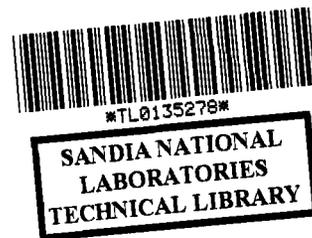


SANDIA REPORT
SAND2004-6577
Unlimited Release
Printed December 2004

SAND 2004-6577 C. 2
REFERENCE COPY

FDTD Simulation Tools for UWB Antenna Analysis

Robert W. Brocato



Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation, A Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under Contract DE-AC04-94AL85000.

Approved for public release; further dissemination unlimited.



LIBRARY DOCUMENT
DO NOT DESTROY
RETURN TO
LIBRARY VAULT

SCANNED

TOTAL PAGES: 53
COPY _____

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof or any of their contractors or subcontractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from

U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-mail: reports@adonis.osti.gov
Online ordering: <http://www.doe.gov/bridge>

Available to the public from

U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd.
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online ordering: <http://www.ntis.gov/ordering.htm>



Sandia National Laboratories

**SAND2004-6577
Unlimited Release
Printed December 2004**

FDTD Simulation Tools for UWB Antenna Analysis

**Robert W. Brocato
Sandia National Laboratories
Opto and RF Microsystems
P.O. Box 5800
Albuquerque, NM 87185**

Abstract:

This paper describes the development of a set of software tools useful for analyzing ultra-wideband (UWB) antennas and structures. These tools are used to perform finite difference time domain (FDTD) simulation of a conical antenna with continuous wave (CW) and UWB pulsed excitations. The antenna is analyzed using spherical coordinate-based FDTD equations that are derived from first principles. The simulation results for CW excitation are compared to simulation and measured results from published sources; the results for UWB excitation are new.

**LIBRARY DOCUMENT
DO NOT DESTROY
RETURN TO
LIBRARY VAULT**

This page is left intentionally blank.

Contents

Section	Page
Nomenclature	6
Figures and Tables	6
Introduction	7
Conical Antenna Geometry and Simulation Parameters	7
Derivation of Spherical FDTD Equations	10
General 3-D Spherical FDTD Equations	12
FDTD Equations for a Conical Antenna	14
Driving Signal	17
Perfectly Matched Layer (PML) Spherical Equations	19
Perfectly Matched Layer (PML) Parameters	22
Convergence Criteria	24
Results: Closed Form Simulations	25
FDTD Simulation Results, CW Excitation	27
Simulation Results: UWB Pulse Excitation	29
Impedance Simulation Results	30
3-D Plots of 2-D FDTD Simulation Results	32
FDTD Simulation Results: PML Effectiveness	34
Conclusions	36
References	36
Appendix: Software Files	37
Distribution	53

Nomenclature

2-D	-	Two dimensional
3-D	-	Three dimensional
BPSK	-	Binary Phase Shift Keying
CDMA	-	Code Division Multiple Access
cm	-	centimeters
DS-CDMA	-	Direct Sequence Code Division Multiple Access
DSSS	-	Direct Sequence Spread Spectrum
FCC	-	Federal Communications Commission
FDTD	-	Finite difference time domain
GHz	-	Giga Hertz (billion cycles/sec)
MathCAD	-	Equation solving engine available from MathSoft
MATLAB	-	Simulation engine available from MathWorks
MHz	-	Mega Hertz (million cycles/sec)
mm	-	millimeters
NIST	-	National Institute of Standards and Technology
PML	-	Perfectly Matched Layer
RF	-	Radio Frequency
SNR	-	Signal to Noise Ratio
SPICE	-	Simulation Program with Integrated Circuit Emphasis
TEM	-	Transverse electro-magnetic
UWB	-	Ultra-Wide Band

Figures and Tables

Figure 1: Conical Antenna Geometry	-	page 8
Figure 2: Conical Antenna Simulation Space	-	page 9
Figure 3: 3-D FDTD Lattice Unit Cell in Spherical Coordinates	-	page 12
Figure 4: Conical Antenna Coordinates	-	page 15
Table 1: Conductivities for a 20 layer parabolic PML	-	page 23
Table 2: Conductivities for a 20 layer geometric profile PML	-	page 24
Figure 5: E-field for infinite cone antenna, closed form solution	-	page 26
Figure 6: 2-D view of E-field for infinite cone antenna, closed form solution	-	page 26
Figure 7: E_{θ} -field for long cone antenna, FDTD solution	-	page 27
Figure 8: E_{θ} -field for long cone antenna, side view, FDTD solution	-	page 28
Figure 9: Comparison of FDTD far field pattern with published data	-	page 28
Figure 10: Sine and cosine driven UWB pulses with $f_c=6.5\text{GHz}$	-	page 29
Figure 11: 3-D view of 2-D E_{θ} Field from UWB pulse propagating out...	-	page 30
Table 3: Antenna Impedance Simulation Results	-	page 31
Series 1: 3-D plots of 2-D FDTD simulation results	-	pages 32-33
Series 2: FDTD Simulation Results: PML Effectiveness	-	pages 34-35

Introduction

In support of ongoing communication work in UWB at Sandia National Laboratories, efforts at building UWB antennas have been undertaken in the recent past with only a modicum of success. This was done because of a pressing need for an adequate UWB antenna, and because it is impossible to buy a high performance UWB antenna, as of this writing. The primary purpose of this work is to provide a thorough understanding of one high performance UWB antenna to aid in Sandia's ongoing communication work.

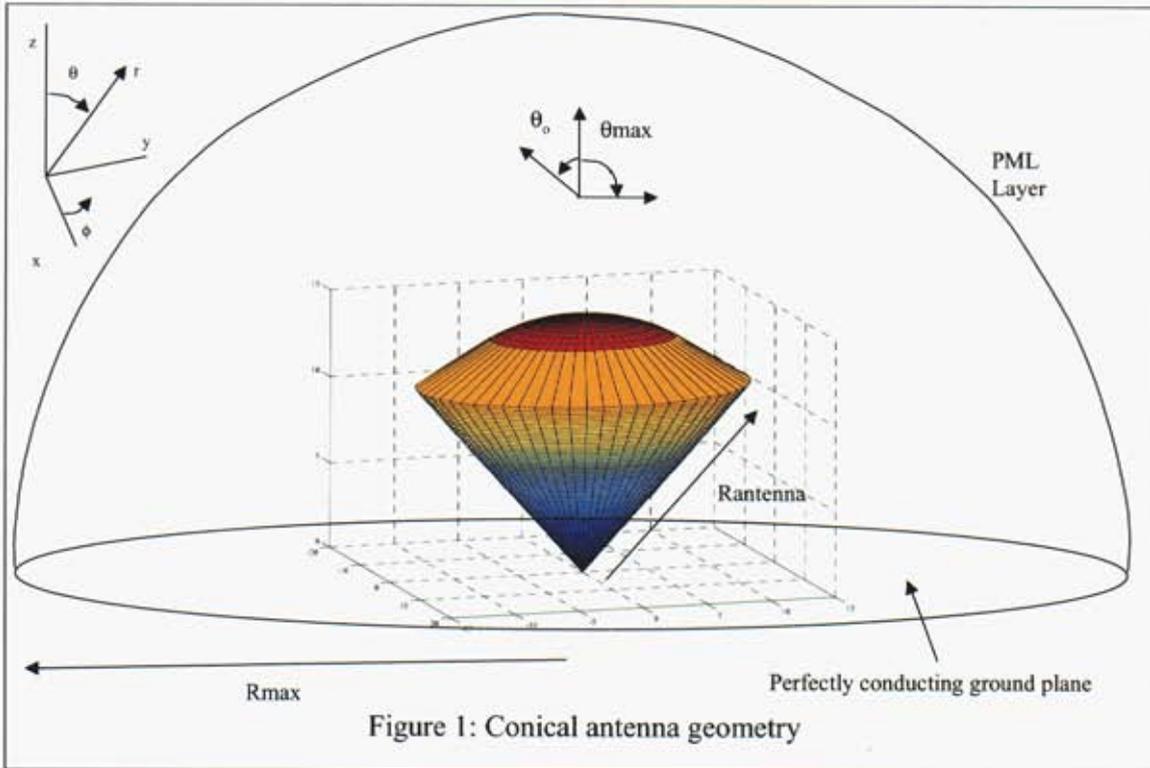
It was recently pointed out by Andrews that "there are almost no companies selling commercial UWB antennas" [1]. He also mentions that "a very important but frequently overlooked concept about UWB antennas is that the commonly accepted principle of antenna transmit-receive reciprocity does not exactly hold true for their time domain performance." As a result, the proper transmission and reception of UWB signals is difficult. To confront half of this problem, the conical antenna is recommended by NIST for transmission of UWB signals [2]. The apparent difficulties of transmitting and receiving UWB signals pose a simulation problem that is perhaps best solved by a time domain technique. Simulation results using the FDTD method for the conical antenna were found for CW excitation [3]. Simulation plots have not been published for the case of FCC-compatible UWB pulse excitation, though a large body of literature exists for this antenna.

The second motivation for this work is to develop general-purpose software tools for analyzing antennas in the UWB operating regime. For the conical antenna, these tools require the development of propagation equations for solving problems with spherical symmetry in FDTD. A number of papers have been written on the problem of FDTD in spherical coordinates [4] [5]. The most comprehensive work on this was done by Holland [6]. However, in spite of a reasonable basis of work establishing the capabilities of FDTD in spherical coordinates, complex spherically symmetric propagation problems continue to be conducted in FDTD using rectangular coordinates. Hertel and Smith recently reported on the FDTD analysis of a complex conical spiral antenna using rectangular coordinates [7]. As a result, the antenna geometry was discretized with a jagged appearance. The antenna design lends itself to working in spherical coordinates, but the authors chose to work in rectangular coordinates. So, the second objective of this work is to develop a complete set of software tools for the spherical FDTD, perfectly matched layer (PML), and wave generation equations. This has been done by deriving these equations from first principles wherever possible, and by testing these equations and comparing the simulation results with published results.

This work will describe the geometry of the conical antenna problem, will cover the derivation of the necessary 3-D and 2-D FDTD and PML equations and the UWB wave generation equations, and will test these equations using the conical antenna geometry. The conical antenna FDTD results for CW excitation will be compared to reported simulation results and related closed form solution results. The FDTD results for UWB pulse excitation will be compared to reported measurement results.

Conical Antenna Geometry and Simulation Parameters

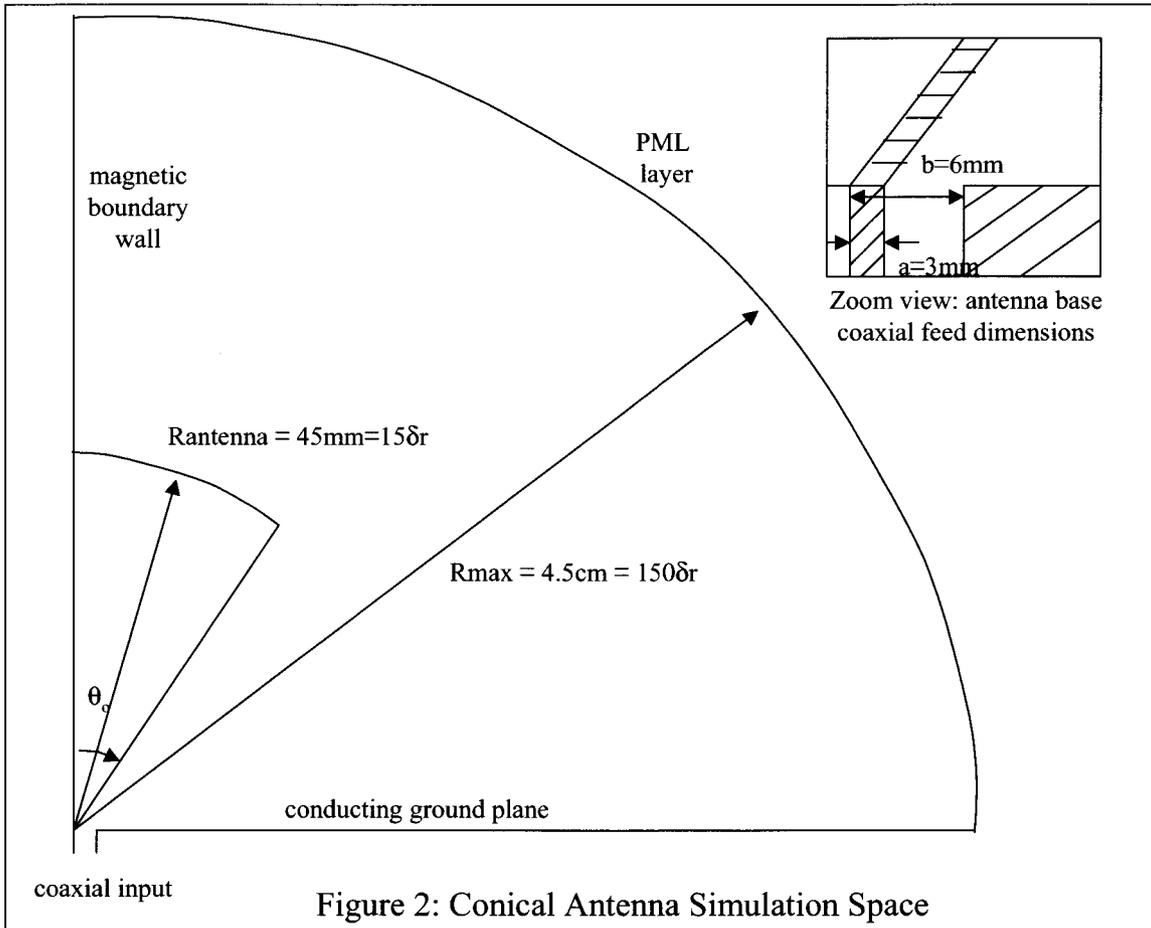
The conical antenna geometry is shown in figure 1. It consists of a conductor that extends from the origin radially out at an angle θ_0 for a distance of R_{antenna} . The angle θ_0 and the length of the antenna affect both the antenna radiation pattern and the input impedance of the antenna. The top of the antenna consists of a conductor that follows the sweep of the θ axis at a constant distance, R_{antenna} , from the origin. The antenna sits on top of a perfectly conducting ground plane that extends 360° in all directions for a distance of R_{max} . Just before the maximum radial distance, R_{max} , is reached, the problem space is terminated in a PML section of thickness N_{pml} .



The FCC mandated band for UWB transmission is 3.1-10.6 GHz [8]. This means that the maximum frequency of operation of a UWB antenna must be 10.6 GHz. For choice of the minimum time and spatial steps and to enable convergence of the FDTD algorithm, 10.6 GHz is considered the frequency of operation. The minimum wavelength, λ_{min} , is therefore given by $\lambda_{\text{min}} = c/f_{\text{min}} = c/10.6\text{GHz} = 28\text{mm}$. The minimum radial step is taken to be about one tenth of this minimum wavelength, $\delta r = \lambda_{\text{min}}/10 = 3 \text{ mm}$, and $\delta\theta = 1^\circ$ was also chosen.

The nominal or center frequency of a UWB pulse is about 6.5 GHz; the nominal wavelength is then $\lambda_{\text{nom}} = c/f_{\text{nom}} = c/6.5\text{GHz} \approx 45\text{mm} = 15\delta r$. The antenna dimensions are taken from the nominal frequency. The antenna length for most calculations is taken to be about $1\lambda_{\text{nom}}$, or about $10\delta r = 45 \text{ mm}$. The simulation space, defined radially by R_{max} , is taken to be $R_{\text{max}} = 10\lambda_{\text{nom}} = 4.5 \text{ cm}$. Both antenna length and semi-angle, θ_0 , are varied in a series of simulations to enable analysis of input impedance for varying conditions.

The problem can be reduced to a two dimensional simulation and can be further cut in half (figure2). The reduction from three to two dimensions can be accomplished, since the antenna and its solution are symmetric in ϕ . The cutting in half of the two dimensional problem can be done by splitting the solution space along the z-axis. The solution space is terminated by a perfectly conducting ground plane on the bottom and by an absorbing PML layer starting at $R_{\max} - N_{\text{pml}}$ and extending to R_{\max} . The PML layer absorbs the outward traveling waves and provides a means of stopping them with a minimal amount of reflection.



A coaxial input signal line feeds the base of the antenna. The output of the coaxial line and the base of the antenna are taken to be the origin of the spherical solution space. The UWB signal is input to the antenna as a voltage across the coaxial line from the inner to the outer conductor. As will be elaborated on in another section, the coaxial input signal produces a spherical wavefront that propagates up the antenna and radiates out into space. The dimensions of the coaxial feed line are shown in an insert in figure 2. The inner (signal) conductor has a diameter 'a' and the outer (ground) conductor is at a distance 'b' from the inner conductor edge. These dimensions are close to values obtained from a standard SMA-type connector and were chosen so that $a = \delta r$ and $b = 2\delta r$. This gives an initiating wave front represented by 90 points from the $\delta\theta$ steps.

Derivation of Spherical FDTD Equations

To derive the field equations for \mathbf{E} and \mathbf{H} , one must start with two of Maxwell's equations and the medium dependent equations as follows:

$$1) \quad \nabla \times \mathbf{E} = -\delta\mathbf{B}/\delta t - \mathbf{J}_m$$

$$2) \quad \nabla \times \mathbf{H} = \delta\mathbf{D}/\delta t + \mathbf{J}_e$$

$$3) \quad \mathbf{B} = \mu \mathbf{H}$$

$$4) \quad \mathbf{D} = \epsilon \mathbf{E}$$

$$5) \quad \mathbf{J}_e = \sigma \mathbf{E}$$

$$6) \quad \mathbf{J}_m = \sigma^* \mathbf{M}$$

Combining these, one obtains two vector versions of Maxwell's equations:

$$7) \quad \nabla \times \vec{E} = -\sigma^* \cdot \vec{H} - \mu \cdot \frac{\delta \vec{H}}{\delta t}$$

$$8) \quad \nabla \times \vec{H} = \sigma \cdot \vec{E} + \epsilon \cdot \frac{\delta \vec{E}}{\delta t}$$

These two vector equations can be expanded using the spherical ∇ operator:

$$9) \quad \nabla = \frac{\delta}{\delta r} \cdot \vec{r} + \frac{1}{r} \cdot \frac{\delta}{\delta \theta} \cdot \vec{\theta} + \frac{1}{r \cdot \sin \theta} \cdot \frac{\delta}{\delta \phi} \cdot \vec{\phi}$$

with \mathbf{r} , $\boldsymbol{\theta}$, and $\boldsymbol{\phi}$ being the spherical unit vectors. These are applied using the following forms:

$$10) \quad \nabla \times \mathbf{E} = \frac{1}{r \cdot \sin \theta} \cdot \begin{vmatrix} \mathbf{r} & \boldsymbol{\theta} & \boldsymbol{\phi} \\ \frac{\partial}{\partial r}(r \cdot \sin \theta) & \frac{\partial}{\partial \theta}(\sin \theta) & \frac{\partial}{\partial \phi} \\ E_r & E_\theta & E_\phi \end{vmatrix}$$

and

$$11) \quad \nabla \times \mathbf{H} = \frac{1}{r \cdot \sin \theta} \cdot \begin{vmatrix} \mathbf{r} & \boldsymbol{\theta} & \boldsymbol{\phi} \\ \frac{\partial}{\partial r}(r \cdot \sin \theta) & \frac{\partial}{\partial \theta}(\sin \theta) & \frac{\partial}{\partial \phi} \\ H_r & H_\theta & H_\phi \end{vmatrix}$$

These two cross products each produce a vector equation which is another way of writing equations 10 and 11:

$$12) \quad -\mu \delta \mathbf{H} / \delta t = \sigma^* \mathbf{H} + \nabla \times \mathbf{E} \quad \text{or, expanding...}$$

$$13) \quad -\mu \delta \mathbf{H} / \delta t = 1/(r \sin \theta) ((\delta / \delta \theta (\sin \theta E_\phi) - \delta E_\theta / \delta \phi) \mathbf{r} + (\delta E_r / \delta \phi - \sin \theta \delta / \delta r (r E_\phi)) \boldsymbol{\theta} + (\sin \theta \delta / \delta r (r E_\theta) - \delta / \delta \theta (\sin \theta E_r)) \boldsymbol{\phi}) + \sigma^* \mathbf{H}$$

$$14) \quad \epsilon \delta \mathbf{E} / \delta t = -\sigma \mathbf{E} + \nabla \times \mathbf{E} \quad \text{or, expanding...}$$

$$15) \quad \epsilon \delta \mathbf{E} / \delta t = 1/(r \sin \theta) ((\delta / \delta \theta (\sin \theta H_\phi) - \delta H_\theta / \delta \phi) \mathbf{r} + (\delta H_r / \delta \phi - \sin \theta \delta / \delta r (r H_\phi)) \boldsymbol{\theta} + (\sin \theta \delta / \delta r (r H_\theta) - \delta / \delta \theta (\sin \theta H_r)) \boldsymbol{\phi}) - \sigma \mathbf{E}$$

The vector equations (13) and (15) produce six scalar Maxwell's equations from equating the \mathbf{r} , $\boldsymbol{\theta}$, and $\boldsymbol{\phi}$ vector terms each into a separate scalar equation:

$$16) \quad \delta H_r / \delta t = 1/\mu (\delta E_\theta / (r \sin \theta \delta \phi) - 1/(\sin \theta) \delta / (r \delta \theta) (\sin \theta E_\phi)) + (\sigma^* / \mu) H_r$$

$$17) \quad \delta H_\theta / \delta t = 1/\mu (1/r \delta / \delta r (r E_\phi) - \delta E_r / (r \sin \theta \delta \phi) + (\sigma^* / \mu) H_\theta)$$

$$18) \quad \delta H_\phi / \delta t = 1/\mu (1/(\sin \theta) \delta / (r \delta \theta) (\sin \theta E_r) - 1/r \delta / \delta r (r E_\theta)) + (\sigma^* / \mu) H_\phi$$

$$19) \quad \delta E_r / \delta t = 1/\epsilon (1/(\sin \theta) \delta / (r \delta \theta) (\sin \theta H_\phi) - \delta H_\theta / (r \sin \theta \delta \phi)) - (\sigma / \epsilon) E_r$$

$$20) \quad \delta E_\theta / \delta t = 1/\epsilon (\delta H_r / (r \sin \theta \delta \phi) - 1/r \delta / \delta r (r H_\phi)) - (\sigma / \epsilon) E_\theta$$

$$21) \quad \delta E_\phi / \delta t = 1/\epsilon (1/r \delta / \delta r (r H_\theta) - 1/(\sin \theta) \delta / (r \delta \theta) (\sin \theta H_r)) - (\sigma / \epsilon) E_\phi$$

These six scalar equations (16)-(21) must be converted into six FDTD equations to enable iterative time-stepping. The FDTD equations are found by referring to the unit cell diagram (fig.3) and using the following spherical, central finite difference equations:

$$22) \quad \frac{\partial F^n}{\partial t}(i, j, k) = \left[\frac{F^{n+\frac{1}{2}}(i, j, k) - F^{n-\frac{1}{2}}(i, j, k)}{\partial t} \right] + O(\partial t^2) \quad \dots \text{ for time derivatives}$$

$$23) \quad \frac{\partial F^n}{\partial r}(i, j, k) = \left[\frac{F^n(i + \frac{1}{2}, j, k) - F^n(i - \frac{1}{2}, j, k)}{\partial r} \right] + O(\partial r^2) \quad \dots \text{ for radial derivatives}$$

$$24) \quad \frac{\partial F^n}{(r \cdot \partial \theta)}(i, j, k) = \left[\frac{F^n(i, j + \frac{1}{2}, k) - F^n(i, j - \frac{1}{2}, k)}{i \cdot \partial r \cdot \partial \theta} \right] + O(\partial \theta^2) \quad \dots \text{ for elevation derivatives}$$

$$25) \frac{\partial F^n}{(r \cdot \sin \theta \cdot \partial \phi)}(i, j, k) = \left[\frac{F^n(i, j, k + 1/2) - F^n(i, j, k - 1/2)}{i \cdot \partial r \cdot \sin(j \cdot \partial \theta) \cdot \partial \phi} \right] + O(\partial \phi^2)$$

.... for azimuthal derivatives

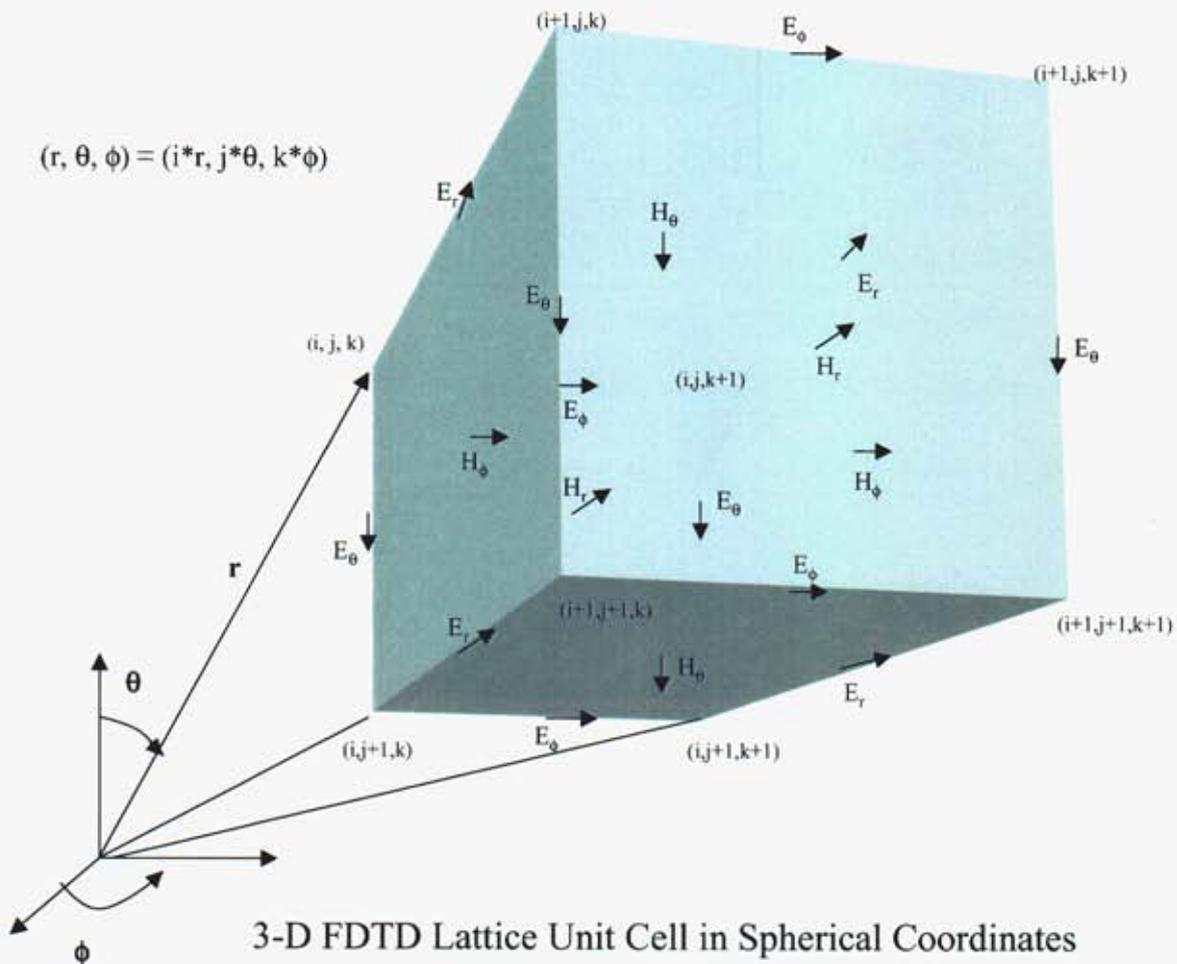


Figure 3

General 3-D Spherical FDTD Equations

Using the finite difference equations in the six scalar Maxwell's equations and referring to the unit cell diagram gives the following six finite difference equations in spherical coordinates:

$$26) H_r^{n+1/2}(i, j+1/2, k+1/2) = [1 + \delta t \sigma^*(i, j+1/2, k+1/2) / \mu(i, j+1/2, k+1/2)] H_r^{n-1/2}(i, j+1/2, k+1/2) + \delta t / \mu(i, j+1/2, k+1/2) * [\{E_\theta^n(i, j+1/2, k+1) - E_\theta^n(i, j+1/2, k)\} / (i \delta r \delta \phi \sin((j+1/2) \delta \theta)) - \{\sin((j+1) \delta \theta) / \sin(j \delta \theta) E_\phi^n(i, j+1, k+1/2) - E_\phi^n(i, j, k+1/2)\} / (i \delta r \delta \theta)].$$

$$\begin{aligned}
27) \quad H_{\theta}^{n+1/2}(i+1/2, j, k+1/2) &= [1 + \delta t \sigma^*(i+1/2, j, k+1/2) / \mu(i+1/2, j, k+1/2)] H_{\theta}^{n-1/2}(i+1/2, j, k+1/2) \\
&+ \delta t / \mu(i+1/2, j, k+1/2) * [\{(i+1) E_{\phi}^n(i+1, j, k+1/2) - i E_{\phi}^n(i, j, k+1/2)\} / i \delta r \\
&- \{E_r^n(i+1/2, j, k+1) - E_r^n(i+1/2, j, k)\} / ((i+1/2) \delta r \delta \phi \sin((j+1/2) \delta \theta))]. \\
28) \quad H_{\phi}^{n+1/2}(i+1/2, j+1/2, k) &= [1 + \delta t \sigma^*(i+1/2, j+1/2, k) / \mu(i+1/2, j+1/2, k)] H_{\phi}^{n-1/2}(i+1/2, j+1/2, k) \\
&+ \delta t / \mu(i+1/2, j+1/2, k) * [\{\sin((j+1) \delta \theta) / \sin(j \delta \theta) E_r^n(i+1/2, j+1, k) - E_r^n(i+1/2, j, k)\} / ((i+1/2) \delta r \delta \theta) \\
&- \{(i+1) E_{\theta}^n(i+1, j+1/2, k) - i E_{\theta}^n(i, j+1/2, k)\} / i \delta r]. \\
29) \quad E_r^{n+1}(i+1/2, j, k) &= [1 - \delta t \sigma(i+1/2, j, k) / \epsilon(i+1/2, j, k)] E_r^n(i+1/2, j, k) + \delta t / \epsilon(i+1/2, j, k) * \\
&[\{\sin((j+1/2) \delta \theta) / \sin((j-1/2) \delta \theta) H_{\phi}^{n+1/2}(i+1/2, j+1/2, k) - H_{\phi}^{n+1/2}(i+1/2, j-1/2, k)\} / ((i+1/2) \delta r \delta \theta) \\
&- \{H_{\theta}^{n+1/2}(i+1/2, j, k+1/2) - H_{\theta}^{n+1/2}(i+1/2, j, k-1/2)\} / ((i+1/2) \delta r \delta \phi \sin(j \delta \theta))]. \\
30) \quad E_{\theta}^{n+1}(i, j+1/2, k) &= [1 - \delta t \sigma(i, j+1/2, k) / \epsilon(i, j+1/2, k)] E_{\theta}^n(i, j+1/2, k) + \delta t / \epsilon(i, j+1/2, k) * \\
&[\{H_r^{n+1/2}(i, j+1/2, k+1/2) - H_r^{n+1/2}(i, j+1/2, k-1/2)\} / (i \delta r \delta \phi \sin((j+1/2) \delta \theta)) \\
&- \{(i+1/2) H_{\phi}^{n+1/2}(i+1/2, j+1/2, k) - (i-1/2) H_{\phi}^{n+1/2}(i-1/2, j+1/2, k)\} / (i-1/2) \delta r]. \\
31) \quad E_{\phi}^{n+1}(i, j, k+1/2) &= [1 - \delta t \sigma(i, j, k+1/2) / \epsilon(i, j, k+1/2)] E_{\phi}^n(i, j, k+1/2) + \delta t / \epsilon(i, j, k+1/2) * \\
&[\{(i+1/2) H_{\theta}^{n+1/2}(i+1/2, j, k+1/2) - (i-1/2) H_{\theta}^{n+1/2}(i-1/2, j, k+1/2)\} / (i-1/2) \delta r \\
&- \{\sin((j+1/2) \delta \theta) / \sin((j-1/2) \delta \theta) H_r^{n+1/2}(i, j+1/2, k+1/2) - H_r^{n+1/2}(i, j-1/2, k+1/2)\} / (i \delta r \delta \theta)].
\end{aligned}$$

Equations (26)-(31) are in the traditional Yee form, using half steps for all time and spatial steps. In order to both use these equations in a simulation and compare the results to published equations, they need to be simplified and re-arranged. First, the equations can be simplified with the use of some replacement coefficients. The following, simplifying coefficients are applied in a manner similar to that given in [9]:

$$\begin{aligned}
32) \quad C_a &= \frac{1 - \frac{\sigma \cdot \delta t}{2 \cdot \epsilon}}{1 + \frac{\sigma \cdot \delta t}{2 \cdot \epsilon}} & 33) \quad C_b &= \frac{\frac{\delta t}{\epsilon \cdot \delta r}}{1 + \frac{\sigma \cdot \delta t}{2 \cdot \epsilon}}
\end{aligned}$$

$$\begin{aligned}
34) \quad D_a &= \frac{1 - \frac{\sigma^* \cdot \delta t}{2 \cdot \mu}}{1 + \frac{\sigma^* \cdot \delta t}{2 \cdot \mu}} & 35) \quad D_b &= \frac{\frac{\delta t}{\mu \cdot \delta r}}{1 + \frac{\sigma^* \cdot \delta t}{2 \cdot \mu}}
\end{aligned}$$

Applying the coefficients in (32)-(35) and converting the spatial half steps to whole steps for matrix indexing gives the following 3-D spherical FDTD equations. Half time steps and spatial steps that are not used for matrix indexing are kept since these can be implemented in the simulation.

$$\begin{aligned}
36) \quad H_r^{n+1/2}(i, j, k) &= D_a H_r^{n-1/2}(i, j, k) + D_b [\{E_{\theta}^n(i, j+1, k) - E_{\theta}^n(i, j, k)\} / (i \delta \phi \sin((j+1/2) \delta \theta)) \\
&- \{\sin((j+1) \delta \theta) / \sin(j \delta \theta) E_{\phi}^n(i, j+1, k) - E_{\phi}^n(i, j, k)\} / (i \delta \theta)].
\end{aligned}$$

- 37) $H_\theta^{n+1/2}(i, j, k) = D_a H_\theta^{n-1/2}(i, j, k) + D_b [\{((i+1)/i) E_\phi^n(i+1, j, k) - E_\phi^n(i, j, k) - \{E_r^n(i, j, k+1) - E_r^n(i, j, k)\}/((i+1/2) \delta\phi \sin((j+1/2)\delta\theta))\}].$
- 38) $H_\phi^{n+1/2}(i, j, k) = D_a H_\phi^{n-1/2}(i, j, k) + D_b [\{\sin((j+1)\delta\theta)/\sin(j\delta\theta) E_r^n(i, j+1, k) - E_r^n(i, j, k)\}/((i+1/2) \delta\theta) - \{(i+1)/i E_\theta^n(i+1, j, k) - E_\theta^n(i, j, k)\}].$
- 39) $E_r^{n+1}(i, j, k) = C_a E_r^n(i, j, k) + C_b [\{\sin((j+1/2)\delta\theta)/\sin((j-1/2)\delta\theta) H_\phi^{n+1/2}(i, j, k) - H_\phi^{n+1/2}(i, j-1, k)\}/((i+1/2) \delta\theta) - \{H_\theta^{n+1/2}(i, j, k) - H_\theta^{n+1/2}(i, j, k-1)\}/((i+1/2) \delta\phi \sin(j\delta\theta))].$
- 40) $E_\theta^{n+1}(i, j, k) = C_a E_\theta^n(i, j, k) + C_b [\{H_r^{n+1/2}(i, j, k) - H_r^{n+1/2}(i, j, k-1)\}/(i \delta\phi \sin((j+1/2)\delta\theta)) - \{(i+1/2)/(i-1/2) H_\phi^{n+1/2}(i, j, k) - H_\phi^{n+1/2}(i-1, j, k)\}].$
- 41) $E_\phi^{n+1}(i, j, k) = C_a E_\phi^n(i, j, k) + C_b [\{((i+1/2)/(i-1/2)) H_\theta^{n+1/2}(i, j, k) - H_\theta^{n+1/2}(i-1, j, k)\} - \{\sin((j+1/2)\delta\theta)/\sin((j-1/2)\delta\theta) H_r^{n+1/2}(i, j, k) - H_r^{n+1/2}(i, j-1, k)\}/(i \delta\theta)].$

Equations (36)-(41) are the full 3-D spherical FDTD equations and are very similar to the 6 spherical FDTD equations given by Holland [6]. There are slight differences between the way Holland implements time and spatial stepping. He drops the half spatial steps throughout his equations. The equations implemented in (36)-(41) convert the half steps to whole steps only as necessitated for matrix indexing. The additional accuracy lost by Holland's conversion of these terms is slight but may be noticeable in features that are small relative to the spatial steps. Experimentation with these equations demonstrated that there are situations near boundaries and the origin where these small inaccuracies can be noticeable. Also, equations (36)-(41) differ in that the $\sin((j+1/2)\delta\theta)/\sin((j-1/2)\delta\theta)$ terms in the E_ϕ and H_ϕ field equations are kept but do not appear in equations (6) or (9) of [6]. Experimentation with the FDTD simulations revealed that these terms provide only a small variation to the field values. In the simulation of the conical antenna, H_ϕ was found to change by less than 2% with the addition of the $\sin((j+1/2)\delta\theta)/\sin((j-1/2)\delta\theta)$ term.

FDTD Equations for a Conical Antenna

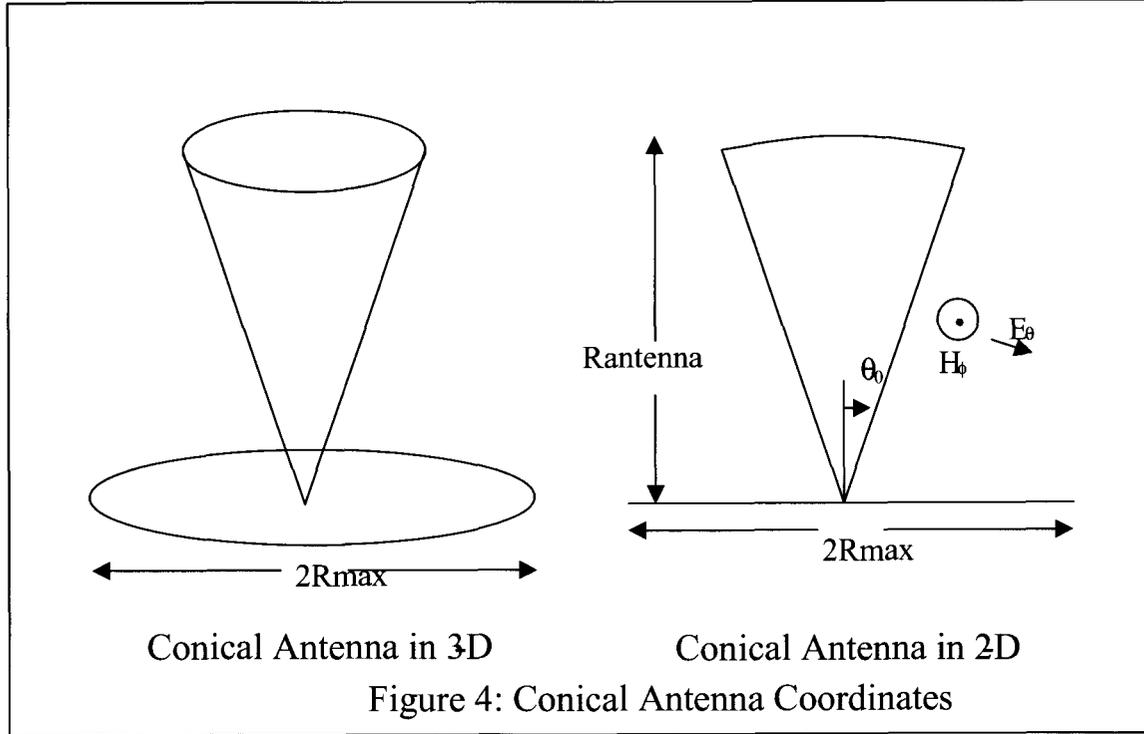
The conical antenna is rotationally symmetric, so it can be reduced to a two dimensional problem. Using the rotational symmetry, the $\delta/\delta\phi$ terms are zero, and the only field components that exist are E_r , E_θ , and H_ϕ . The six scalar Maxwell's equations then reduce to the following three simplified equations:

$$42) \quad \frac{\delta H_\phi}{\delta t} = \frac{1}{\mu} \cdot \left[-\sigma^* \cdot H_\phi + \frac{1}{\sin \theta} \cdot \frac{\delta}{r \cdot \delta\theta} (\sin \theta \cdot E_r) - \frac{1}{r} \cdot \frac{\delta}{\delta r} \cdot (r \cdot E_\theta) \right]$$

$$43) \quad \frac{\delta E_r}{\delta t} = \frac{1}{\varepsilon} \cdot \left[-\sigma \cdot E_r + \frac{1}{\sin \theta} \cdot \frac{\delta}{r \cdot \delta\theta} \cdot (\sin \theta \cdot H_\phi) \right]$$

$$44) \quad \frac{\delta E_\theta}{\delta t} = \frac{1}{\epsilon} \cdot \left[-\sigma \cdot E_\theta - \frac{1}{r} \cdot \frac{\delta}{\delta r} (r \cdot H_\phi) \right]$$

In (42)-(44), as in the 3-D versions, the magnetic and electric conductivities are kept to enable computation of a PML layer at the periphery of the problem space. These three scalar equations produce three FDTD equations in spherical coordinates.



Applying the central difference equations (22)-(25) to the 2-D spherical Maxwell's equations in (42)-(44) gives the following equations, using the traditional half time step notations in all locations. Here the coordinates for σ , μ , σ^* , and ϵ are the same as for the field value on the left hand side of the equation:

$$45) \quad H_\phi^{n+1/2}(i + \frac{1}{2}, j + \frac{1}{2}) = \left(1 - \frac{\sigma^* \cdot \delta t}{\mu} \right) H_\phi^{n-1/2}(i + \frac{1}{2}, j + \frac{1}{2}) + \frac{\delta t}{\delta r \cdot \mu} \left[\frac{\left(\frac{\sin((j+1) \cdot \delta\theta)}{\sin(j \cdot \delta\theta)} \right) \cdot E_r^n(i + \frac{1}{2}, j + 1) - E_r^n(i + \frac{1}{2}, j)}{((i + \frac{1}{2})\delta\theta)} - \left(\frac{i+1}{i} \right) \cdot E_\theta^n(i + 1, j + \frac{1}{2}) + E_\theta^n(i, j + \frac{1}{2}) \right]$$

$$46) \quad E_r^{n+1}(i + \frac{1}{2}, j) = \left(1 - \frac{\sigma \cdot \delta t}{\epsilon}\right) E_r^n(i + \frac{1}{2}, j) + \frac{\delta t}{\delta r \cdot \epsilon} \cdot \left[\frac{\left(\frac{\sin((j + \frac{1}{2}) \cdot \delta \theta)}{\sin((j - \frac{1}{2}) \cdot \delta \theta)}\right) \cdot H_\phi^{n+1/2}(i + \frac{1}{2}, j + \frac{1}{2}) - H_\phi^{n+1/2}(i + \frac{1}{2}, j - \frac{1}{2})}{((i + \frac{1}{2}) \delta \theta)} \right]$$

$$47) \quad E_\theta^{n+1}(i, j + \frac{1}{2}) = \left(1 - \frac{\sigma \cdot \delta t}{\epsilon}\right) E_\theta^n(i, j + \frac{1}{2}) + \frac{\delta t}{\delta r \cdot \epsilon} \cdot \left[H_\phi^{n+1/2}(i - \frac{1}{2}, j + \frac{1}{2}) - \left(\frac{i + \frac{1}{2}}{i - \frac{1}{2}}\right) \cdot H_\phi^{n+1/2}(i + \frac{1}{2}, j + \frac{1}{2}) \right]$$

Applying the simplifying coefficients in (32)-(35) to equations 45-47 and eliminating the half spatial steps in the matrix indexing gives the following, final 2-D FDTD equations:

$$48) \quad H_\phi^{n+1/2}(i, j) = D_a \cdot H_\phi^{n-1/2}(i, j) + D_b \cdot \left[\frac{\left(\frac{\sin((j+1) \cdot \delta \theta)}{\sin(j \cdot \delta \theta)}\right) \cdot E_r^n(i, j+1) - E_r^n(i, j)}{((i + \frac{1}{2}) \delta \theta)} - \left(\frac{i+1}{i}\right) \cdot E_\theta^n(i+1, j) + E_\theta^n(i, j) \right]$$

$$49) \quad E_r^{n+1}(i, j) = C_a \cdot E_r^n(i, j) + C_b \cdot \left[\frac{\left(\frac{\sin((j + \frac{1}{2}) \cdot \delta \theta)}{\sin((j - \frac{1}{2}) \cdot \delta \theta)}\right) \cdot H_\phi^{n+1/2}(i, j) - H_\phi^{n+1/2}(i, j - 1)}{((i + \frac{1}{2}) \delta \theta)} \right]$$

$$50) \quad E_\theta^{n+1}(i, j) = C_a \cdot E_\theta^n(i, j) + C_b \cdot \left[H_\phi^{n+1/2}(i - 1, j) - \left(\frac{i + \frac{1}{2}}{i - \frac{1}{2}}\right) \cdot H_\phi^{n+1/2}(i, j) \right]$$

In (48)-(50) the coordinates for σ , μ , σ^* , and ϵ are the same as the coordinates of the field value on the left side of the equations. These are the equations used in the FDTD simulation of the conical antenna. They are almost identical to the 2-D FDTD equations used by Liu and Grimes [3]. The equations in [3] have at least one typographical error and appear to use values for I and J that start are incremented by 1. This is a standard Matlab requirement for matrix indexing, and

it appears that the equations shown in [3] are intended to use 1 as the origin. The equations shown in (48)-(50) use 0 as the origin, and the Matlab implementation of these equations must be incremented by 1 for all I and J.

Another small difference with the FDTD equations in [3] is the absence of the $\sin((j+1/2)\delta\theta)/\sin((j-1/2)\delta\theta)$ term in the equation for H_ϕ . This produces a only small difference to the simulated field values. Its absence from the equations in [3] leads me to believe that its presence in (48) may be an error. I convinced myself that it belongs in the equation only after going through the derivation for the equation three times.

Driving Signal

The base of the antenna is driven by a voltage signal through a coaxial line. The coaxial line has an inner conductor diameter “a” and an outer conductor diameter “b”. The voltage on the coaxial line is a solution to the wave equation in cylindrical coordinates:

$$51) \quad \frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial V}{\partial r} \right) + \frac{1}{r^2} \frac{\partial^2 V}{\partial \theta^2} = 0$$

Note that the θ units of this equation are different from the θ units of the spherical coordinates of the antenna. This equation is subject to the boundary conditions $V(a, \theta) = V_0$ and $V(b, \theta) = 0$. In actuality, V_0 will vary with time, but the time variation can be introduced once the static solution is found. Since the coaxial line is axially symmetric, the solution is independent of θ . The wave equation in cylindrical coordinates simplifies to:

$$52) \quad \frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial V}{\partial r} \right) = 0$$

This has the general solution:

$$53) \quad V(r, \theta) = C \cdot \ln(r) + D$$

Applying the boundary conditions gives the final solution for the coaxial driving voltage:

$$54) \quad V(r, \theta) = \frac{V_0 \cdot \ln\left(\frac{r}{b}\right)}{\ln\left(\frac{b}{a}\right)}$$

The electric field of the incoming wave is the gradient of this potential:

$$55) \quad E = -\nabla V$$

Applying the spherical gradient operator gives:

$$56) \quad E_r = -\frac{V_0}{r \cdot \ln\left(\frac{b}{a}\right)}$$

With the E field being independent of θ . This is the electric field across the input to the base of the cone. Here the radial component of electric field from this TEM wave transforms from the

cylindrical coordinate system of the coaxial line into the E_θ component driving the base of the antenna in the spherical coordinate system in which the antenna is to be simulated. This change is made by taking $r = b \sin\theta$ in the source equation. Then the driving electric field at the base of the antenna is:

$$57) \quad E_\theta = \frac{-V_0}{b \cdot \sin\theta \cdot \ln\left(\frac{b}{a}\right)}$$

In accordance with the example set by Liu and Grimes [3], to reduce nonphysical reflections, the voltage seen by the antenna must take into consideration any impedance mismatch between the input line and the antenna. This voltage seen by the antenna at its base is then given by:

$$58) \quad V_m = V_s(t) - R_s \cdot I_m(t)$$

Here, $V_s(t)$ is the time varying input voltage from the voltage source, R_s is the impedance of the voltage source (which is assumed to be matched to the coaxial line), and $I_m(t)$ is the input current at the base of the antenna produced by the input voltage. The input current is given by Ramo, Whinnery, and Van Duzer [10] as:

$$59) \quad I_m = 2 \cdot \pi \cdot r \cdot \sin(\theta) \cdot H_\phi$$

The electric driving field, E_θ , resulting from the input voltage, $V_m(t)$, is found by replacing the constant voltage, V_0 , from the static solution (57) with the time varying voltage, $V_m(t)$. The driving electric field at the base of the antenna is then:

$$60) \quad E_\theta = -\frac{V_s(t) - R_s \cdot 2\pi \cdot r \sin(\theta) \cdot H_\phi}{b \cdot \sin(\theta) \cdot \ln\left(\frac{b}{a}\right)}$$

This can be converted into an FDTD equation to drive the base of the symmetrically reduced structure.

$$61) \quad E_\theta^n(i, j(\theta) - j(90^\circ)) = -\frac{V_s(n \cdot \Delta t) - R_s \cdot 2\pi \cdot i \Delta r \cdot \sin((j + \frac{1}{2}) \cdot \Delta\theta) \cdot H_\phi^{n-1/2}(i + \frac{1}{2}, j + \frac{1}{2})}{b \cdot \sin(j(\theta) \cdot \Delta\theta) \cdot \ln\left(\frac{b}{a}\right)}$$

This field is a small, spherical source extending from the inner conductor of the coaxial line to the outer, grounded conductor, which forms part of the ground plane. Then, as the voltage, $V_s(t)$, steps forward in time it drives the base of the antenna with this spherical electric field. The antenna radiates the resulting wave in accordance with the FDTD equations given in (48)-(50).

The input impedance of the antenna is given by the ratio of $V_m(t)$ to $I_m(t)$. The input voltage is found from the driving field given by (61) by using the integral equation version of (55) across the base of the antenna:

$$62) \quad V_m = \int_{\theta_0}^{\pi/2} b \cdot E_\theta \Delta\theta$$

The current, $I_{in}(t)$, is found by averaging the current from (59) across the input to the antenna at its base. The input impedance is then given by:

$$63) \quad Z_{in} = \frac{V_{in}}{I_{in}}$$

This input impedance is calculated for each value of $R_{antenna}$ and θ_0 during the simulation and is compared to theoretical and measured values in a later section.

Perfectly Matched Layer (PML) Spherical Equations

The artificially imposed boundary at the edge of the simulation space needs to terminate in an absorbing layer to prevent spurious reflections from interfering with the outgoing wave. Because the problem is in spherical coordinates, an absorbing boundary layer is spherically symmetric. That is, the PML layer only needs to vary with the radius, r . In order to form a PML, the following steps must be taken [11] [12]:

- 1) Resolve H_ϕ into the $H_{\phi r}$ and $H_{\phi\theta}$ components in the coupled Maxwell equations.
- 2) Create FDTD equations from the revised Maxwell equations.
- 3) Modify the C_a , C_b , D_a , and D_b (equations 32-35) time constants to include exponential difference time advance [13].
- 4) Calculate conductivity values for the PML layers using the matching condition.

Performing the first step involves splitting the magnetic field into one component due to r and one due to θ . Starting with the Maxwell equations given in (42)-(44), and using a PML with electric and magnetic conductivities for r and θ respectively given by $(\sigma_r, \sigma_r^*, \sigma_\theta, \sigma_\theta^*)$:

$$64) \quad \frac{\delta H_{\phi\theta}}{\delta t} = \frac{1}{\mu} \cdot \left[-\sigma_\theta^* \cdot H_{\phi\theta} + \frac{1}{\sin \theta} \cdot \frac{\delta}{r \cdot \delta\theta} (\sin \theta \cdot E_r) \right]$$

$$65) \quad \frac{\delta H_{\phi r}}{\delta t} = \frac{1}{\mu} \cdot \left[-\sigma_r^* \cdot H_{\phi r} - \frac{1}{r} \cdot \frac{\delta}{\delta r} \cdot (r \cdot E_\theta) \right]$$

$$66) \quad \frac{\delta E_r}{\delta t} = \frac{1}{\varepsilon} \cdot \left[-\sigma_\theta \cdot E_r + \frac{1}{\sin \theta} \cdot \frac{\delta}{r \cdot \delta\theta} \left\{ \sin \theta \cdot (H_{\phi r} + H_{\phi\theta}) \right\} \right]$$

$$67) \quad \frac{\delta E_\theta}{\delta t} = \frac{1}{\varepsilon} \cdot \left[-\sigma_r \cdot E_\theta - \frac{1}{r} \cdot \frac{\delta}{\delta r} \left\{ r \cdot (H_{\phi r} + H_{\phi\theta}) \right\} \right]$$

These four Maxwell's equations can be converted into four FDTD equations using (22)-(25) in accordance with the second step given above. The results are as follows:

$$68) \quad H_{\phi\theta}^{n+1/2}(i + \frac{1}{2}, j + \frac{1}{2}) = \left(1 - \frac{\sigma_{\theta}^* \cdot \delta t}{\mu}\right) H_{\phi\theta}^{n-1/2}(i + \frac{1}{2}, j + \frac{1}{2}) + \frac{\delta t}{\delta r \cdot \mu} \cdot \left[\frac{\left(\frac{\sin((j+1) \cdot \delta\theta)}{\sin(j \cdot \delta\theta)}\right) \cdot E_r^n(i + \frac{1}{2}, j+1) - E_r^n(i + \frac{1}{2}, j)}{((i + \frac{1}{2})\delta\theta)} \right]$$

$$69) \quad H_{\phi r}^{n+1/2}(i + \frac{1}{2}, j + \frac{1}{2}) = \left(1 - \frac{\sigma_r^* \cdot \delta t}{\mu}\right) H_{\phi r}^{n-1/2}(i + \frac{1}{2}, j + \frac{1}{2}) + \frac{\delta t}{\delta r \cdot \mu} \cdot \left[-\left(\frac{i+1}{i}\right) \cdot E_{\theta}^n(i+1, j + \frac{1}{2}) + E_{\theta}^n(i, j + \frac{1}{2}) \right]$$

$$70) \quad E_r^{n+1}(i + \frac{1}{2}, j) = \left(1 - \frac{\sigma_{\theta} \cdot \delta t}{\mu}\right) E_r^n(i + \frac{1}{2}, j) + \frac{\delta t}{\delta r \cdot \epsilon} \cdot \left[\frac{\left(\frac{\sin((j + \frac{1}{2}) \cdot \delta\theta)}{\sin((j - \frac{1}{2}) \cdot \delta\theta)}\right) \cdot [H_{\phi r}^{n+1/2}(i + \frac{1}{2}, j + \frac{1}{2}) + H_{\phi\theta}^{n+1/2}(i + \frac{1}{2}, j + \frac{1}{2})]}{((i + \frac{1}{2})\delta\theta)} - \frac{[H_{\phi r}^{n+1/2}(i + \frac{1}{2}, j - \frac{1}{2}) + H_{\phi\theta}^{n+1/2}(i + \frac{1}{2}, j - \frac{1}{2})]}{((i + \frac{1}{2})\delta\theta)} \right]$$

$$71) \quad E_{\theta}^{n+1}(i, j + \frac{1}{2}) = \left(1 - \frac{\sigma_r \cdot \delta t}{\epsilon}\right) E_{\theta}^n(i, j + \frac{1}{2}) + \frac{\delta t}{\delta r \cdot \epsilon} \cdot \left[\begin{aligned} & H_{\phi r}^{n+1/2}(i - \frac{1}{2}, j + \frac{1}{2}) + H_{\phi\theta}^{n+1/2}(i - \frac{1}{2}, j + \frac{1}{2}) \\ & - \left(\frac{i + \frac{1}{2}}{i - \frac{1}{2}}\right) \cdot [H_{\phi r}^{n+1/2}(i + \frac{1}{2}, j + \frac{1}{2}) + H_{\phi\theta}^{n+1/2}(i + \frac{1}{2}, j + \frac{1}{2})] \end{aligned} \right]$$

These four equations can be simplified using the following definitions where x is replaced by 'r' or 'θ', as appropriate:

$$72) \quad \hat{C}_{ax} = \frac{1 - \frac{\sigma_x \cdot \delta t}{2 \cdot \epsilon}}{1 + \frac{\sigma_x \cdot \delta t}{2 \cdot \epsilon}}$$

$$73) \quad \hat{C}_{bx} = \frac{\frac{\delta t}{\epsilon \cdot \delta r}}{1 + \frac{\sigma_x \cdot \delta t}{2 \cdot \epsilon}}$$

$$74) \quad \widehat{D}_{ax} = \frac{1 - \frac{\sigma_x^* \cdot \delta t}{2 \cdot \mu}}{1 + \frac{\sigma_x^* \cdot \delta t}{2 \cdot \mu}} \quad 75) \quad \widehat{D}_{bx} = \frac{\frac{\delta t}{\mu \cdot \delta r}}{1 + \frac{\sigma_x^* \cdot \delta t}{2 \cdot \mu}}$$

These are similar to (32)-(35), except that the conductivities have been split into their respective r and θ components. Using these simplifying coefficients, the PML iteration equations become:

$$76) \quad H_{\phi\theta}^{n+1/2}(i + 1/2, j + 1/2) = \widehat{D}_{a\theta} \cdot H_{\phi\theta}^{n-1/2}(i + 1/2, j + 1/2) + \widehat{D}_{b\theta} \cdot \left[\frac{\left(\frac{\sin((j+1) \cdot \delta\theta)}{\sin(j \cdot \delta\theta)} \right) \cdot E_r^n(i + 1/2, j+1) - E_r^n(i + 1/2, j)}{((i + 1/2)\delta\theta)} \right]$$

$$77) \quad H_{\phi r}^{n+1/2}(i + 1/2, j + 1/2) = \widehat{D}_{ar} \cdot H_{\phi r}^{n-1/2}(i + 1/2, j + 1/2) + \widehat{D}_{br} \cdot \left[- \left(\frac{i+1}{i} \right) \cdot E_\theta^n(i+1, j + 1/2) + E_\theta^n(i, j + 1/2) \right]$$

$$78) \quad E_r^{n+1}(i + 1/2, j) = \widehat{C}_{a\theta} \cdot E_r^n(i + 1/2, j) + \widehat{C}_{b\theta} \cdot \left[\frac{\left(\frac{\sin((j + 1/2) \cdot \delta\theta)}{\sin((j - 1/2) \cdot \delta\theta)} \right) \cdot [H_\phi^{n+1/2}(i + 1/2, j + 1/2)]}{((i + 1/2)\delta\theta)} - \frac{[H_\phi^{n+1/2}(i + 1/2, j - 1/2)]}{((i + 1/2)\delta\theta)} \right]$$

$$79) \quad E_\theta^{n+1}(i, j + 1/2) = \widehat{C}_{ar} E_\theta^n(i, j + 1/2) + \widehat{C}_{br} \cdot \left[H_\phi^{n+1/2}(i - 1/2, j + 1/2) - \left(\frac{i + 1/2}{i - 1/2} \right) \cdot [H_\phi^{n+1/2}(i + 1/2, j + 1/2)] \right]$$

...with $H_\phi = H_{\phi\theta} + H_{\phi r}$ in (78) and (79). Next, a step recommended in [13] is attempted. This involves converting to exponential time stepping in the PML region. This is done to accommodate the rapid decrease in wave amplitude in the PML region. Equations 71-74 still hold in the PML region, but new constants must be calculated that take the rapid time stepping into consideration. The new constants that arise from this approach are as follows:

$$80) \quad \hat{C}_{ar} = \exp\left(\frac{-\sigma_r \cdot \delta r}{\varepsilon}\right) \quad \hat{C}_{br} = \frac{\exp\left(\frac{-\sigma_r \cdot \delta r}{\varepsilon}\right) - 1}{\sigma_r}$$

$$81) \quad \hat{C}_{a\theta} = \exp\left(\frac{-\sigma_\theta \cdot \delta \theta}{\varepsilon}\right) \quad \hat{C}_{b\theta} = \frac{\exp\left(\frac{-\sigma_\theta \cdot \delta \theta}{\varepsilon}\right) - 1}{\sigma_\theta}$$

$$82) \quad \hat{D}_{ar} = \exp\left(\frac{-\sigma_r^* \cdot \delta r}{\mu}\right) \quad \hat{D}_{br} = \frac{\exp\left(\frac{-\sigma_r^* \cdot \delta r}{\mu}\right) - 1}{\sigma_r^*}$$

$$83) \quad \hat{D}_{a\theta} = \exp\left(\frac{-\sigma_\theta^* \cdot \delta \theta}{\mu}\right) \quad \hat{D}_{b\theta} = \frac{\exp\left(\frac{-\sigma_\theta^* \cdot \delta \theta}{\mu}\right) - 1}{\sigma_\theta^*}$$

In summary, equations (76)-(79) are used to compute the field components in the PML region with coefficients given by either equations (72)-(75) or (80)-(83). It was found that coefficients (72)-(75) gave good PML performance, whereas use of the exponential coefficients in (80)-(83) caused excessive reflections from the input edge of the PML region.

Perfectly Matched Layer (PML) Parameters:

In order to implement the PML iteration equations calculated in the previous section, it is necessary to calculate the conductivity profiles for each layer. In each layer, from free space to the last PML layer, the impedance matching condition given by Berenger must be kept [11]:

$$84) \quad \frac{\sigma}{\varepsilon} = \frac{\sigma^*}{\mu}$$

In this two dimensional problem, only σ_r and σ_r^* are really needed, as the outgoing spherical wave is expected to be normally incident on the outer layer. The outer layer forms a boundary in r with θ being represented from 0 to 90°. However, for completeness, the σ_θ and σ_θ^* terms can be kept, since they contribute little to additional computational overhead in the PML region.

The parameters that need to be calculated to compute the PML layer are the number of cells in the PML layer, N , the free space conductivities, σ_0 and σ_0^* , the final conductivities, σ_N and σ_N^* , and the profile of conductivities for each layer, σ_m and σ_m^* . The desired reflection for a normally incident wave is taken to be $R(0)$. Berenger gives the reflection factor for a wave at arbitrary angle of incidence, θ , as [12]:

$$85) \quad R(\theta) = R(0)^{\cos\theta}$$

The conductivity profile in the PML region can be linear, parabolic, or geometric. All three methods were experimented with in this work, although results from all the different cases can not be presented for lack of space. For a linear conductivity profile, $n = 1$, for a parabolic

conductivity profile, $n = 2$, and for a geometric profile, $g = 2-4$, are used. For a desired conductivity profile of thickness δ , the reflection factor $R(0)$ is given by:

$$86) \quad R(0) = e^{-\frac{1}{2\epsilon_0 c} \int_0^\delta \sigma(\rho) d\rho}$$

Adapting the results of the manipulation of equation (86) from [12] into spherical coordinates, so that δx is replaced by δr , the free space conductivity for the three different cases is given by

$$87) \quad \begin{array}{ll} \text{Linear or parabolic} & \text{geometric} \\ \sigma_0 = -\frac{\epsilon_0 \cdot c \cdot \ln(R(0))}{2^{n+2} \cdot \partial r \cdot N^{n+1}} & \sigma_0 = -\frac{\epsilon_0 \cdot c \cdot \ln(g) \cdot \ln(R(0))}{2 \cdot \partial r \cdot (g^N - 1)} \end{array}$$

Using equation (87) to calculate the electrical conductivity of free space, the magnetic conductivity of free space can be calculated from equation (84). The final layer conductivity is calculated from

$$88) \quad \begin{array}{ll} \text{Linear or parabolic} & \text{geometric} \\ \sigma_N = \sigma_0 \cdot (n+1) \cdot 2^{n+1} \cdot N^n & \sigma_N = \sigma_0 \cdot g^N \cdot \frac{g-1}{\sqrt{g} \cdot \ln(g)} \end{array}$$

The conductivity value for each layer, i , of the PML section of total thickness, δ , for i from 0 to N is then given by

$$89) \quad \begin{array}{ll} \text{Linear or parabolic} & \text{geometric} \\ \sigma_i = \sigma_N \cdot \left(\frac{i}{\delta}\right)^n & \sigma_i = \sigma_N \cdot g^{i-N} \end{array}$$

Using equations 87-89 and 84, one can calculate the conductivity profile given a desired reflection factor, $R(0)$, and number of layers, N . As an example, for a reflection factor of 0.01 with 20 layers in a parabolic conductivity profile, some of the conductivities are:

Conductivity type	layer 0	layer 1	layer 2	layer 10	layer 19	layer 20
electric	3.20E-05	0.0008	0.0069	0.0763	0.2756	0.3054
magnetic	4.52	109	434	10851	39171	43403

Table 1: Conductivities for a 20 layer parabolic PML

The correct profile is calculated automatically in the program for any desired reflection factor, number of layers, and for either a linear, parabolic, or geometric conductivity profile. Much lower reflection factors than 0.01 were typically used for parabolic or geometric profiles, as the lower reflection factor leads to a much higher final conductivity but only slightly higher free space conductivity. Thinner PML layers or linear layers with high reflection factors require higher free space conductivities to achieve the desired reflection factor. High free space conductivity leads to undesirable wave attenuation in the propagation region. To avoid this, PML regions with fairly large numbers of layers were used and with either parabolic or geometric conductivity profiles.

Some layer conductivities for a geometric conductivity profile with 20 layers and the same reflection factor are shown in table 2:

Conductivity type	layer 0	layer 1	layer 2	layer 10	layer 19	layer 20
electric	1.85E-12	1.10E-11	4.40E-11	2.90E-06	0.76	3.05
magnetic	2.63E-07	1.57E-06	6.30E-06	0.414	10851	434030

Table 2: Some conductivities for a 20 layer geometric conductivity profile PML

Note the much lower free space and higher final layer conductivities for the geometric profile. The results from the PML simulation testing were very good. Either a parabolic or a geometric profile gave adequate performance to enable antenna evaluation. The performance of the PML to a narrowband CW signal was found to be better than to the wideband UWB signal. It is believed that this is due to a discretization effect that arises from the stepping of the layer conductivities. The results from both CW and Gaussian pulse excitation are shown in a separate section of the appendix entitled, *FDTD Simulation Results: PML Effectiveness*.

Convergence Criteria

There are three general convergence criteria that the FDTD algorithm must satisfy, as described in [14]. The first is that the boundary of the computational space must be sufficiently far from the surface of a scatterer in accordance with the requirements of the particular radiation boundary condition used. In the case of the PML boundary, fairly close terminations are tolerable, so the $10\lambda_{\text{nom}}$ distance used for this problem is probably much more than is needed. The second criterion is that the cell sides must not be longer than $0.1\lambda_{\text{min}}$. As was described in the section on antenna geometry, the choice for δr satisfies this criterion. The third criterion is that the time step must not exceed a certain minimum determined by the cell dimensions. The coordinate invariant stability condition for the time step in two dimensions is given by [14]:

$$90) \quad c\delta t \leq \min \frac{\text{cell_area}}{\text{cell_diagonal_length}}$$

In the spherical coordinates used for this problem, the minimum cell dimensions occur at the spatial step closest to the origin. Since the coaxial line feeds a spherical wave at the $r = 2\delta r$ position (the origin is at the base of the antenna and is not part of the iteration), the minimum cell dimensions occur between $r = 2\delta r$ and $r = 3\delta r$. For these cells, the minimum time step criterion is the most critical. Using (90) for these minimum cells one obtains:

$$91) \quad \delta t \leq \min \frac{2 \cdot \delta r^2 \cdot \delta \theta}{c \cdot \sqrt{\delta r^2 + \left(\frac{5}{2} \cdot \delta r \cdot \delta \theta\right)^2}} \approx \frac{2 \cdot \delta r \cdot \delta \theta}{c}$$

Using the already chosen values for δr and $\delta \theta$, one obtains, $\delta t(\text{max}) = 0.35$ psec. As an additional check, the value for the minimum time step for the farthest cells in the simulation is given by:

$$92) \quad \delta t \leq \min \frac{r \cdot \delta r \cdot \delta \theta}{c \cdot \sqrt{\delta r^2 + (r \cdot \delta \theta)^2}}$$

For the farthest cells where $r = 150\delta r$, $\delta r(\max) = 9.3$ psec. The minimum time step for the smallest cell will be sufficient for all other cells, also.

Results

Closed Form Simulations

A closed form solution exists for an infinite biconical antenna and is given in [10]. This closed form solution is for steady state, rather than transient, voltage excitations, and it is for an infinite, rather than a finite, antenna. Nevertheless, the closed form solution can provide a benchmark against which to compare the FDTD results, for steady state excitation. The three closed form field components in free space are given as [10]:

$$93) \quad E_r = 0.$$

$$94) \quad E_\theta = \frac{\eta}{r \cdot \sin \theta} \cdot [A \cdot e^{j(\omega t - kr)} + B \cdot e^{j(\omega t + kr)}]$$

$$95) \quad H_\phi = \frac{1}{r \cdot \sin \theta} \cdot [A \cdot e^{j(\omega t - kr)} + B \cdot e^{j(\omega t + kr)}]$$

The equations for E_θ and H_ϕ appear with two different propagating wave terms. The first term represents a wave that is traveling radially outward with the velocity of light, and the second term represents a wave that is traveling radially inward with the same velocity. There is no field component in the radial direction, as this is the direction of propagation of the wave. The antenna input impedance for an infinite cone antenna over a ground plane with semi-angle θ_0 is given as [1] [10]

$$96) \quad Z_o = \frac{\eta \cdot \ln \left[\cot \left(\frac{\theta_0}{2} \right) \right]}{2 \cdot \pi}$$

A MATLAB simulation entitled, *IdealCone.m*, was run to display the real portion of the outward traveling wave component for both E_θ and H_ϕ . The code for the simulation is included in the software appendix. The parameters of the simulation such as δr and $\delta \theta$ are the same as are used in the FDTD simulations. The antenna cone angle was chosen as 30° . The simulation results were converted from spherical to Cartesian coordinates and are plotted in figures 4 and 5 from two different angles. The printed plots appear somewhat choppy due to discretization that is not present in the images when seen on a computer monitor. The image is a 3-D projection of a 2-D cross section of the E_θ field. Two different angular views are shown for clarity. The closed form H_ϕ field component differs from the E_θ field only in magnitude and is not shown.

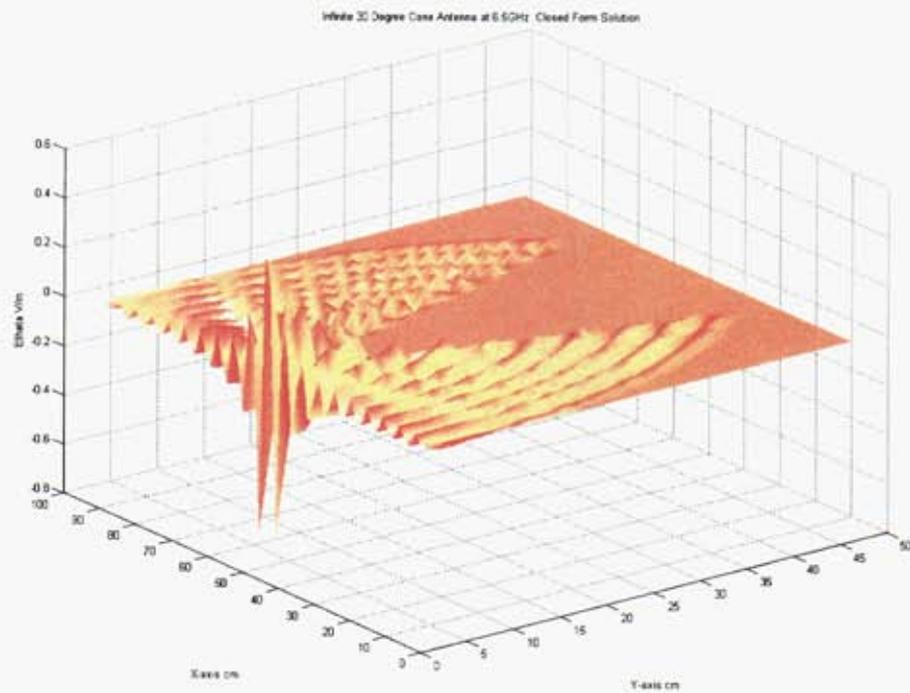


Figure 5: E-field for infinite cone antenna, closed form solution.

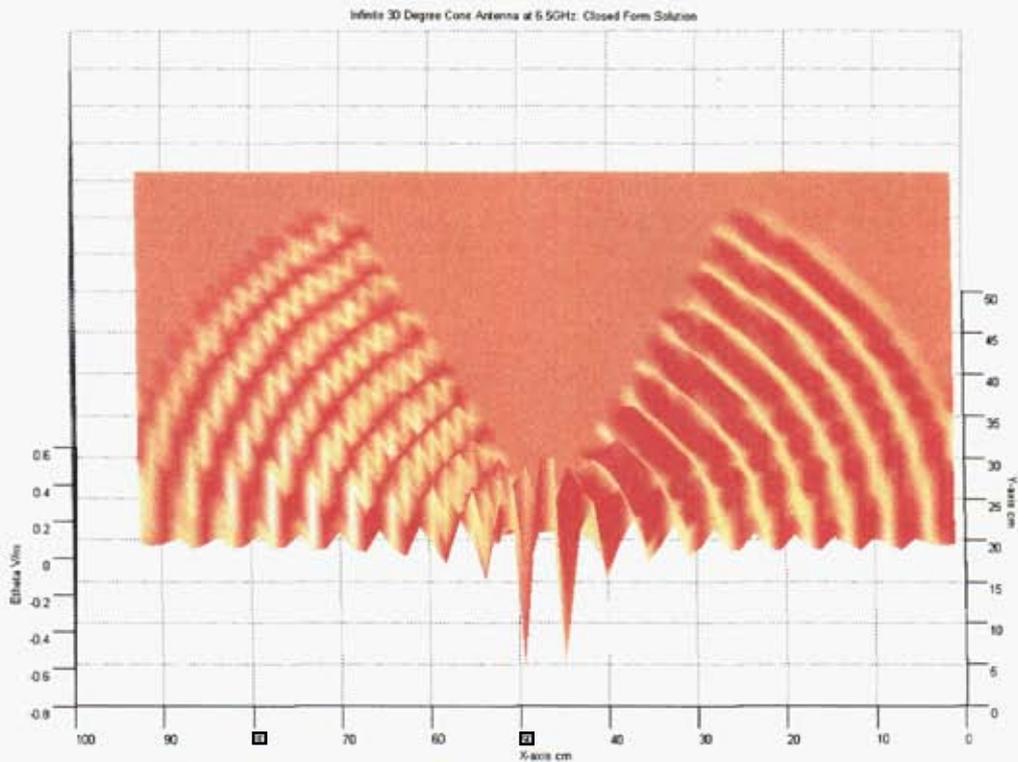


Figure 6: 2-D view of E-field for infinite cone antenna, closed form solution

FDTD Simulation Results, CW Excitation

The FDTD simulation was written in MATLAB and is called *FDTD1.m*. This simulation uses the equations and parameters generated in the previous sections. It is written as a MATLAB function that is callable from the command line with a number of time steps as an input parameter. To compare results with the closed form simulations, an FDTD simulation with a steady state voltage source was used for the results shown in figures 7 and 8.

The closed form results in figures 5 and 6 compare well both qualitatively and quantitatively to the FDTD results in figures 7 and 8. The FDTD results terminate about $1\lambda_{\text{nom}}$ before the outer edge of the simulation space due to the rapid attenuation of the PML layer used here. The scales in the two sets of figures are slightly different, but it can be seen that in both sets of figures the amplitude changes from about 0.2v/m about $1\lambda_{\text{nom}}$ from the base of the antenna to about 0.05v/m about $8\lambda_{\text{nom}}$ from the base. Both sets of images exhibit some apparent variation in θ that is a result of the conversion to Cartesian coordinates from spherical coordinates for plotting. These variations exist as visual artifacts in the images, but are not present in the spherical data.

The FDTD simulation was performed with an $8.2\lambda_{\text{nom}}$ tall antenna with a 30° degree semi-angle. Overall, the CW-FDTD simulation results compare very well to the theoretical results for an infinite antenna.

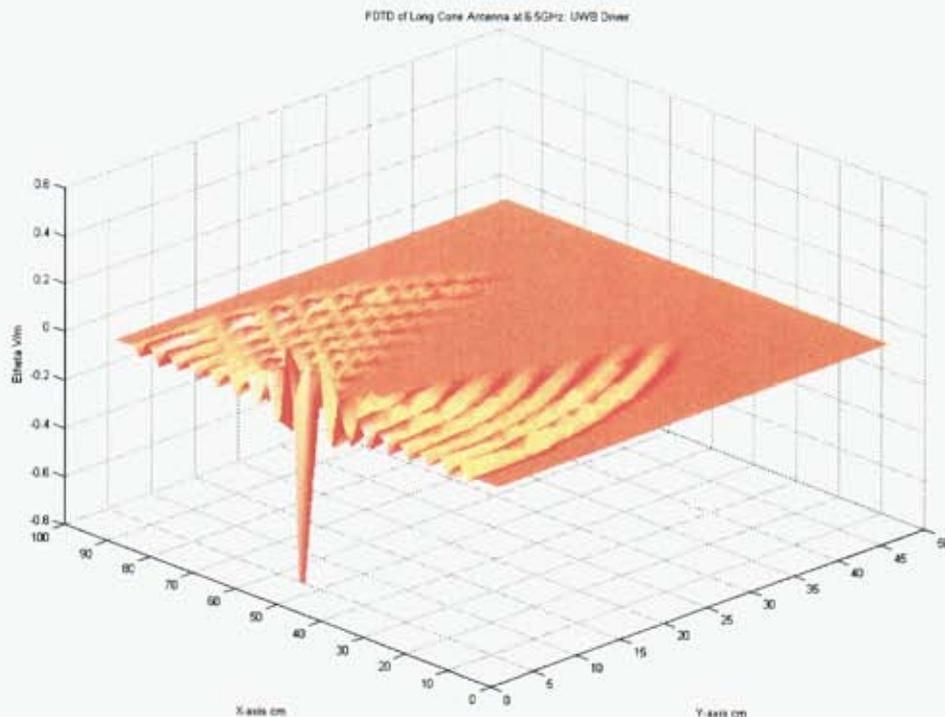


Figure 7: E_θ -field for long cone antenna, FDTD solution.

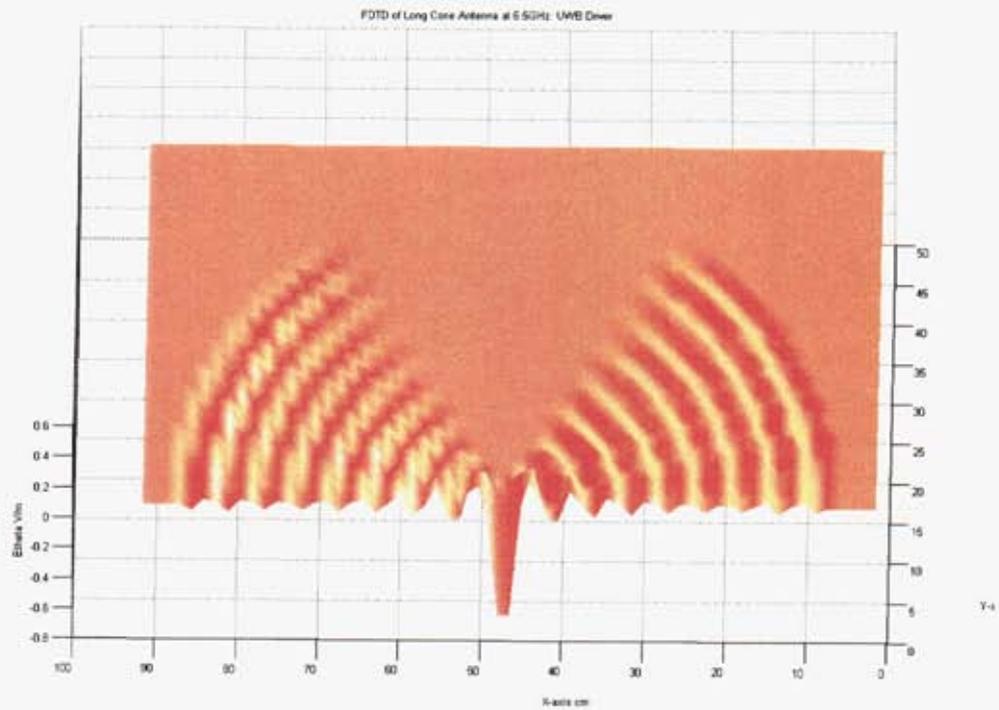


Figure 8: E_0 -field for long cone antenna, side view, FDTD solution.

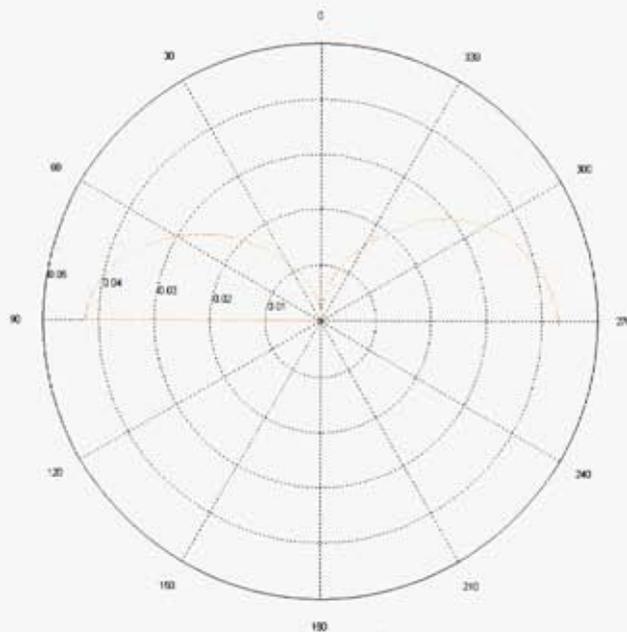


Figure 9: Comparison of FDTD far field pattern (left) with published data (right)

Figure 9 shows a comparison of the far field radiation pattern from the FDTD simulation with CW excitation (left side) against published far field data for a comparable antenna (right side)

[3]. The antenna was 1λ tall with a 30° semi-angle over a ground plane that covered the entire simulation space. The simulation was run for 8000 steps with 0.2 psec/step. The comparison was generated from the E_θ field at a distance of 37 cm (8.2λ) from the origin. This distance is shortly before the PML region begins to attenuate the outward traveling wave. The E_θ field simulation data was collected as output from the *FDTD1.m* MATLAB routine. The polar plot comparing the published data to the simulated data was generated using the MATLAB routine *Pplot.m*. The simulated and measured data compare fairly well with the simulation, faithfully reproducing the field everywhere with only slightly lower levels in the 30 - 60° region.

Simulation Results: UWB Pulse Excitation

The MATLAB simulation *FDTD1.m* can be run with a UWB pulse selected as the voltage drive signal $V_s(t)$ in equations (60) and (61). The UWB pulse is modeled as a Gaussian-shaped sinusoid of peak amplitude $0.7V_o$. It is described by

$$97) \quad V_s(t) = -V_o \cdot \sin[2 \cdot \pi \cdot (n - n_{start}) \cdot t] \cdot \exp\left[-\left(\frac{n - n_{start}}{n_{decay}}\right)^2\right]$$

where n_{start} determines the zero-phase point of the bipolar pulse, and n_{decay} determines the $1/e$ time constant of the pulse. The simulation results of a short MATLAB program *Vsrc.m* to simulate this pulse and a cosine variant of it are shown in figure 10. The cosine version is described by

$$98) \quad V_s(t) = V_o \cdot \cos[2 \cdot \pi \cdot (n - n_{start}) \cdot t] \cdot \exp\left[-\left(\frac{n - n_{start}}{n_{decay}}\right)^2\right]$$

Equation (97) describes a unipolar Gaussian-shaped UWB pulse and (98) describes a bipolar UWB pulse. Both the sine and cosine versions of the UWB pulses described by (97) and (98) were used as excitation sources for the antenna in figure 1.

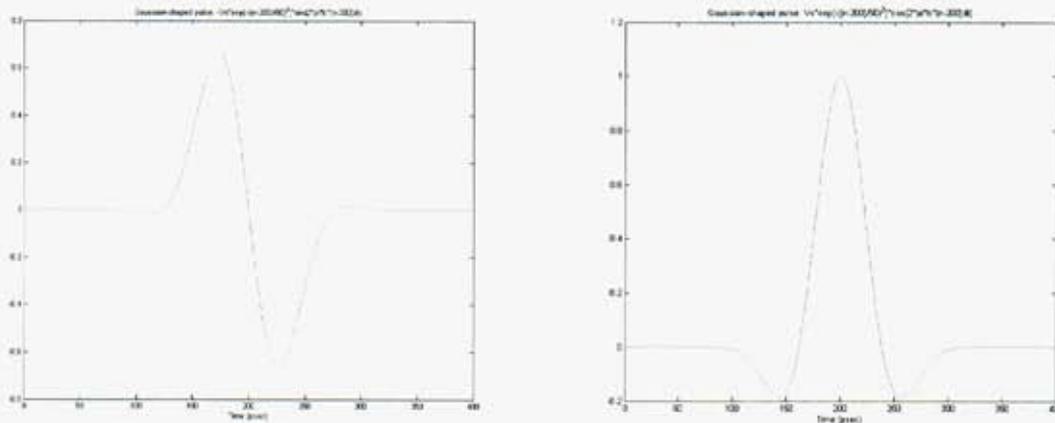


Figure 10: Sine and cosine driven UWB pulses with $f_c=6.5\text{GHz}$.

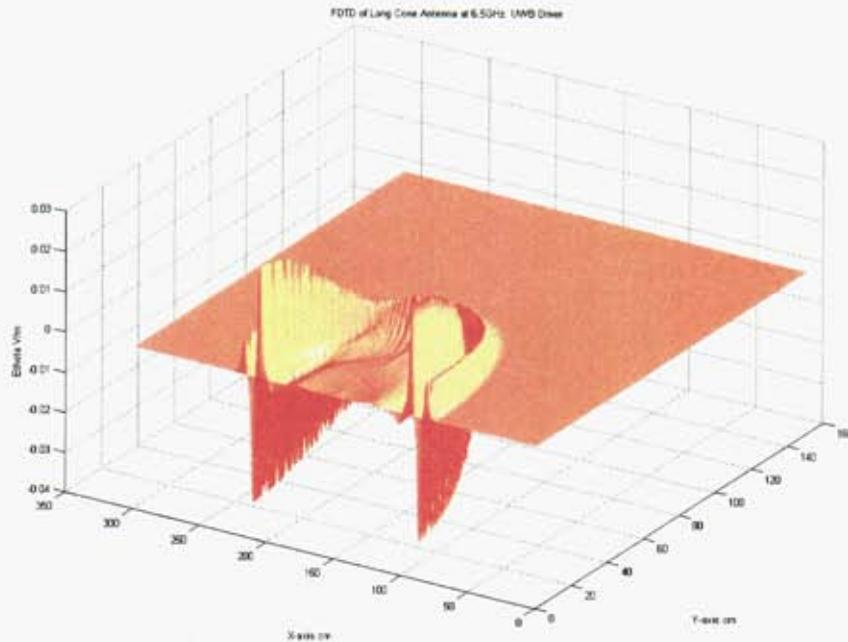


Figure 11: 3-D view of 2-D E_{θ} Field from UWB pulse propagating out from the conical antenna

The sine version of the UWB pulse was used to drive the antenna in figure 11. The E_{θ} field component after 2.0 nsec (4000 time steps) is shown. The FDTD simulation shows detailed structure of the wavefront as it advances radially out from the antenna, although some of the finest variations in this image are artifacts arising from the conversion from spherical to Cartesian coordinates for plotting. The spiky appearance of the wave along the tops of the crests is due to the conversion artifacts. The distortion of the unipolar UWB input voltage pulse into a bipolar E_{θ} field pulse compares well with results from [17]. The ability to reproduce this kind of wave distortion using FDTD analysis is a valuable benefit of this analysis for difficult UWB design problems. In addition, some ringing from the impedance mismatch between the 74Ω antenna and the 50Ω coaxial input line is visible in this image and in the images in the appendix.

A separate section of time-lapsed images is included in the appendix providing detailed images of the propagation of a Gaussian-shaped pulse from the conical antenna. These plots give an effective visual review of the propagation of the wave, its far-field characteristics, and its final absorption in the PML region. The MATLAB function *Film.m* was created to generate a time-stepped movie of the UWB pulse as it propagates from the base of the antenna until it is absorbed in the PML layer.

Impedance Simulation Results

The FDTD simulation was also used to compute antenna input impedance data for variations in antenna semi-angle, θ_0 , and height, a . The impedance was computed during the CW-excitation FDTD simulation using equations (59) and (62)-(63). The simulation-generated results are shown in table 3 in the fourth column, as compared with measured results in the fifth column, theoretical results for an infinite antenna in the sixth column, and calculated results using varying antenna parameters in the seventh column.

Antenna angle (degrees)	Antenna height (a) (mm) (dr = 3mm)	ka (k = 0.14/mm)	Zsimulated (ohms)	Zreported[3] (ohms)	Zinfinite[10] (ohms)	Zcalc[16] (ohms)
30	372	52	73	-	79	80
30	45	6.3	73	74	-	85
30	33	4.6	73	72	-	70
30	24	3.25	72	70	-	79
30	15	2.1	77	91	-	100
30	9	1.3	58	50	-	49
30	6	0.85	32	22	-	29
4	372	52	225	-	201	209
10	372	52	146	-	146	150
20	372	52	93	-	104	106
45	372	52	53	-	53	61
60	372	52	27	-	33	34

Table 3: Antenna Impedance Simulation Results

Measured impedance data is available from [3] which refers to [15] as the original source of the measurements. The measured data all exists for an antenna semi-angle of 30° and for varying antenna heights. The tallest antenna for which measurement data is available is only about $1\lambda_{nom}$. The simulated impedances correlate with the measured results very well, and only diverge significantly for very short antennas. This is probably due to discretization-induced errors, as the antenna dimensions approach the dimensions of the unit cell.

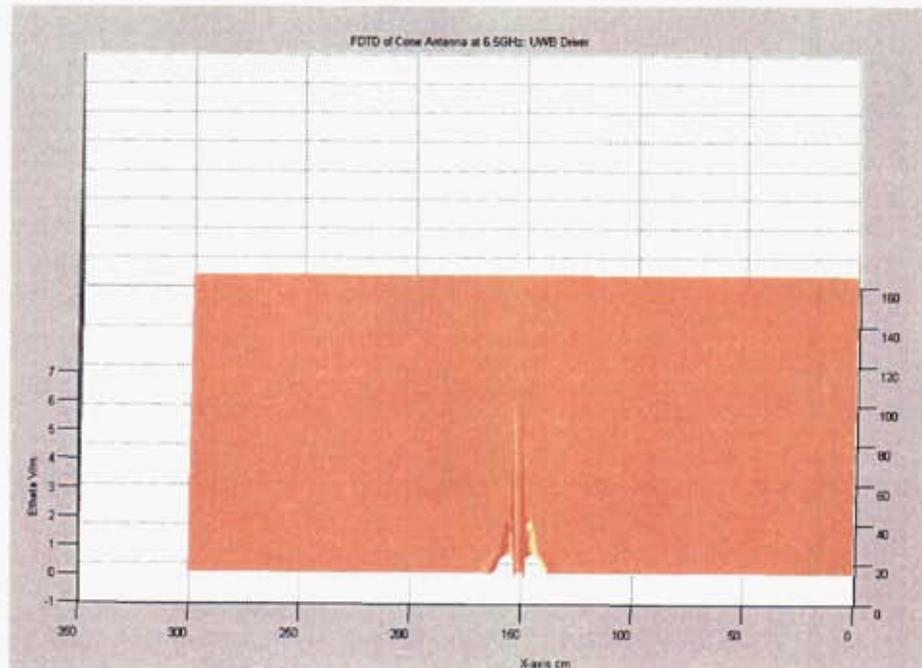
The theoretical impedance values included in the table are for an infinite, conical antenna [10] and [1]. The theoretical model is based on an infinite cone antenna over an infinite ground plane, and the impedance is calculated using equation (96). The comparable simulation data was obtained using an $8.2\lambda_{nom}$ tall antenna with varying antenna angles. The antenna height of $8.2\lambda_{nom}$ is long relative to a wavelength but is not infinite. The simulated impedances do correlate very well with the corresponding theoretical values, albeit for an infinite antenna.

The calculated impedance values were obtained using a MathCAD worksheet, *ConeAntZ.mcd*, included in the appendix. The methodology and equations used in this worksheet are described in [16]. The formulae used to calculate the impedances give results that agree fairly well with both the measured and the FDTD simulated values. The greatest discrepancies between the simulated and the calculated values occur for very short antennas. Again, it may be possible to obtain better simulation accuracy using a smaller cell size for the smaller antennas. For angles of 60° or more, the simulated impedances were found to be unpredictable and increasingly less reliable. This is probably due to phase differences between the simulated currents and voltages. Large variations in impedance for small changes in sample time were found at these large angles. Simulation results were most reliable for antenna heights of $\frac{1}{2}\lambda_{nom}$ or more and for antenna angles of 45° or less.

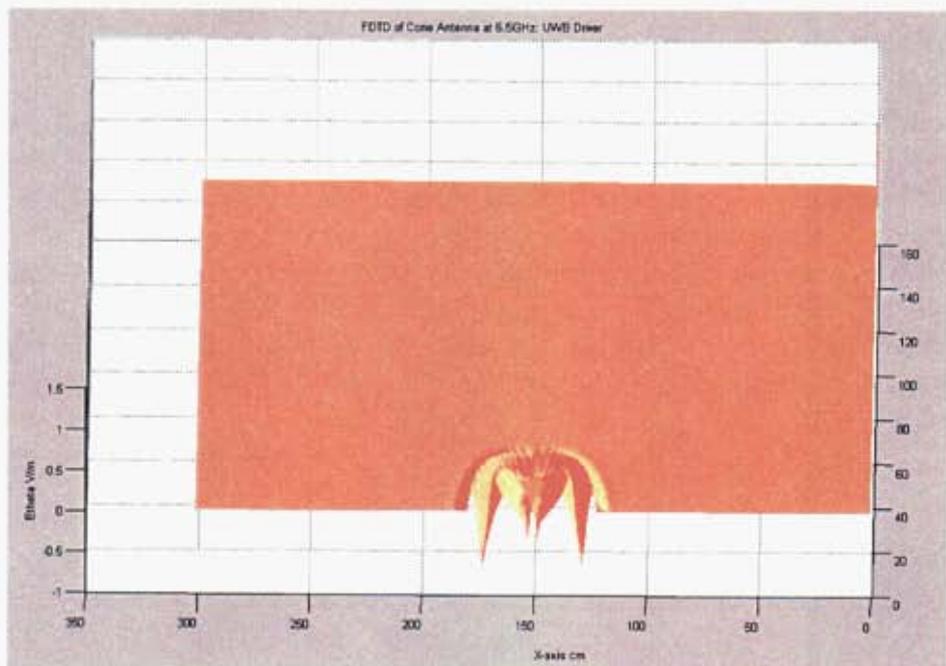
Three-dimensional plots of two-dimensional FDTD simulation results

These plots are of FDTD simulation results for a 15mm tall conical antenna with a semi-angle of 30° in a spherical field with an outer radius of 150mm. The excitation waveform is a Gaussian-shaped sinusoid centered around 6.5GHz. These plots are of the E_θ field, converted back from the spherical simulation space into Cartesian coordinates for plotting. The coordinate conversion process and plotting process can introduce some small-scale visual artifacts; otherwise, the data is faithfully reproduced.

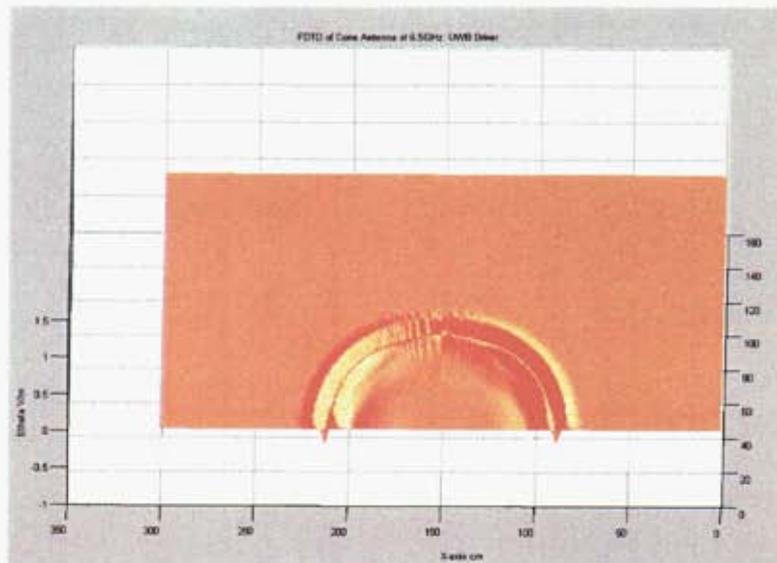
1) Plot of simulation results after 1000 time-steps (200 psec). The sine version of the Gaussian pulse used for excitation is starting to travel out from the base of the antenna.



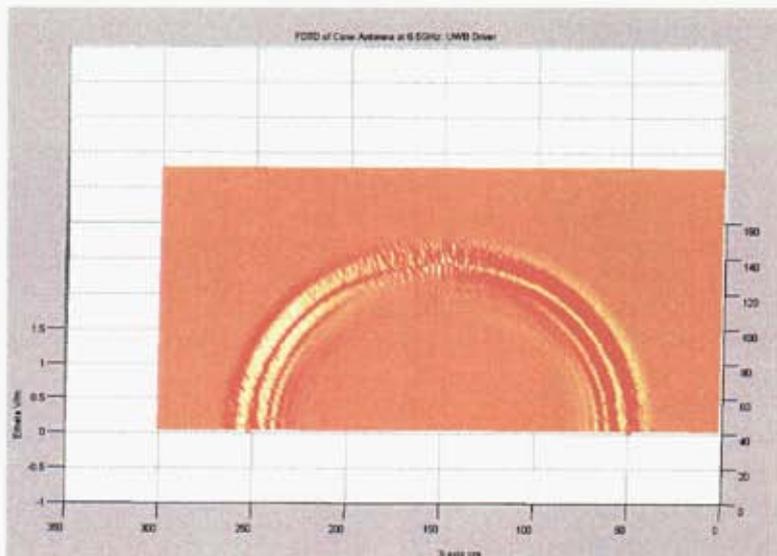
2) Plot of simulation results after 2000 time-steps (400 psec). The conical cross-section of the antenna is clearly visible, surrounded by the wave excitation.



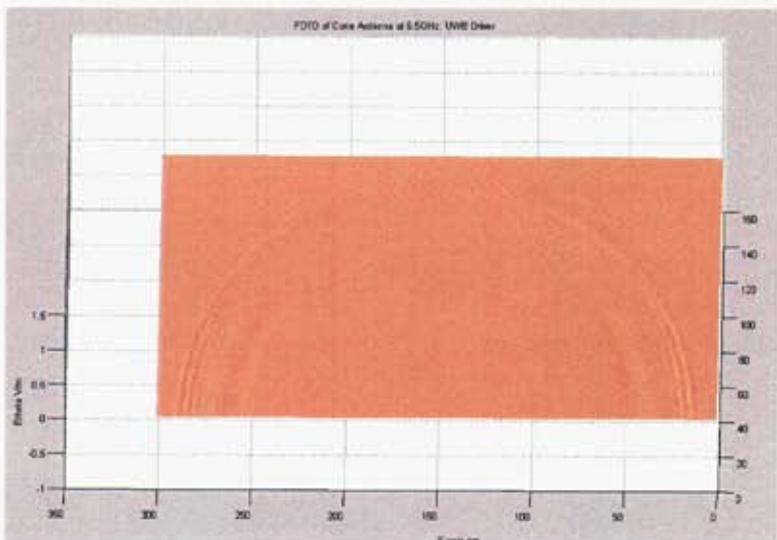
3) Plot of simulation results after 4000 time-steps (800 psec). The wave is traveling away from the antenna and starting to assume the far-field pattern shape.



4) Simulation results after 6000 time-steps (1.2 nsec). The wave is still traveling spherically outward, as expected, with an amplitude distribution that has assumed the far-field pattern. The main Gaussian pulse is followed by some small ringing arising from the coaxial input line-antenna impedance mismatch.



5) Simulation results after 9000 time-steps (1.8 nsec). The Gaussian pulse has been entirely absorbed by the PML region. The small remaining waves are the residual ringing and are not reflected from the PML. These waves are also absorbed by the PML region within another 0.2nsec, and the simulation space is left with no visible waves.

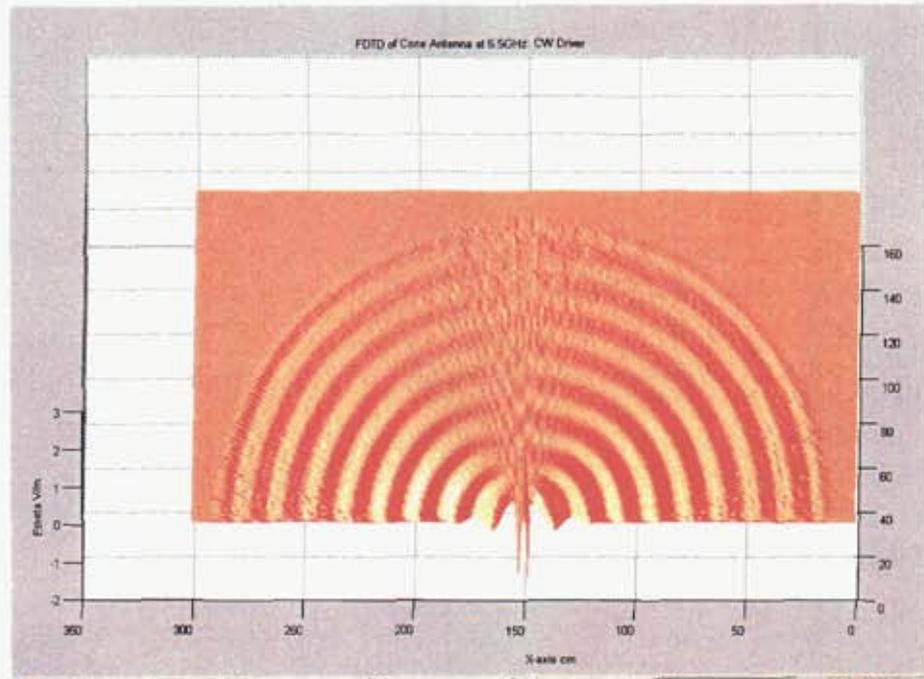


FDTD Simulation Results: PML Effectiveness

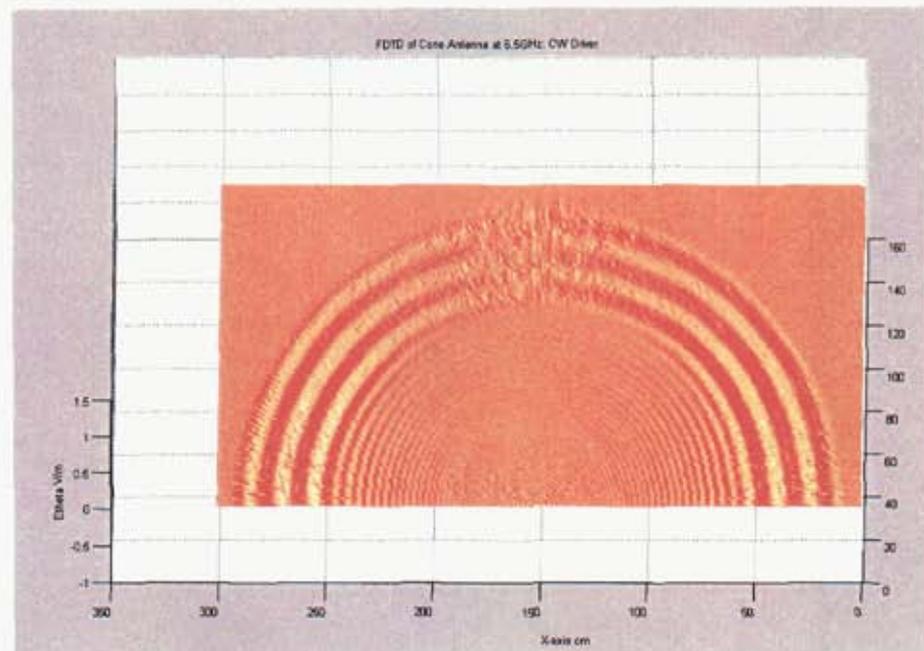
To assess the effectiveness of the PML region, a series of snapshots of the simulation space for both CW and Gaussian pulse excitation are presented. These images demonstrate that the PML technique, as implemented, is adequately effective for absorbing both the narrowband CW signal and the UWB signal. The PML used here is a 20-layer thick region with parabolic conductivity profile; it is less effective against the wideband signal than the CW signal.

CW Excitation

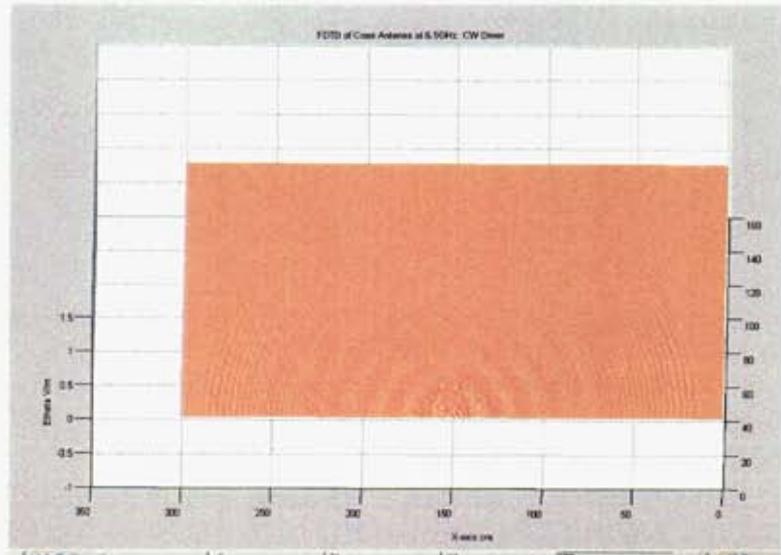
1) FDTD simulation with CW excitation for 7000 time-steps (1.4nsec). The wave has reached the PML and is about to be absorbed. The PML used here was a 20-layer parabolic stepped conductivity region.



2) FDTD simulation with CW excitation after the excitation source has been turned off, 12000 time-steps (2.4nsec). The tail end of the wave is about to be absorbed in the PML. The residual, small waves trailing the main waves are due to antenna/ input coaxial line mismatch induced ringing.

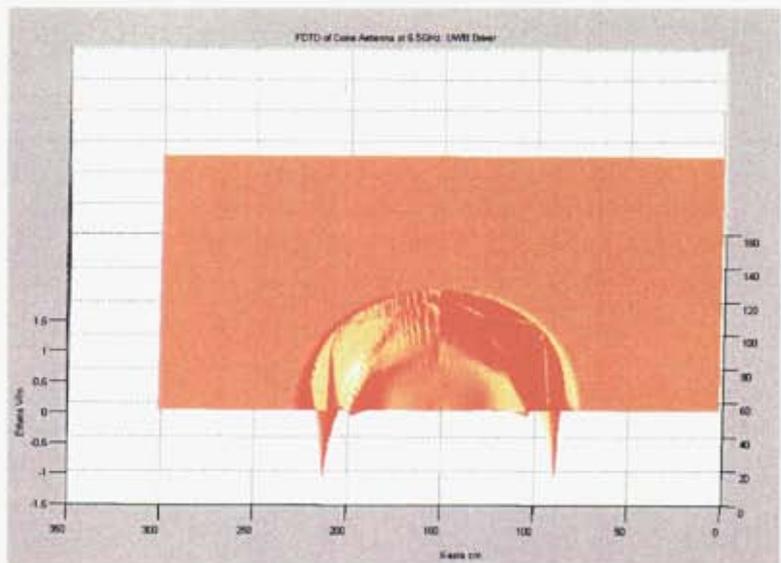


3) FDTD simulation with CW excitation after the excitation source has been turned off, and the wave has been absorbed by the PML region (16000 time-steps, 3.2nsec). A faint wave at the 6.5GHz frequency is still visible and is likely due to a small reflection from the PML region.

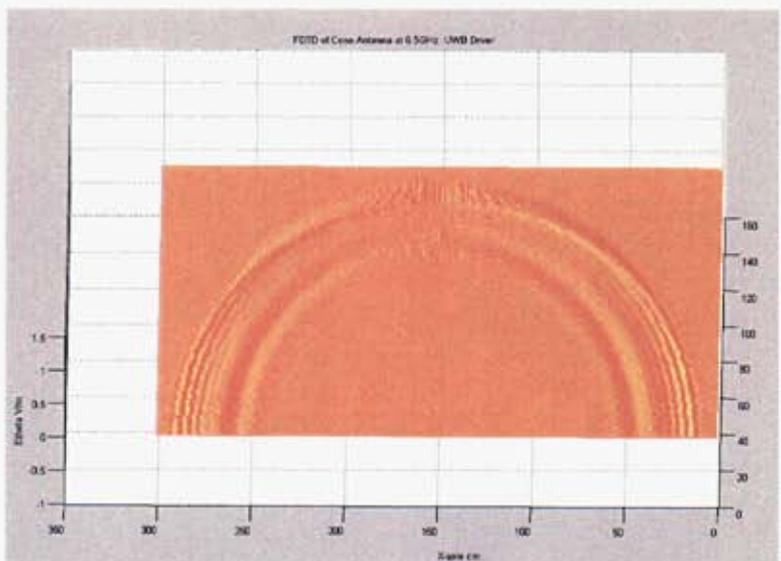


UWB Excitation

1) FDTD simulation with UWB Gaussian shaped sinusoidal pulse excitation. This image is after 4000 time-steps, as the wave is traveling outward and well before it has hit the PML region. Note the faint impression of the antenna in the center of the wave.



2) FDTD simulation with UWB excitation after 9000 time-steps (1.8nsec). The wave has struck and been mostly absorbed by the 20 layer thick PML region. The high frequency ringing is due to the coaxial input line mismatch, and the lower frequency signal is the UWB signal that has been reflected from the PML.



Conclusion

Complete sets of spherical FDTD simulation equations were developed from first principles to govern electromagnetic wave generation, propagation, and absorption. These equations were used to simulate a conical antenna for CW excitation, Gaussian pulse excitation, and input impedance. The simulation results were found to compare well to other published data, both measured and simulated.

References

- 1) J.R. Andrews, "UWB Signal Sources, Antennas, and Propagation", *Picosecond Pulse Labs Appl. Note*, AN-14a, Aug. 2003, pp. 3.
- 2) R. Lawton and A. Ondrejka, "Antennas and the Associated Time-Domain Range for the Measurement of Impulsive Fields", *Nat. Bur. of Stnds. Tech. Note 1008*, Boulder CO, Nov. 1978.
- 3) G. Liu and C.A. Grimes, "Spherical-Coordinate FDTD Analysis of Conical Antennas Mounted Above Finite Ground Planes", *Microwave and Opt. Tech. Let.*, vol. 23, no. 2, Oct. 1999, pp. 78-82.
- 4) M.A. Fusco, et al, "A Three-Dimensional FDTD Algorithm in Curvilinear Coordinates", *IEEE Trans. on Ant. and Prop.*, vol. 39, no. 10, Oct, 1991, pp. 1463-1471.
- 5) F.L. Teixeira and W.C. Chew, "PML-FDTD in Cylindrical and Spherical Coordinates", *IEEE Microwave and Guided Wave Let.*, vol. 7, no. 9, Sep. 1997, pp 285-287.
- 6) R. Holland, "THREDS: A Finite Difference Time-Domain EMP Code in 3D Spherical Coordinates", *IEEE Trans. on Nucl. Sci.*, vol. NS-30, no. 6, Dec. 1983, pp. 4592-4595.
- 7) T.W. Hertel and G.S. Smith, "Analysis and Design of Two-Arm Conical Spiral Antennas", *IEEE Trans. on Electromag. Compat.*, vol. 44, no. 1, Feb. 2002, pp. 25-37.
- 8) FCC 02-48, "Revision of Part 15 of the Commission's Rules Regarding Ultra-Wideband Transmission Systems", First Report and Order, Wash. D.C., adopted Feb. 14, 2002, released April 22, 2002.
- 9) A.Z. Elsherbeni, C.G. Christodoulou, and J. Gomez-Tagle, "The Finite Difference Time Domain Technique for Microstrip Antenna Applications," *unpublished manuscript*, pp. 1-59.
- 10) Ramo, Whinnery, and Van Duzer, *Fields and Waves in Communication Electronics*, John Wiley and Sons, 1965, p. 464.
- 11) J.P. Berenger, "A Perfectly Matched Layer for the Absorption of Electromagnetic Waves," *Journal of Computational Physics*, no. 114, 1994, pp. 185-200.
- 12) J.P. Berenger, "Perfectly Matched Layer for the FDTD Solution of Wave-Structure Interaction Problems," *IEEE Transactions on Antennas and Propagation*, vol 44, no. 1, Jan. 1996, pp. 110-117.
- 13) D.S. Katz, et al, "Validation and Extension to Three Dimensions of the Berenger PML Absorbing Boundary Condition for FD-TD Meshes," *IEEE Microwave and Guided Wave Letters*, vol. 4, no. 8, Aug. 1994, pp. 268-270.
- 14) M. Fusco, "FDTD Algorithm in Curvilinear Coordinates", *IEEE Trans. On Ant. and Prop.*, vol. 38, no. 1, Jan. 1990, pp. 76-89.
- 15) Harvard Radio Research Laboratory Staff, "Very High Frequency Techniques," McGraw-Hill, New York and London, 1947, vol. 1, pp. 102-110.
- 16) S.S. Sandler and R.W.P. King, "Compact Conical Antennas for Wide-Band Coverage", *IEEE Trans. on Ant. and Prop.*, vol. 47, no. 3, Mar. 1994, pp. 436-439.

17) J.G. Maloney, G.S. Smith, and W.R. Scott, "Accurate Computation of the Radiation from Simple Antennas using the Finite-Difference Time-Domain Method", *IEEE Trans. On Ant. and Prop.*, vol. 38, no. 7, Jul. 1990, pp. 1059-1068.

Appendix

Matlab Function: *FDTD1.m* (11/23/04)

```
% Two dimensional, spherical FDTD Simulation of a conical UWB antenna.
% This simulation uses 2-D versions of the 3-D spherical Maxwell's
% equations. The antenna radiation pattern is stepped forward in time,
% and the electric and magnetic field components are calculated for the
% propagation of the wave into space. Excitation waves can either be a CW
% sine wave or a Gaussian-shaped pulse.
%
% R.W. Brocato
%
function [Et] = FDTD1(tmax)
% Initialize standard free space constants
uo = 4*pi*1e-7;
eo = 1e-9/(36*pi);
Z = 120*pi;
c = 3e+8;
% Define the antenna dimensions. It is 15dr (1 lambda_nom) long with an
% apex angle of 30 degrees. Add one for Matlab matrix referencing.
ant_length = 15;
ant_angle = 31;
% Define boundary edges. Rmax occurs at 10(lambda_nominal) where lambda
% nominal is the wavelength for a 6.5GHz wave. The expected bandwidth is
% 3.1-10.6 GHz, with 3.1 GHz being lambda_max and 10.6 GHz being lambda_min.
% This range is defined by the FCC mask for UWB transmission. Theta
% extends from 0 to 90 degrees. Here, dr = 0.003 m (lambda_min/10), and
% dtheta = 1 degree. Using dt = 0.2 psec will satisfy Courant's limit for
% all frequencies.
dr = 0.003; % cell radius
dth = 1.0*pi/180; % cell angle in radians
dt = 0.2e-12; % time step
Rmax = 151; % solution space radius
THmax = 91; % solution space angle
b = 0.006; % coaxial input line outer radius
a = 0.003; % coaxial input line inner radius
% Set up constants for the PML region: Npml = # of layers, R0 = desired
% reflection coefficient at zero angle, sigma_space = conductance of free
```

```

% space (0 layer), sigmaM_space = magnetic conductance of free space, L=1
% for linear conductivity profile, L=2 for parabolic conductivity profile,
% for a geometric conductivity, 3-4 lines must be commented and
% uncommented. All equations used are from Berenger's IEEE Trans. on Ant.
% and Prop., Jan. 1996 paper.
R0 = 1e-14;           % select desired reflection
Npml = 20;           % select number of layers to use
L = 2;               % select conductivity profile
g = 2;               % select geometric factor
% calculate free space conductivity for a linear or parabolic profile
sigma_space = - eo*c*log(R0)/(2^(L+2)*dr*Npml^(L+1));
% calculate free space conductivity for a geometric profile
% sigma_zero = - eo*c*log(g)*log(R0)/(2*dr*(g^(Npml)-1));
% sigma_space = sigma_zero*(sqrt(g) - 1)/log(g);
sigmaM_space = uo*sigma_space/eo; % impedance matching condition
% calculate final conductivity for a linear or parabolic profile
sigmaPML(Npml) = sigma_space*(L+1)*2^(L+1)*Npml^L;
% calculate final conductivity for a geometric profile
% sigmaPML(Npml) = sigma_zero*(g-1)*(g^Npml)/(sqrt(g)*log(g));
sigmaMPML(Npml) = uo*sigmaPML(Npml)/eo;
for I = 1:(Npml-1)
    sigmaPML(I) = sigmaPML(Npml)*(I/Npml)^L; % for linear profile
% sigmaPML(I) = sigmaPML(Npml)*g^(I-Npml); % for geometric profile
    sigmaMPML(I) = uo*sigmaPML(I)/eo;
end
% Initialize the PML region H sub-components
for I = 1:Npml
    for J = 1:THmax
        Hpr(I,J) = 0;
        Hpt(I,J) = 0;
    end
end
% Set up the conducting antenna surface. The antenna is made of copper
% with conductivity of 5.8e+7 mhos/m. Free space is given a small
% electrical conductivity and a matching magnetic conductivity according
% to the PML calculations done above.
sigma_cu = 5.8e+7;
sigmaM_cu = uo*sigma_cu/eo;
for I = 1:Rmax
    for J = 1:THmax
        if (J == ant_angle)
            if (I <= ant_length)
                sigma(I,J) = sigma_cu;
                sigmaM(I,J) = sigmaM_cu;
            elseif (I < Rmax-Npml+1)
                sigma(I,J) = sigma_space;
            end
        end
    end
end

```

```

        sigmaM(I,J) = sigmaM_space;
    else
        sigma(I,J) = sigmaPML(I + Npml - Rmax);
        sigmaM(I,J) = sigmaMPML(I + Npml - Rmax);
    end
elseif (I == ant_length)
    if (J <= ant_angle)
        sigma(I,J) = sigma_cu;
        sigmaM(I,J) = sigmaM_cu;
    else
        sigma(I,J) = sigma_space;
        sigmaM(I,J) = sigmaM_space;
    end
else
    if (I < Rmax-Npml+1)
        sigma(I,J) = sigma_space;
        sigmaM(I,J) = sigmaM_space;
    else
        sigma(I,J) = sigmaPML(I + Npml - Rmax);
        sigmaM(I,J) = sigmaMPML(I + Npml - Rmax);
    end
end
end
end
end
% To initialize, zero the fields for all of the free nodes in the problem.
for I = 1:Rmax
    for J = 1:THmax
        if ((I >= ant_length) | (J >= ant_angle))
            Er(I,J) = 0;
            Et(I,J) = 0;
            Hp(I,J) = 0;
        end
    end
end
end
% Establish the Gaussian driving pulse parameters. Its center is
% 200ps out, it has a decay time of 50ps and a center
% frequency of 6.5 GHz.
tstart = 200e-12;
tdecay = 50e-12;
tsample = 183e-12;
once = 0;
fc = 6.5e+9;
Vocw = 0.01;
Vouwb = 0.06;
Rs = 50;
Zin = 0;

```

```

% Set up some constants for the time iterations
t = 0;
g1 = dt/(2*uo);      % g1 = 7.6e-8
g2 = dt/(2*eo);      % g2 = 0.0113
g3 = dt/(dr*eo);     % g3 = 7.53
g4 = dt/(dr*uo);     % g4 = 5.3e-5
sphere_factor = 1/(b*log(2));
I_factor = 2*pi*b;
volt_factor = (log(sin(ant_angle*pi/180))-log(1-cos(ant_angle*pi/180)))/log(2);
% Begin the time iterations
while (t < (tmax*dt))
    % For the first half timestep, update the Hphi field everywhere.
    % Don't step the singularity at the origin, the ground plane points,
    % the line of symmetry, or the PML region.
    t = t + 0.5*dt;
    for I = 3:(Rmax-Npml)
        for J = 2:(THmax-1)
            if ((I >= ant_length) | (J >= ant_angle))
                g5 = sin(J*dth)/sin((J-1)*dth);
                Da = (1-sigmaM(I,J)*g1)/(1+sigmaM(I,J)*g1);
                Db = g4/(1+sigmaM(I,J)*g1);
                ER1 = (g5*Er(I,J+1)-Er(I,J))/((I-1/2)*dth);
                ET1 = (I/(I-1))*Et(I+1,J) - Et(I,J);
                Hp(I,J) = Da*Hp(I,J) + Db*(ER1 - ET1);
            end
        end
    end
    % Compute the H field at the ground plane, outside the PML region
    J = THmax;
    for I = 3:(Rmax-Npml)
        Da = (1-sigmaM(I,J)*g1)/(1+sigmaM(I,J)*g1);
        Db = g4/(1+sigmaM(I,J)*g1);
        ET1 = (I/(I-1))*Et(I+1,J) - Et(I,J);
        Hp(I,J) = Da*Hp(I,J) - Db*ET1;
    end
    % Compute the H field at the along the line of symmetry.
    for I = ant_length:(Rmax-Npml)
        Hp(I,1) = Hp(I,2);
    end
    % Update the H field in the PML region, excluding the ground plane
    layer = 1;
    for I = (Rmax-Npml+1):Rmax
        Dar = (1-sigmaMPML(layer)*g1)/(1+sigmaMPML(layer)*g1);
        Dbr = g4/(1+sigmaMPML(layer)*g1);
        Dat = Dar;
        Dbt = Dbr;
    end
end

```

```

for J = 2:(THmax-1)
    g5 = sin(J*dth)/sin((J-1)*dth);
    ER1 = (g5*Er(I,J+1)-Er(I,J))/((I-1/2)*dth);
    if (I == Rmax)
        ET1 = 0;
    else
        ET1 = (I/(I-1))*Et(I+1,J) - Et(I,J);
    end
    Hpr(layer,J) = Dar*Hpr(layer,J) - Dbr*ET1;
    Hpt(layer,J) = Dat*Hpt(layer,J) + Dbt*ER1;
    Hp(I,J) = Hpr(layer,J) + Hpt(layer,J);
end
layer = layer + 1;
end
% Update the H field in the PML region along the ground plane
J = THmax;
layer = 1;
for I = (Rmax-Npml+1):Rmax
    Dar = (1-sigmaMPML(layer)*g1)/(1+sigmaMPML(layer)*g1);
    Dbr = g4/(1+sigmaMPML(layer)*g1);
    Dat = Dar;
    if (I == Rmax)
        ET1 = 0;
    else
        ET1 = (I/(I-1))*Et(I+1,J) - Et(I,J);
    end
    Hpr(layer,J) = Dar*Hpr(layer,J) - Dbr*ET1;
    Hpt(layer,J) = Dat*Hpt(layer,J);
    Hp(I,J) = Hpr(layer,J) + Hpt(layer,J);
    layer = layer + 1;
end
end
% For the second half timestep, update the E fields everywhere. Again,
% don't step the driver at the origin or the ground plane points.
t = t + 0.5*dt;
% Step the excitation source, either the Gaussian pulse, or the steady
% state sinusoidal 6.5GHz driver.
% if (t < 1.0e-9)
%     Vsrc = Vocw*cos(2*pi*fc*t);           % CW source
% else
%     Vsrc = 0;
% end
% Vsrc = Vouwb*exp(-((t-tstart)/tdecay)^2)*cos(2*pi*fc*(t-tstart));
% Vsrc = -Vouwb*exp(-((t-tstart)/tdecay)^2)*sin(2*pi*fc*(t-tstart));
% Now update the E fields. First step the driving sphere...
Vin = 0; lins = 0;
for J = ant_angle:THmax

```

```

lin = I_factor*sin((J-1/2)*dth)*Hp(3,J);
Vdrv = Vsrc - Rs*lin;
if ((t >= tsample) & (J > ant_angle))
    Vin = Vin + b*Et(3,J)*dth;
    lins = lins + lin;
    if (J == (THmax - 1))
        tsample = 1e-6;
        lins = lins/(THmax - ant_angle - 1);
        exVin = Vin
        exlin = lin
        Zin = abs(Vin/Iin)
    end
end
Et(3,J) = Vdrv*sphere_factor/sin((J-1/2)*dth);
end
% Now step E fields for the free nodes
for I = 4:(Rmax-Npml)
    for J = 2:(THmax-1)
        if ((I >= ant_length) | (J >= ant_angle))
            g6 = sin((J-1/2)*dth)/sin((J-3/2)*dth);
            Ca = (1 - g2*sigma(I,J))/(1 + g2*sigma(I,J));
            Cb = g3/(1 + g2*sigma(I,J));
            HPHI1 = (g6*Hp(I,J) - Hp(I,J-1))/((I-1/2)*dth);
            HPHI2 = Hp(I-1,J) - ((I-1/2)/(I-3/2))*Hp(I,J);
            Er(I,J) = Ca*Er(I,J) + Cb*HPHI1;
            Et(I,J) = Ca*Et(I,J) + Cb*HPHI2;
        end
    end
end
% Compute the E fields at the ground plane
J = THmax;
for I = 4:(Rmax-Npml)
    Ca = (1 - g2*sigma(I,J))/(1 + g2*sigma(I,J));
    Cb = g3/(1 + g2*sigma(I,J));
    HPHI2 = Hp(I-1,J) - ((I-1/2)/(I-3/2))*Hp(I,J);
    Er(I,J) = 0;
    Et(I,J) = Ca*Et(I,J) + Cb*HPHI2;
end
% Compute the E fields along the line of symmetry.
for I = ant_length:(Rmax-Npml)
    Ca = (1 - g2*sigma(I,1))/(1 + g2*sigma(I,1));
    Cb = g3/(1 + g2*sigma(I,1));
    HPHI1 = (Hp(I,2) - Hp(I,1))/((I-1/2)*dth);
    Er(I,1) = Ca*Er(I,1) + Cb*HPHI1;
    Et(I,1) = 0;
end

```

```

% Update the E fields in the PML region, except for the ground plane
layer = 1;
for I = (Rmax-Npml+1):Rmax
    for J = 2:(THmax-1)
        g6 = sin((J-1/2)*dth)/sin((J-3/2)*dth);
        Car = (1 - g2*sigmaPML(layer))/(1 + g2*sigmaPML(layer));
        Cbr = g3/(1 + g2*sigmaPML(layer));
        Cat = Car;
        Cbt = Cbr;
        HPHI1 = (g6*Hp(I,J) - Hp(I,J-1))/((I-1/2)*dth);
        HPHI2 = (Hp(I-1,J) - ((I-1/2)/(I-3/2))*Hp(I,J));
        Er(I,J) = Cat*Er(I,J) + Cbt*HPHI1;
        Et(I,J) = Car*Et(I,J) + Cbr*HPHI2;
    end
    layer = layer + 1;
end
% Calculate the E field in the PML region at the ground plane
layer = 1;
J = THmax;
for I = (Rmax-Npml+1):Rmax
    Car = (1 - g2*sigmaPML(layer))/(1 + g2*sigmaPML(layer));
    Cbr = g3/(1 + g2*sigmaPML(layer));
    HPHI2 = (Hp(I-1,J) - ((I-1/2)/(I-3/2))*Hp(I,J));
    Er(I,J) = 0;
    Et(I,J) = Car*Et(I,J) + Cbr*HPHI2;
    layer = layer + 1;
end
end
% Plot the results
% Convert the polar coordinates to rectangular coordinates and mirror the
% antenna simulation data to show both +x and -x views of the radiating
% field.
for I = 1:Rmax
    for J = 1:THmax-1
        x = 151 + round((I*sin((J-1)*dth)));
        x2 = 303 - x;
        y = 1 + round((I*cos((J-1)*dth)));
        Ecart(x,y) = Et(I,J);
        Ecart(x2,y) = Et(I,J);
    end
end
% Load a new Cartesian matrix for interpolation of missing elements
for I = 1:(2*Rmax-1)
    for J = 1:Rmax
        EcartNew(I,J) = Ecart(I,J);
    end
end

```

```

end
% Interpolate the missing elements in the Cartesian matrix
Imin = 2; Imax = 2*Rmax - 2; Jmin = 2; Jmax = Rmax - 1;
for I = Imin:Imax
    for J = Jmin:Jmax
        if ((Ecart(I,J) == 0) & (Rmax*cos((I-Rmax)*dth*91/151) + 25 >= J))
            ItempLo = I - 1;
            ItempHi = I + 1;
            while ((Ecart(IntempLo,J) == 0) & (IntempLo > 1))
                ItempLo = ItempLo - 1;
            end
            while ((Ecart(IntempHi,J) == 0) & (IntempHi < 2*Rmax - 1))
                ItempHi = ItempHi + 1;
            end
            M = Ecart(IntempLo,J);
            N = Ecart(IntempHi,J);
            if (M == 0)
                temp1 = N;
            elseif (N == 0)
                temp1 = M;
            else
                temp1 = sign(M+N)*sqrt(abs(M*N));
            end
            JtempLo = J - 1;
            JtempHi = J + 1;
            while ((Ecart(I,JtempLo) == 0) & (JtempLo > 1))
                JtempLo = JtempLo - 1;
            end
            while ((Ecart(I,JtempHi) == 0) & (JtempHi < Rmax))
                JtempHi = JtempHi + 1;
            end
            M = Ecart(I,JtempLo);
            N = Ecart(I,JtempHi);
            if (M == 0)
                temp2 = N;
            elseif (N == 0)
                temp2 = M;
            else
                temp2 = sign(M+N)*sqrt(abs(M*N));
            end
            if (temp1 == 0)
                EcartNew(I,J) = temp2;
            elseif (temp2 == 0)
                EcartNew(I,J) = temp1;
            else
                EcartNew(I,J) = sign(temp1+temp2)*sqrt(abs(temp1*temp2));
            end
        end
    end
end

```

```

        end
    end
end
EcartNew(1,151) = 1.2;    % bogus points added to force scaling
EcartNew(2,151) = -1.0;  % in the final plot
end
I = 1:301;
J = 1:151;
x(I) = I;
y(J) = J;
surf(y(J), x(I), EcartNew(I,J))
xlabel('Y-axis cm'), ylabel('X-axis cm')
zlabel('Etheta V/m')
% title('FDTD of Cone Antenna at 6.5GHz: CW Driver')
title('FDTD of Cone Antenna at 6.5GHz: UWB Driver')
shading interp
colormap pink

```

Matlab Function: *Vsrc.m* (11/3/04)

```
% This function models the UWB Gaussian-shaped voltage pulse to drive
% a UWB antenna.
% It accepts a max. timestep "Nmax"
% It outputs an array of values. It should be called once, and its
% matrix output results should be referenced.
%
% Robert Brocato
% 11/3/04
function [Vout] = Vsrc(Nmax, dt)
% Set the time step at 1psec
dt = 1e-12;
% Take pulse center time (nstart) = 200psec
nstart = 200;
% The 1/e time (ndecay) is taken as 10dt
ndecay = 50;
% Pulse center frequency (fc) is 6.5GHz, center of FCC 3.1-10.6 GHz range
fc = 6.5*10^9;
% Pulse max. amplitude (Vo) is taken as 1 volt
Vo = 1.0;
for n = 1:Nmax
    Vout(n) = Vo*exp(-((n-nstart)/ndecay)^2)*cos(2*pi*fc*(n-nstart)*dt);
end
plot(Vout)
% title('Gaussian-shaped pulse: -Vo*exp(-((n-200)/50)^2)*sin(2*pi*fc*(n-200)dt)')
xlabel('Time (psec)');
title('Gaussian-shaped pulse: Vo*exp(-((n-200)/50)^2)*cos(2*pi*fc*(n-200)dt)')
```

Matlab Function: *IdealCone.m* (10/28/04)

```
% Closed form field solution for an infinite conical antenna
%
% R.W. Brocato
%
% The closed form solution for the infinite conical antenna is taken from
% the book Fields and Waves by Ramo, Whinnery, and Van Duzer, pp. 462-465.
%
function [Et] = IdealCone()
% Use the same discretization parameters (dr, dth, dt, ranges, etc.) as are
% used by the FDTD simulation.
dr = 0.003;
dth = 1;
dt = 5e-12;
Z = 377;
c = 3e+8;
% Range extends to 15 median wavelengths in R and from 0-90 degrees in
% theta. The angle of the infinite cone antenna is the same as for the one
% used in the FDTD simulation.
Rmax = 151;
THmax = 91;
cone_angle = 31;
theta = cone_angle;
r = dr;
% Initialize the wave parameters, frequency (fo), amplitude (Eo),
% wavenumber (k), etc.
Vo = 0.033;
Eo = Vo/(2*log(cot((pi/180)*(cone_angle-1)/2)));
fo = 6.5e+9;           % 6.5GHz
k = 2*pi*fo/c;       % 136.1
t = 0;
for i = 1:Rmax
    theta = cone_angle;
    for j = 1:(THmax-1)
        if (j >= cone_angle)
            wave = cos(2*pi*fo*t - k*r);
            Et(i,j) = (1/(r)*sin((theta-1)*pi/180))*(Eo*wave);
        else
            Et(i,j) = 0.0;
        end
        Hp(i,j) = Et(i,j)/Z;
        theta = theta + dth;
    end
    r = r + dr;
end
```

```

% Plot the results
% Convert the polar coordinates to rectangular coordinates and mirror the
% antenna simulation data to show both +x and -x views of the radiating
% field.
% The solution space must be compressed by 5x to accomplish the polar to
% rectangular conversion.
for i = 1:Rmax
    for j = 1:THmax-1
        x = 32 + round((i*sin((j-1)*dth*pi/180))/5);
        x2 = 63 - x;
        y = 1 + round((i*cos((j-1)*dth*pi/180))/5);
        Ecart(x,y) = Et(i,j);
        Ecart(x2,y) = Et(i,j);
    end
end
i = 1:62;
j = 1:31;
x(i) = i*1.5;
y(j) = j*1.5;
surf(y(j), x(i), Ecart(i,j))
xlabel('Y-axis cm'), ylabel('X-axis cm')
zlabel('Etheta V/m')
title('Infinite 30 Degree Cone Antenna at 6.5GHz: Closed Form Solution')
shading interp
colormap pink

```

Matlab Function: *ConeGeom.m* (10/10/04)

```
% This function plots the physical outline of a conical antenna.
%
% R.W. Brocato
function [Z] = ConeGeom()
ant_angle = 31;
R = 15;
Ha = cos((ant_angle - 1)*pi/180);
Zmax = R*Ha;
N = 60;
Zprev = 0;
for i = 1:N+1
    for j = 1:N+1
        if (Zprev < Zmax)
            Z(i,j) = Ha*R*(i-1)/N;
            term = R*(i-1)/N;
            Zprev = term;
        else
            Z(i,j) = R*cos(pi/2*(2*(i-1)/N - 1) - pi/2);
            term = (1/Ha)*R*cos(pi/2*(2*(i-1)/N - 1));
        end
        arg = 2*pi*(j-1)/N;
        Y(i,j) = term*cos(arg + pi/2);
        X(i,j) = term*cos(arg + pi);
    end
end
surf(X,Y,Z)
hidden on
```

Matlab Function: *Pplot.m* (11/11/04)

```
% Create a polar plot to show antenna far field radiation pattern.
%
% R.W. Brocato
%
% This function accepts the input matrix from a simulation run of FDTD1 and
% generates a 2-D polar plot. The run of FDTD1 should be for 8000
% timesteps at dt = 0.2psec.
%
function [] = Pplot(Ft)
% Define some constants: Jmax is the maximum angle to view up to, Rview is
% the far field radius value at which to plot the field.
Jmax = 91;
Rview = 121;
tstep = pi/180;
tmax = pi/2;
t = tstep:tstep:tmax+tstep;
% Manually enter the radiation field data from the Liu and Grimes paper.
B(1:10) = [0 0.02 0.03 0.04 0.06 0.07 0.08 0.09 0.10 0.11];
B(11:21) = [0.12 0.13 0.15 0.16 0.18 0.19 0.2 0.21 0.23 0.24 0.25];
B(22:33) = [0.26 0.27 0.29 0.3 0.31 0.32 0.33 0.35 0.36 0.37 0.38 0.4];
B(34:45) = [0.41 0.42 0.44 0.45 0.46 0.48 0.49 0.5 0.52 0.53 0.55 0.59];
B(46:57) = [0.60 0.62 0.63 0.65 0.66 0.67 0.69 0.7 0.71 0.73 0.74 0.75];
B(58:69) = [0.77 0.78 0.79 0.8 0.81 0.82 0.83 0.85 0.86 0.87 0.88 0.88];
B(70:81) = [0.89 0.9 0.91 0.91 0.92 0.93 0.94 0.94 0.95 0.96 0.96 0.97];
B(82:91) = [0.97 0.98 0.98 0.99 0.99 0.99 0.99 1.00 1.00 1.00];
% Scale the data in amplitude
B = B * 0.043;
% Plot the simulation results vs. the paper results
% polar(t, Ft(Rview,1:Jmax),'r')
C(1:Jmax) = Ft(Rview, 1:Jmax);
C(Jmax+1:2*Jmax) = B;
t2(1:Jmax) = t;
t2(Jmax+1:2*Jmax) = -t;
polar(t2, C, 'r')
```

Matlab Function: *Film.m* (11/25/04)

```
% Film.m : Creates a movie of the conical antenna FDTD simulation by
% calling FDTD1 repeatedly with different time intervals. Call this
% routine by m = film; if it is desired to save the movie file, m, for
% replay later.
% To execute the results of this function, open the Matlab data file
% GaussianPulse. This loads the workspace with the struct array, m. Then
% run movie(m,10,3); in the command window.
%
% R.W. Brocato
%
function [m] = Film()
tcount = 0;
for I = 1:23
    if (I < 7)
        tinc = 250;
    else
        tinc = 500;
    end
    tcount = tcount + tinc;
    FDTD1(tcount);
    view(-90,60);
    axis([0 200 0 300 -1 2]);
    m(I) = getframe;
end
cla
movie(m,10,3)
```

MathCAD Worksheet for Calculation of
Conical Antenna Driving Point Impedance [16]

Prepare some general antenna parameters

$$\theta_0 := 30 \cdot \frac{\pi}{180} \quad \text{semi-angle of the antenna in radians}$$

$$a := 45 \quad \text{height of the antenna in mm}$$

$$f_c := 6.5 \cdot 10^9 \quad \text{center frequency of interest}$$

$$\lambda := \frac{3 \cdot 10^{11}}{f_c} \quad \lambda = 46.154 \quad \text{wavelength in mm.}$$

$$k := \frac{2 \cdot \pi}{\lambda} \quad ka := k \cdot a \quad ka = 6.126$$

Calculate the impedance of an infinite cone antenna

$$Z_c := 60 \cdot \ln \left(\cot \left(\frac{\theta_0}{2} \right) \right) \quad Z_c = 79.017$$

Calculate the impedance of the finite cone antenna

$$\zeta(n, ka) := \frac{H_2(n, ka)}{H_2(n-1, ka) - \frac{n}{ka} \cdot H_2(n, ka)}$$

$$n := 1, 3..17$$

$$S := \frac{60}{Z_c} \left[\sum_n \left[\frac{2 \cdot n + 1}{n \cdot (n + 1)} \cdot \left(\text{Leg}(n, \cos(\theta_0)) \right)^2 \cdot \zeta(n, ka) \right] \right] \quad S = -0.157 + 0.964i$$

$$\beta_{\alpha} := e^{-i \cdot 2 \cdot ka} \cdot \frac{(1 + iS)}{iS - 1} \quad \beta_{\alpha} = -0.037 + 0.073i$$

$$Z_{in} := Z_c \cdot \frac{1 - \beta_{\alpha}}{1 + \beta_{\alpha}}$$

$$Z_{in} = 84.063 - 12.396i$$

$$|Z_{in}| = 84.972$$

Reference [16]: S.S. Sandler and R.W.P. King, "Compact Conical Antenna for Wide-Band Coverage",
IEEE Trans. on Ant. and Prop., Vol. 42, No. 3, Mar. 1994, pp. 436-439.

Distribution

1	MS0874	Robert W. Brocato, 1751
1	MS0874	David W. Palmer, 1751
1	MS9154	Jack Skinner, 8245
1	MS9154	Ron Kyker, 8245
1	MS0874	Gregg A. Wouters, 1751
1	MS0874	Vincent M. Hietala, 1738
1	MS1071	Michael G. Knoll, 1730
1	MS9018	Central Technical Files, 8945-1
2	MS0899	Technical Library, 9616

LIBRARY DOCUMENT
DO NOT DESTROY
RETURN TO
LIBRARY VAULT