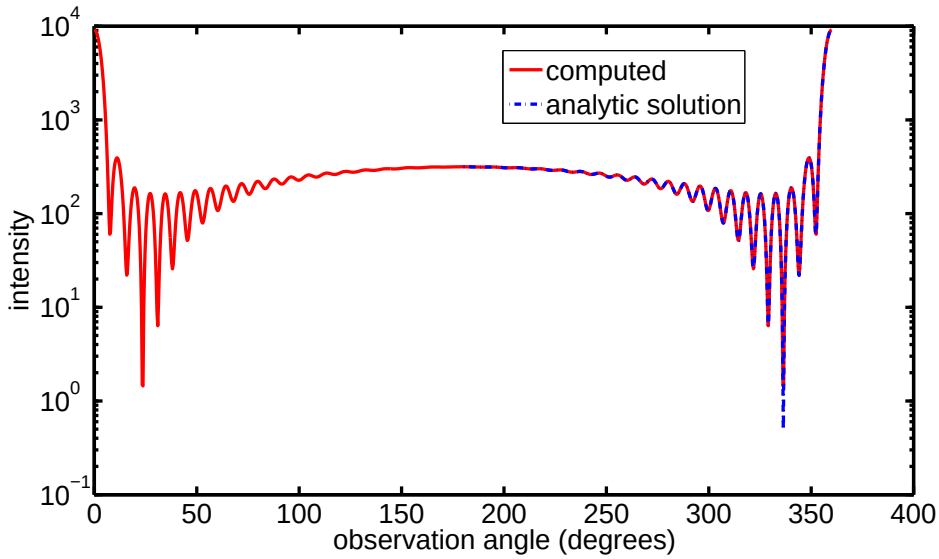

Figure 5.7 Comparison of computed solution with analytic solution for a cylindrical perfect conductor. `len` = 0, `patches` = 0, `t` = 8, `tpatches` = 50, `n` = 10^{10}, `k` = 0, and `thetadeg` = 0. The polarization is TE. Note the logarithmic y-axis scale.

www.manaraa.com

tpatches	$\phi = 180°$
10	-1.8420
20	-1.8423
40	-1.8425
80	-1.8425
160	-1.8426
exact	-1.8426

Figure 5.8 Comparison of computed and exact TM scattering cross section at $\phi = 180°$ of a homogeneous circular dielectric cylinder with a circumference of 0.5137λ and $\epsilon_r = 10$.

of a dielectric of arbitrary index of refraction. The scattering cross section is defined as

$$\sigma(\phi, \phi^{inc}) = \lim_{\rho \to \infty} 2\pi\rho \frac{|E_z^s(\rho, \phi)|^2}{|E_z^{inc}(0,0)|^2} \tag{5.20}$$

where the units of this quantity are given in wavelengths. Applying our normalization convention, this takes the form

$$\sigma(\phi) = \frac{|f(\phi)|^2}{8\pi} \tag{5.21}$$

Comparison data obtained from the literature [19] is given in terms of decibels, which is defined as

$$\sigma_{dB} = 10 \log_{10} \sigma \tag{5.22}$$

So, substituting (5.21) into (5.22) produces the quantity we will use for comparison.

$$\sigma_{dB}(\phi) = 10 \log_{10} \frac{|f(\phi)|^2}{8\pi} \tag{5.23}$$

Figures 5.8-5.10 show comparisons of exact [19] and computed scattering cross sections as expressed in (5.23). The computed row with the highest supposed accuracy, `tpatches` = 160, matches perfectly with the exact solution to the precision

www.manaraa.com

tpatches	$\phi = 0°$	$\phi = 30°$	$\phi = 60°$	$\phi = 90°$	$\phi = 120°$	$\phi = 150°$	$\phi = 180°$
10	5.5793	2.3621	-5.9468	-7.9133	-15.0969	-8.8516	-6.4536
20	5.5782	2.3624	-5.9418	-7.9203	-15.1078	-8.8509	-6.454
40	5.5776	2.3625	-5.9391	-7.9238	-15.1132	-8.8504	-6.454
80	5.5772	2.3625	-5.9378	-7.9255	-15.1159	-8.8501	-6.4541
160	5.5771	2.3625	-5.9371	-7.9264	-15.1172	-8.8499	-6.4541
exact	5.58	2.36	-5.94	-7.94	-15.12	-8.85	-6.45

Figure 5.9 Comparison of computed and exact TE scattering cross section at multiple ϕ angles of a homogeneous circular dielectric cylinder with a circumference of 2λ and $\epsilon_r = 2.56 + 2.56i$.

tpatches	$\phi = 0°$	$\phi = 30°$	$\phi = 60°$	$\phi = 90°$	$\phi = 120°$	$\phi = 150°$	$\phi = 180°$
10	-22.3142	-23.6361	-28.6524	-37.7197	-26.6924	-22.5859	-21.4615
20	-22.3154	-23.6375	-28.6543	-37.7118	-26.6902	-22.5846	-21.4604
40	-22.3161	-23.6382	-28.6554	-37.7079	-26.6890	-22.5840	-21.4599
80	-22.3164	-23.6385	-28.6559	-37.7059	-26.6885	-22.5837	-21.4596
160	-22.3165	-23.6387	-28.6561	-37.7049	-26.6882	-22.5835	-21.4595
exact	-22.32	-23.64	-28.66	-37.70	-26.69	-22.58	-21.46

Figure 5.10 Comparison of computed and exact TE scattering cross section at multiple ϕ angles of a homogeneous circular dielectric cylinder with a circumference of 0.248λ and $\epsilon_r = 2 + 50i$.

www.manaraa.com

given in the literature. The one exception to this occurs in Figure 5.9. The rounded computed value at $\phi = 90°$ is still 0.01 greater than the exact value, but is converging towards it as `tpatches` is increased. There is no cause for alarm because of the small nature of the error. If `tpatches` were to be further increased, it appears as though the program would converge to the exact solution.

5.4 Fresnel Coefficients

Perhaps the most convincing validation of the code is a comparison with Fresnel theory. (see section 1.2.2) A convenient program (`rt.m`) was written in `Matlab` by R. Steven Turley and Amy Baker which predicts the reflection and transmission of a specified multilayer stack at a given angle. This program uses the Fresnel coefficients, and thus drags along the associated approximations. The Fresnel model assumes an infinitely long interface so that the reflected wave only has a component at the specular angle [6]. The nonspecular reflection and edge effects associated with our finite surface are not present. This notwithstanding, if a method can be developed to compute the approximate reflectance $R = \frac{I}{I_0}$ of our scatterer, then a comparison with the Fresnel prediction is a powerful validation tool.

For a flat scatterer of length 50λ, the full width at half maximum (FWHM) of the specular peak is still relatively large. As roughness is introduced or the incident angle is changed, both the height and FWHM of the specular peak can visibly change. This seems to suggest that it is the area of this peak which is relevant for measuring reflection. Because a perfect conductor reflects 100% the light incident on it, the reflected specular peak at the same angle of the perfect conductor can be used as a normalization I_0.

Generalizing this technique, it is possible to form a crude estimate of reflectance

www.manaraa.com

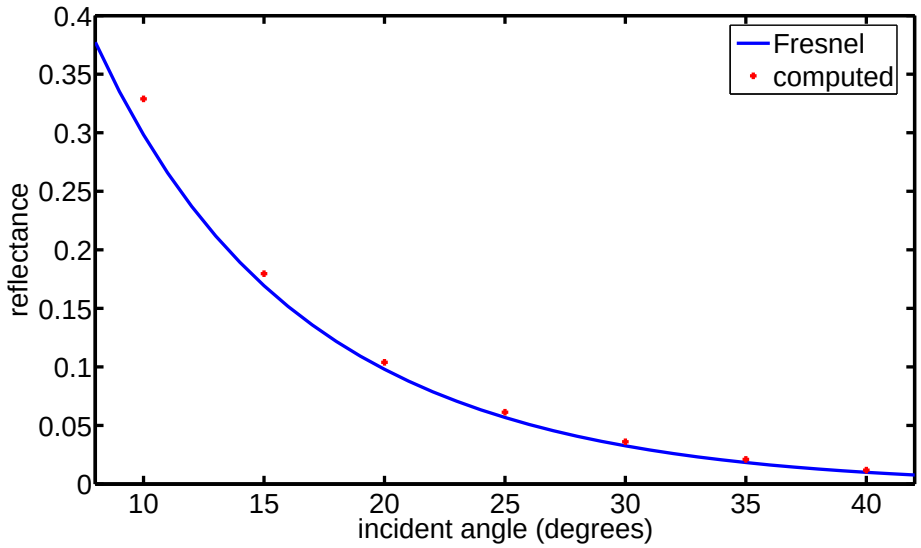

Figure 5.11 Comparison of TM computed reflectance and Fresnel model predicted reflectance as a function of incident angle from $\theta = 10°$ to $\theta = 40°$. `len` = 50, `patches` = 250, `t` = 0.5, `tpatches` = 10, `n` = 1.05, and `k` = 0.05.

by simply taking the ratio of the specular peak heights. With no roughness involved, the widths of the two peaks should be comparable. Reflectance is then approximately equal to the height of the dielectric's specular peak divided by the height of the perfect conductor's specular peak. The latter is easily determined by looking at the peak of the analytic physical optics solution for a perfect conductor, which has already been demonstrated to be in agreement with the code. (Section 5.1.2)

The agreement between the Fresnel prediction and the computed solution is quantitatively very good throughout most of the angular range. This provides yet another verification of the accuracy of the code if a sufficient number of patches are used. The only significant discrepancy occurs near $\theta = 90°$ where the computed reflection takes an unpredicted jump. In Figure 5.12, where the index of refraction is $n = 1.05$ and $k = 0.05$, the relative error is greater than an order of magnitude. The relative error is much less extreme in Figure 5.14, where $n = 0.8$ and $k = 0.2$. Even though the values are both very small, the error is significant. There is most likely nothing

www.manaraa.com

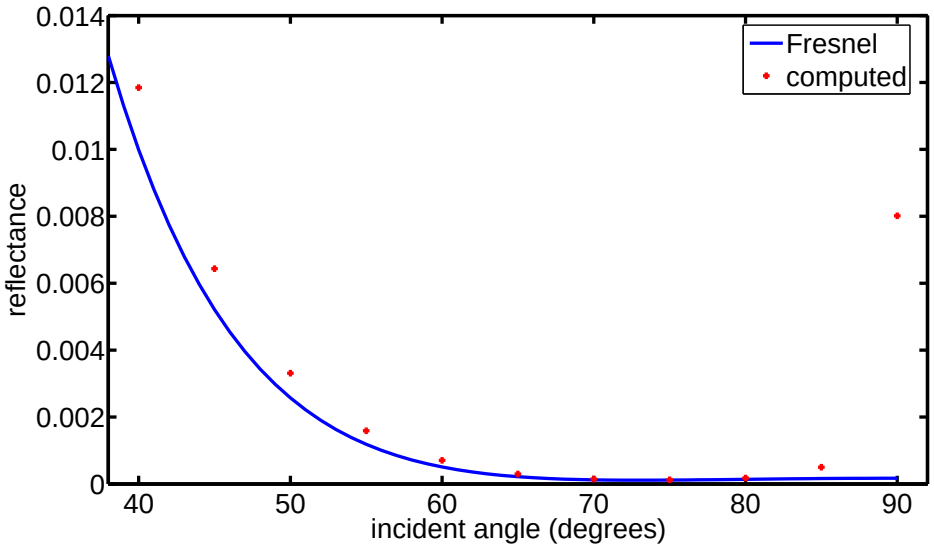

Figure 5.12 A continuation of the comparison of TM computed reflectance and Fresnel model predicted reflectance as a function of incident angle from $\theta = 40°$ to $\theta = 90°$. `len` $= 50$, `patches` $= 250$, `t` $= 0.5$, `tpatches` $= 10$, `n` $= 1.05$, and `k` $= 0.05$.

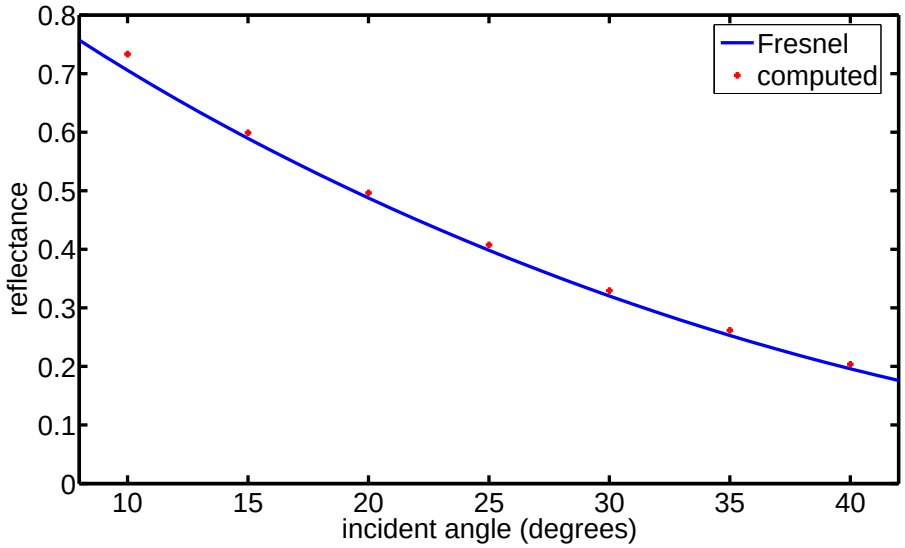

Figure 5.13 Comparison of TM computed reflectance and Fresnel model predicted reflectance as a function of incident angle from $\theta = 10°$ to $\theta = 40°$. `len` $= 50$, `patches` $= 250$, `t` $= 0.5$, `tpatches` $= 10$, `n` $= 0.8$, and `k` $= 0.2$.

www.manaraa.com

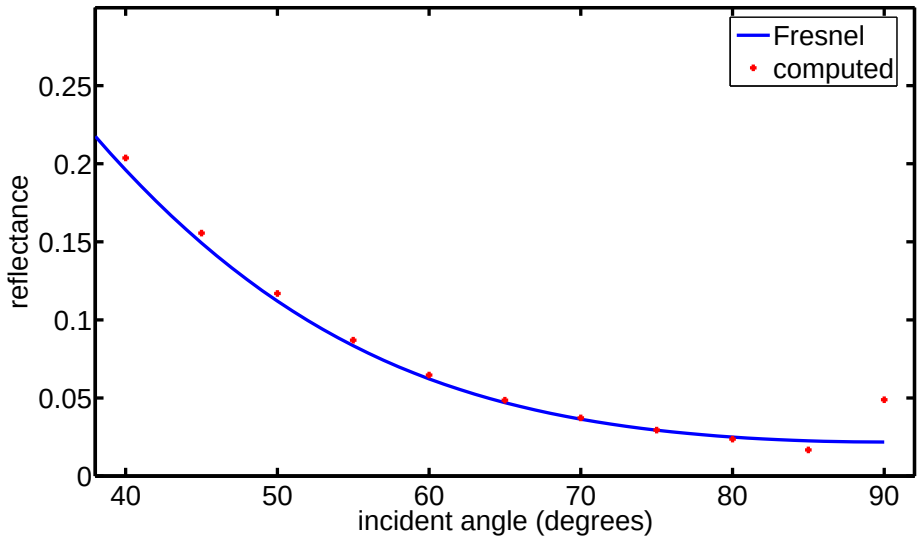

Figure 5.14 A continuation of the comparison of TM computed reflectance and Fresnel model predicted reflectance as a function of incident angle from $\theta = 40°$ to $\theta = 90°$. `len` $= 50$, `patches` $= 250$, `t` $= 0.5$, `tpatches` $= 10$, `n` $= 0.8$, and `k` $= 0.2$.

wrong with the calculation or convergence. At incident angles near 90°, the scattered wave solution when added to the incident wave will reproduce the correct total field outside the scatterer. In this specific instance, the scattered wave does not entirely correspond to our particular notion of the 'reflected beam.'

www.manaraa.com

Chapter 6

Results

Now that the program has been shown to work for known geometries, we can add different types of roughness and analyze the corresponding reflection. As a first experiment, the comparison between the predicted reflectance of a rough surface using the Nevot-Croce factor and its computed analog will be performed.

Figure 6.1 depicts the rough scatterer with index of refraction $n = 1.05$, $k = 0.05$. The smooth scatterer is identical except the top section is perfectly flat. The rough section was generated by setting `rfreq` $= 5$ and `sigma` $= 0.05$. The length of these long sections was set at `len` $= 20$, shorter than the surfaces constructed in Section 5.4. Note that the scale of the y-axis is much different than the x-axis scale in Figure 6.1. This has been done to more clearly see the shape of the top rough section.

Figure 6.2 plots the reflected angular intensity distribution of the two surfaces as a function of the observation angle. The width of both specular peaks is large because the scatterer is only 20λ long. The maximum intensity of the specular peak is significantly smaller for the rough surface as expected. The other nonspecular side peaks, however, have roughly the same amplitude and shape. This suggests that the total reflected power from the rough surface has been both diminished and scattered

www.manaraa.com

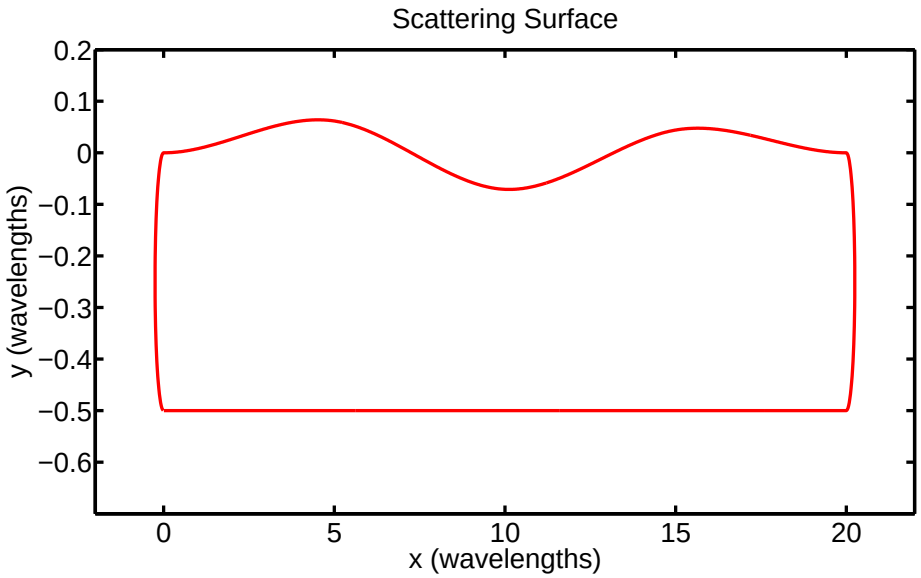

Figure 6.1 The rough scattering surface whose reflectance is compared to the Nevot-Croce prediction. `len` = 20, `patches` = 250, `t` = 0.5, `tpatches` = 10, `n` = 1.05, `k` = 0.05, `rfreq` = 5, `rheighttop` = 0.05, `rheightbot` = 0, and `inpstatetop` = 44.

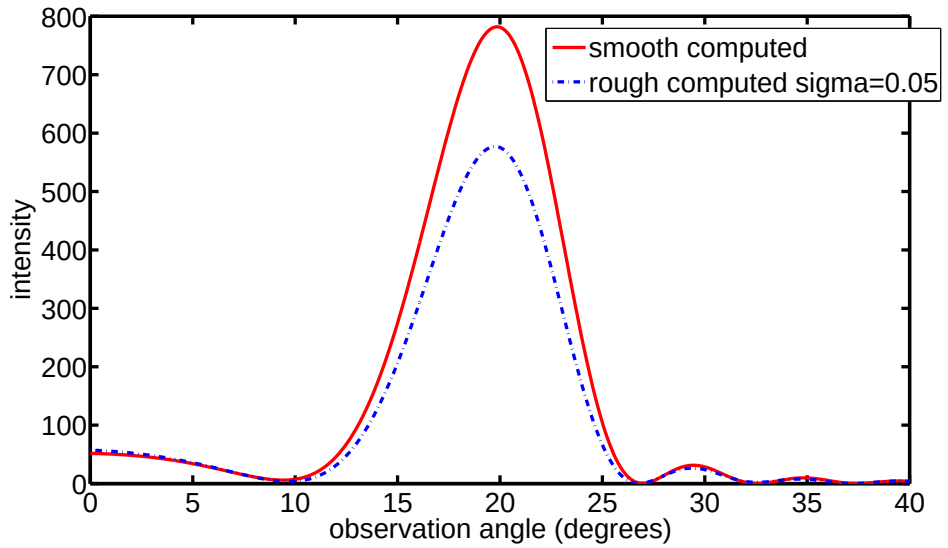

Figure 6.2 Comparison of TM computed angular intensity distribution of a rough and smooth surface. `len` = 20, `patches` = 250, `t` = 0.5, `tpatches` = 10, `n` = 1.05, and `k` = 0.05 for both surfaces. Additionally, `rfreq` = 5, `rheighttop` = 0.05, `rheightbot` = 0, and `inpstatetop` = 44 for the rough surface.

www.manaraa.com

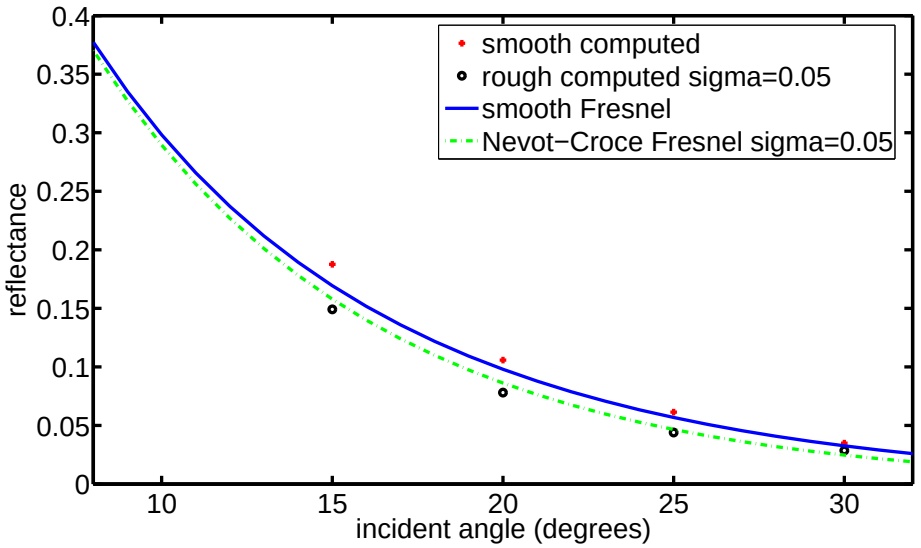

Figure 6.3 Comparison of TM computed reflectance and Fresnel model predicted reflectance modified by the Nevot-Croce factor as a function of incident angle from $\theta = 10°$ to $\theta = 30°$. `len` = 20, `patches` = 250, `t` = 0.5, `tpatches` = 10, `n` = 1.05, `k` = 0.05, `rfreq` = 5, `rheighttop` = 0.05, `rheightbot` = 0, and `inpstatetop` = 44.

into nonspecular angles.

Figure 6.3 presents a plot of reflectance vs. incident angle. The reflectance is calculated in the same manner as Section 5.4. The solid blue curve shows the predicted reflectance of the smooth scatterer. The red dots are the computed data points which are close to those in Figure 5.11. The discrepancy arises from the difference in length between the two scatterers. The black circles are the computed data points for the corresponding rough surface, while the green dash-dot line is the predicted value using the Nevot-Croce factor. The Nevot-Croce factor only modifies the Fresnel coefficient of the top interface when using the recursive Parratt formula because the bottom surface is perfectly smooth.

The data appears to be qualitatively consistent with the Nevot-Croce factor prediction. The smooth and rough data points are both monotonically decreasing functions of incident angle, with the rough points shifted down in intensity due to the

www.manaraa.com

roughness. This difference is on order of the predicted shift seen by the difference between the solid blue and dash-dot green lines. In this respect, the Nevot-Croce factor looks like it adequately estimates the approximate drop in reflectance. But, looking more carefully at the relative difference between the computed smooth and rough points, this quantity evolves from greater than to less than predicted. Furthermore, at $\theta = 15°$, the rough data point is below the predicted green line, while at $\theta = 30°$, it is greater than the prediction. This supports the idea that the Nevot-Croce factor doesn't capture the true decrease in reflectance as a function of incident angle.

While this data is a start, the scatterer we have used (Figure 6.1) is far from ideal for studying the Nevot-Croce factor. In particular, the frequency of the roughness is low compared with the length of the plate. The length is small enough that the reflectance is highly dependent on the particular random numbers used to generate the surface. In fact, only three non-zero points were used to create the surface. The reflectance can significantly change based on the surface these three random points define. In other words, the sample size is small enough that the averaging effects of a long surface do not take place. The Nevot-Croce factor is based on the sample size being large enough that random statistical phenomena are insignificant. I have attempted to create a surface which is representative of a typical undulating rough surface.

The length and amount of roughness of the scattering surface is limited by computing power. It was shown in Section 4.1 that the introduction of roughness into the model increases the number of necessary quadrature points. If the surface is too long, or if the roughness frequency or amplitude is too high, then the computed accuracy of the surface currents suffers. One symptom of this is rapid oscillations of the surface currents which is suppressed as the number of quadrature points is increased. Figure 6.1 represents a surface whose reflectance could be accurately modeled us-

www.manaraa.com

ing my present workstation. When the program is adapted to run on the Marylou supercomputing cluster at BYU, longer, more realistic surfaces will be studied.

Longer surfaces will narrow the width of the specular intensity peak. The peak width in this analysis is dominated by diffraction from the relatively short 20λ length. The expected change in the width of this peak due to roughness will be more apparent when the peak is very narrow to begin with. Thus, the current approximation of taking a simple ratio of the peak heights to determine reflection could need modification. Because the reflected wave is smeared out over an angular range, the definition of what angles corresponds to 'reflection' will need to be addressed.

www.manaraa.com

www.manaraa.com

Chapter 7

Conclusions

Starting from Maxwell's Equations, surface integral equations were developed to simulate the scattering of electromagnetic radiation from two-dimensional closed bodies. It was shown that the scatterers could be created in multilayer mirror-like forms. This allowed us to specify the parameters of the mirror and analyze the associated reflection.

The source code used for calculating reflectance has been shown to correctly predict the scattered angular intensity of several common test cases. The scattering data obtained from both conductors and dielectrics agreed well with analytic solutions and approximations.

Based on the limited amount of data which has been acquired, the Nevot-Croce correction factor appears to reasonably estimate the decrease in reflectance due to roughness. However, the data indicates that it does not manifest the correct angular behavior. If this is truly the case, it could be a possible explanation for the discrepancy between the measured data and the fits presented in Section 1.4.

In order to verify this preliminary conclusion, more runs with different randomly generated surfaces need to be acquired and analyzed. This would allow us to average

www.manaraa.com

over many surfaces to predict the average effects of roughness. Also, the length of the plate needs to be increased so that the roughness can be treated more as a general macroscopic property of the surface rather than being dependent on the particular surface. The tools are all in place to run the program on the Marylou supercomputing cluster at BYU. The reflectance of surfaces which are longer, rougher, and more representative of our actual mirrors will then be computed to yield more meaningful data. When this has been accomplished, the data will be fit to find an efficient formula which will be adapted to our optical constant fits. This will allow us to determine the optical constants of materials in the extreme ultraviolet much more accurately.

This research approaches the problem of multilayer roughness differently than others in the field have. We have solved for the reflected intensity of a two interface multilayer up to an arbitrary accuracy dependent upon the number of quadrature points used. The specific effects of RMS amplitude, spatial frequency, and correlation between surfaces can now be analyzed and merged to correct for the particular characteristics of a mirror. The prevailing techniques, pioneered by Stearns and others, rely on the use of approximations which are grossly violated in our experiments. Rather than estimating the specific effects of roughness from an analytic perspective, this code has been developed to extract the roughness corrections from the actual reflectance data itself.

www.manaraa.com

Bibliography

[1] G. L.-T. Chiu and J. M. Shaw, "Optical Lithography: Introduction," http://www.research.ibm.com/journal/rd/411/chiu.html (Accessed July 27, 2006).

[2] K. Takemoto, et. al., "Transmission x-ray microscopy with 50 nm resolution in Ritsumeikan synchrotron radiation center," X-ray Microscopy AIP conference proceedings 446-51 (2-6 Aug 1999).

[3] B. R. Sandel, et. al., "The Extreme Ultraviolet Imager Investigation for the IMAGE Mission," Space Science Reviews 91 (1-2), 197-242 (2000).

[4] D. D. Allred, M. B. Squires, R. S. Turley, W. Cash, and A. Shipley, "Highly reflective uranium mirrors for astrophysics applications," Proc. SPIE 4782, 212-223 (2002).

[5] S. Lunt, "Determining the indices of refraction of reactively sputtered uranium dioxide thin films from 46 to 584 angstroms," M.S. Thesis (Brigham Young University, Provo, UT, 2000).

[6] J. D. Jackson, *Classical Electrodynamics*, 3rd ed., Wiley, New York, 1998.

[7] D. Attwood, *Soft X-Rays and Extreme Ultraviolet Radiation*, Cambridge University Press, Cambridge, 1999.

www.manaraa.com

[8] N. Brimhall, "Thorium-Based Mirrors in the Extreme Ultraviolet," Honors Thesis (Brigham Young University, Provo, UT, 2005).

[9] V.G. Kohn, "On the theory of reflectivity by an x-ray multilayer mirror," Phys. Stat. Sol. 185(61), 61-70 (1995).

[10] L.G. Parratt, "Surface studies of solids by total reflection of x-rays," Physical Review 95 (2), 359-369 (1954).

[11] M. Ohring, *Materials Science of Thin Films* Academic Press, San Diego, 2001.

[12] V. Holy, J. Kubena, and I. Ohlidal, Phys. Rev. B 47, 23 (1993).

[13] L. Nevot, B. Pardo, and J. Corno, Revue Phys. Appl. 23, (1988).

[14] P. Debye, Verh. D. Deutsch. Phys. Ges. 15, 22 (1913).

[15] P. Croce and L. Nevot, J. De Physique Appliquee 11, 5 (1976).

[16] D.G. Stearns "The Scattering of x rays from nonideal multilayer structures," J. Appl. Phys. 65(2), 491-506 (1988).

[17] J.J.H. Wang, *Generalized Moment Methods in Electromagnetics*, John Wiley & Sons, Inc., New York, 1991.

[18] R.S. Turley, "Scalar Physical Optics," Brigham Young University Internal Report. (2005)

[19] A.F. Peterson, et. al., *Computational Methods for Electromagnetics*, IEEE Press, New York, 1998.

[20] L.F. Canino, et. al., "Numerical Solution of the Helmholtz Equation in 2D and 3D Using a High-Order Nystrom Discretization," Journal of Computational Physics 146, 627-663 (1998).

www.manaraa.com

[21] W.H. Press, et. al., *Numerical Recipes in C: The Art of Scientific Computing* Cambridge University Press, Cambridge, 1992.

[22] R.S. Turley, "Path Integrals," Brigham Young University Internal Report. (2005)

[23] R.S. Turley, "Using the Nystrom Method to Solve the Scalar Electric Field Integral Equation," Brigham Young University Internal Report. (2005)

[24] P. MacDonald, R.S. Turley, Hughes Research Labs Internal Report.

[25] S. M. Wandzura, Hughes Research Labs Internal Report.

[26] S. M. Wandzura, "Scattering from Surfaces, Version 2.0," Hughes Research Labs Internal Report. (1995)

[27] J.J. Bowman, et. al., *Electromagnetic and Acoustic Scattering by Simple Shapes*, North-Holland Publishing Co., Amsterdam, 1969.

www.manaraa.com

www.manaraa.com

Appendix A

Matlab Source Code

A.1 Sample Run

This section contains a simple example of how to run the program. Suppose we want to look at the reflection from a scatterer which is rough at both the top and bottom interfaces. The scatterer is 100λ long and 2λ thick. We determine that we want to use 200 patches per long section of the scatterer and 10 patches per circular cap section. The complex index of refraction will be $\mathcal{N} = 1.05 + 0.05i$. The incident angle of the radiation will be 20° from grazing. Now, we decide parameters which will be used to model the rough sections. We would like the random points surface points which serve as the framework of the surface spline to be spaced by 5λ. We will shoot for the rms roughness of the top surface to be 0.03λ and for the bottom surface to be 0.02λ. Each time the run is performed, we would like to randomize the surfaces, and the roughness will be uncorrelated.

To perform this run, we first open the `.m` file `userinputs.m` and change the parameters accordingly. It should look something like this:

```
len = 100; % length of the straight sections of the scatterer
patches = 200; % num patches (4 pts each) for each straight section
t = 2; % thickness of the scatterer
tpatches = 10; % num patches (4 pts each) for each curved section
n = 1.05; % real part of complex index of refraction
beta = 0.05; % imaginary part of complex index of refraction
thetadeg = 20; % incident angle from grazing
rfreq = 5; % spacing per random surface point (units of wavelengths/point)
rheightbot = 0.02; % std dev of normal random number distribution (bot)
rheighttop = 0.03; % std dev of normal random number distribution (top)
fixsurfacebot = 2; % if 1, WON'T generate new random surface, else will
```


www.manaraa.com

```
inpstatebot = 25; % random generator seed (when fixsurface = 1)
fixsurfacetop = 2; % if 1, WON'T generate new random surface, else will
inpstatetop = 71; % random generator seed (when fixsurface = 1)
correlatedroughness = 2; %if 1, surfaces ARE CORRELATED. If 2, not.
thrange = 20; % range of theta scan, centered around thetadeg in degrees
thsteps = 1000; % number of points to evaluate theta scan at
```

Once these have been set, the `.m` file can be saved and closed. The run for either polarization is performed by calling:

`TM`; (s polarization)
`TE`; (p polarization)

A.2 `cartJ.m`

This function calls `llquadr.m` which performs the integrations of the weight integral found in (3.47).

```
% *** Specifies quadrature weight integration (cartesian J (TM case)) ***

function f = cartJ(a,b,sing,n,spp,dpp,offset,k) f =
llquadr(@(x)cartJfunc(x,n,sing,spp,dpp,offset,k),...
    linlogOrder(b-sing,n),sing,b)-...
    llquadr(@(x)cartJfunc(x,n,sing,spp,dpp,offset,k),...
    linlogOrder(sing-a,n),sing,a);
```

A.3 `cartJfunc.m`

This function specifies the integrand of (3.47).

```
% *** Specifies quadrature weight function (cartesian J (TM case)) ***

function y = cartJfunc(x, n, sing, spp, dpp, offset, k) y =
1./4.*besselh(0,k*sqrt((x-sing).^2+...
    (ppval(spp,(x+offset))-ppval(spp,(sing+offset))).^2))...
    .*x.^n.*sqrt(1+(ppval(dpp,(x+offset))).^2);
```

A.4 `cartK.m`

This function calls `llquadr.m` which performs the integrations of the weight integrals found in (3.48).

www.manaraa.com

```
% *** Specifies quadrature weight integration (cartesian K (TM case)) ***

function f = cartK(a,b,sing,n,spp,dpp,offset,k) f =
llquadr(@(x)cartKfunc(x,n,sing,spp,dpp,offset,k),...
    linlogOrder(b-sing,n),sing,b)-...
    llquadr(@(x)cartKfunc(x,n,sing,spp,dpp,offset,k),...
    linlogOrder(sing-a,n),sing,a);
```

A.5 cartKfunc.m

This function specifies the integrand of (3.48).

```
% *** Specifies quadrature weight function (cartesian K (TM case)) ***

function y = cartKfunc(x, n, sing, spp, dpp, offset, k) y =
i./4.*besselh(1,k*sqrt((x-sing).^2+...
    (ppval(spp,(sing+offset))-ppval(spp,(x+offset))).^2))...
    .*x.^n.*sqrt(1+(ppval(dpp,(x+offset))).^2)...
    ./sqrt((x-sing).^2+(ppval(spp,(sing+offset))-...
    ppval(spp,(x+offset))).^2).*k...
    .*(cos(atan(ppval(dpp,(x+offset)))))...
    .*(ppval(spp,(sing+offset))-ppval(spp,(x+offset)))-...
    sin(atan(ppval(dpp,(x+offset)))).*(sing-x));
```

A.6 constants.m

```
% *** defines constants and index of refraction ***

c = 2.99792458E8;
mu0 = 4*pi*1E-7;
N = n + i*beta;
epsilon0 = 1/c^2/mu0;
epsilon = N^2/c^2/mu0;
omega = 2*pi*c;
k0 = omega*sqrt(epsilon0*mu0);
k = omega*sqrt(epsilon*mu0);
eta0 = sqrt(mu0/epsilon0);
eta = sqrt(mu0/epsilon);
```

A.7 dsurface.m

```
% *** Computes the derivative of a piecewise polynomial ***
```

www.manaraa.com

```
% The function returns another piecewise polynomial with the
% derivative.

function dpp = dsurface(pp) [breaks coefs l] = unmkpp(pp); for i=1:l
    dcoefs(i,:)=polyder(coefs(i,:));
end dpp = mkpp(breaks, dcoefs);
```

A.8 FFTsurf.m

```
% *** computes the power spectral density of both surfaces ***

numpts = 8196;
tau = len/numpts;
xf = 0:tau:len;
yftop = ppval(spptop,xf);
yfbot = ppval(sppbot,xf);
amptop = fft(yftop);
ampbot = fft(yfbot);
Ptop = abs(amptop).^2;
Pbot = abs(ampbot).^2;
df = 1/(numpts*tau);
f = 0:df:1/tau;
figure plot(f,Pbot,'c-');
xlabel('f (inverse wavelengths)')
ylabel('P(f)')
title('Power spectrum bottom surface')
axis([tau max(f)/2,0 max(Pbot(2:length(Pbot)))]);
figure plot(f,Ptop,'c-');
xlabel('f (inverse wavelengths)')
ylabel('P(f)') title('Power spectrum top surface')
axis([tau max(f)/2,0 max(Ptop(2:length(Ptop)))]);
```

A.9 lin_log_weights.m

This function determines the quadrature zeros and weights for the integrals of the form of (3.18).

```
% *** Returns a matrix with quadrature zeros and weights... ***
% for integrating polynomials or polynomials times
% logs exactly.  Results are exact for polynomials
% up to order (order-1)
%
```

www.manaraa.com

```
% This function assumes these weights have already
% been loaded into the matrix llquadzw by the
% commands:
% global llquadzw
% load /(directory)/llquadzw.txt
% This permits multiple calls without having
% to reload this each time.

function zw = lin_log_weights(order)
global llquadzw
offset = (order)*(order-1)/2 + 1;
zw = llquadzw(offset:offset+order-1,:);
return
switch order
    case (1)
        zw=[0.3678794411714423216 1];
    case (2)
        zw=[0.08829686513765301176 0.298499893705524914708;...
            0.6751864909098872010 4 0.70150010629447508529];
    case (3)
        zw=[0.028811662530951831 1743, 0.103330707964928646769;...
            0.304063729612137652611, 0.45463652597009870884;...
            0.81166922534407811686, 0.44203276606497264439];
    case (4)
        zw=[0.011802590997844918 2649, 0.043391028778414391102;...
            0.14282567997748369513 7, 0.240452097659460675978;...
            0.48920152265457447872, 0.42140345225977593198;...
            0.87867997406918370281, 0.294753421302349000941];
    case (5)
        zw=[0.005652228205080097 1359, 0.021046945791854629119;...
            0.07343037174265227340 6, 0.130705540744446697591;...
            0.28495740446255815371 5, 0.289702301671314156842;...
            0.61948226408477838141, 0.35022037012039871029;...
            0.91575808300469833378, 0.208324841671985806163];
end
% This could also be done with cell arrays
% example a{1}=[0.37 1]
% a{2}=[0.09 .298; 0.675 .702]
% etc.
```

www.manaraa.com

A.10 `linlogOrder.m`

This function determines the order of the linlog integration necessary to integrate (??) to a certain accuracy.

```
% *** Returns the order to user for 7 digit precision in linlog... ***
% quadratures (llquadr) for a patch of the given length with
% a monomial of the given order.

function ord = linlogOrder(length, order)
ord = ceil(4.15+0.53*length+0.43*order);
% In general, this overestimates the order a little
```

A.11 `llquad.m`

```
% *** Calling syntax: llquad(@func, order)... ***
% integrates func using a order-order rule on
% the interval 0..1.  It can have a log singularity
% at 0.

function r = llquad(func, order)
zw = lin_log_weights(order);
r = sum(func(zw(:,1)).*zw(:,2));
```

A.12 `llquadr.m`

This function is called to perform perform the linlog integration of (3.18).

```
% *** Calling syntax: llquadr(@func, order)... ***
% integrates func using a order-order rule on
% the interval a..b.  It can have a log singularity
% at a.

function r = llquadr(func, order, a, b)
zw = lin_log_weights(order);
dx = b-a;
r = dx*sum(func(a+dx*zw(:,1)).*zw(:,2));
```

A.13 `makesurface.m`

This function constructs the rough sections of the scatterer as described in Section 3.5.2.

www.manaraa.com

```
% *** Constructs a surface described by a piece-wise cubic... ***
% polynomial with random heights.
% Parameters:
%   length:  length of surface in wavelengths
%   spacing: spacing between random points
%   sigma:  rms deviation of surface from 0 (gaussian noise)
%   fixsurface: see user inputs
%   inpstate: seed of random generator
%   t: mean of distribution
% Return values
%   surfpp: piecewise cubic interpolating polynomial for surface
%   y: random height values for each surface point x

function [surfpp,y] = makesurface(length, spacing,...
    sigma, fixsurface, inpstate, t)
if fixsurface == 1
    state = inpstate;
    randn('state', state);
end x = 0+spacing:spacing:length-spacing; y =
normrnd(t,sigma,size(x));
surfpp = spline([0 x length],[0 t y t 0]);
```

A.14 nystromconstants.m

The constants including the numbers 1 and 2 in the variable name are a combination of the quadrature weights found in (3.8) - (3.11) and the factors of $\frac{1}{4}$ found in (3.40) and (3.41). The constants including 3 and 4 are only the quadrature weights.

```
% *** constants used in solving Nystrom problem ***

C1 = 13*dx/48; % flat sections constants
C2 = 11*dx/48;
C3 = 13*dx/12;
C4 = 11*dx/12;
C1t = 13*dtheta*r/48; % rounded section constants
C2t = 11*dtheta*r/48;
C3t = 13*dtheta*r/12;
C4t = 11*dtheta*r/12;
```

www.manaraa.com

A.15 `polJ.m`

This function calls `llquadr.m` which performs the polar coordinate analog of the integrations of the weight integral found in (3.47).

```
% *** Specifies quadrature weight integration (polar J (TM case)) ***

function f = polJ(a,b,sing,n,offset,k,r)
f = llquadr(@(x)polJfunc(x,n,sing,offset,k,r),...
    linlogOrder(b-sing,n),sing,b)-...
    llquadr(@(x)polJfunc(x,n,sing,offset,k,r),...
    linlogOrder(sing-a,n),sing,a);
```

A.16 `polJfunc.m`

This function specifies the polar coordinate analog of the integrand of (3.47).

```
% *** Specifies quadrature weight function (cartesian J (TM case)) ***

function y = polJfunc(x, n, sing, offset, k, r)
y = r./4.*besselh(0,k*sqrt((r*cos(x+offset)-r*cos(sing+offset)).^2+...
    (r*sin(x+offset)-r*sin(sing+offset)).^2)).*x.^n;
```

A.17 `polK.m`

This function calls `llquadr.m` which performs the polar coordinate analog of the integrations of the weight integral found in (3.48).

```
% *** Specifies quadrature weight integration (polar K (TM case)) ***

function f = polK(a,b,sing,n,offset,k,r)
f = llquadr(@(x)polKfunc(x,n,sing,offset,k,r),...
    linlogOrder(b-sing,n),sing,b)-...
    llquadr(@(x)polKfunc(x,n,sing,offset,k,r),...
    linlogOrder(sing-a,n),sing,a);
```

A.18 `polKfunc.m`

This function specifies the polar coordinate analog of the integrand of (3.48).

```
% *** Specifies quadrature weight function (cartesian K (TM case)) ***
```

www.manaraa.com

```
function y = polKfunc(x, n, sing, offset, k, r)
y = i.*r./4.*besselh(1,k*sqrt((r*cos(x+offset)-r*cos(sing+offset)).^2+...
    (r*sin(x+offset)-r*sin(sing+offset)).^2)).*x.^n...
    ./sqrt((r*cos(x+offset)-r*cos(sing+offset)).^2+...
    (r*sin(x+offset)-r*sin(sing+offset)).^2).*k...
    .*(cos(x+pi/2+offset).*(r*sin(sing+offset)-r*sin(x+offset))...
    -sin(x+pi/2+offset).*(r*cos(sing+offset)-r*cos(x+offset)));
```

A.19 storedata.m

```
% *** stores the output angles and intensity in a text file ***

intensityout=[thevaldeg;intensity];
fid = fopen('intensityoutfile','w');
fprintf(fid,'%7.3f %24.20f \r\n',intensityout);
st = fclose(fid);
```

A.20 surfacesetup.m

This function is responsible for constructing the scatterer according to the information found in Sections 3.5 and 3.6.2. It calls `makesurface.m`.

```
% *** set up surface and quadrature points ***

pl = len/patches; % patch length = tot length div number of patches
dx = pl/4; % distance between quadrature points
thetar = thetadeg*pi/180; % switching to radians
if len ~= 0 % when scatterer is a circle (no flat sections)
    [sppbot,randybot] = makesurface(len,rfreq,rheightbot,fixsurfacebot,...
        inpstatebot,-t); % generates random piecewise spline surface (bot)
    dppbot = dsurface(sppbot); % computes p.w. dy/dx of surface (bot)
end
x = (.5+(0:((4*patches)-1)))*dx; % array of evenly spaced quad points
if len ~= 0 % when scatterer is a circle (no flat sections)
    ybot = ppval(sppbot,x); % y values for each x quadrature point (bot)
    ypbot = ppval(dppbot,x); % dy/dx at each quadrature point (bot)
    randn('state',sum(100*clock)); % initializes rand number distribution
    if correlatedroughness ~= 1 % if surfaces aren't correlated,...
        % then mean of top distribution is centered at ideal interface
        randybot = -t;
    end
    [spptop,randytop] = makesurface(len,rfreq,rheighttop,fixsurfacetop,...
```

www.manaraa.com

```
        inpstatetop,randybot+t); % generates top surface
    dpptop = dsurface(spptop); % generates p.w. dy/dx of surface (top)
    ytop = fliplr(ppval(spptop,x)); % y values for each x quad point (top)
    yptop = fliplr(ppval(dpptop,x)); % dy/dx at each quad point (top)
end
if len ~= 0 % when scatterer is a circle (no flat sections)
    stermtop = sqrt(1+yptop.^2); % path integral Jacobian (top)
    stermbot = sqrt(1+ypbot.^2); % path integral Jacobian (bot)
end
r=t/2; % radius is half of scatterer thickness
cir=pi*t; % half circle circumference
tl = pi/tpatches; % theta length = pi / theta patches
dtheta = tl/4; % delta theta between each quadrature point on sides
theta = (.5+(0:((4*tpatches)-1)))*dtheta; % setup for side (theta) quad
thetaa = theta-pi/2; % right side points (theta)
thetab = theta+pi/2; % left side points (theta)
xr = r*cos(thetaa)+len; % x values for right side
yr = r*sin(thetaa)-r; % y values for right side
xl = r*cos(thetab); % x values for left side
yl = r*sin(thetab)-r; % y values for left side
for bb=1:4*patches % these loops create an array for the theta values...
    % (theta = 0 for flat sections, not included in calculation)
    theta(bb)=0;
end
for bb=1:4*tpatches
    theta(4*patches+bb)=thetaa(bb);
end
for bb=1:4*patches
    theta(4*(patches+tpatches)+bb)=0;
end
for bb=1:4*tpatches
    theta(4*(2*patches+tpatches)+bb)=thetab(bb);
end
x1=x; % storing the initial x values before x is altered
x = [x xr len-x xl]; % x values for entire scattering surface
if len ~= 0 % when scatterer is a circle (no flat sections)
    y = [ybot yr ytop yl]; % y values for entire scattering surface
    yp = [ypbot zeros(1,4*tpatches) yptop zeros(1,4*tpatches)]; % dy/dx ...
        % at each quadrature point (0 = side sections)
    sterm = [stermbot zeros(1,4*tpatches) stermtop zeros(1,4*tpatches)];
        % surface term for flat sections only
else % when scatterer is a circle (no flat sections)
    y = [yr yl];
```

www.manaraa.com

```
end
figure % plots the surface
plot(x,y,'r-');
axis equal;
title('Scattering Surface');
xlabel('x (wavelengths)');
ylabel('y (wavelengths)');
```

A.21 TE.m

This file is called to execute a run for TE polarization. It references all the other subfunctions which are called.

```
function TE % function so outputs and inputs can be specified by user
t0=cputime; % marks time at start of program
global llquadzw;
load llquadzw.txt; % weights and quadrature points for the linlog int

userinputs; % contains program parameters specified by user
constants; % defines constants
surfacesetup; % constructs scattering surfaces and quadrature points
% FFTsurf; % computes and displays power spectral density
nystromconstants; % defines constants used in filling and solving matricies
t1=cputime;
TEnystromfill; % fills Nystrom matrix
t2=cputime;
TEsolvematrix; % solves matrix equation currents / defines inc. beam
t3=cputime;
TEfarfield; % solves for far field intensity as a function of angle
t4=cputime;
% TEcylcon; % analytic solution cylinder (change thevaldeg in far field)
% storedata; % stores the data in an output text file
t5=cputime;
timeinfo; % displays the elapsed time information
```

A.22 TEcylcon.m

This routine plots (5.19).

```
% *** Analytic solution for TE cylindrical conductor ***

a = r;
```

www.manaraa.com

```
f0=besselj(1,2*pi*a)/besselh(1,2*pi*a);
an = 0:pi/1800:pi;
for ss = 1:length(an)
    for mm = 1:100
        fsum(mm)=(-1).^mm.*2*cos(mm.*an(ss)).*...
            (-besselj(mm+1,2*pi*a)+mm/2/pi/a.*besselj(mm,2*pi*a))./...
            (-besselh(mm+1,2*pi*a)+mm/2/pi/a.*besselh(mm,2*pi*a));
    end
    ftheta(ss)=-4*(f0+sum(fsum)); % factor of 4 makes...
        % normalization consistent w/ our notation
end
andeg = an*180/pi+180-thetadeg;
fintensity = abs(ftheta).^2;
figure
plot(thevaldeg,intensity,'r-',andeg,fintensity,'g-');
xlabel('observation angle (degrees)');
ylabel('intensity');
title(['Plane wave at ',num2str(thetadeg),' degrees from grazing.']);
legend('intensity','analytic solution TE cylinder conductor)');
```

A.23 `TEfarfield.m`

This `.m` file calculates the TE analog of (3.73), which contains (3.69) and (3.70), for many observation angles.

```
% *** solving for the far field amplitude as a function of angle ***

% commented line is for viewing 360 degree view of scattered intensity
% active when looking at analytic cylinder solution

if len == 0 % far field theta range (for circle & flat scatters)
    thevaldeg = 0:360/thsteps:360;
else
    thevaldeg = thetadeg-thrange/2:thrange/thsteps:thetadeg+thrange/2;
end
thevalr = thevaldeg*pi/180;

for v=1:thsteps+1
    for q=1:patches
        pvq = 4*(q-1);
        B(v,pvq+1)=C3.*sterm(pvq+1).*...
            exp(-i*k0*(x(pvq+1)*cos(thevalr(v))+...
            y(pvq+1)*sin(thevalr(v)))).*...
```

www.manaraa.com

```
            (1/eta0*K(pvq+1)-J(pvq+1).*...
            (sin(thevalr(v)).*cos(atan(yp(pvq+1)))-...
            cos(thevalr(v)).*sin(atan(yp(pvq+1)))));
        B(v,pvq+2)=C4.*sterm(pvq+2).*...
            exp(-i*k0*(x(pvq+2)*cos(thevalr(v))+...
            y(pvq+2)*sin(thevalr(v)))).*...
            (1/eta0*K(pvq+2)-J(pvq+2).*...
            (sin(thevalr(v)).*cos(atan(yp(pvq+2)))-...
            cos(thevalr(v)).*sin(atan(yp(pvq+2)))));
        B(v,pvq+3)=C4.*sterm(pvq+3).*...
            exp(-i*k0*(x(pvq+3)*cos(thevalr(v))+...
            y(pvq+3)*sin(thevalr(v)))).*...
            (1/eta0*K(pvq+3)-J(pvq+3).*...
            (sin(thevalr(v)).*cos(atan(yp(pvq+3)))-...
            cos(thevalr(v)).*sin(atan(yp(pvq+3)))));
        B(v,pvq+4)=C3.*sterm(pvq+4).*...
            exp(-i*k0*(x(pvq+4)*cos(thevalr(v))+...
            y(pvq+4)*sin(thevalr(v)))).*...
            (1/eta0*K(pvq+4)-J(pvq+4).*...
            (sin(thevalr(v)).*cos(atan(yp(pvq+4)))-...
            cos(thevalr(v)).*sin(atan(yp(pvq+4)))));
    end
end

for v=1:thsteps+1
    for q=patches+1:patches+tpatches
        pvq = 4*(q-1);
        B(v,pvq+1)=C3t.*...
            exp(-i*k0*(x(pvq+1)*cos(thevalr(v))+...
            y(pvq+1)*sin(thevalr(v)))).*...
            (1/eta0*K(pvq+1)-J(pvq+1).*...
            (sin(thevalr(v)).*cos(theta(pvq+1)+pi/2)-...
            cos(thevalr(v)).*sin(theta(pvq+1)+pi/2)));
        B(v,pvq+2)=C4t.*...
            exp(-i*k0*(x(pvq+2)*cos(thevalr(v))+...
            y(pvq+2)*sin(thevalr(v)))).*...
            (1/eta0*K(pvq+2)-J(pvq+2).*...
            (sin(thevalr(v)).*cos(theta(pvq+2)+pi/2)-...
            cos(thevalr(v)).*sin(theta(pvq+2)+pi/2)));
        B(v,pvq+3)=C4t.*...
            exp(-i*k0*(x(pvq+3)*cos(thevalr(v))+...
            y(pvq+3)*sin(thevalr(v)))).*...
            (1/eta0*K(pvq+3)-J(pvq+3).*...
```

www.manaraa.com

```
            (sin(thevalr(v)).*cos(theta(pvq+3)+pi/2)-...
            cos(thevalr(v)).*sin(theta(pvq+3)+pi/2)));
        B(v,pvq+4)=C3t.*...
            exp(-i*k0*(x(pvq+4)*cos(thevalr(v))+...
            y(pvq+4)*sin(thevalr(v)))).*...
            (1/eta0*K(pvq+4)-J(pvq+4).*...
            (sin(thevalr(v)).*cos(theta(pvq+4)+pi/2)-...
            cos(thevalr(v)).*sin(theta(pvq+4)+pi/2)));
    end
end

for v=1:thsteps+1
    for q=patches+tpatches+1:2*patches+tpatches
        pvq = 4*(q-1);
        B(v,pvq+1)=-C3.*sterm(pvq+1).*...
            exp(-i*k0*(x(pvq+1)*cos(thevalr(v))+...
            y(pvq+1)*sin(thevalr(v)))).*...
            (1/eta0*K(pvq+1)-J(pvq+1).*...
            (sin(thevalr(v)).*cos(atan(yp(pvq+1)))-...
            cos(thevalr(v)).*sin(atan(yp(pvq+1)))));
        B(v,pvq+2)=-C4.*sterm(pvq+2).*...
            exp(-i*k0*(x(pvq+2)*cos(thevalr(v))+...
            y(pvq+2)*sin(thevalr(v)))).*...
            (1/eta0*K(pvq+2)-J(pvq+2).*...
            (sin(thevalr(v)).*cos(atan(yp(pvq+2)))-...
            cos(thevalr(v)).*sin(atan(yp(pvq+2)))));
        B(v,pvq+3)=-C4.*sterm(pvq+3).*...
            exp(-i*k0*(x(pvq+3)*cos(thevalr(v))+...
            y(pvq+3)*sin(thevalr(v)))).*...
            (1/eta0*K(pvq+3)-J(pvq+3).*...
            (sin(thevalr(v)).*cos(atan(yp(pvq+3)))-...
            cos(thevalr(v)).*sin(atan(yp(pvq+3)))));
        B(v,pvq+4)=-C3.*sterm(pvq+4).*...
            exp(-i*k0*(x(pvq+4)*cos(thevalr(v))+...
            y(pvq+4)*sin(thevalr(v)))).*...
            (1/eta0*K(pvq+4)-J(pvq+4).*...
            (sin(thevalr(v)).*cos(atan(yp(pvq+4)))-...
            cos(thevalr(v)).*sin(atan(yp(pvq+4)))));
    end
end

for v=1:thsteps+1
    for q=2*patches+tpatches+1:2*(patches+tpatches)
```

www.manaraa.com

```
        pvq = 4*(q-1);
        B(v,pvq+1)=C3t.*...
            exp(-i*k0*(x(pvq+1)*cos(thevalr(v))+...
            y(pvq+1)*sin(thevalr(v)))).*...
            (1/eta0*K(pvq+1)-J(pvq+1).*...
            (sin(thevalr(v)).*cos(theta(pvq+1)+pi/2)-...
            cos(thevalr(v)).*sin(theta(pvq+1)+pi/2)));
        B(v,pvq+2)=C4t.*...
            exp(-i*k0*(x(pvq+2)*cos(thevalr(v))+...
            y(pvq+2)*sin(thevalr(v)))).*...
            (1/eta0*K(pvq+2)-J(pvq+2).*...
            (sin(thevalr(v)).*cos(theta(pvq+2)+pi/2)-...
            cos(thevalr(v)).*sin(theta(pvq+2)+pi/2)));
        B(v,pvq+3)=C4t.*...
            exp(-i*k0*(x(pvq+3)*cos(thevalr(v))+...
            y(pvq+3)*sin(thevalr(v)))).*...
            (1/eta0*K(pvq+3)-J(pvq+3).*...
            (sin(thevalr(v)).*cos(theta(pvq+3)+pi/2)-...
            cos(thevalr(v)).*sin(theta(pvq+3)+pi/2)));
        B(v,pvq+4)=C3t.*...
            exp(-i*k0*(x(pvq+4)*cos(thevalr(v))+...
            y(pvq+4)*sin(thevalr(v)))).*...
            (1/eta0*K(pvq+4)-J(pvq+4).*...
            (sin(thevalr(v)).*cos(theta(pvq+4)+pi/2)-...
            cos(thevalr(v)).*sin(theta(pvq+4)+pi/2)));
    end
end

B=B.*i.*sqrt(i).*k0; % gets normalization and phase correct for field
field=sum(B.'); % computes far field amplitude
intensity = abs(field).^2; % computes far field intensity
figure
plot(thevaldeg,intensity,'r-');
xlabel('observation angle (degrees)');
ylabel('intensity');
title(['Plane wave at ',num2str(thetadeg),' degrees from grazing.']);
legend('intensity');
```

A.24 TEnystromfill.m

This .m file fills the Nystrom matrix of the TE analog of (3.46). It is broken up into patches which contain singularities and patches which don't. The patches which con-

www.manaraa.com

tain singularities use (3.47) and (3.48). The patches which don't contain singularities use (3.40) and (3.41).

```
% *** fills Nystrom matrix ***

% Bottom J outside
for j=1:(2*(patches+tpatches)) % looping through observation points
    pv = 4*(j-1);
    for l=1:patches % looping through the patches
        pvl = 4*(l-1);
        offset = (l-1)*pl;
        if(j==l) % if the observation point is on the patch,
                 % we need to use a quadrature which
                 % integrates over the green function
                 % (see numerical methods section 18.3)
            for m=1:4 % these different m's correspond to observation pts
                pvm = pv + m;
                singpt = x(m);
                W0 = cartK(0,pl,singpt,0,sppbot,dppbot,offset,k0);
                W1 = cartK(0,pl,singpt,1,sppbot,dppbot,offset,k0)./dx;
                W2 = cartK(0,pl,singpt,2,sppbot,dppbot,offset,k0)./dx.^2;
                W3 = cartK(0,pl,singpt,3,sppbot,dppbot,offset,k0)./dx.^3;
                A(pvm,pv+1)=-(13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(pvm,pv+2)=-(-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(pvm,pv+3)=-(2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(pvm,pv+4)=-(-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
                A(pvm,pvm)=A(pvm,pvm)-.5;
            end
        else
            for n=1:4 % now filling off diagonal blocks...
                pvn = pv + n;
                A(pvn,pvl+1)=-i*C1*sterm(pvl+1)*...
                    besselh(1,k0*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2))*k0/sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2).*((y(pvn)-y(pvl+1)).*...
                    cos(atan(yp(pvl+1)))-(x(pvn)-x(pvl+1)).*...
                    sin(atan(yp(pvl+1))));
                A(pvn,pvl+2)=-i*C2*sterm(pvl+2)*...
                    besselh(1,k0*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2))*k0/sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2).*((y(pvn)-y(pvl+2)).*...
                    cos(atan(yp(pvl+2)))-(x(pvn)-x(pvl+2)).*...
                    sin(atan(yp(pvl+2))));
```

www.manaraa.com

```
                A(pvn,pvl+3)=-i*C2*sterm(pvl+3)*...
                    besselh(1,k0*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2))*k0/sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2).*((y(pvn)-y(pvl+3)).*...
                    cos(atan(yp(pvl+3)))-(x(pvn)-x(pvl+3)).*...
                    sin(atan(yp(pvl+3))));
                A(pvn,pvl+4)=-i*C1*sterm(pvl+4)*...
                    besselh(1,k0*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2))*k0/sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2).*((y(pvn)-y(pvl+4)).*...
                    cos(atan(yp(pvl+4)))-(x(pvn)-x(pvl+4)).*...
                    sin(atan(yp(pvl+4))));
            end
        end
    end
end

bp = 8*(patches+tpatches);

% Bottom J inside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=1:patches
        pvl = 4*(l-1);
        offset = (l-1)*pl;
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = x(m);
                W0 = cartK(0,pl,singpt,0,sppbot,dppbot,offset,k);
                W1 = cartK(0,pl,singpt,1,sppbot,dppbot,offset,k)./dx;
                W2 = cartK(0,pl,singpt,2,sppbot,dppbot,offset,k)./dx.^2;
                W3 = cartK(0,pl,singpt,3,sppbot,dppbot,offset,k)./dx.^3;
                A(pvm+bp,pv+1)=-(13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(pvm+bp,pv+2)=-(-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(pvm+bp,pv+3)=-(2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(pvm+bp,pv+4)=-(-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
                A(pvm+bp,pvm)=A(pvm+bp,pvm)+.5;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(pvn+bp,pvl+1)=-i*C1*sterm(pvl+1)*...
```

www.manaraa.com

```
                    besselh(1,k*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2))*k/sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2).*((y(pvn)-y(pvl+1)).*...
                    cos(atan(yp(pvl+1)))-(x(pvn)-x(pvl+1)).*...
                    sin(atan(yp(pvl+1))));
                A(pvn+bp,pvl+2)=-i*C2*sterm(pvl+2)*...
                    besselh(1,k*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2))*k/sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2).*((y(pvn)-y(pvl+2)).*...
                    cos(atan(yp(pvl+2)))-(x(pvn)-x(pvl+2)).*...
                    sin(atan(yp(pvl+2))));
                A(pvn+bp,pvl+3)=-i*C2*sterm(pvl+3)*...
                    besselh(1,k*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2))*k/sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2).*((y(pvn)-y(pvl+3)).*...
                    cos(atan(yp(pvl+3)))-(x(pvn)-x(pvl+3)).*...
                    sin(atan(yp(pvl+3))));
                A(pvn+bp,pvl+4)=-i*C1*sterm(pvl+4)*...
                    besselh(1,k*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2))*k/sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2).*((y(pvn)-y(pvl+4)).*....
                    cos(atan(yp(pvl+4)))-(x(pvn)-x(pvl+4)).*...
                    sin(atan(yp(pvl+4))));
            end
        end
    end
end

% Bottom K Outside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=1:patches
        pvl = 4*(l-1);
        offset = (l-1)*pl;
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = x(m);
                W0 = cartJ(0,pl,singpt,0,sppbot,dppbot,offset,k0);
                W1 = cartJ(0,pl,singpt,1,sppbot,dppbot,offset,k0)./dx;
                W2 = cartJ(0,pl,singpt,2,sppbot,dppbot,offset,k0)./dx.^2;
                W3 = cartJ(0,pl,singpt,3,sppbot,dppbot,offset,k0)./dx.^3;
                A(pvm,bp+pv+1)=k0/eta0*...
```

www.manaraa.com

```
                    (13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(pvm,bp+pv+2)=k0/eta0*...
                    (-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(pvm,bp+pv+3)=k0/eta0*...
                    (2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(pvm,bp+pv+4)=k0/eta0*...
                    (-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(pvn,bp+pvl+1)=k0/eta0*C1*sterm(pvl+1)*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2));
                A(pvn,bp+pvl+2)=k0/eta0*C2*sterm(pvl+2)*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2));
                A(pvn,bp+pvl+3)=k0/eta0*C2*sterm(pvl+3)*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2));
                A(pvn,bp+pvl+4)=k0/eta0*C1*sterm(pvl+4)*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2));
            end
        end
    end
end

% Bottom K Inside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=1:patches
        pvl = 4*(l-1);
        offset = (l-1)*pl;
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = x(m);
                W0 = cartJ(0,pl,singpt,0,sppbot,dppbot,offset,k);
                W1 = cartJ(0,pl,singpt,1,sppbot,dppbot,offset,k)./dx;
                W2 = cartJ(0,pl,singpt,2,sppbot,dppbot,offset,k)./dx.^2;
                W3 = cartJ(0,pl,singpt,3,sppbot,dppbot,offset,k)./dx.^3;
                A(bp+pvm,bp+pv+1)=k/eta*...
```

www.manaraa.com

```
                    (13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(bp+pvm,bp+pv+2)=k/eta*...
                    (-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(bp+pvm,bp+pv+3)=k/eta*...
                    (2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(bp+pvm,bp+pv+4)=k/eta*...
                    (-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(bp+pvn,bp+pvl+1)=k/eta*C1*sterm(pvl+1)*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2));
                A(bp+pvn,bp+pvl+2)=k/eta*C2*sterm(pvl+2)*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2));
                A(bp+pvn,bp+pvl+3)=k/eta*C2*sterm(pvl+3)*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2));
                A(bp+pvn,bp+pvl+4)=k/eta*C1*sterm(pvl+4)*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2));
            end
        end
    end
end

% Right Semicircle J Outside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(patches+1):(patches+tpatches)
        pvl = 4*(l-1);
        offset=(l-patches-1)*tl-pi/2;
        if (j==l)
            for m=1:4
                pvm = pv + m;
                singpt = theta(4*patches+m)+pi/2;
                W0 = polK(0,tl,singpt,0,offset,k0,r);
                W1 = polK(0,tl,singpt,1,offset,k0,r)./dtheta;
                W2 = polK(0,tl,singpt,2,offset,k0,r)./dtheta.^2;
                W3 = polK(0,tl,singpt,3,offset,k0,r)./dtheta.^3;
                A(pvm,pv+1)=-(13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
```

www.manaraa.com

```
                A(pvm,pv+2)=-(-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(pvm,pv+3)=-(2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(pvm,pv+4)=-(-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
                A(pvm,pvm)=A(pvm,pvm)-.5;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(pvn,pvl+1)=-i*C1t*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+1)).^2+(y(pvn)-y(pvl+1)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+1)).^2+(y(pvn)-y(pvl+1)).^2).*...
                    ((y(pvn)-y(pvl+1)).*cos(theta(pvl+1)+pi/2)-...
                    (x(pvn)-x(pvl+1)).*sin(theta(pvl+1)+pi/2));
                A(pvn,pvl+2)=-i*C2t*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+2)).^2+(y(pvn)-y(pvl+2)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+2)).^2+(y(pvn)-y(pvl+2)).^2).*...
                    ((y(pvn)-y(pvl+2)).*cos(theta(pvl+2)+pi/2)-...
                    (x(pvn)-x(pvl+2)).*sin(theta(pvl+2)+pi/2));
                A(pvn,pvl+3)=-i*C2t*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+3)).^2+(y(pvn)-y(pvl+3)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+3)).^2+(y(pvn)-y(pvl+3)).^2).*...
                    ((y(pvn)-y(pvl+3)).*cos(theta(pvl+3)+pi/2)-...
                    (x(pvn)-x(pvl+3)).*sin(theta(pvl+3)+pi/2));
                A(pvn,pvl+4)=-i*C1t*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+4)).^2+(y(pvn)-y(pvl+4)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+4)).^2+(y(pvn)-y(pvl+4)).^2).*...
                    ((y(pvn)-y(pvl+4)).*cos(theta(pvl+4)+pi/2)-...
                    (x(pvn)-x(pvl+4)).*sin(theta(pvl+4)+pi/2));
            end
        end
    end
end

% Right Semicircle J Inside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(patches+1):(patches+tpatches)
        pvl = 4*(l-1);
        offset=(l-patches-1)*tl-pi/2;
        if (j==l)
            for m=1:4
                pvm = pv + m;
                singpt = theta(4*patches+m)+pi/2;
```

www.manaraa.com

```
                W0 = polK(0,tl,singpt,0,offset,k,r);
                W1 = polK(0,tl,singpt,1,offset,k,r)./dtheta;
                W2 = polK(0,tl,singpt,2,offset,k,r)./dtheta.^2;
                W3 = polK(0,tl,singpt,3,offset,k,r)./dtheta.^3;
                A(pvm+bp,pv+1)=-(13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(pvm+bp,pv+2)=-(-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(pvm+bp,pv+3)=-(2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(pvm+bp,pv+4)=-(-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
                A(pvm+bp,pvm)=A(pvm+bp,pvm)+.5;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(pvn+bp,pvl+1)=-i*C1t*besselh...
                    (1,k*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2))*k/sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2).*((y(pvn)-y(pvl+1)).*...
                    cos(theta(pvl+1)+pi/2)-(x(pvn)-x(pvl+1)).*...
                    sin(theta(pvl+1)+pi/2));
                A(pvn+bp,pvl+2)=-i*C2t*besselh...
                    (1,k*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2))*k/sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2).*((y(pvn)-y(pvl+2)).*...
                    cos(theta(pvl+2)+pi/2)-(x(pvn)-x(pvl+2)).*...
                    sin(theta(pvl+2)+pi/2));
                A(pvn+bp,pvl+3)=-i*C2t*besselh...
                    (1,k*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2))*k/sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2).*((y(pvn)-y(pvl+3)).*...
                    cos(theta(pvl+3)+pi/2)-(x(pvn)-x(pvl+3)).*...
                    sin(theta(pvl+3)+pi/2));
                A(pvn+bp,pvl+4)=-i*C1t*besselh...
                    (1,k*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2))*k/sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2).*((y(pvn)-y(pvl+4)).*...
                    cos(theta(pvl+4)+pi/2)-(x(pvn)-x(pvl+4)).*...
                    sin(theta(pvl+4)+pi/2));
            end
        end
    end
end

% Right Semicircle K Outside
```

www.manaraa.com

```
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(patches+1):(patches+tpatches)
        pvl = 4*(l-1);
        offset=(l-patches-1)*tl-pi/2;
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = theta(4*patches+m)+pi/2;
                W0 = polJ(0,tl,singpt,0,offset,k0,r);
                W1 = polJ(0,tl,singpt,1,offset,k0,r)./dtheta;
                W2 = polJ(0,tl,singpt,2,offset,k0,r)./dtheta.^2;
                W3 = polJ(0,tl,singpt,3,offset,k0,r)./dtheta.^3;
                A(pvm,bp+pv+1)=k0/eta0*...
                    (13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(pvm,bp+pv+2)=k0/eta0*...
                    (-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(pvm,bp+pv+3)=k0/eta0*...
                    (2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(pvm,bp+pv+4)=k0/eta0*...
                    (-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(pvn,bp+pvl+1)=k0/eta0*C1t*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2));
                A(pvn,bp+pvl+2)=k0/eta0*C2t*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2));
                A(pvn,bp+pvl+3)=k0/eta0*C2t*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2));
                A(pvn,bp+pvl+4)=k0/eta0*C1t*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2));
            end
        end
    end
end

% Right Semicircle K Inside
```

www.manaraa.com

```
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(patches+1):(patches+tpatches)
        pvl = 4*(l-1);
        offset=(l-patches-1)*tl-pi/2;
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = theta(4*patches+m)+pi/2;
                W0 = polJ(0,tl,singpt,0,offset,k,r);
                W1 = polJ(0,tl,singpt,1,offset,k,r)./dtheta;
                W2 = polJ(0,tl,singpt,2,offset,k,r)./dtheta.^2;
                W3 = polJ(0,tl,singpt,3,offset,k,r)./dtheta.^3;
                A(bp+pvm,bp+pv+1)=k/eta*...
                    (13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(bp+pvm,bp+pv+2)=k/eta*...
                    (-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(bp+pvm,bp+pv+3)=k/eta*...
                    (2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(bp+pvm,bp+pv+4)=k/eta*...
                    (-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(bp+pvn,bp+pvl+1)=k/eta*C1t*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2));
                A(bp+pvn,bp+pvl+2)=k/eta*C2t*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2));
                A(bp+pvn,bp+pvl+3)=k/eta*C2t*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2));
                A(bp+pvn,bp+pvl+4)=k/eta*C1t*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2));
            end
        end
    end
end

% Top J outside
```

www.manaraa.com

```
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(patches+tpatches+1):(2*patches+tpatches)
        pvl = 4*(l-1);
        offset = (2*patches+tpatches-l)*pl;
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = x(5-m);
                W0 = cartK(0,pl,singpt,0,spptop,dpptop,offset,k0);
                W1 = cartK(0,pl,singpt,1,spptop,dpptop,offset,k0)./dx;
                W2 = cartK(0,pl,singpt,2,spptop,dpptop,offset,k0)./dx.^2;
                W3 = cartK(0,pl,singpt,3,spptop,dpptop,offset,k0)./dx.^3;
                A(pvm,pv+1)=(13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(pvm,pv+2)=(-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(pvm,pv+3)=(2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(pvm,pv+4)=(-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
                A(pvm,pvm)=A(pvm,pvm)-.5;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(pvn,pvl+1)=i*C1*sterm(pvl+1)*...
                    besselh(1,k0*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2))*k0/sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2).*((y(pvn)-y(pvl+1)).*...
                    cos(atan(yp(pvl+1)))-(x(pvn)-x(pvl+1)).*...
                    sin(atan(yp(pvl+1))));
                A(pvn,pvl+2)=i*C2*sterm(pvl+2)*...
                    besselh(1,k0*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2))*k0/sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2).*((y(pvn)-y(pvl+2)).*...
                    cos(atan(yp(pvl+2)))-(x(pvn)-x(pvl+2)).*...
                    sin(atan(yp(pvl+2))));
                A(pvn,pvl+3)=i*C2*sterm(pvl+3)*...
                    besselh(1,k0*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2))*k0/sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2).*((y(pvn)-y(pvl+3)).*...
                    cos(atan(yp(pvl+3)))-(x(pvn)-x(pvl+3)).*...
                    sin(atan(yp(pvl+3))));
                A(pvn,pvl+4)=i*C1*sterm(pvl+4)*...
                    besselh(1,k0*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2))*k0/sqrt((x(pvn)-x(pvl+4)).^2+...
```

www.manaraa.com

```
                    (y(pvn)-y(pvl+4)).^2).*((y(pvn)-y(pvl+4)).*...
                    cos(atan(yp(pvl+4)))-(x(pvn)-x(pvl+4)).*...
                    sin(atan(yp(pvl+4))));
            end
        end
    end
end

% Top J Inside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(patches+tpatches+1):(2*patches+tpatches)
        pvl = 4*(l-1);
        offset = (2*patches+tpatches-l)*pl;
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = x(5-m);
                W0 = cartK(0,pl,singpt,0,spptop,dpptop,offset,k);
                W1 = cartK(0,pl,singpt,1,spptop,dpptop,offset,k)./dx;
                W2 = cartK(0,pl,singpt,2,spptop,dpptop,offset,k)./dx.^2;
                W3 = cartK(0,pl,singpt,3,spptop,dpptop,offset,k)./dx.^3;
                A(bp+pvm,pv+1)=(13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(bp+pvm,pv+2)=(-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(bp+pvm,pv+3)=(2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(bp+pvm,pv+4)=(-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
                A(bp+pvm,pvm)=A(bp+pvm,pvm)+.5;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(bp+pvn,pvl+1)=i*C1*sterm(pvl+1)*...
                    besselh(1,k*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2))*k/sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2).*((y(pvn)-y(pvl+1)).*...
                    cos(atan(yp(pvl+1)))-(x(pvn)-x(pvl+1)).*...
                    sin(atan(yp(pvl+1))));
                A(bp+pvn,pvl+2)=i*C2*sterm(pvl+2)*...
                    besselh(1,k*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2))*k/sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2).*((y(pvn)-y(pvl+2)).*...
                    cos(atan(yp(pvl+2)))-(x(pvn)-x(pvl+2)).*...
                    sin(atan(yp(pvl+2))));
```

www.manaraa.com

```
                A(bp+pvn,pvl+3)=i*C2*sterm(pvl+3)*...
                    besselh(1,k*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2))*k/sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2).*((y(pvn)-y(pvl+3)).*...
                    cos(atan(yp(pvl+3)))-(x(pvn)-x(pvl+3)).*...
                    sin(atan(yp(pvl+3))));
                A(bp+pvn,pvl+4)=i*C1*sterm(pvl+4)*...
                    besselh(1,k*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2))*k/sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2).*((y(pvn)-y(pvl+4)).*...
                    cos(atan(yp(pvl+4)))-(x(pvn)-x(pvl+4)).*...
                    sin(atan(yp(pvl+4))));
            end
        end
    end
end

% Top K Outside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(patches+tpatches+1):(2*patches+tpatches)
        pvl = 4*(l-1);
        offset = (2*patches+tpatches-1)*pl;
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = x(5-m);
                W0 = cartJ(0,pl,singpt,0,spptop,dpptop,offset,k0);
                W1 = cartJ(0,pl,singpt,1,spptop,dpptop,offset,k0)./dx;
                W2 = cartJ(0,pl,singpt,2,spptop,dpptop,offset,k0)./dx.^2;
                W3 = cartJ(0,pl,singpt,3,spptop,dpptop,offset,k0)./dx.^3;
                A(pvm,bp+pv+1)=-k0/eta0*...
                    (13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(pvm,bp+pv+2)=-k0/eta0*...
                    (-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(pvm,bp+pv+3)=-k0/eta0*...
                    (2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(pvm,bp+pv+4)=-k0/eta0*...
                    (-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
            end
        else
            for n=1:4
                pvn = pv + n;
```

www.manaraa.com

```
                A(pvn,bp+pvl+1)=-k0/eta0*C1*sterm(pvl+1)*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2));
                A(pvn,bp+pvl+2)=-k0/eta0*C2*sterm(pvl+2)*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2));
                A(pvn,bp+pvl+3)=-k0/eta0*C2*sterm(pvl+3)*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2));
                A(pvn,bp+pvl+4)=-k0/eta0*C1*sterm(pvl+4)*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2));
            end
        end
    end
end

% Top K Inside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(patches+tpatches+1):(2*patches+tpatches)
        pvl = 4*(l-1);
        offset = (2*patches+tpatches-1)*pl;
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = x(5-m);
                W0 = cartJ(0,pl,singpt,0,spptop,dpptop,offset,k);
                W1 = cartJ(0,pl,singpt,1,spptop,dpptop,offset,k)./dx;
                W2 = cartJ(0,pl,singpt,2,spptop,dpptop,offset,k)./dx.^2;
                W3 = cartJ(0,pl,singpt,3,spptop,dpptop,offset,k)./dx.^3;
                A(bp+pvm,bp+pv+1)=-k/eta*...
                    (13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(bp+pvm,bp+pv+2)=-k/eta*...
                    (-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(bp+pvm,bp+pv+3)=-k/eta*...
                    (2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(bp+pvm,bp+pv+4)=-k/eta*...
                    (-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
            end
        else
            for n=1:4
                pvn = pv + n;
```

www.manaraa.com

```
                A(bp+pvn,bp+pvl+1)=-k/eta*C1*sterm(pvl+1)*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2));
                A(bp+pvn,bp+pvl+2)=-k/eta*C2*sterm(pvl+2)*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2));
                A(bp+pvn,bp+pvl+3)=-k/eta*C2*sterm(pvl+3)*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2));
                A(bp+pvn,bp+pvl+4)=-k/eta*C1*sterm(pvl+4)*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2));
            end
        end
    end
end

% Left Semicircle J Outside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(2*patches+tpatches+1):2*(patches+tpatches)
        pvl = 4*(l-1);
        offset=(l-2*patches-tpatches-1)*tl+pi/2;
        if (j==l)
            for m=1:4
                pvm = pv + m;
                singpt = theta(4*patches+m)+pi/2;
                W0 = polK(0,tl,singpt,0,offset,k0,r);
                W1 = polK(0,tl,singpt,1,offset,k0,r)./dtheta;
                W2 = polK(0,tl,singpt,2,offset,k0,r)./dtheta.^2;
                W3 = polK(0,tl,singpt,3,offset,k0,r)./dtheta.^3;
                A(pvm,pv+1)=-(13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(pvm,pv+2)=-(-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(pvm,pv+3)=-(2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(pvm,pv+4)=-(-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
                A(pvm,pvm)=A(pvm,pvm)-.5;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(pvn,pvl+1)=-i*C1t*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+1)).^2+(y(pvn)-y(pvl+1)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+1)).^2+(y(pvn)-y(pvl+1)).^2).*...
```

www.manaraa.com

```
                    ((y(pvn)-y(pvl+1)).*cos(theta(pvl+1)+pi/2)-...
                    (x(pvn)-x(pvl+1)).*sin(theta(pvl+1)+pi/2));
                A(pvn,pvl+2)=-i*C2t*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+2)).^2+(y(pvn)-y(pvl+2)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+2)).^2+(y(pvn)-y(pvl+2)).^2).*...
                    ((y(pvn)-y(pvl+2)).*cos(theta(pvl+2)+pi/2)-...
                    (x(pvn)-x(pvl+2)).*sin(theta(pvl+2)+pi/2));
                A(pvn,pvl+3)=-i*C2t*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+3)).^2+(y(pvn)-y(pvl+3)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+3)).^2+(y(pvn)-y(pvl+3)).^2).*...
                    ((y(pvn)-y(pvl+3)).*cos(theta(pvl+3)+pi/2)-...
                    (x(pvn)-x(pvl+3)).*sin(theta(pvl+3)+pi/2));
                A(pvn,pvl+4)=-i*C1t*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+4)).^2+(y(pvn)-y(pvl+4)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+4)).^2+(y(pvn)-y(pvl+4)).^2).*...
                    ((y(pvn)-y(pvl+4)).*cos(theta(pvl+4)+pi/2)-...
                    (x(pvn)-x(pvl+4)).*sin(theta(pvl+4)+pi/2));
            end
        end
    end
end

% Left Semicircle J Inside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(2*patches+tpatches+1):2*(patches+tpatches)
        pvl = 4*(l-1);
        offset=(l-2*patches-tpatches-1)*tl+pi/2;
        if (j==l)
            for m=1:4
                pvm = pv + m;
                singpt = theta(4*patches+m)+pi/2;
                W0 = polK(0,tl,singpt,0,offset,k,r);
                W1 = polK(0,tl,singpt,1,offset,k,r)./dtheta;
                W2 = polK(0,tl,singpt,2,offset,k,r)./dtheta.^2;
                W3 = polK(0,tl,singpt,3,offset,k,r)./dtheta.^3;
                A(bp+pvm,pv+1)=-(13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(bp+pvm,pv+2)=-(-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(bp+pvm,pv+3)=-(2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(bp+pvm,pv+4)=-(-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
                A(bp+pvm,pvm)=A(bp+pvm,pvm)+.5;
            end
        else
```

www.manaraa.com

```
            for n=1:4
                pvn = pv + n;
                A(bp+pvn,pvl+1)=-i*C1t*besselh(1,k*sqrt...
                    ((x(pvn)-x(pvl+1)).^2+(y(pvn)-y(pvl+1)).^2))*k/...
                    sqrt((x(pvn)-x(pvl+1)).^2+(y(pvn)-y(pvl+1)).^2).*...
                    ((y(pvn)-y(pvl+1)).*cos(theta(pvl+1)+pi/2)-...
                    (x(pvn)-x(pvl+1)).*sin(theta(pvl+1)+pi/2));
                A(bp+pvn,pvl+2)=-i*C2t*besselh(1,k*sqrt...
                    ((x(pvn)-x(pvl+2)).^2+(y(pvn)-y(pvl+2)).^2))*k/...
                    sqrt((x(pvn)-x(pvl+2)).^2+(y(pvn)-y(pvl+2)).^2).*...
                    ((y(pvn)-y(pvl+2)).*cos(theta(pvl+2)+pi/2)-...
                    (x(pvn)-x(pvl+2)).*sin(theta(pvl+2)+pi/2));
                A(bp+pvn,pvl+3)=-i*C2t*besselh(1,k*sqrt...
                    ((x(pvn)-x(pvl+3)).^2+(y(pvn)-y(pvl+3)).^2))*k/...
                    sqrt((x(pvn)-x(pvl+3)).^2+(y(pvn)-y(pvl+3)).^2).*...
                    ((y(pvn)-y(pvl+3)).*cos(theta(pvl+3)+pi/2)-...
                    (x(pvn)-x(pvl+3)).*sin(theta(pvl+3)+pi/2));
                A(bp+pvn,pvl+4)=-i*C1t*besselh(1,k*sqrt...
                    ((x(pvn)-x(pvl+4)).^2+(y(pvn)-y(pvl+4)).^2))*k/...
                    sqrt((x(pvn)-x(pvl+4)).^2+(y(pvn)-y(pvl+4)).^2).*...
                    ((y(pvn)-y(pvl+4)).*cos(theta(pvl+4)+pi/2)-...
                    (x(pvn)-x(pvl+4)).*sin(theta(pvl+4)+pi/2));
            end
        end
    end
end

% Left Semicircle K Outside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(2*patches+tpatches+1):2*(patches+tpatches)
        pvl = 4*(l-1);
        offset=(l-2*patches-tpatches-1)*tl+pi/2;
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = theta(4*patches+m)+pi/2;
                W0 = polJ(0,tl,singpt,0,offset,k0,r);
                W1 = polJ(0,tl,singpt,1,offset,k0,r)./dtheta;
                W2 = polJ(0,tl,singpt,2,offset,k0,r)./dtheta.^2;
                W3 = polJ(0,tl,singpt,3,offset,k0,r)./dtheta.^3;
                A(pvm,bp+pv+1)=k0/eta0*...
                    (13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
```

www.manaraa.com

```
                A(pvm,bp+pv+2)=k0/eta0*...
                    (-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(pvm,bp+pv+3)=k0/eta0*...
                    (2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(pvm,bp+pv+4)=k0/eta0*...
                    (-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(pvn,bp+pvl+1)=k0/eta0*C1t*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2));
                A(pvn,bp+pvl+2)=k0/eta0*C2t*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2));
                A(pvn,bp+pvl+3)=k0/eta0*C2t*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2));
                A(pvn,bp+pvl+4)=k0/eta0*C1t*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2));
            end
        end
    end
end

% Left Semicircle K Inside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(2*patches+tpatches+1):2*(patches+tpatches)
        pvl = 4*(l-1);
        offset=(l-2*patches-tpatches-1)*tl+pi/2;
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = theta(4*patches+m)+pi/2;
                W0 = polJ(0,tl,singpt,0,offset,k,r);
                W1 = polJ(0,tl,singpt,1,offset,k,r)./dtheta;
                W2 = polJ(0,tl,singpt,2,offset,k,r)./dtheta.^2;
                W3 = polJ(0,tl,singpt,3,offset,k,r)./dtheta.^3;
                A(bp+pvm,bp+pv+1)=k/eta*(13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(bp+pvm,bp+pv+2)=k/eta*(-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
```

www.manaraa.com

```
                A(bp+pvm,bp+pv+3)=k/eta*(2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(bp+pvm,bp+pv+4)=k/eta*(-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(bp+pvn,bp+pvl+1)=k/eta*C1t*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2));
                A(bp+pvn,bp+pvl+2)=k/eta*C2t*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2));
                A(bp+pvn,bp+pvl+3)=k/eta*C2t*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2));
                A(bp+pvn,bp+pvl+4)=k/eta*C1t*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2));
            end
        end
    end
end
```

A.25 TEsolvematrix.m

This .m file solves the TE analog of (3.46).

```
% *** solve matrix and define incident beam ***

rhs=exp(i*k0.*(cos(thetar).*x-sin(thetar).*y)); % incident plane wave

% the following comments section is the code for a tapered beam

% rhs = erf(20/len*((len/2-5E-8)-abs((y+t/2)/tan(thetar)+len/2-x)));
% for g=1:length(rhs)
%     if rhs(g)< 0
%         rhs(g)=0;
%     end
% end
% rhs=rhs.*exp(i*k0.*(cos(thetar).*x+sin(thetar).*y));

for j=1:bp % bottom half of incident block matrix is null
    rhs(bp+j)=0;
```

www.manaraa.com

```
end

KJ=A\rhs.'; % invert matrix and solve for surface currents
J=KJ(1:bp);
K=KJ(bp+1:2*bp);

figure % plots surface current J and K magnitudes
plot(x,abs(J),'r-',x,abs(K),'b-');
xlabel('x');
ylabel('magnitude');
title(['Plane wave at ',num2str(thetadeg),' degrees from grazing.']);
legend('abs(J)','abs(K)');
```

A.26 timeinfo.m

```
% *** Displays the elapsed time information ***

fprintf('\nElapsed time of indicated calculations: (in seconds) \n\n')
fprintf('Problem Setup: %g',t1-t0)
fprintf('\nNystrom Matrix Fill: %g',t2-t1)
fprintf('\nMatrix Inversion: %g',t3-t2)
fprintf('\nFar Field Intensity: %g',t4-t3)
fprintf('\nComparison Plots and Power Spectral Density: %g',t5-t4)
fprintf('\nTotal: %g',t5-t0)
fprintf('\n')
fprintf('\n')
```

A.27 TM.m

This file is called to execute a run for TM polarization. It references all the other subfunctions which are called.

```
function TM % function so outputs and inputs can be specified by user
t0=cputime; % marks time at start of program
global llquadzw;
load llquadzw.txt; % weights and quadrature points for the linlog int

userinputs; % contains program parameters specified by user
constants; % defines constants
surfacesetup; % constructs scattering surfaces and quadrature points
% FFTsurf; % computes and displays power spectral density
nystromconstants; % defines constants used in filling and solving matricies
```

www.manaraa.com

```
t1=cputime;
TMnystromfill; % fills Nystrom matrix
t2=cputime;
TMsolvematrix; % solves matrix equation currents / defines inc. beam
t3=cputime;
TMfarfield; % solves for far field intensity as a function of angle
t4=cputime;
% TMcylcon; % analytic solution cylinder (change thevaldeg in far field)
% TMphysoptcompare; % compares plate solution to physical optics
% storedata; % stores the data in an output text file
t5=cputime;
timeinfo; % displays the elapsed time information
```

A.28 TMcylcon.m

This routine plots (5.19).

```
% *** Analytic solution for TM cylindrical conductor ***

a = r;
f0=besselj(0,2*pi*a)/besselh(0,2*pi*a);
an = 0:pi/1800:pi;
for ss = 1:length(an)
    for mm = 1:100
        fsum(mm)=(-1).^mm.*2*cos(mm.*an(ss)).*...
            besselj(mm,2*pi*a)./besselh(mm,2*pi*a);
    end
    ftheta(ss)=-4*(f0+sum(fsum)); % factor of 4 makes normalization...
    % consistent w/ our notation
end
andeg = an*180/pi+180-thetadeg;
fintensity = abs(ftheta).^2;
figure
plot(thevaldeg,intensity,'r-',andeg,fintensity,'g-');
xlabel('observation angle (degrees)');
ylabel('intensity');
title(['Plane wave at ',num2str(thetadeg),' degrees from grazing.']);
legend('intensity','analytic solution TM cylinder conductor)');
```

www.manaraa.com

A.29 `TMfarfield.m`

This `.m` file calculates (3.73), which contains (3.69) and (3.70), for many observation angles.

```
% *** solving for the far field amplitude as a function of angle ***

% commented line is for viewing 360 degree view of scattered intensity
% active when looking at analytic cylinder solution

if len == 0 % far field theta range (for circle & flat scatters)
    thevaldeg = 0:360/thsteps:360;
else
    thevaldeg = thetadeg-thrange/2:thrange/thsteps:thetadeg+thrange/2;
end
thevalr = thevaldeg*pi/180;

for v=1:thsteps+1
    for q=1:patches
        pvq = 4*(q-1);
        B(v,pvq+1)=C3.*sterm(pvq+1).*...
            exp(-i*k0*(x(pvq+1)*cos(thevalr(v))+...
            y(pvq+1)*sin(thevalr(v)))).*...
            (eta0*J(pvq+1)+K(pvq+1).*...
            (sin(thevalr(v)).*cos(atan(yp(pvq+1)))-...
            cos(thevalr(v)).*sin(atan(yp(pvq+1)))));
        B(v,pvq+2)=C4.*sterm(pvq+2).*...
            exp(-i*k0*(x(pvq+2)*cos(thevalr(v))+...
            y(pvq+2)*sin(thevalr(v)))).*...
            (eta0*J(pvq+2)+K(pvq+2).*...
            (sin(thevalr(v)).*cos(atan(yp(pvq+2)))-...
            cos(thevalr(v)).*sin(atan(yp(pvq+2)))));
        B(v,pvq+3)=C4.*sterm(pvq+3).*...
            exp(-i*k0*(x(pvq+3)*cos(thevalr(v))+...
            y(pvq+3)*sin(thevalr(v)))).*...
            (eta0*J(pvq+3)+K(pvq+3).*...
            (sin(thevalr(v)).*cos(atan(yp(pvq+3)))-...
            cos(thevalr(v)).*sin(atan(yp(pvq+3)))));
        B(v,pvq+4)=C3.*sterm(pvq+4).*...
            exp(-i*k0*(x(pvq+4)*cos(thevalr(v))+...
            y(pvq+4)*sin(thevalr(v)))).*...
            (eta0*J(pvq+4)+K(pvq+4).*...
            (sin(thevalr(v)).*cos(atan(yp(pvq+4)))-...
```

www.manaraa.com

```
            cos(thevalr(v)).*sin(atan(yp(pvq+4)))));
    end
end

for v=1:thsteps+1
    for q=patches+1:patches+tpatches
        pvq = 4*(q-1);
        B(v,pvq+1)=C3t.*...
            exp(-i*k0*(x(pvq+1)*cos(thevalr(v))+...
            y(pvq+1)*sin(thevalr(v)))).*...
            (eta0*J(pvq+1)+K(pvq+1).*...
            (sin(thevalr(v)).*cos(theta(pvq+1)+pi/2)-...
            cos(thevalr(v)).*sin(theta(pvq+1)+pi/2)));
        B(v,pvq+2)=C4t.*...
            exp(-i*k0*(x(pvq+2)*cos(thevalr(v))+...
            y(pvq+2)*sin(thevalr(v)))).*...
            (eta0*J(pvq+2)+K(pvq+2).*...
            (sin(thevalr(v)).*cos(theta(pvq+2)+pi/2)-...
            cos(thevalr(v)).*sin(theta(pvq+2)+pi/2)));
        B(v,pvq+3)=C4t.*...
            exp(-i*k0*(x(pvq+3)*cos(thevalr(v))+...
            y(pvq+3)*sin(thevalr(v)))).*...
            (eta0*J(pvq+3)+K(pvq+3).*...
            (sin(thevalr(v)).*cos(theta(pvq+3)+pi/2)-...
            cos(thevalr(v)).*sin(theta(pvq+3)+pi/2)));
        B(v,pvq+4)=C3t.*...
            exp(-i*k0*(x(pvq+4)*cos(thevalr(v))+...
            y(pvq+4)*sin(thevalr(v)))).*...
            (eta0*J(pvq+4)+K(pvq+4).*...
            (sin(thevalr(v)).*cos(theta(pvq+4)+pi/2)-...
            cos(thevalr(v)).*sin(theta(pvq+4)+pi/2)));
    end
end

for v=1:thsteps+1
    for q=patches+tpatches+1:2*patches+tpatches
        pvq = 4*(q-1);
        B(v,pvq+1)=-C3.*sterm(pvq+1).*...
            exp(-i*k0*(x(pvq+1)*cos(thevalr(v))+...
            y(pvq+1)*sin(thevalr(v)))).*...
            (eta0*J(pvq+1)+K(pvq+1).*...
            (sin(thevalr(v)).*cos(atan(yp(pvq+1)))-...
            cos(thevalr(v)).*sin(atan(yp(pvq+1)))));
```

www.manaraa.com

```
        B(v,pvq+2)=-C4.*sterm(pvq+2).*...
            exp(-i*k0*(x(pvq+2)*cos(thevalr(v))+...
            y(pvq+2)*sin(thevalr(v)))).*...
            (eta0*J(pvq+2)+K(pvq+2).*...
            (sin(thevalr(v)).*cos(atan(yp(pvq+2)))-...
            cos(thevalr(v)).*sin(atan(yp(pvq+2)))));
        B(v,pvq+3)=-C4.*sterm(pvq+3).*...
            exp(-i*k0*(x(pvq+3)*cos(thevalr(v))+...
            y(pvq+3)*sin(thevalr(v)))).*...
            (eta0*J(pvq+3)+K(pvq+3).*...
            (sin(thevalr(v)).*cos(atan(yp(pvq+3)))-...
            cos(thevalr(v)).*sin(atan(yp(pvq+3)))));
        B(v,pvq+4)=-C3.*sterm(pvq+4).*...
            exp(-i*k0*(x(pvq+4)*cos(thevalr(v))+...
            y(pvq+4)*sin(thevalr(v)))).*...
            (eta0*J(pvq+4)+K(pvq+4).*...
            (sin(thevalr(v)).*cos(atan(yp(pvq+4)))-...
            cos(thevalr(v)).*sin(atan(yp(pvq+4)))));
    end
end

for v=1:thsteps+1
    for q=2*patches+tpatches+1:2*(patches+tpatches)
        pvq = 4*(q-1);
        B(v,pvq+1)=C3t.*...
            exp(-i*k0*(x(pvq+1)*cos(thevalr(v))+...
            y(pvq+1)*sin(thevalr(v)))).*...
            (eta0*J(pvq+1)+K(pvq+1).*...
            (sin(thevalr(v)).*cos(theta(pvq+1)+pi/2)-...
            cos(thevalr(v)).*sin(theta(pvq+1)+pi/2)));
        B(v,pvq+2)=C4t.*...
            exp(-i*k0*(x(pvq+2)*cos(thevalr(v))+...
            y(pvq+2)*sin(thevalr(v)))).*...
            (eta0*J(pvq+2)+K(pvq+2).*...
            (sin(thevalr(v)).*cos(theta(pvq+2)+pi/2)-...
            cos(thevalr(v)).*sin(theta(pvq+2)+pi/2)));
        B(v,pvq+3)=C4t.*...
            exp(-i*k0*(x(pvq+3)*cos(thevalr(v))+...
            y(pvq+3)*sin(thevalr(v)))).*...
            (eta0*J(pvq+3)+K(pvq+3).*...
            (sin(thevalr(v)).*cos(theta(pvq+3)+pi/2)-...
            cos(thevalr(v)).*sin(theta(pvq+3)+pi/2)));
        B(v,pvq+4)=C3t.*...
```

www.manaraa.com

```
            exp(-i*k0*(x(pvq+4)*cos(thevalr(v))+...
            y(pvq+4)*sin(thevalr(v)))).*...
            (eta0*J(pvq+4)+K(pvq+4).*...
            (sin(thevalr(v)).*cos(theta(pvq+4)+pi/2)-...
            cos(thevalr(v)).*sin(theta(pvq+4)+pi/2)));
    end
end

B=B.*i.*sqrt(i).*k0; % gets normalization and phase correct for field
field=sum(B.'); % computes far field amplitude
intensity = abs(field).^2; % computes far field intensity
figure
plot(thevaldeg,intensity,'r-');
xlabel('observation angle (degrees)');
ylabel('intensity');
title(['Plane wave at ',num2str(thetadeg),' degrees from grazing.']);
legend('intensity');
```

A.30 TMnystromfill.m

This .m file fills the Nystrom matrix of (3.46). It is broken up into patches which contain singularities and patches which don't. The patches which contain singularities use (3.47) and (3.48). The patches which don't contain singularities use (3.40) and (3.41).

```
% *** fills Nystrom matrix ***

% Bottom K outside
for j=1:(2*(patches+tpatches)) % looping through observation points
    pv = 4*(j-1);
    for l=1:patches % looping through the patches
        pvl = 4*(l-1);
        offset = (l-1)*pl;
        if(j==l) % if the observation point is on the patch,
                 % we need to use a quadrature which
                 % integrates over the green function
                 % (see numerical methods section 18.3)
            for m=1:4 % these different m's correspond to observation pts
                pvm = pv + m;
                singpt = x(m);
                W0 = cartK(0,pl,singpt,0,sppbot,dppbot,offset,k0);
                W1 = cartK(0,pl,singpt,1,sppbot,dppbot,offset,k0)./dx;
                W2 = cartK(0,pl,singpt,2,sppbot,dppbot,offset,k0)./dx.^2;
```

www.manaraa.com

```
                W3 = cartK(0,pl,singpt,3,sppbot,dppbot,offset,k0)./dx.^3;
                A(pvm,pv+1)=(13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(pvm,pv+2)=(-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(pvm,pv+3)=(2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(pvm,pv+4)=(-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
                A(pvm,pvm)=A(pvm,pvm)+.5;
            end
        else
            for n=1:4 % now filling off diagonal blocks...
                pvn = pv + n;
                A(pvn,pvl+1)=i*C1*sterm(pvl+1)*...
                    besselh(1,k0*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2))*k0/sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2).*((y(pvn)-y(pvl+1)).*...
                    cos(atan(yp(pvl+1)))-(x(pvn)-x(pvl+1)).*...
                    sin(atan(yp(pvl+1))));
                A(pvn,pvl+2)=i*C2*sterm(pvl+2)*...
                    besselh(1,k0*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2))*k0/sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2).*((y(pvn)-y(pvl+2)).*...
                    cos(atan(yp(pvl+2)))-(x(pvn)-x(pvl+2)).*...
                    sin(atan(yp(pvl+2))));
                A(pvn,pvl+3)=i*C2*sterm(pvl+3)*...
                    besselh(1,k0*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2))*k0/sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2).*((y(pvn)-y(pvl+3)).*...
                    cos(atan(yp(pvl+3)))-(x(pvn)-x(pvl+3)).*...
                    sin(atan(yp(pvl+3))));
                A(pvn,pvl+4)=i*C1*sterm(pvl+4)*...
                    besselh(1,k0*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2))*k0/sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2).*((y(pvn)-y(pvl+4)).*...
                    cos(atan(yp(pvl+4)))-(x(pvn)-x(pvl+4)).*...
                    sin(atan(yp(pvl+4))));
            end
        end
    end
end

bp = 8*(patches+tpatches);

% Bottom K inside
for j=1:(2*(patches+tpatches))
```

www.manaraa.com

```
pv = 4*(j-1);
for l=1:patches
    pvl = 4*(l-1);
    offset = (l-1)*pl;
    if(j==l)
        for m=1:4
            pvm = pv + m;
            singpt = x(m);
            W0 = cartK(0,pl,singpt,0,sppbot,dppbot,offset,k);
            W1 = cartK(0,pl,singpt,1,sppbot,dppbot,offset,k)./dx;
            W2 = cartK(0,pl,singpt,2,sppbot,dppbot,offset,k)./dx.^2;
            W3 = cartK(0,pl,singpt,3,sppbot,dppbot,offset,k)./dx.^3;
            A(pvm+bp,pv+1)=(13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
            A(pvm+bp,pv+2)=(-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
            A(pvm+bp,pv+3)=(2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
            A(pvm+bp,pv+4)=(-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
            A(pvm+bp,pvm)=A(pvm+bp,pvm)-.5;
        end
    else
        for n=1:4
            pvn = pv + n;
            A(pvn+bp,pvl+1)=i*C1*sterm(pvl+1)*...
                besselh(1,k*sqrt((x(pvn)-x(pvl+1)).^2+...
                (y(pvn)-y(pvl+1)).^2))*k/sqrt((x(pvn)-x(pvl+1)).^2+...
                (y(pvn)-y(pvl+1)).^2).*((y(pvn)-y(pvl+1)).*...
                cos(atan(yp(pvl+1)))-(x(pvn)-x(pvl+1)).*...
                sin(atan(yp(pvl+1))));
            A(pvn+bp,pvl+2)=i*C2*sterm(pvl+2)*...
                besselh(1,k*sqrt((x(pvn)-x(pvl+2)).^2+...
                (y(pvn)-y(pvl+2)).^2))*k/sqrt((x(pvn)-x(pvl+2)).^2+...
                (y(pvn)-y(pvl+2)).^2).*((y(pvn)-y(pvl+2)).*...
                cos(atan(yp(pvl+2)))-(x(pvn)-x(pvl+2)).*...
                sin(atan(yp(pvl+2))));
            A(pvn+bp,pvl+3)=i*C2*sterm(pvl+3)*...
                besselh(1,k*sqrt((x(pvn)-x(pvl+3)).^2+...
                (y(pvn)-y(pvl+3)).^2))*k/sqrt((x(pvn)-x(pvl+3)).^2+...
                (y(pvn)-y(pvl+3)).^2).*((y(pvn)-y(pvl+3)).*...
                cos(atan(yp(pvl+3)))-(x(pvn)-x(pvl+3)).*...
                sin(atan(yp(pvl+3))));
            A(pvn+bp,pvl+4)=i*C1*sterm(pvl+4)*...
                besselh(1,k*sqrt((x(pvn)-x(pvl+4)).^2+...
                (y(pvn)-y(pvl+4)).^2))*k/sqrt((x(pvn)-x(pvl+4)).^2+...
                (y(pvn)-y(pvl+4)).^2).*((y(pvn)-y(pvl+4)).*...
```

www.manaraa.com

```
                    cos(atan(yp(pvl+4)))-(x(pvn)-x(pvl+4)).*...
                    sin(atan(yp(pvl+4))));
            end
        end
    end
end

% Bottom J Outside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=1:patches
        pvl = 4*(l-1);
        offset = (l-1)*pl;
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = x(m);
                W0 = cartJ(0,pl,singpt,0,sppbot,dppbot,offset,k0);
                W1 = cartJ(0,pl,singpt,1,sppbot,dppbot,offset,k0)./dx;
                W2 = cartJ(0,pl,singpt,2,sppbot,dppbot,offset,k0)./dx.^2;
                W3 = cartJ(0,pl,singpt,3,sppbot,dppbot,offset,k0)./dx.^3;
                A(pvm,bp+pv+1)=k0*eta0*...
                    (13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(pvm,bp+pv+2)=k0*eta0*...
                    (-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(pvm,bp+pv+3)=k0*eta0*...
                    (2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(pvm,bp+pv+4)=k0*eta0*...
                    (-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(pvn,bp+pvl+1)=k0*eta0*C1*sterm(pvl+1)*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2));
                A(pvn,bp+pvl+2)=k0*eta0*C2*sterm(pvl+2)*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2));
                A(pvn,bp+pvl+3)=k0*eta0*C2*sterm(pvl+3)*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2));
                A(pvn,bp+pvl+4)=k0*eta0*C1*sterm(pvl+4)*...
```

www.manaraa.com

```
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2));
            end
        end
    end
end

% Bottom J Inside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=1:patches
        pvl = 4*(l-1);
        offset = (l-1)*pl;
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = x(m);
                W0 = cartJ(0,pl,singpt,0,sppbot,dppbot,offset,k);
                W1 = cartJ(0,pl,singpt,1,sppbot,dppbot,offset,k)./dx;
                W2 = cartJ(0,pl,singpt,2,sppbot,dppbot,offset,k)./dx.^2;
                W3 = cartJ(0,pl,singpt,3,sppbot,dppbot,offset,k)./dx.^3;
                A(bp+pvm,bp+pv+1)=k*eta*...
                    (13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(bp+pvm,bp+pv+2)=k*eta*...
                    (-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(bp+pvm,bp+pv+3)=k*eta*...
                    (2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(bp+pvm,bp+pv+4)=k*eta*...
                    (-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(bp+pvn,bp+pvl+1)=k*eta*C1*sterm(pvl+1)*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2));
                A(bp+pvn,bp+pvl+2)=k*eta*C2*sterm(pvl+2)*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2));
                A(bp+pvn,bp+pvl+3)=k*eta*C2*sterm(pvl+3)*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2));
                A(bp+pvn,bp+pvl+4)=k*eta*C1*sterm(pvl+4)*...
```

www.manaraa.com

```
                    besselh(0,k*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2));
            end
        end
    end
end

% Right Semicircle K Outside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(patches+1):(patches+tpatches)
        pvl = 4*(l-1);
        offset=(l-patches-1)*tl-pi/2;
        if (j==l)
            for m=1:4
                pvm = pv + m;
                singpt = theta(4*patches+m)+pi/2;
                W0 = polK(0,tl,singpt,0,offset,k0,r);
                W1 = polK(0,tl,singpt,1,offset,k0,r)./dtheta;
                W2 = polK(0,tl,singpt,2,offset,k0,r)./dtheta.^2;
                W3 = polK(0,tl,singpt,3,offset,k0,r)./dtheta.^3;
                A(pvm,pv+1)=(13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(pvm,pv+2)=(-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(pvm,pv+3)=(2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(pvm,pv+4)=(-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
                A(pvm,pvm)=A(pvm,pvm)+.5;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(pvn,pvl+1)=i*C1t*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+1)).^2+(y(pvn)-y(pvl+1)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+1)).^2+(y(pvn)-y(pvl+1)).^2).*...
                    ((y(pvn)-y(pvl+1)).*cos(theta(pvl+1)+pi/2)-...
                    (x(pvn)-x(pvl+1)).*sin(theta(pvl+1)+pi/2));
                A(pvn,pvl+2)=i*C2t*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+2)).^2+(y(pvn)-y(pvl+2)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+2)).^2+(y(pvn)-y(pvl+2)).^2).*...
                    ((y(pvn)-y(pvl+2)).*cos(theta(pvl+2)+pi/2)-...
                    (x(pvn)-x(pvl+2)).*sin(theta(pvl+2)+pi/2));
                A(pvn,pvl+3)=i*C2t*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+3)).^2+(y(pvn)-y(pvl+3)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+3)).^2+(y(pvn)-y(pvl+3)).^2).*...
```

www.manaraa.com

```
                    ((y(pvn)-y(pvl+3)).*cos(theta(pvl+3)+pi/2)-...
                    (x(pvn)-x(pvl+3)).*sin(theta(pvl+3)+pi/2));
                A(pvn,pvl+4)=i*C1t*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+4)).^2+(y(pvn)-y(pvl+4)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+4)).^2+(y(pvn)-y(pvl+4)).^2).*...
                    ((y(pvn)-y(pvl+4)).*cos(theta(pvl+4)+pi/2)-...
                    (x(pvn)-x(pvl+4)).*sin(theta(pvl+4)+pi/2));
            end
        end
    end
end

% Right Semicircle K Inside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(patches+1):(patches+tpatches)
        pvl = 4*(l-1);
        offset=(l-patches-1)*tl-pi/2;
        if (j==l)
            for m=1:4
                pvm = pv + m;
                singpt = theta(4*patches+m)+pi/2;
                W0 = polK(0,tl,singpt,0,offset,k,r);
                W1 = polK(0,tl,singpt,1,offset,k,r)./dtheta;
                W2 = polK(0,tl,singpt,2,offset,k,r)./dtheta.^2;
                W3 = polK(0,tl,singpt,3,offset,k,r)./dtheta.^3;
                A(pvm+bp,pv+1)=(13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(pvm+bp,pv+2)=(-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(pvm+bp,pv+3)=(2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(pvm+bp,pv+4)=(-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
                A(pvm+bp,pvm)=A(pvm+bp,pvm)-.5;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(pvn+bp,pvl+1)=i*C1t*besselh(1,k*sqrt...
                    ((x(pvn)-x(pvl+1)).^2+(y(pvn)-y(pvl+1)).^2))*k/...
                    sqrt((x(pvn)-x(pvl+1)).^2+(y(pvn)-y(pvl+1)).^2).*...
                    ((y(pvn)-y(pvl+1)).*cos(theta(pvl+1)+pi/2)-...
                    (x(pvn)-x(pvl+1)).*sin(theta(pvl+1)+pi/2));
                A(pvn+bp,pvl+2)=i*C2t*besselh(1,k*sqrt...
                    ((x(pvn)-x(pvl+2)).^2+(y(pvn)-y(pvl+2)).^2))*k/...
                    sqrt((x(pvn)-x(pvl+2)).^2+(y(pvn)-y(pvl+2)).^2).*...
```

www.manaraa.com

```
                    ((y(pvn)-y(pvl+2)).*cos(theta(pvl+2)+pi/2)-...
                    (x(pvn)-x(pvl+2)).*sin(theta(pvl+2)+pi/2));
                A(pvn+bp,pvl+3)=i*C2t*besselh(1,k*sqrt...
                    ((x(pvn)-x(pvl+3)).^2+(y(pvn)-y(pvl+3)).^2))*k/...
                    sqrt((x(pvn)-x(pvl+3)).^2+(y(pvn)-y(pvl+3)).^2).*...
                    ((y(pvn)-y(pvl+3)).*cos(theta(pvl+3)+pi/2)-...
                    (x(pvn)-x(pvl+3)).*sin(theta(pvl+3)+pi/2));
                A(pvn+bp,pvl+4)=i*C1t*besselh(1,k*sqrt...
                    ((x(pvn)-x(pvl+4)).^2+(y(pvn)-y(pvl+4)).^2))*k/...
                    sqrt((x(pvn)-x(pvl+4)).^2+(y(pvn)-y(pvl+4)).^2).*...
                    ((y(pvn)-y(pvl+4)).*cos(theta(pvl+4)+pi/2)-...
                    (x(pvn)-x(pvl+4)).*sin(theta(pvl+4)+pi/2));
            end
        end
    end
end

% Right Semicircle J Outside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(patches+1):(patches+tpatches)
        pvl = 4*(l-1);
        offset=(l-patches-1)*tl-pi/2;
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = theta(4*patches+m)+pi/2;
                W0 = polJ(0,tl,singpt,0,offset,k0,r);
                W1 = polJ(0,tl,singpt,1,offset,k0,r)./dtheta;
                W2 = polJ(0,tl,singpt,2,offset,k0,r)./dtheta.^2;
                W3 = polJ(0,tl,singpt,3,offset,k0,r)./dtheta.^3;
                A(pvm,bp+pv+1)=k0*eta0*...
                    (13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(pvm,bp+pv+2)=k0*eta0*...
                    (-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(pvm,bp+pv+3)=k0*eta0*...
                    (2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(pvm,bp+pv+4)=k0*eta0*...
                    (-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
            end
        else
            for n=1:4
                pvn = pv + n;
```

www.manaraa.com

```
                A(pvn,bp+pvl+1)=k0*eta0*C1t*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2));
                A(pvn,bp+pvl+2)=k0*eta0*C2t*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2));
                A(pvn,bp+pvl+3)=k0*eta0*C2t*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2));
                A(pvn,bp+pvl+4)=k0*eta0*C1t*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2));
            end
        end
    end
end

% Right Semicircle J Inside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(patches+1):(patches+tpatches)
        pvl = 4*(l-1);
        offset=(l-patches-1)*tl-pi/2;
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = theta(4*patches+m)+pi/2;
                W0 = polJ(0,tl,singpt,0,offset,k,r);
                W1 = polJ(0,tl,singpt,1,offset,k,r)./dtheta;
                W2 = polJ(0,tl,singpt,2,offset,k,r)./dtheta.^2;
                W3 = polJ(0,tl,singpt,3,offset,k,r)./dtheta.^3;
                A(bp+pvm,bp+pv+1)=k*eta*...
                    (13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(bp+pvm,bp+pv+2)=k*eta*...
                    (-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(bp+pvm,bp+pv+3)=k*eta*...
                    (2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(bp+pvm,bp+pv+4)=k*eta*...
                    (-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
            end
        else
            for n=1:4
                pvn = pv + n;
```

www.manaraa.com

```
                A(bp+pvn,bp+pvl+1)=k*eta*C1t*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2));
                A(bp+pvn,bp+pvl+2)=k*eta*C2t*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2));
                A(bp+pvn,bp+pvl+3)=k*eta*C2t*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2));
                A(bp+pvn,bp+pvl+4)=k*eta*C1t*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2));
            end
        end
    end
end

% Top K outside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(patches+tpatches+1):(2*patches+tpatches)
        pvl = 4*(l-1);
        offset = (2*patches+tpatches-l)*pl;
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = x(5-m);
                W0 = cartK(0,pl,singpt,0,spptop,dpptop,offset,k0);
                W1 = cartK(0,pl,singpt,1,spptop,dpptop,offset,k0)./dx;
                W2 = cartK(0,pl,singpt,2,spptop,dpptop,offset,k0)./dx.^2;
                W3 = cartK(0,pl,singpt,3,spptop,dpptop,offset,k0)./dx.^3;
                A(pvm,pv+1)=-(13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(pvm,pv+2)=-(-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(pvm,pv+3)=-(2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(pvm,pv+4)=-(-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
                A(pvm,pvm)=A(pvm,pvm)+.5;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(pvn,pvl+1)=-i*C1*sterm(pvl+1)*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+1)).^2+(y(pvn)-y(pvl+1)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+1)).^2+(y(pvn)-y(pvl+1)).^2).*...
```

www.manaraa.com

```
                    ((y(pvn)-y(pvl+1)).*cos(atan(yp(pvl+1)))-...
                    (x(pvn)-x(pvl+1)).*sin(atan(yp(pvl+1))));
                A(pvn,pvl+2)=-i*C2*sterm(pvl+2)*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+2)).^2+(y(pvn)-y(pvl+2)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+2)).^2+(y(pvn)-y(pvl+2)).^2).*...
                    ((y(pvn)-y(pvl+2)).*cos(atan(yp(pvl+2)))-...
                    (x(pvn)-x(pvl+2)).*sin(atan(yp(pvl+2))));
                A(pvn,pvl+3)=-i*C2*sterm(pvl+3)*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+3)).^2+(y(pvn)-y(pvl+3)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+3)).^2+(y(pvn)-y(pvl+3)).^2).*...
                    ((y(pvn)-y(pvl+3)).*cos(atan(yp(pvl+3)))-...
                    (x(pvn)-x(pvl+3)).*sin(atan(yp(pvl+3))));
                A(pvn,pvl+4)=-i*C1*sterm(pvl+4)*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+4)).^2+(y(pvn)-y(pvl+4)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+4)).^2+(y(pvn)-y(pvl+4)).^2).*...
                    ((y(pvn)-y(pvl+4)).*cos(atan(yp(pvl+4)))-...
                    (x(pvn)-x(pvl+4)).*sin(atan(yp(pvl+4))));
            end
        end
    end
end

% Top K Inside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(patches+tpatches+1):(2*patches+tpatches)
        pvl = 4*(l-1);
        offset = (2*patches+tpatches-l)*pl;
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = x(5-m);
                W0 = cartK(0,pl,singpt,0,spptop,dpptop,offset,k);
                W1 = cartK(0,pl,singpt,1,spptop,dpptop,offset,k)./dx;
                W2 = cartK(0,pl,singpt,2,spptop,dpptop,offset,k)./dx.^2;
                W3 = cartK(0,pl,singpt,3,spptop,dpptop,offset,k)./dx.^3;
                A(bp+pvm,pv+1)=-(13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(bp+pvm,pv+2)=-(-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(bp+pvm,pv+3)=-(2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(bp+pvm,pv+4)=-(-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
                A(bp+pvm,pvm)=A(bp+pvm,pvm)-.5;
            end
        else
```

www.manaraa.com

```
            for n=1:4
                pvn = pv + n;
                A(bp+pvn,pvl+1)=-i*C1*sterm(pvl+1)*besselh(1,k*sqrt...
                    ((x(pvn)-x(pvl+1)).^2+(y(pvn)-y(pvl+1)).^2))*k/...
                    sqrt((x(pvn)-x(pvl+1)).^2+(y(pvn)-y(pvl+1)).^2).*...
                    ((y(pvn)-y(pvl+1)).*cos(atan(yp(pvl+1)))-...
                    (x(pvn)-x(pvl+1)).*sin(atan(yp(pvl+1))));
                A(bp+pvn,pvl+2)=-i*C2*sterm(pvl+2)*besselh(1,k*sqrt...
                    ((x(pvn)-x(pvl+2)).^2+(y(pvn)-y(pvl+2)).^2))*k/...
                    sqrt((x(pvn)-x(pvl+2)).^2+(y(pvn)-y(pvl+2)).^2).*...
                    ((y(pvn)-y(pvl+2)).*cos(atan(yp(pvl+2)))-...
                    (x(pvn)-x(pvl+2)).*sin(atan(yp(pvl+2))));
                A(bp+pvn,pvl+3)=-i*C2*sterm(pvl+3)*besselh(1,k*sqrt...
                    ((x(pvn)-x(pvl+3)).^2+(y(pvn)-y(pvl+3)).^2))*k/...
                    sqrt((x(pvn)-x(pvl+3)).^2+(y(pvn)-y(pvl+3)).^2).*...
                    ((y(pvn)-y(pvl+3)).*cos(atan(yp(pvl+3)))-...
                    (x(pvn)-x(pvl+3)).*sin(atan(yp(pvl+3))));
                A(bp+pvn,pvl+4)=-i*C1*sterm(pvl+4)*besselh(1,k*sqrt...
                    ((x(pvn)-x(pvl+4)).^2+(y(pvn)-y(pvl+4)).^2))*k/...
                    sqrt((x(pvn)-x(pvl+4)).^2+(y(pvn)-y(pvl+4)).^2).*...
                    ((y(pvn)-y(pvl+4)).*cos(atan(yp(pvl+4)))-...
                    (x(pvn)-x(pvl+4)).*sin(atan(yp(pvl+4))));
            end
        end
    end
end

% Top J Outside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(patches+tpatches+1):(2*patches+tpatches)
        pvl = 4*(l-1);
        offset = (2*patches+tpatches-1)*pl;
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = x(5-m);
                W0 = cartJ(0,pl,singpt,0,spptop,dpptop,offset,k0);
                W1 = cartJ(0,pl,singpt,1,spptop,dpptop,offset,k0)./dx;
                W2 = cartJ(0,pl,singpt,2,spptop,dpptop,offset,k0)./dx.^2;
                W3 = cartJ(0,pl,singpt,3,spptop,dpptop,offset,k0)./dx.^3;
                A(pvm,bp+pv+1)=-k0*eta0*...
                    (13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
```

www.manaraa.com

```
                A(pvm,bp+pv+2)=-k0*eta0*...
                    (-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(pvm,bp+pv+3)=-k0*eta0*...
                    (2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(pvm,bp+pv+4)=-k0*eta0*...
                    (-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(pvn,bp+pvl+1)=-k0*eta0*C1*sterm(pvl+1)*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2));
                A(pvn,bp+pvl+2)=-k0*eta0*C2*sterm(pvl+2)*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2));
                A(pvn,bp+pvl+3)=-k0*eta0*C2*sterm(pvl+3)*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2));
                A(pvn,bp+pvl+4)=-k0*eta0*C1*sterm(pvl+4)*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2));
            end
        end
    end
end

% Top J Inside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(patches+tpatches+1):(2*patches+tpatches)
        pvl = 4*(l-1);
        offset = (2*patches+tpatches-1)*pl;
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = x(5-m);
                W0 = cartJ(0,pl,singpt,0,spptop,dpptop,offset,k);
                W1 = cartJ(0,pl,singpt,1,spptop,dpptop,offset,k)./dx;
                W2 = cartJ(0,pl,singpt,2,spptop,dpptop,offset,k)./dx.^2;
                W3 = cartJ(0,pl,singpt,3,spptop,dpptop,offset,k)./dx.^3;
                A(bp+pvm,bp+pv+1)=-k*eta*...
                    (13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
```

www.manaraa.com

```
                A(bp+pvm,bp+pv+2)=-k*eta*...
                    (-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(bp+pvm,bp+pv+3)=-k*eta*...
                    (2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(bp+pvm,bp+pv+4)=-k*eta*...
                    (-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(bp+pvn,bp+pvl+1)=-k*eta*C1*sterm(pvl+1)*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2));
                A(bp+pvn,bp+pvl+2)=-k*eta*C2*sterm(pvl+2)*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2));
                A(bp+pvn,bp+pvl+3)=-k*eta*C2*sterm(pvl+3)*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2));
                A(bp+pvn,bp+pvl+4)=-k*eta*C1*sterm(pvl+4)*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2));
            end
        end
    end
end

% Left Semicircle K Outside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(2*patches+tpatches+1):2*(patches+tpatches)
        pvl = 4*(l-1);
        offset=(l-2*patches-tpatches-1)*tl+pi/2;
        if (j==l)
            for m=1:4
                pvm = pv + m;
                singpt = theta(4*patches+m)+pi/2;
                W0 = polK(0,tl,singpt,0,offset,k0,r);
                W1 = polK(0,tl,singpt,1,offset,k0,r)./dtheta;
                W2 = polK(0,tl,singpt,2,offset,k0,r)./dtheta.^2;
                W3 = polK(0,tl,singpt,3,offset,k0,r)./dtheta.^3;
                A(pvm,pv+1)=(13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(pvm,pv+2)=(-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
```

www.manaraa.com

```
                A(pvm,pv+3)=(2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(pvm,pv+4)=(-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
                A(pvm,pvm)=A(pvm,pvm)+.5;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(pvn,pvl+1)=i*C1t*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+1)).^2+(y(pvn)-y(pvl+1)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+1)).^2+(y(pvn)-y(pvl+1)).^2).*...
                    ((y(pvn)-y(pvl+1)).*cos(theta(pvl+1)+pi/2)-...
                    (x(pvn)-x(pvl+1)).*sin(theta(pvl+1)+pi/2));
                A(pvn,pvl+2)=i*C2t*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+2)).^2+(y(pvn)-y(pvl+2)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+2)).^2+(y(pvn)-y(pvl+2)).^2).*...
                    ((y(pvn)-y(pvl+2)).*cos(theta(pvl+2)+pi/2)-...
                    (x(pvn)-x(pvl+2)).*sin(theta(pvl+2)+pi/2));
                A(pvn,pvl+3)=i*C2t*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+3)).^2+(y(pvn)-y(pvl+3)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+3)).^2+(y(pvn)-y(pvl+3)).^2).*...
                    ((y(pvn)-y(pvl+3)).*cos(theta(pvl+3)+pi/2)-...
                    (x(pvn)-x(pvl+3)).*sin(theta(pvl+3)+pi/2));
                A(pvn,pvl+4)=i*C1t*besselh(1,k0*sqrt...
                    ((x(pvn)-x(pvl+4)).^2+(y(pvn)-y(pvl+4)).^2))*k0/...
                    sqrt((x(pvn)-x(pvl+4)).^2+(y(pvn)-y(pvl+4)).^2).*...
                    ((y(pvn)-y(pvl+4)).*cos(theta(pvl+4)+pi/2)-...
                    (x(pvn)-x(pvl+4)).*sin(theta(pvl+4)+pi/2));
            end
        end
    end
end

% Left Semicircle K Inside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(2*patches+tpatches+1):2*(patches+tpatches)
        pvl = 4*(l-1);
        offset=(l-2*patches-tpatches-1)*tl+pi/2;
        if (j==l)
            for m=1:4
                pvm = pv + m;
                singpt = theta(4*patches+m)+pi/2;
                W0 = polK(0,tl,singpt,0,offset,k,r);
```

www.manaraa.com

```
                W1 = polK(0,tl,singpt,1,offset,k,r)./dtheta;
                W2 = polK(0,tl,singpt,2,offset,k,r)./dtheta.^2;
                W3 = polK(0,tl,singpt,3,offset,k,r)./dtheta.^3;
                A(bp+pvm,pv+1)=(13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(bp+pvm,pv+2)=(-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(bp+pvm,pv+3)=(2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(bp+pvm,pv+4)=(-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
                A(bp+pvm,pvm)=A(bp+pvm,pvm)-.5;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(bp+pvn,pvl+1)=i*C1t*besselh(1,k*sqrt...
                    ((x(pvn)-x(pvl+1)).^2+(y(pvn)-y(pvl+1)).^2))*k/...
                    sqrt((x(pvn)-x(pvl+1)).^2+(y(pvn)-y(pvl+1)).^2).*...
                    ((y(pvn)-y(pvl+1)).*cos(theta(pvl+1)+pi/2)-...
                    (x(pvn)-x(pvl+1)).*sin(theta(pvl+1)+pi/2));
                A(bp+pvn,pvl+2)=i*C2t*besselh(1,k*sqrt...
                    ((x(pvn)-x(pvl+2)).^2+(y(pvn)-y(pvl+2)).^2))*k/...
                    sqrt((x(pvn)-x(pvl+2)).^2+(y(pvn)-y(pvl+2)).^2).*...
                    ((y(pvn)-y(pvl+2)).*cos(theta(pvl+2)+pi/2)-...
                    (x(pvn)-x(pvl+2)).*sin(theta(pvl+2)+pi/2));
                A(bp+pvn,pvl+3)=i*C2t*besselh(1,k*sqrt...
                    ((x(pvn)-x(pvl+3)).^2+(y(pvn)-y(pvl+3)).^2))*k/...
                    sqrt((x(pvn)-x(pvl+3)).^2+(y(pvn)-y(pvl+3)).^2).*...
                    ((y(pvn)-y(pvl+3)).*cos(theta(pvl+3)+pi/2)-...
                    (x(pvn)-x(pvl+3)).*sin(theta(pvl+3)+pi/2));
                A(bp+pvn,pvl+4)=i*C1t*besselh(1,k*sqrt...
                    ((x(pvn)-x(pvl+4)).^2+(y(pvn)-y(pvl+4)).^2))*k/...
                    sqrt((x(pvn)-x(pvl+4)).^2+(y(pvn)-y(pvl+4)).^2).*...
                    ((y(pvn)-y(pvl+4)).*cos(theta(pvl+4)+pi/2)-...
                    (x(pvn)-x(pvl+4)).*sin(theta(pvl+4)+pi/2));
            end
        end
    end
end

% Left Semicircle J Outside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(2*patches+tpatches+1):2*(patches+tpatches)
        pvl = 4*(l-1);
        offset=(l-2*patches-tpatches-1)*tl+pi/2;
```

www.manaraa.com

```
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = theta(4*patches+m)+pi/2;
                W0 = polJ(0,tl,singpt,0,offset,k0,r);
                W1 = polJ(0,tl,singpt,1,offset,k0,r)./dtheta;
                W2 = polJ(0,tl,singpt,2,offset,k0,r)./dtheta.^2;
                W3 = polJ(0,tl,singpt,3,offset,k0,r)./dtheta.^3;
                A(pvm,bp+pv+1)=k0*eta0*...
                    (13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(pvm,bp+pv+2)=k0*eta0*...
                    (-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(pvm,bp+pv+3)=k0*eta0*...
                    (2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(pvm,bp+pv+4)=k0*eta0*...
                    (-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(pvn,bp+pvl+1)=k0*eta0*C1t*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2));
                A(pvn,bp+pvl+2)=k0*eta0*C2t*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2));
                A(pvn,bp+pvl+3)=k0*eta0*C2t*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2));
                A(pvn,bp+pvl+4)=k0*eta0*C1t*...
                    besselh(0,k0*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2));
            end
        end
    end
end

% Left Semicircle J Inside
for j=1:(2*(patches+tpatches))
    pv = 4*(j-1);
    for l=(2*patches+tpatches+1):2*(patches+tpatches)
        pvl = 4*(l-1);
        offset=(l-2*patches-tpatches-1)*tl+pi/2;
```

www.manaraa.com

```
        if(j==l)
            for m=1:4
                pvm = pv + m;
                singpt = theta(4*patches+m)+pi/2;
                W0 = polJ(0,tl,singpt,0,offset,k,r);
                W1 = polJ(0,tl,singpt,1,offset,k,r)./dtheta;
                W2 = polJ(0,tl,singpt,2,offset,k,r)./dtheta.^2;
                W3 = polJ(0,tl,singpt,3,offset,k,r)./dtheta.^3;
                A(bp+pvm,bp+pv+1)=k*eta*...
                    (13.125.*W0-17.75.*W1+7.5.*W2-W3)./6;
                A(bp+pvm,bp+pv+2)=k*eta*...
                    (-4.375.*W0+11.75.*W1-6.5.*W2+W3)./2;
                A(bp+pvm,bp+pv+3)=k*eta*...
                    (2.625.*W0-7.75.*W1+5.5.*W2-W3)./2;
                A(bp+pvm,bp+pv+4)=k*eta*...
                    (-1.875.*W0+5.75.*W1-4.5.*W2+W3)./6;
            end
        else
            for n=1:4
                pvn = pv + n;
                A(bp+pvn,bp+pvl+1)=k*eta*C1t*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+1)).^2+...
                    (y(pvn)-y(pvl+1)).^2));
                A(bp+pvn,bp+pvl+2)=k*eta*C2t*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+2)).^2+...
                    (y(pvn)-y(pvl+2)).^2));
                A(bp+pvn,bp+pvl+3)=k*eta*C2t*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+3)).^2+...
                    (y(pvn)-y(pvl+3)).^2));
                A(bp+pvn,bp+pvl+4)=k*eta*C1t*...
                    besselh(0,k*sqrt((x(pvn)-x(pvl+4)).^2+...
                    (y(pvn)-y(pvl+4)).^2));
            end
        end
    end
end
```

A.31 `TMphysoptcompare.m`

This function calls `TMphysoptics.m`.

```
% *** compares plat solution to physical optics (TM) ***
```

www.manaraa.com

```
[physang, physint] = TMphysoptics(len, thetar, thevalr);
reflect=max(intensity)/max(physint);
figure
plot(thevaldeg,intensity,'r-',physang,physint,'g-');
xlabel('observation angle (degrees)');
ylabel('intensity');
title(['Plane wave at ',num2str(thetadeg),' degrees from grazing.']);
legend('intensity','phys optics TM plate conductor');
```

A.32 TMphysoptics.m

This plots the TM physical optics solution given by 5.17).

```
% *** analytic physical optics approximation flat plate (TM) ***

function [angle intensity] = TMphysoptics(len, thetar, thevalr)
angle = thevalr*180/pi;
field = i*sqrt(i)*4*sin(thetar).*sin(pi*len.*...
    (cos(thetar)-cos(thevalr))./(cos(thetar)-cos(thevalr));
intensity = abs(field).^2;
```

A.33 TMsolvematrix.m

This .m file solves (3.46).

```
% *** solve matrix and define incident beam ***

rhs=exp(i*k0.*(cos(thetar).*x-sin(thetar).*y)); % incident plane wave

% the following comments section is the code for a tapered beam

% rhs = erf(20/len*((len/2-5E-8)-abs((y+t/2)/tan(thetar)+len/2-x)));
% for g=1:length(rhs)
%     if rhs(g)< 0
%         rhs(g)=0;
%     end
% end
% rhs=rhs.*exp(i*k0.*(cos(thetar).*x+sin(thetar).*y));

for j=1:bp % bottom half of incident block matrix is null
    rhs(bp+j)=0;
end
```

www.manaraa.com

```
KJ=A\rhs.'; % invert matrix and solve for surface currents
K=KJ(1:bp);
J=KJ(bp+1:2*bp);

figure % plots surface current J and K magnitudes
plot(x,abs(J),'r-',x,abs(K),'b-');
xlabel('x');
ylabel('magnitude');
title(['Plane wave at ',num2str(thetadeg),' degrees from grazing.']);
legend('abs(J)','abs(K)');
```

A.34 userinputs.m

```
% *** parameters to be toggled by user ***

len = 50; % length of the straight sections of the scatterer
patches = 100; % num patches (4 pts each) for each straight section
t = 0.5; % thickness of the scatterer
tpatches = 10; % num patches (4 pts each) for each curved section
n = 1E10; % real part of complex index of refraction
beta = 0; % imaginary part of complex index of refraction
thetadeg = 90; % incident angle from grazing
rfreq = 2; % spacing per random surface point (units of wavelengths/point)
rheightbot = 0; % std dev of normal random number distribution (bot)
rheighttop = 0; % std dev of normal random number distribution (top)
fixsurfacebot = 1; % if 1, WON'T generate new random surface, else will
inpstatebot = 25; % random generator seed (when fixsurface = 1)
fixsurfacetop = 1; % if 1, WON'T generate new random surface, else will
inpstatetop = 5; % random generator seed (when fixsurface = 1)
correlatedroughness = 2; %if 1, surfaces ARE CORRELATED. If 2, not.
thrange = 20; % range of theta scan, centered around thetadeg in degrees
thsteps = 1000; % number of points to evaluate theta scan at
```

A.35 llquadzw.txt

These are the quadrature points and weights of the linlog quadrature rule. The first column contains the quadrature positions on the interval from 0 to 1. The second column contains the weights.

```
0.36787944117144233 1.0
0.08829686513765302 0.29849989370552493
```

www.manaraa.com

```
0.6751864909098872  0.7015001062944751
0.028811662530951833    0.10333070796492864
0.3040637296121377  0.4546365259700987
0.8116692253440781  0.44203276606497266
0.011802590997844918    0.043391028778414394
0.1428256799774837  0.24045209765946068
0.4892015226545745  0.42140345225977593
0.8786799740691837  0.294753421302349
0.0056522282050800975   0.021046945791854628
0.07343037174265227 0.1307055407444467
0.28495740446255813 0.28970230167131417
0.6194822640847784  0.3502203701203987
0.9157580830046983  0.2083248416719858
0.003025802137546259    0.01135133881727261
0.04097825415595061 0.07524106995491653
0.1708632955268773  0.18879004161541635
0.41325570884479323 0.2858207218272273
0.7090951467906286  0.2844864278914088
0.9382395903771671  0.1543103998937584
0.0017596521184657743   0.006632666319025705
0.024469650712513367    0.04579970797847534
0.10674805685878895 0.1238402080713182
0.2758076412959174  0.21210192602381192
0.5178551421518337  0.2613906456720077
0.7718154853623849  0.23163618029090938
0.9528413405810906  0.11859866564445172
0.0010906939419218229   0.004124301185198343
0.01544065354637409 0.029270379674687295
0.06943486210070215 0.08311406745317
0.18744324425543704 0.15372167034228773
0.373304421343093   0.21349760952226085
0.6004940136993973  0.23187027244357505
0.8168773397346666  0.19053623904036773
0.962839759269448   0.09386546033845297
7.110732887084292E-4    0.0026948911490209727
0.010194533026254907    0.019498064752635148
0.04683386722112451 0.05727368794913122
0.13036783136513153 0.11155101434875819
0.27044725718891177 0.1671748768632406
0.45831945709512795 0.20369711869471113
0.6654446330703512  0.20338245316419987
0.850116729849269   0.15865527983010627
0.9699770448705807  0.07607261324819661
```

www.manaraa.com

```
4.829617106896295E-4     0.001833400073789845
0.006988629214315765     0.013453122345991789
0.03261139659467763 0.040497194316958335
0.09282575738916596 0.0818223696589036
0.1983272568954038  0.12919234277013755
0.3488801429793532  0.16954531954725874
0.5304405557879561  0.1891002165329956
0.7167646485116551  0.17796575396147055
0.8752345575062336  0.1337247706154615
0.9752456986843929  0.06286551017703246
3.3932415013438556E-4    0.0012897363421949574
0.00494463807238221 0.009564701864401512
0.023346589135108006     0.02932398914942156
0.06757652642306404 0.06087210046049704
0.14759528165052643 0.09980233008453562
0.26692816830234756 0.13791440941591387
0.41984102340899343 0.1653115850838504
0.5909773692263063  0.17310985679474128
0.757700134502519   0.15600833092113442
0.8946267613136245  0.11400004591177339
0.9792436163211115  0.05280291397153589
2.452842649772222E-4     9.331998830671429E-4
0.003593698020213691 6    0.006977495915143716
0.017122925951596143     0.021694689021998535
0.050205612323188195     0.04597069439559112
0.11152358555736257 0.07752703869583179
0.20600300290512397 0.1112525181837396
0.3324626975136065  0.14027310600858334
0.48260670096593195 0.15751713008682963
0.6416794701239928  0.15740834222364897
0.7907090871833554  0.13728383271778585
0.9098838287655626  0.09819669547418951
0.9823479377157619  0.04496525739359079
1.8165420441651812E-4    6.91641401275959E-4
0.002672848790349611     0.005204839576802011
0.012826941530277098     0.016362516262130273
0.03799312075154549 0.03523203708791365
0.08552172002897618 0.060722504576943524
0.1606063564168977  0.08966498999672999
0.26444144362260075 0.11735480146848429
0.3931134303360546  0.1384519468445716
0.5374984949295463  0.14805886310048344
0.6842354027020987  0.14275543615167516
```

www.manaraa.com

```
0.8176214868732855  0.12137759295266985
0.9220884082406747  0.08537824790248698
0.984805887437127   0.03874458267783327
1.3736861500404895E-4   5.233455038223549E-4
0.002028188140373322    0.003958788126865641
0.009789363220505977    0.012555495593581581
0.029232593990232005    0.02738139830137573
0.06650290968077247 0.0480077023917984
0.12654623966481734 0.07248603484128054
0.21170110252714167 0.09761767696993331
0.3206896875245348  0.11947159388777363
0.4482154478386415  0.13406137156047615
0.5852679227476645  0.13806138726285247
0.7201097110240036  0.12943411529068025
0.839797854076572   0.10785514448265467
0.9319950782450486  0.0748594992983696
0.9867849013246821  0.03372644648853567
1.0578454845862939E-4   4.0321772464846156E-4
0.0015662438361678174   0.0030629784347870025
0.007595218903207093    0.009784212118766145
0.022831067393986233    0.02155875222558126
0.05238863015682001 0.03832306737088916
0.10075868520121296 0.05889819902630038
0.1707407688499433  0.08111702993925952
0.2625912061189931  0.10212210197206864
0.37353650518455805 0.1187890590304013
0.49774635841453346 0.12821031644669392
0.6267890313923734  0.1281633274170932
0.7505161034614076  0.11748946588849166
0.8582553352078606  0.09632301856959041
0.9401412912123458  0.06613453983189342
0.9884015959863418  0.029620714003535515
8.277309234714502E-5    3.1563510123328075E-4
0.0012283757008173626   0.0024060271597610428
0.005798715382018705    0.0077311486995085
0.018073696699772948    0.01717968538296474
0.0417672109064258  0.030884140854647815
0.08103861272100588 0.04815232117974444
0.1387787920651679  0.06752091080786918
0.21609636938067214 0.08692587041361871
0.31184282304458977 0.10396721868051911
0.4224350419037976  0.1162322159076941
0.5420119053274876  0.12163241491114929
```

www.manaraa.com

```
0.6629152260330544  0.11870766643887921
0.7764400251249938  0.10685433735889731
0.8737610371066735  0.08644529923711167
0.9469174698312383  0.05882560951438403
0.9897391882007426  0.026219498352017572
6.568899591380737E-5    2.505751920292486E-4
9.767296947584137E-4    0.0019156446286214377
0.004770213811772724    0.006185767294481286
0.014483378596946176    0.013842606285452683
0.03366846568173805 0.025117728612661108
0.06580382970117091 0.0396275282685842
0.1136797763609382  0.05638903969171848
0.17884094206560217 0.07391620129934323
0.2611609472708127  0.09038549529736682
0.3586093877225763  0.10385104268102938
0.4672478745480024  0.11248285338012624
0.5814636731877431  0.11479879714106866
0.6944200042692705  0.10986137432079289
0.7986749243362264  0.09741449492093224
0.8868993430301315  0.07794283737137313
0.9526123374696345  0.05264724880336799
0.9908583352563146  0.02337076481105095
5.279174605173557E-5    2.0143620079126075E-4
7.862427879699046E-4    0.0015437564620981766
0.003850381597898164    0.005005598717650643
0.011735527485174807    0.011267981518477735
0.027416783911094433    0.020606018212952788
0.05391531339051689 0.032831537629343106
0.09382888936436151 0.047290168614680925
0.14888764448116143 0.06291429809565553
0.2195887631010483  0.07832660836044944
0.3049556004234778  0.09198312367908176
0.40245324539033106 0.1023401559882098
0.5080753662724753  0.10802672394946217
0.6165981885890037  0.10800324833769989
0.7219787660623742  0.10168869527476027
0.8178584094586451  0.08904171811697714
0.8981201147555153  0.07058647416878219
0.9574428883656851  0.04738116764423372
0.9918040867027748  0.02096128902869345
4.2908722143159395E-5    1.6376626336989358E-4
6.399412468571565E-4    0.0012576852207736755
0.003141196425123936    0.00409240535186896
```

www.manaraa.com

```
0.009605331610612182    0.00925861935019232
0.022535293549746142    0.0170435140893301
0.04454764347164635 0.027381973208816104
0.07801122192897776 0.0398450224190629
0.12469388145940183 0.05366746933818868
0.18545563981132124 0.06781238201563217
0.26002231379497626 0.08106625922465209
0.3468658977314563  0.09215557770315286
0.4432074269523883  0.09987316079136845
0.545145865365175   0.10320172675452165
0.6479038981239168  0.10142215672587788
0.746169668402419   0.09419546019374829
0.8345036726013908  0.08160998405397668
0.9077732648074391  0.0641889491274348
0.9615746157343289  0.04285851482015456
0.9926104619879788  0.018905373347877945
3.523304530334011E-5    1.3449967646775757E-4
5.26093982517407E-4 0.0010347769229506144
0.002587519540581395    0.0033772636772332024
0.00793447194838037 0.0076735561935946444
0.018682888137445623    0.01420549628554197
0.03709767336975037 0.022984438463208613
0.06531248867402127 0.03373636055771364
0.10504850471155062 0.04591476307345218
0.15735969181900195 0.058740479942804
0.22243006276745467 0.071265013161102
0.2994437656540999  0.08245180897758317
0.38654244694388196 0.09126820151638737
0.4808764538267896  0.09677971590916136
0.5787479322055069  0.09823814334008972
0.6758354758400374  0.09515530305402967
0.7674824608725643  0.0873556504104574
0.84902525397032    0.07500277721227173
0.9161337032416644  0.058597295808233696
0.9651354279002556  0.038947250549611435
0.9933035364569542  0.01713720526810586
2.9198869642679487E-5   1.1148510786015597E-4
4.36443370253067E-4 8.590435637852216E-4
0.0021502879959421823   0.002811037685952878
0.006609722908765695    0.006410719227402426
0.01561241671661910l    0.011925410826761842
0.031120646748728466    0.019412825297658448
0.055042321981125235    0.028705568024102703
```

www.manaraa.com

```
0.08900640515086539 0.03941555546842202
0.1341527191510696  0.0509581964531954
0.19095511908468396 0.06259450344943483
0.25909440044489024 0.07348778073743406
0.3373933103178151  0.08277029065832578
0.42382096097702643 0.08961443188115334
0.5155682314581671  0.09330254759644417
0.6091898551232197  0.09328955264231968
0.7008033182707466  0.0892531227282597
0.7863299013823843  0.08112718381475857
0.8617595877784768  0.06911580568275878
0.9234194878883017  0.053686305684134114
0.9682253921920038  0.03554305697770823
0.9939035659323987  0.015605576492127649
2.4402521399573383E-5   9.318692630869644E-5
3.6507944429259006E-4   7.190169831866503E-4
0.0018013902301466142   0.0023581777966596414
0.005548943408984983    0.005395282384452488
0.013142593595173398    0.010078787241052649
0.026285511885560955    0.016493260809598013
0.04667674970885309 0.024544612428883075
0.07583118022660221 0.033959748365955056
0.11490643276584327 0.04430083577368803
0.1645510428434398  0.054993219153106135
0.22478654700703865 0.06536497682666716
0.2949337703271305  0.07469519836217622
0.3735902709329145  0.08226737663631158
0.4586620676893956  0.087423892942487
0.5474486738626697  0.0896174624275711
0.6367764021950574  0.08845560949871825
0.7231712035678026  0.08373474461901895
0.8030592357508834  0.07546117992967818
0.8729811715589184  0.06385740460171091
0.9298051527431974  0.04935313810311771
0.9709236690882349  0.03256272225300607
0.9944264772852992  0.014270165936646451
2.0551505076564973E-5   7.849188995113796E-5
3.077079974427594E-4    6.063482983550677E-4
0.0015203030019569552   0.0019926116661367677
0.0046917489033226414   0.004571749719998125
0.011138890110293766    0.008571805046734143
0.02234353420472049 0.014091587272414559
0.03981564567992313     0.021087216042145633
```

www.manaraa.com

```
0.06494830227465159 0.029369224044565847
0.09887519856277688 0.03861053558793372
0.14234140501622397 0.048364858571851944
0.1955972541283589  0.05809428838573062
0.25832368634835906 0.06720426525837617
0.32959525321869065 0.07508349755234922
0.40788439061292214 0.08114609956101425
0.4911077203374904  0.08487301925356212
0.5767122118260433  0.08584986165088726
0.6617962409911 0.0837984455957582
0.7432581158310259  0.0785998503881924
0.8179626721513269  0.0703072873258274
0.8829152220427237  0.059147839451851404
0.9354315904479398  0.045512966265512635
0.9732935211577413  0.029939271424517307
0.9948849321987526  0.013098879746334065
1.7430855208890156E-5   6.658148594493893E-5
2.6116539584667943E-4   5.148764046936282E-4
0.0012918438382756683   0.001694975431462046
0.003993200278433083    0.003898505876634727
0.009500373326025564    0.00733313740514584
0.019106106170368717    0.012103808129234147
0.03415141921374505 0.018200774206819818
0.05590818306366681     0.025495466765698947
0.08546162071519225 0.033744506600532666
0.1236008471359436  0.042601688682000635
0.1707251037176106  0.05163722772410114
0.2267723931765664  0.060363079872193685
0.291175827553381   0.06826275441702777
0.36285125641375177 0.0748237291954233
0.4402177554391799  0.0795704034874551
0.5212504542905049  0.08209547456859137
0.6035631015380272  0.08208771199994334
0.6845158372490904  0.07935432201613525
0.7613419948433096  0.07383642972000128
0.8312864907782004  0.06561663975718274
0.8917475710767295  0.054918148116936176
0.9404134509575381  0.042095495334097024
0.9753859815525255  0.027618344074969492
0.9952890981970386  0.01206591872777494
1.4880520564672535E-5   5.684606602517E-5
2.2309115957696828E-4   4.3999758576933795E-4
0.001104643649055819 3  0.001450718904759958
```

www.manaraa.com

```
0.0034194694688859215    0.0033440187381737836
0.00815052929503389 0.006308099547359186
0.016428937494797746     0.010448872310353418
0.02944598355986496 0.015779503663136192
0.048357569707933624     0.02221579084737619
0.07418709391973236 0.029577702414101157
0.1077329558835264   0.03759704560718378
0.14948663825808736 0.04593085159498192
0.19956673095928773 0.0541797236657
0.25767335583132533 0.06191009152230725
0.3230662664067205   0.06867907489284757
0.3945685120691865   0.07406049616510227
0.4705960495534081   0.0776705045127631
0.5492121464331801   0.0791912877674548
0.6282039422434296   0.07839145250881502
0.7051772010693157   0.07514184161385325
0.7776641844158139   0.06942582121576334
0.8432387621385732   0.06134339190482064
0.8996324161061799   0.05110885129800288
0.9448447334057146   0.03904218956400919
0.9772425752266931   0.0255554713626328
0.9956472154564406   0.011150354726707824
1.277990502112353E-5     4.882611712282679E-5
1.9170336510757313E-4    3.782320213653593E-4
9.500925875734715E-4     0.0012487899305349283
0.002944820612684768 7   0.0028841618355420155
0.007030788372688508     0.005454421567072764
0.014200696263368381     0.009063246013396312
0.025513771712971107     0.01373885468562794
0.042017594016869324     0.019429339475101854
0.0646675823368833   0.026002689786322773
0.09424888355050776 0.03325155125950687
0.1313053020728959   0.04090249161832604
0.17607955943706774 0.048629294616392926
0.22846857633414125 0.05606957634515007
0.2879966662326581   0.06284382980996468
0.35380856138343675 0.06857586410909634
0.42468311188288105 0.07291352152786823
0.4990673656711279   0.0755485340615198
0.5751296140526366   0.07623442162994834
0.6508289353325925   0.07480143545947288
0.7239978465131847   0.07116770644247822
0.792433930956133    0.06534596163723708
```

www.manaraa.com

```
0.8539957907440346  0.05744541288463155
0.9066984099774426  0.04766869427099276
0.9488030513266553  0.036304065578360405
0.9788973614378855  0.02371400879458207
0.9959660189202048  0.010335068522384952
1.103713551352541E-5    4.217146793863346E-5
1.656422417964755E-4    3.269212624093412E-4
8.216020415021419E-4    0.0010807084457197438
0.0025494796682884553   0.0025003046963233046
0.006095904770325674    0.004739171432038364
0.012334777002751235    0.007896829806912422
0.022209158531864254    0.012011048604997274
0.036666991657632465    0.017053368291382928
0.056594127368088484    0.022928227196037255
0.0827483203197991  0.029475764709293927
0.1156978250002815  0.036478185329406325
0.1557680181484857  0.043669382101920916
0.20299908429874483 0.05074734775687771
0.25711725407828756 0.05738875540187234
0.3175214062954865  0.06326497549488891
0.3832860693360594  0.06805871794109461
0.45318102438344593 0.07148045157960195
0.5257068649530954  0.07328375999578278
0.5991450474842351  0.073278842259786
0.6716202182254626  0.07134345742516422
0.7411719608883939  0.06743073798782853
0.8058326107457122  0.061573453965540766
0.8637074507970288  0.05388448929429862
0.9130534659849433  0.044553493780651626
0.9523529147278067  0.03383993365602405
0.9803784836831633  0.022063551998144925
0.9962510564210735  0.009605948118062533
9.581504540294826E-6    3.6612548401125436E-5
1.43859730972342E-4 2.8401420203780887E-4
7.140810418985653E-4    9.399065298739146E-4
0.0022181124051196626   0.0021779359871614895
0.005310623349551285    0.0041364975218919535
0.010763282863543742    0.006909881006477732
0.01941709664552334 0.010541554467408827
0.03212928039570467 0.015020003587664315
0.04971735113505941 0.020277354552636222
0.07290327583815621 0.026190869079152104
0.10226053605612309 0.03258726143914562
```

www.manaraa.com

```
0.13816684818777977 0.039249650541058674
0.18076502709328449 0.0459268320477723
0.22993410759763303 0.0523444421975684
0.2852723616537579  0.05821749187755426
0.3460932910462451  0.06326368104720421
0.4114350629380239  0.06721686312534288
0.4800832173648081  0.06984001841328775
0.5506058423079049  0.07093711558630479
0.6213998134008274  0.07036328985729079
0.690746160233329   0.06803284332190654
0.7568721752752096  0.06392467371306317
0.818017546303043   0.05808485789633293
0.8725015859898183  0.05062625156229036
0.9187885672794224  0.04172511689981153
0.9555482769476675  0.03161499219387206
0.9817093578807328  0.020578711781648277
0.9965069310786334  0.008951277015839957
8.358077509026045E-6    3.1939873035105296E-5
1.2554048888479551E-4   2.479137244802797E-4
6.235601768550263E-4    8.212524499953166E-4
0.001938724421546737    0.0019056604571111278
0.004647250066169768    0.0036259591906748503
0.009432588534094975    0.006070685456033144
0.017046103765818593    0.009286323199827869
0.028262948282374228    0.013273509464399598
0.04383521630338818 0.017985406000957865
0.06444459322880657 0.02332832701324242
0.09065676039683991 0.029164611502398678
0.12287975151077384 0.03531762714177794
0.16132829512679062 0.041578695929541984
0.20599592315026294 0.047715644349988956
0.256636289558867   0.0534826063313559
0.31275474135296133 0.058630649252800406
0.3736107328053689  0.06291875423514844
0.4382311944183664  0.06612466371556447
0.5054344808808607  0.06805511270134333
0.5738640494069412  0.06855498501014845
0.6420305821915061  0.06751498117035128
0.7083608838863567  0.06487744852011802
0.7712515741745153  0.06064010368525596
0.8291253708925305  0.05485746982062305
0.8804876317530744  0.04763995291815405
0.9239808022776038  0.03915059474494858
```

www.manaraa.com

```
0.9584345268713504  0.029599700389502312
0.982909591902549   0.019238158990629358
0.9967374878834097  0.008361262760591268
7.323797442726057E-6    2.798921543095474E-5
1.1004470045777488E-4   2.1736552650254156E-4
5.469183261839674E-4    7.207035865343896E-4
0.0017018575191016412   0.0016744609650549897
0.0040838636097143746   0.0031912824064114654
0.00830004117688234 0.005353788313529335
0.01502978156079995     0.008209621368081956
0.02495392361575455 0.011768029213084813
0.03878338617106294 0.015998143504891402
0.05715089848117639 0.020829041093624295
0.08060574147265517 0.026151597661309315
0.10957039423451795 0.03182206826945638
0.14430837300150082 0.03766725599980672
0.18489794942753196 0.0434910622632234
0.2312130015482552  0.04908215322840304
0.28291196019720777 0.05422242866126262
0.33943548119085265 0.058695944290976916
0.4000131131094502  0.06229791811493648
0.46367885693930627 0.06484344570723305
0.5292951427410363  0.06617555985125438
0.595584395325645   0.06617229527720048
0.661167040396915 2  0.06475245892292092
0.7246045281760328  0.06187985834995464
0.7844457347349415  0.05756580366344204
0.839274951478663   0.05186976919246568
0.8877595976173431  0.04489817849556722
0.9286957959632276  0.03680136250036949
0.9610500591874097  0.02776887037741359
0.983995703521289   0.018023873784160743
0.9969459586797631  0.007827670195496758
6.44463908909534E-6 2.4630773027179577E-5
9.686621774891527E-5    1.9137662273558862E-4
4.816819582518684E-4    6.350507043469919E-4
0.0014999965889179307   0.001477150434040435
0.0036029866934082482   0.0028194250315611673
0.007331487526183152    0.004738642849324728
0.013288790937388884    0.007282334606375033
0.02210977445477698 0.010465740311830184
0.03442763716738958 0.014270093541274297
0.05083962342542433 5   0.018642321184494808
```

www.manaraa.com

```
0.07187316553209268 0.02349618057236937
0.09795386996515765 0.028714798829564063
0.12937670842642685 0.034154523525701135
0.16628165755792673 0.039649941386525764
0.20863486398623857 0.045019875028788243
0.25621620091483305 0.050074128236226916
0.3086138369326747  0.0546207197521741
0.36522616658480095 0.058473325133734516
0.4252711661093298  0.061458636706516925
0.4878029477403128  0.0634233534654301
0.5517350034300943  0.06424052583346931
0.6158693648089452  0.06381500402639148
0.6789306710234296  0.06208777247066567
0.7396039390316426  0.05903899500849397
0.7965746798072114  0.05468964493437383
0.848569904905668   0.04910164859116934
0.8943985257099043  0.042376530257259334
0.9329896675388936  0.034652612463584274
0.963427521027151   0.026100932902171414
0.9849816856202325  0.0169205511250949
0.9971350758159141  0.007343533691284935
5.693507553617974E-6    2.1761164055045618E-5
8.560159573353658E-5    1.6915499261886826E-4
4.258762287233882E-4    5.61727684501124E-4
0.00132712765917052 0.0013079612243402968
0.0031905900860272943   0.0024998693909955893
0.006499405687857923    0.004208572973776043
0.011795827238539608    0.006480636971386528
0.019655221294758576    0.00933535740654361
0.03065788112328044 0.012763108992411112
0.04535974839690379 0.016724823114855496
0.06426319118538036 0.02115262339348098
0.08778899146107959 0.025951602901735246
0.11625065798021576 0.03100268446615335
0.14983210129530916 0.03616646488949099
0.18856958964372547 0.0412879069464199
0.23233875181548602 0.046201710948873466
0.28084721138506386 0.05073817175613247
0.33363323252701743 0.05472930827349748
0.3900705384885743  0.058015041444713075
0.4493792376890749  0.06044919391374767
0.5106425676490226  0.06190509002704642
0.5728289517763088  0.062228054847920882
```

www.manaraa.com

```
0.6348186663715512  0.06150208118747326
0.695434242349751   0.05952814015136212
0.7534735845173743  0.056351288112238974
0.8077446861041707  0.05199920760257677
0.8571007517459341  0.046534505296957185
0.9004745214141074  0.040053313906039494
0.9369106149155023  0.032682745583628214
0.9655948084867538  0.024577340675839082
0.9858794546214816  0.015915126617754603
0.9973071619647179  0.006902929510146736
5.048671225530156E-6    1.9297457666551922E-5
7.592696444468548E-5    1.5006445217311603E-4
3.7791346241676326E-4   4.986686581544643E-4
0.0011784055984128848   0.001162234866824391
0.0028353397690508535   0.0022240829455816833
0.00578148310299468 0.003749972268157451
0.0105052367158275  0.005784944764260339
0.0175286495906909  0.008350921956018357
0.02738344639395215 0.011445148512244447
0.04058594875834076 0.01503954315240933
0.05761181455802881 0.01908098582077551
0.07887153221800249 0.023492537167566538
0.10468758668630647 0.02817555358251664
0.13527416100979434 0.03301262945615323
0.1707201542655499  0.03787126909017844
0.21097618574336136 0.042608164756552866
0.25584612054364847 0.04707393571733511
0.30498349777577727 0.051118166320363916
0.3578930746118391  0.054594570163980935
0.41393752335865586 0.057366102169505585
0.47234914053418436 0.05930984139960413
0.5322462528324465  0.06032147458829721
0.5926538408606599  0.060319223367216575
0.6525277533213859  0.059247076643813706
0.7107817570461654  0.05707721288726724
0.7663165664167193  0.05381152442969862
0.8180499228546598  0.04948218642062711
0.8649467539730966  0.04415124596242478
0.9060484348228406  0.03790924201808314
0.9405002037372857  0.03087290746358595
0.9675758684112931  0.023182081392524697
0.986699208859247   0.014996395777839542
0.9974642016568276  0.006500794370598407
```

www.manaraa.com

```
4.492582250114292E-6    1.7172679304899794E-5
6.758059575572542E-5    1.3359060384817906E-4
3.3650888809959485E-4   4.4419980374195845E-4
0.0010499021353660143   0.0010361857468708133
0.002528023320435537    0.001985104225513129
0.005159526268716118    0.0033516821742054104
0.009385217676274153    0.0051790914708461176
0.015679386318356966    0.007490825055190534
0.02452934266645772 0.010289252380423388
0.036413876595788465    0.013554906946019076
0.05178138156178547 0.017246534361831555
0.07102840952273803 0.021301904199598883
0.09447941145287024 0.02563936355540312
0.12236838379790164 0.030160085596936913
0.15482308276190737 0.03475094320851927
0.1918523877533697  0.03928791693340649
0.23333729525727967 0.04363992843320197
0.2790259080769515  0.04767297622908651
0.3285326560770279  0.05125444003568264
0.38134184744374316 0.054257413892054684
0.4368155085220992  0.05656492674060197
0.4942053300856662  0.05807391216022255
0.5526684030184446  0.05869879653191566
0.6112863012478005  0.05837458676305215
0.6690869584611931  0.05705935444292948 4
0.7250686913244835  0.054736032445896546
0.7782256487003623  0.05141346193984086
0.8275739162158773  0.047126651847918026
0.8721774802963752  0.04193623848569642
0.9111732569795943  0.03592716041210877
0.9437944212805213  0.029206596081483273
0.969391347490734   0.02190127843922232
0.9874497169493747  0.014154706370706323
0.9976078986595814  0.006132779806720094
4.010980937010523E-6    1.5332397234438866E-5
6.034964939613452E-5    1.1931492593617741E-4
3.006164317823871E-4    3.9695672760045017E-4
9.384132431425842E-4    9.267197421896937E-4
0.0022611095400135563   0.0017772227746893745
0.00461861900484087 0.0030045066022608246
0.008409605067284434    0.004649674798059875
0.014065562778553399    0.006737018509498977
0.022033294995912316    0.009272689375873869
```

www.manaraa.com

```
0.03275635525132841 0.012243966032232172
0.04665606589350032 0.015619141021227308
0.06411273725125864 0.019348035517579257
0.08544771113246052 0.023363129600666766
0.11090682999411454 0.02758127649152916
0.14064589224618732 0.03190595079768488
0.1747185956171352  0.03622996395123618
0.21306739609436043 0.04043856520513074
0.25551762193067923 0.04441283422919665
0.30177508327446023 0.04803326190503012
0.35142731117230114 0.05118340965187706
0.4039484482916271  0.05375353472157103
0.4587077011448546  0.05564406947289345
0.5149811533355105  0.056768846661298424
0.5719666347923079  0.05705797013951586
0.6288012463411473  0.056460240836605585
0.6845810552492498  0.05494506115149926
0.7383824081635055  0.052503756563515094
0.7892842553283274  0.04915027086621056
0.8363908458024856  0.04492121049117853
0.8788541388474076  0.03987523352541093
0.9158952828215542  0.034091800463071815
0.9468245423000019  0.027669330040016784
0.9710591208350612  0.020722862042010938
0.9881385513295243  0.013381708290320484
0.9977397223132375  0.005795134478147276
3.5922101619651165E-6   1.3732108891794563E-5
5.406001459928744E-5    1.0689489750098045E-4
2.6937940675786E-4  3.558208922532232E-4
8.413107806345918E-4    8.312940284225005E-4
0.0020284079291637675   0.0015957301491226373
0.004146468278634052    0.0027008309644590083
0.00755766009472711     0.004185537950438294
0.012652429214120192    0.006074377251209505
0.01984338042479044 0.008376249172915429
0.02954030576611134 0.011083701267245666
0.04213814986781696 0.01417270963432014
0.057999650819164236    0.01760297802293338
0.07743943135708277 0.021318748957966288
0.10070914907829896 0.02525010592337761
0.12798418012139726 0.02931473101863138
0.15935226828463847 0.033420068914478684
0.19480451615015862 0.03746583575917197
```

www.manaraa.com

```
0.23422902807460846 0.041346801285878976
0.27740743881254554 0.044955764052191434
0.32401447827047725 0.04818663374863231
0.37362063486703984 0.05093753102273438
0.4256978897607274  0.05311381438350015
0.4796284044303055  0.05463094550681008
0.5347159573745233  0.0554171086077809
0.5901998445712299  0.055415506358023135
0.6452708851764309  0.054586263909582745
0.6990891108983395  0.05290788368346951
0.7508026664090912  0.050378206374442715
0.7995674105808424  0.04701484776216947
0.8445666854181124  0.042855096046912816
0.8850307121526595  0.03795527030005922
0.9202550828887067  0.032389557570530275
0.9496178437251875  0.026248367723604223
0.9725947248110737  0.019636297231851126
0.9887722785483385  0.012670149050527267
0.9978609451874788  0.00548460846796077
3.226686457573206E-6    1.2335224180663856E-5
4.856846483460022E-5    9.604863746104893E-5
2.4209236129432013E-4   3.1987034393523916E-4
7.564273981732722E-4    7.478079625638684E-4
0.0018248035783040375   0.00143672498324935
0.0037328925541173907   0.0024343215700164734
0.006808730931906212    0.0037773559183020733
0.011411019240858909    0.00549018318278394
0.01791613780358366 0.007583656813751746
0.026704259511674152    0.010054425686042047
0.038144967371106696    0.01288464925126189
0.05258280108487706 0.0160421412109823
0.07032316143181556 0.019480940172855267
0.09161899364155222 0.023142293741918948
0.1166586515486491  0.026956030848005148
0.14555531355510146 0.030842286140396913
0.17833828017559233 0.034713530297428644
0.21494643231180102 0.038476851379244704
0.2552240707893379  0.04203642513933591
0.2989192927047948  0.04529610670615138
0.34568499056051055 0.04816207240433957
0.39508248792942063 0.05054543880893071
0.44658775248622856 0.05236478646371404
0.49960005567948124 0.05354851804026866
```

www.manaraa.com

```
0.553452880082619   0.05403698500426674
0.6074268124420659  0.053784322974890726
0.6607641043869704  0.05275994374549067
0.712684535239432   0.05094964116820868
0.7624021736971873  0.048356278544815536
0.8091426084236563  0.045000036540351424
0.8521602025742459  0.040918212684588604
0.8907549245945378  0.036164576100614985
0.9242883178858846  0.0308082946070695
0.9521981972744588  0.02493246919618202
0.9740117133643122  0.018632357847543392
0.9893566152067715  0.012013705660225
0.9979726738672983  0.005198374998632167
2.9064889999183383E-6   1.1111498020814642E-5
4.3756555187085526E-5   8.654294758211389E-5
2.181713507684719E-4    2.8834125901866986E-4
6.819666388585224E-4    6.745175994636655E-4
0.001646049417416248    0.0012969594683643958
0.0033694191725911838   0.0021996871941245932
0.006150506259632302    0.0034173041840497888
0.010317086394059708    0.004973706399681155
0.016215050305720364    0.006881088664859645
0.024196340364669933    0.009139276349461293
0.0346063902196419  0.011735402262553579
0.047771418759439216    0.014643939392870741
0.06398593561749387 0.017827090650581124
0.08350081182350523 0.02123552723740389
0.10651225614558955 0.02480945795227101
0.13315201549217368 0.02848000287227122
0.16347908702545033 0.03217083666729067
0.19747319117339482 0.03580005952746861
0.2350302095155078  0.039282247497665006
0.2759597407374179  0.04253062908633532
0.3199848728276893  0.0454593314783604
0.36674421186597206 0.047985637626957824
0.41579614863955794 0.05003219498442467
0.4666252854755458  0.051529117670423626
0.5186508886331845  0.05241592644438983
0.5712371778649886  0.05264327487993322
0.6237052157405819  0.052174415530143306
0.6753461163269572  0.05098636648452949
0.7254352569711802  0.04907074638115896
0.7732471491921629  0.04643425457355204
```

www.manaraa.com

```
0.818070605810994    0.04309878139267165
0.8592238319863987   0.03910114632866111
0.8960690681770235   0.034492465463456454
0.9280264237498098   0.029337168468064296
0.9545865629965556   0.023711693964504958
0.9753219528347452   0.017702938002118817
0.989896556448008    0.011406845694533636
0.9980758742640766   0.004933966036975236
2.6250380049060306E-6    1.0035803672763636E-5
3.952584288194972E-5     7.818394206024305E-5
1.9713061908584883E-4    2.6059773706434814E-4
6.164322644568964E-4     6.099683057997869E-4
0.001488602407152558     0.0011737177499974122
0.0030489656799804646    0.0019924890020021325
0.005569527805236629     0.0030987923460615717
0.00935025335166042 0.004515864554034832
0.014709322438596299     0.006256772260140321 5
0.02197262337704276 0.008323783969298953
0.03146275924687664 0.010708028253918334
0.04348786037097826 0.01338944844877953
0.0583305012168030 4 0.016337056645075194
0.07623701926160918 0.01950948219633557
0.09740752481387079 0.02285580247012381
0.12198687489377129 0.026316636397053096
0.15005686142734578 0.029825474662758533
0.18162983476345546 0.033310214327070375
0.21664394860243608 0.03669486038992738
0.2549601726946185   0.03990135248790202
0.2963611761116979   0.042851471607851035
0.3405521375964034   0.04546877952859435
0.3871634916079997   0.04768054270318155
0.4357555703956822   0.04941959249943179
0.4858250549530007   0.050626075121258635
0.5368131022194043   0.051249047104168
0.5881149735369208   0.05124787595258945
0.6390909512014589   0.05059341017435854
0.6890782969271264   0.04926888855385195
0.7374039790003694   0.04727056485385428
0.783397874526041    0.044608031095245575
0.8264061399911663   0.04130422997546781
0.8658044377747142   0.03739515471431906
0.9010107084983299   0.03292924263209471
0.931497189691723    0.02796647754976951
```

www.manaraa.com

```
0.9568014020396931  0.022577228847915024
0.976535867057191   0.01684089416657822
0.9903964823751481  0.010844711927275194
0.9981713925366468  0.004689219043119115
2.376841438791435E-6    9.087166484795518E-6
3.5794119865953565E-5   7.080965702766152E-5
1.7856419152485403E-4   2.361079242496758E-4
5.585723523572761E-4    5.52941324982586E-4
0.0013494936126551916   0.0010647190766918551
0.002765586187887195    0.0018089882755908076
0.00505522381417743 0.00281624906827513
0.008493328161829778    0.004108945006397529
0.01337289057267608 0.005700655250238588
0.019995795834284628    0.007595509475343502
0.02866317888553044 0.009787840935178711
0.039665557957306426    0.012262087395473313
0.05327299439566269 0.014992943683671163
0.06972553012579277 0.01794576352818758
0.08922414836933326 0.02107720234078324
0.11192249194361464 0.02433608676462221
0.13791955651040663 0.027664491309405653
0.1672535538992926  0.030998997354574984
0.19989711362906382 0.03427210534230175
0.2357539596251301  0.03741376721044341
0.2746571646162645  0.04035300312219564
0.31636904760570544 0.04301956440574828
0.3605827410273972  0.04534560337566032
0.40692541463682524 0.04726731039834552
0.45496310378579513 0.0487264791953468
0.5042070514270982  0.04967196293601935
0.554121436898801   0.050060986119925356
0.6041323311128448  0.04986028053168121
0.65363768801575    0.04904701759057694
0.7020181568119037  0.04760951411983075
0.748648479051785   0.04554769381702903
0.7929092197838029  0.042873292399205315
0.8341985729238538  0.039609800403023025
0.8719439780738345  0.035792143845247024
0.9056132894254163  0.03146610940093139
0.9347252475155071  0.02668752792915317
0.9588590232728008  0.021521241144567268
0.9776626420648422  0.016039912381280514
0.9908602466200785  0.010323026080510124
```

www.manaraa.com

```
0.9982599724712418  0.004462232713798836
2.157294294967718E-6    8.247998064669447E-6
3.249242492341324E-5    6.428418613566664E-5
1.6213125702082334E-4   2.144250288537577E-4
5.073348171894436E-4    5.024111649828837E-4
0.0012262245702609825   9.680402542894589E-4
0.0025142684157429956   0.0016460238429515082
0.0045986639271920916   0.0025649477663168556
0.007731752222114276    0.003746377524778571
0.012183619308440531    0.005204131180138624
0.01823406169532701 0.006943737612674629
0.026164108774468106    0.00896209369114927
0.036247305768803896    0.011247329342493178
0.04874096765935924 0.013778884736250585
0.06387761628347004 0.016527798925958622
0.08185680970236266 0.019457204425379484
0.10283756502684768 0.02252301745216903
0.12693156333985806 0.025674809052852143
0.15419730844217597 0.02885683813354116
0.18463539025057193 0.03200922365594455
0.21818497926401015 0.0350692300035839
0.2547216511319526  0.037972636852987865
0.29405661062529    0.04065516286000546
0.33593735289505877 0.04305391113875743
0.3800497675156456  0.0451088039009851
0.4260216581791441  0.04676397375119811
0.473427618773648   0.04796907999595092
0.5217951756678397  0.04868051990506759
0.5706120770376905  0.04886250712378974
0.6193345836662097  0.04848799232739538
0.6673965924243118  0.047539404669456824
0.7142194041296029  0.04600919652477553
0.7592219321277744  0.043900178381311154
0.8018311371054893  0.04122563539905801
0.8414924675881453  0.03800922203646386
0.8776800844821679  0.03428463617907871
0.9099066520432526  0.030095079426688518
0.9377324871519848  0.025492516100898983
0.9607738754999374  0.020536753082482755
0.9787103978021419  0.015294396174353337
0.991291250400277   0.009838008165234459
0.9983422699889635  0.004251330025551719
```

www.manaraa.com