

A Student-accessible Graphical User Interface-based FDTD Algorithm for Visualization of Electromagnetic Fields*

S. SINGH, S. BHATT, S. KAPOOR, A. BHAVSAR and B. P. KUMAR

Department of Electrical and Electronic Engineering, California State University, 6000 J Street, Sacramento, CA-95819-6019, USA. E-mail: kumarp@ecs.csus.edu

The topic of Maxwell's equations and applications is always an interesting, though challenging area for faculty and students. This paper outlines a fast and user-interactive MATLAB-based graphical user interface (GUI) software for electromagnetic field calculations using a finite difference time domain (FDTD) algorithm. The significant reduction in computation time is achieved through a vector-based MATLAB algorithm, and additionally, the GUI user interface facilitates the understanding and modeling of electromagnetic field effects. This paper is primarily directed at teachers and students involved in advanced electromagnetic and RF design studies at the graduate level, for example, courses in microwave engineering, power engineering and biomedical engineering. Typical electromagnetic simulation studies are presented in this paper to illustrate the efficacy and user-friendly nature of the software.

INTRODUCTION

NUMERICAL CALCULATION and visualization of electromagnetic fields (EMF) and applications is essential in the graduate electrical engineering or biomedical curriculum, especially for students involved in study of electromagnetic fields and electromagnetic field effects. Sophisticated algorithms [1–6] currently exist, which can accurately simulate EMF interactions with media, for example, the effects of EMF from cellular phones on the human head. These algorithms use numerical techniques like FDTD, method of moments and the method of finite elements. Commercially available software, such as Remcom's X-FDTD[®], Ansoft's HFSS[®], Comsol's FEMLAB[®] are powerful but expensive, and require advanced computing platforms, hence being out of reach of many university budgets.

The EMF simulation MATLAB-based software presented in this paper [7] is very suitable for faculty and students involved in advanced graduate level microwave, power or biomedical curricula. The educational contribution and impact of this paper, and its expected audience, are described in the next section.

EDUCATIONAL BENEFIT OF SOFTWARE FOR ELECTROMAGNETIC SIMULATION

The main goal of this contribution is to provide an inexpensive but easily interactive electromagnetic simulation tool to teachers and students

alike. Some of the specific areas and courses that would benefit by this software are listed below.

Microwave Engineering (senior undergraduate and graduate level)

Specific areas within microwave engineering curriculum that would be applicable are the following:

- Generation and visualization of electromagnetic fields of transmission line structures such as antennas, coaxial lines, waveguide structures and more currently, microstrip circuits. Applicable courses include the following:
 - Transmission Lines and Fields (undergraduate)
 - Microwave Engineering I—Analysis and Passive devices (graduate)
 - Microwave Engineering II—Active devices (graduate)

Power Engineering (senior undergraduate and graduate level)

Specific areas within Power Engineering curriculum that would be applicable are the following:

- Generation and visualization of electromagnetic fields of electric generators, transformers and power transmission lines. Applicable courses include the following:
 - Power System Laboratory (undergraduate)
 - Power Electronics Laboratory (undergraduate)
 - Advanced Power Electronic Systems Applications (graduate)
 - Advanced Topics in Power Systems (graduate)

* Accepted 2 April 2005.

Biomedical Engineering (senior undergraduate and graduate level)

Students studying practical radiology and radiotherapy courses will find this software very useful in the following applications:

- Modeling of electromagnetic field (EMF) interaction with biological media. This would simulate, for example, the effects of cellular phone radiation on human subjects in close proximity [5, 6], which is an area of significant interest in the present wireless age.
- Modeling and study of radiation therapy systems such as microwave hyperthermia [1, 4, 7], which involves the application of focused microwave radiation to treat tumors in the body.

DETAILS OF MATLAB-BASED FDTD SOFTWARE FOR EMF SIMULATION

The structure of the integrated FDTD simulation software program, presented in this paper, is shown in Fig. 1. The main components of the software, the FDTD simulation program and the source/object models, are explained below.

FDTD simulation program

The FDTD simulation program, presented in this paper, consists of 500 lines of MATLAB code that was developed over many years at the California State University, Sacramento. The key sequential steps in the FDTD based field calculation program are given below:

- Input fundamental constants, such as frequency of operation, velocity of light, and electric/magnetic parameters of medium.
- Input cells per wavelength (at least 10 cells per wavelength recommended)
- Input total number of time steps (usually taken as 500)
- Input source center position w.r.t. origin (x-y-z coordinates)
- Input distance between source and object (x-y-z coordinates)
- Calculate the dimensions of the simulation space (using parameters listed above)

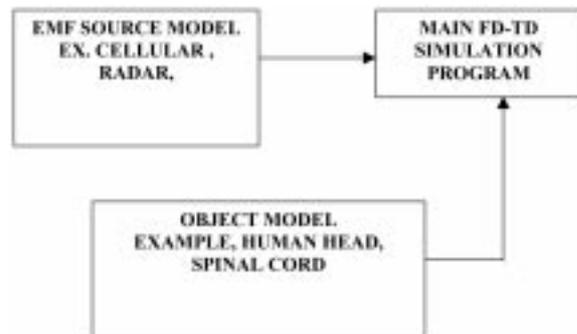


Fig. 1. Block diagram of integrated EMF simulation software.

- Input source and object model (as a file property. mat)
- Initialize electric and magnetic fields
- Start time loop
- Apply boundary conditions
- Update electric fields
- Update sources
- Update magnetic fields
- Calculate and plot electric field, specific absorption rate (SAR) or induced current density values, over the simulation space.

Advantages of present FDTD calculation program

The most significant feature of this section of the software is the vectorized form of programming in MATLAB. This advantage is illustrated below, where two methods of programming are compared, the first being a direct **nested FOR LOOP structure**, and the second being the **vectorized structure**. Both these samples represent the FDTD programming for the same calculation of electric and magnetic fields. As seen from the two samples of programming below, the **FOR LOOP** programming approach has one time-stepping loop, and three space-stepping loops, for the x-y-z coordinates of space. On the other hand, the **vectorized** programming approach uses only one time-stepping loop, and the spatial-stepping is done in parallel for all elements of the x-y-z space vectors. In summary, the vectorized approach is almost 12 times faster for 50 time steps, and almost 30 times faster for 300 time steps.

Sample section of nested FOR LOOP programming

```

for n=3:Nmax;
    FOR loop for time increment
    for k=2:Dmax;
        FOR loop for z-axis increment
        for j=2:Dmax;
            FOR loop for y-axis increment
            for i=2:Dmax;
                FOR loop for x-axis increment
                Hx(i,j,k,n)=
                    Hx(i,j,k,n-1) + Ey(i,j,k+1,n-1)
                    -Ey(i,j,k,n-1) - Ez(i,j+1,k,n-1)
                    + Ez(i,j,k,n-1) );
                Hy(i,j,k,n)=
                    (Hy(i,j,k,n-1) + Ez(i+1,j,k,n-1)
                    -Ez(i,j,k,n-1) - Ex(i,j,k+1,n-1)
                    + Ex(i,j,k,n-1) );
                Hz(i,j,k,n)=
                    (Hz(i,j,k,n-1) + Ex(i,j+1,k,n-1)
                    -Ex(i,j,k,n-1) - Ey(i+1,j,k,n-1)
                    + Ey(i,j,k,n-1) );
                Ex(i,j,k,n)=
                    Ca(i,j,k) * Ex(i,j,k,n-1)
                    +Cb(i,j,k) * (Hz(i,j,k,n-1)
                    -Hz(i,j-1,k,n-1)
                    -Hy(i,j,k,n-1) + Hy(i,j,k-1,n-1) );
                Ey(i,j,k,n)=
                    Ca(i,j,k) * Ey(i,j,k,n-1)
  
```

```

+Cb(i, j, k) * (Hx(i, j, k, n-1)
-Hx(i, j, k-1, n-1)
-Hz(i, j, k, n-1) +Hz(i-1, j, k, n-1) );
Ez(i, j, k, n) =
Ca(i, j, k) * Ez(i, j, k, n-1)
+Cb(i, j, k) * (Hy(i, j, k, n-1)
-Hy(i-1, j, k, n-1)
-Hx(i, j, k, n-1) +Hx(i, j-1, k, n-1) );
end;
end;
end;
end;

```

Sample section of vectorized programming:

```

for n=1:nmax;
  FOR loop for time increment
ex(1:ie, 2:je, 2:ke, ncur) =
  ca(1:ie, 2:je, 2:ke, 1) .
  *ex(1:ie, 2:je, 2:ke, npr1)
+cb(1:ie, 2:je, 2:ke, 1) .
  *hz(1:ie, 2:je, 2:ke, npr1)
-hz(1:ie, 1:je-1, 2:ke, npr1)
+hy(1:ie, 2:je, 1:ke-1, npr1)
-hy(1:ie, 2:je, 2:ke, npr1) );
ey(2:ie, 1:je, 2:ke, ncur) =
  ca(2:ie, 1:je, 2:ke, 1) .
  *ey(2:ie, 1:je, 2:ke, npr1)
+cb(2:ie, 1:je, 2:ke, 1) .
  *hx(2:ie, 1:je, 2:ke, npr1)
-hx(2:ie, 1:je, 1:ke-1, npr1)
+hz(1:ie-1, 1:je, 2:ke, npr1)
-hz(2:ie, 1:je, 2:ke, npr1) );
ez(2:ie, 2:je, 1:ke, ncur) =
  ca(2:ie, 2:je, 1:ke, 1) .
  *ez(2:ie, 2:je, 1:ke, npr1)
+cb(2:ie, 2:je, 1:ke, 1) .
  * (hx(2:ie, 1:je-1, 1:ke, npr1)
-hx(2:ie, 2:je, 1:ke, npr1)
+hy(2:ie, 2:je, 1:ke, npr1)
-hy(1:ie-1, 2:je, 1:ke, npr1) );
hx(2:ie, 1:je, 1:ke, ncur) =
  da*hx(2:ie, 1:je, 1:ke, npr1)
+db*(ey(2:ie, 1:je, 2:kb, ncur)
-ey(2:ie, 1:je, 1:ke, ncur)
+ez(2:ie, 1:je, 1:ke, ncur)
-ez(2:ie, 2:jb, 1:ke, ncur) );
hy(1:ie, 2:je, 1:ke, ncur) =
  da*hy(1:ie, 2:je, 1:ke, npr1)
+db*(ex(1:ie, 2:je, 1:ke, ncur)
-ex(1:ie, 2:je, 2:kb, ncur)
+ez(2:ib, 2:je, 1:ke, ncur)
-ez(1:ie, 2:je, 1:ke, ncur) );
hz(1:ie, 1:je, 2:ke, ncur) =
  da*hz(1:ie, 1:je, 2:ke, npr1)
+db*(ex(1:ie, 2:jb, 2:ke, ncur)
-ex(1:ie, 1:je, 2:ke, ncur)
+ey(1:ie, 1:je, 2:ke, ncur)
-ey(2:ib, 1:je, 2:ke, ncur) );
end;

```

Source and object models

The **source model** represents the origin of the electromagnetic field radiation, such as:

- an incident plane wave from cellular, television or radio sources;
- a single antenna, or an antenna array for medical therapy applications.

The **object model** represents the medium around the source model, such as:

- lossless medium (air or vacuum);
- lossy media, for example, biological tissue of the human body.

To use the FDTD method, a computational domain must be specified, which includes both the source and object models described above. The computational domain, which is comprised by a large number of small cells, is the space where the simulation will be performed.

The Graphic Property Editor generates a graphical user interface (GUI) for the user. The GUI implementation facilitates a user to specify or edit properties of any cell within computational domain easily.

OVERVIEW AND FUNCTIONING OF THE GRAPHIC PROPERTY EDITOR

The design of the Graphic Property Editor tool was implemented using MATLAB [9] programming language. The design of the tool can be divided into three subsections:

- ‘Main’ window of the program, which is displayed when the MATLAB script is invoked from the MATLAB command prompt.
- ‘Horizontal View’ window.
- ‘Generation Matrix’ window.

These sub-sections of the GUI program are detailed in the sections below.

‘Main’ window

A ‘Main’ window is displayed when the project is run from the MATLAB command prompt by typing ‘project’ as shown in Fig. 2.

The main components of the ‘Main’ window are as given below:

- ‘X Length’ and ‘Y Length’ editable textboxes are used to enter the dimensions of the new horizontal cross section view.
- ‘Enter initial Z value’ editable textbox. In this editable textbox, z-axis location of the first layer of the horizontal cross section view is entered.
- ‘Enter Maximum number of layers you want to enter’ textbox. In this textbox, maximum number of layers a user wants to create is entered. A matrix structure, **topviewCellProperties** is created according to the dimension entered in this textbox.
- ‘Top View’ pushbutton. When the ‘Top View’ pushbutton is pressed, a window ‘Horizontal View’ is opened, which will be explained shortly.
- ‘Generate Matrix’ pushbutton. When the ‘Generate Matrix’ pushbutton is pressed, a matrix is

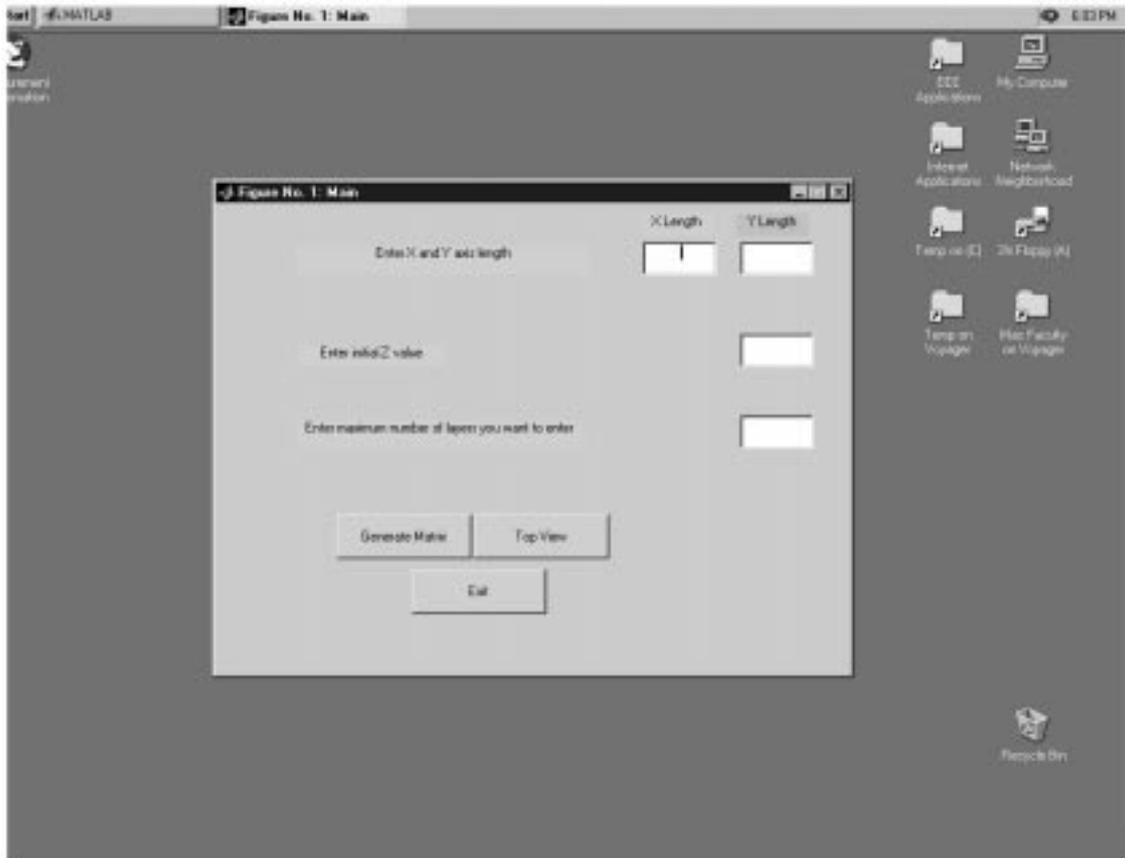


Fig. 2. 'Main' window.

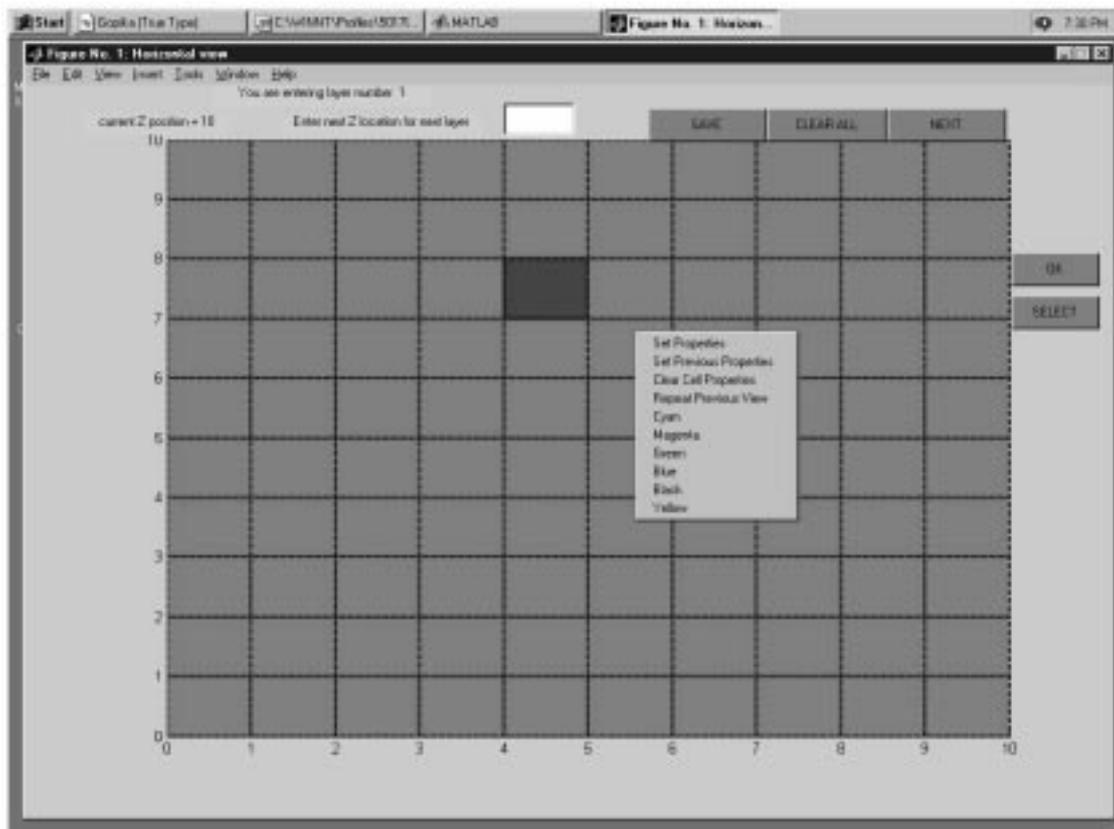


Fig. 3. 'Horizontal View' window.

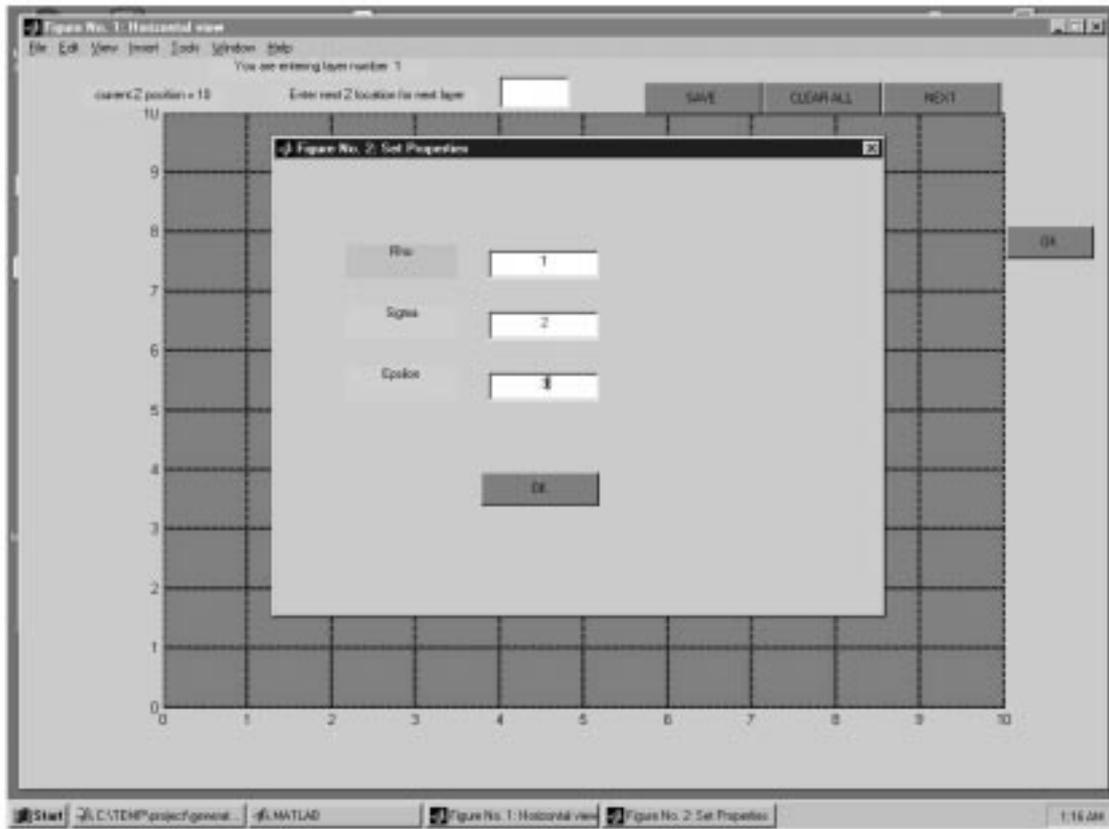


Fig. 4. 'Set Properties' window.

created which contains x, y and z locations of each cell, and their properties.

- *'EXIT' pushbutton.* When the 'EXIT' pushbutton is pressed, all variables are cleared from the workspace and 'Main' window is closed.

'Horizontal View' window

The 'Horizontal View' window is opened when 'Top View' pushbutton is pressed as shown in Fig. 3. This window opens with an empty view if the new view is being created or with the existing view if the latter is being updated.

The 'Horizontal View' window has the following components:

- *Rectangle cell area.* A rectangle cell area contains all the cells whose properties can be set or changed. Properties of any cell can be set or changed by clicking the right mouse button on that cell. When the right mouse button is pressed on any cell, a popup menu with the following items: 'Set Properties', 'Set Previous properties', 'Clear Cell Properties', 'Repeat Previous View' and with all the color options appears.
- *'Set Properties' item.* When 'Set Properties' menu item is clicked, a new window, 'Set Properties' appears as shown in Fig. 4. In this window, properties 'Rho' (density), 'Sigma' (electrical conductivity) and 'Epsilon' (relative permittivity) can be entered.
- *'Set Previous Properties' item.* When 'Set Previous Properties' menu item is clicked an

appropriate cell is assigned the same properties as the properties of the last edited cell.

- *'Clear Cell Properties' item.* When 'Clear Cell Properties' menu item is clicked, properties of the appropriate cell are reset and the color of the cell is changed to red, indicating that the cell's properties have not been set.
- *'Repeat Previous View' item.* When 'Repeat Previous View' menu item is clicked, view of the previously edited layer is copied to the current layer.

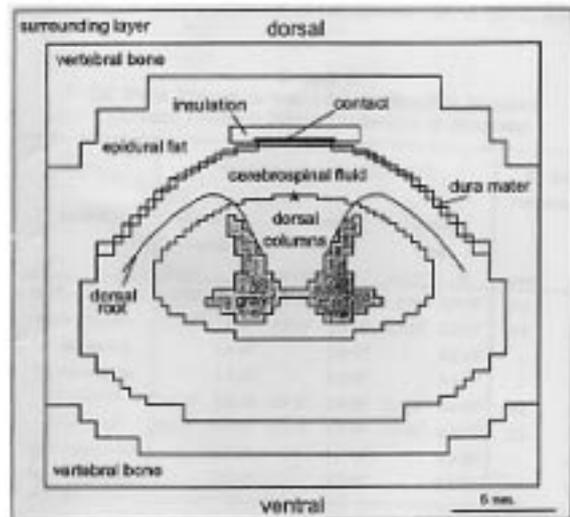


Fig. 5. Transverse section of midcervical spinal cord model.

Table 1. Material properties and dimensions of spinal model.

No.	Medium	Thickness (mm)	Dielectric constant (ϵ_r)	Permeability (μ H/m)	Conductivity (σ S/m)
1	Air	—	1	$4\pi \times 10^{-7}$	0.0
2	Skin	5/3	44.5	$4\pi \times 10^{-7}$	0.01
3	Bone	15	6.4	$4\pi \times 10^{-7}$	0.02
4	Epidural Fat	8/3	6.4	$4\pi \times 10^{-7}$	0.04
5	Cerebro-spinal fluid (CSF)	4	60	$4\pi \times 10^{-7}$	1.7
6	White matter (WM)	2	15	$4\pi \times 10^{-7}$	0.6
7	Grey matter (GM)	2	10	$4\pi \times 10^{-7}$	0.23
Layer Number 8: Central canal of the spinal column					

- *Color options.* The popup menu also contains color options like 'Cyan', 'Magenta', 'Green', 'Blue', 'Black' and 'Yellow'. When any of these color items is clicked for a particular cell, the cell's color is changed accordingly.
- *'SAVE' pushbutton.* When 'SAVE' pushbutton is pressed, the current 'Horizontal View' properties are saved on the hard disk with file name **topviewCellProperties.mat**.
- *'CLEAR All' pushbutton.* When 'CLEAR ALL' pushbutton is pressed, properties of all the cells in the current layer are cleared.
- *'Enter the Z location for next layer' editable text.* In this text box, the z-axis position for the next layer is entered. If a new view is being created and if this textbox is empty and 'NEXT' pushbutton is pressed, control of the program does not go to the next layer.
- *'NEXT' pushbutton.* When the 'NEXT' pushbutton is pressed, a new 'Horizontal View' window is opened for the next.
- *'OK' pushbutton.* When the 'OK' pushbutton is pressed, all cell properties are saved into the **topviewCellProperties.mat** file and 'Top View' window is closed and 'Main' window is opened again.
- *'SELECT' pushbutton.* A user can select a rectangle area of any size using the 'SELECT' pushbutton. This is a very useful feature because a user can select any number of cells by drawing a rectangle and can apply all the functions in the popup menu.

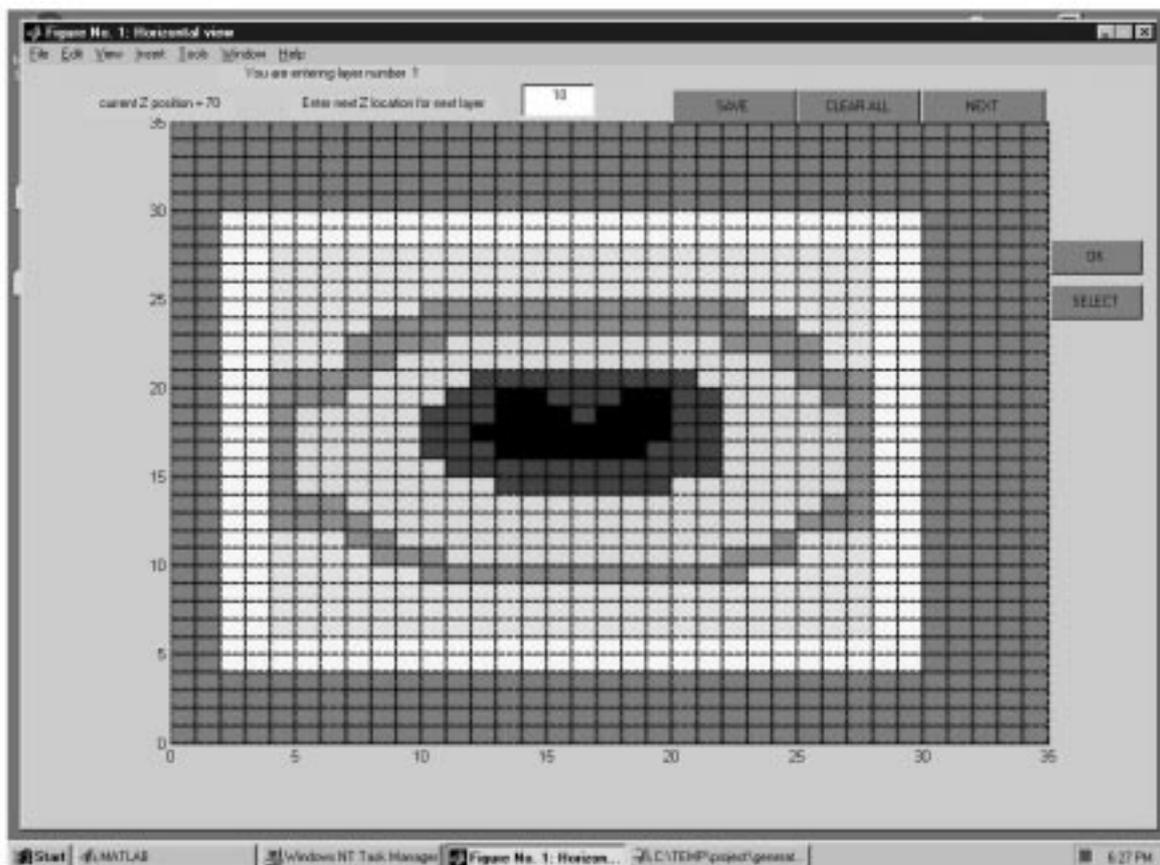


Fig. 6. Horizontal cross section of spinal cord.

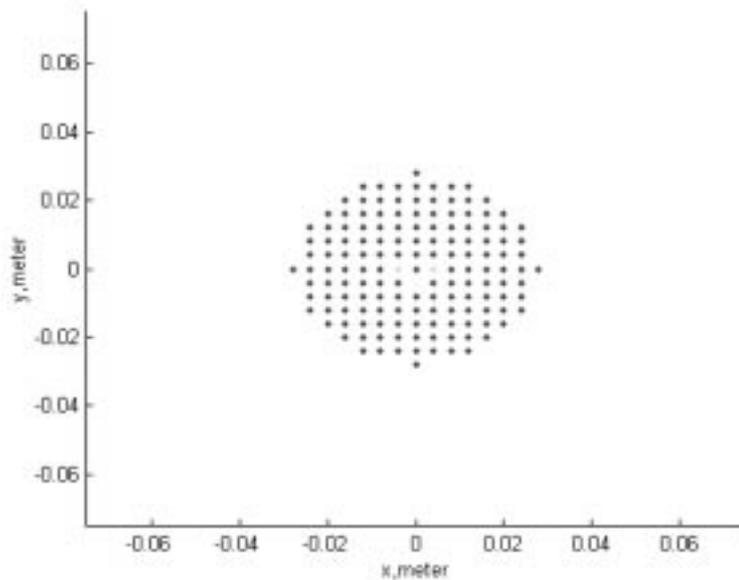
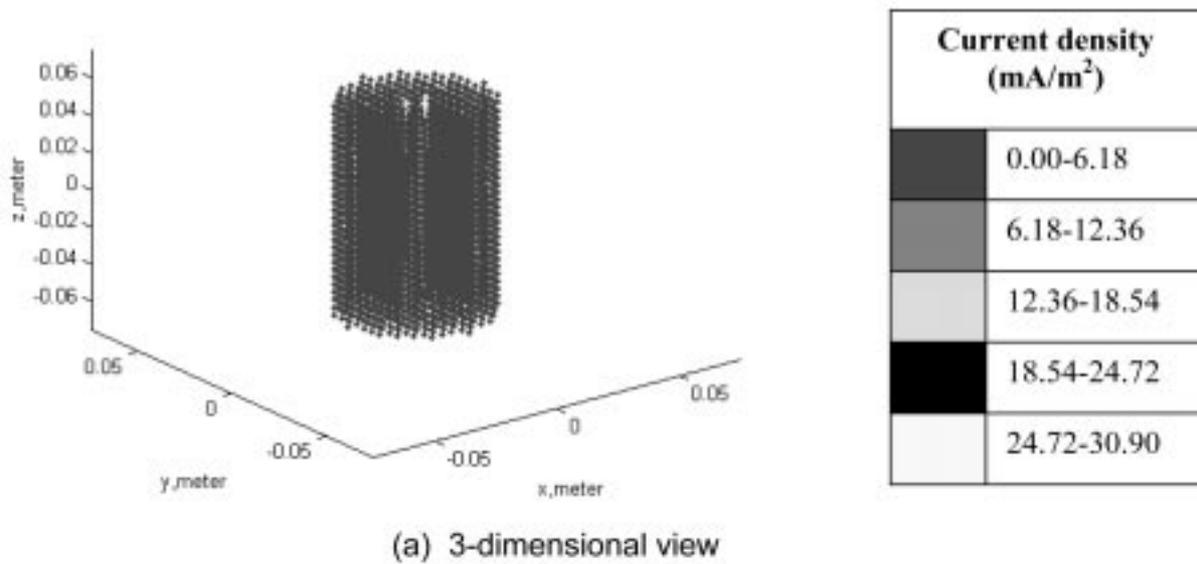


Fig. 7. Induced current density distribution in spinal cord for plane wave incidence at 100 MHz.

'Generate Matrix' pushbutton

When the 'Generate Matrix' pushbutton is pressed a matrix containing position and properties of all the cells is generated. This matrix serves as the object model in the main FDTD program, which was described above. The matrix generated by this final step of the GUI has 6 columns of data, corresponding to each cell in the object.

These first three columns contain the x-coordinate, y-coordinate and z-coordinate, respectively, of a particular object cell. The last three columns contain, respectively, the relative permittivity (no units), electric conductivity (Siemens/meter) and density (Kilograms/meter³) value at that particular cell. This matrix is saved as a **property.mat** file, and is input into the main FDTD program, described above through the following command: \gg **load property.mat**. Once

the property matrix, which defines the material properties of the simulation space, is loaded into the main FDTD program, the latter can be run to obtain the electromagnetic field (EMF) distribution.

SIMULATION EXAMPLES

The simulation study performed in this paper involves the EMF interaction of a uniform plane wave with the human spinal cord.

Cross-sections of spinal cord

To create various horizontal cross-sections of spinal cord using the Graphic Property Editor tool, a model of spinal cord in [8] was used,

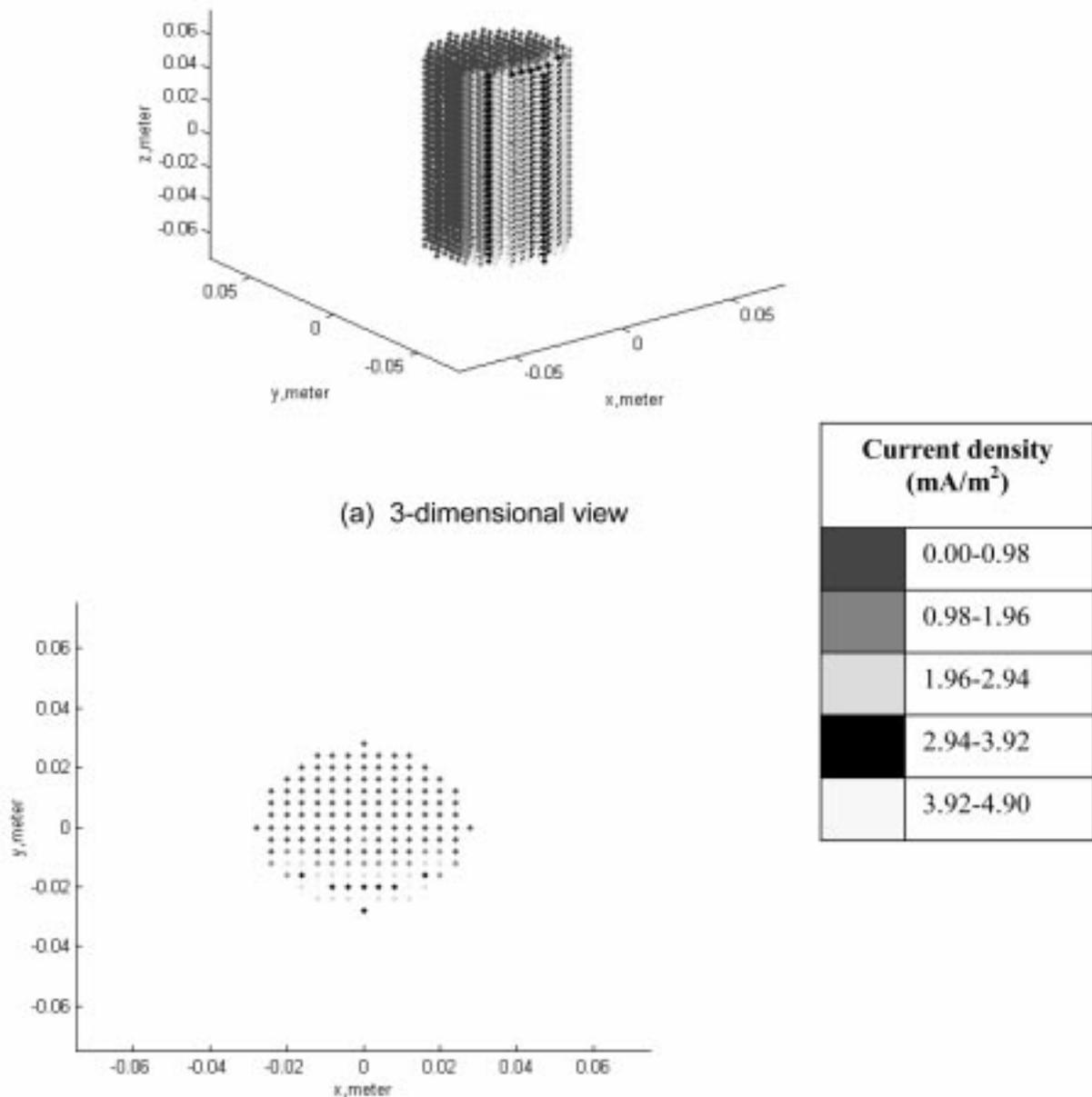


Fig. 8. Induced current density distribution in spinal cord for plane wave incidence at 700 MHz.

whose transverse section of midcervical section is shown in Fig. 5.

In this simulation study, the following assumptions are made on the cross-section of the spinal cord:

- The spinal cord cross-section is assumed to be uniform throughout the length of the spinal cord.
- The spinal cord is modeled as a concentric structure, whose dimensions and electrical, magnetic and material properties are given in Table 1, taken from reference [8].

Table 1 gives the thickness of the different layers of the spinal cord, starting with the outermost layer of air, and going inwards to the innermost layer, which is the central canal of the spinal cord. Figure 6 shows the cross-sections entered through the

Graphic Property Editor tool for the model shown in Fig. 5.

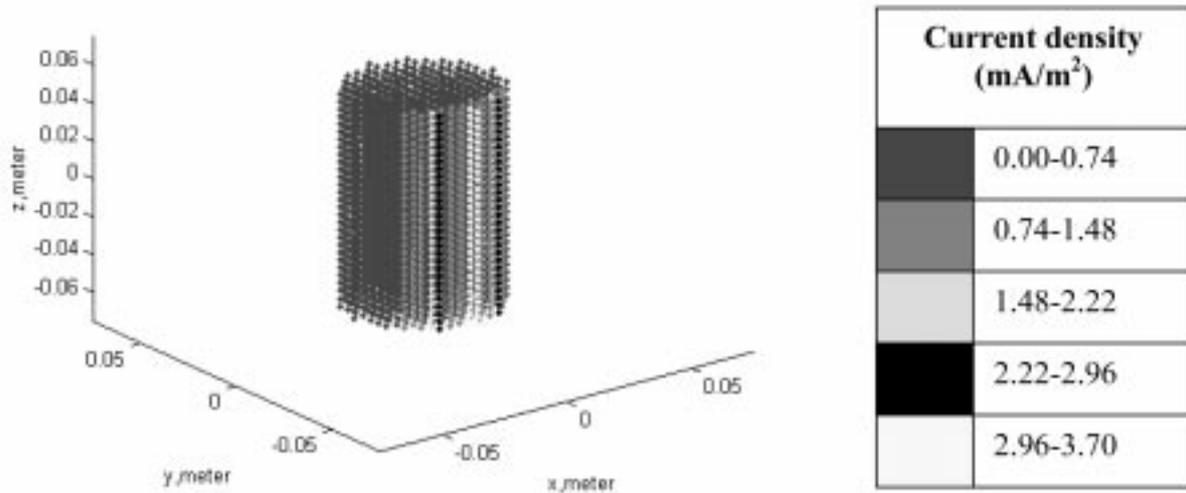
The current density induced in the spinal cord is given by:

$$\mathbf{J} = \sigma \mathbf{E} \quad (1)$$

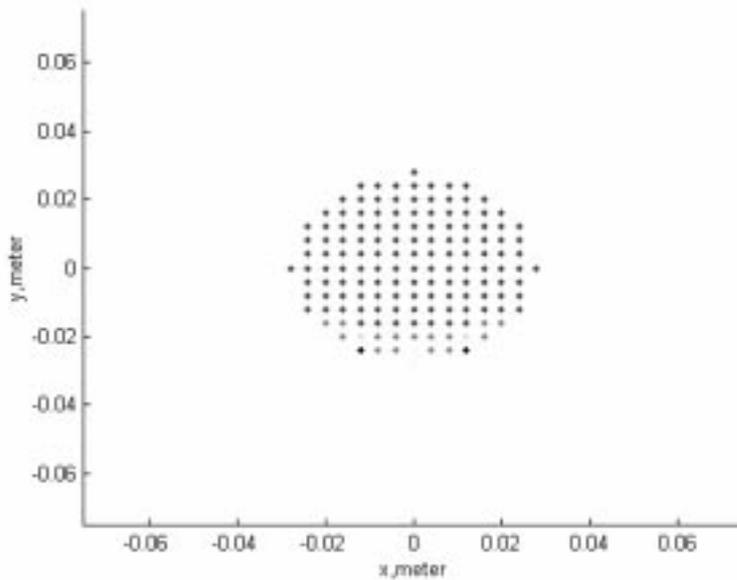
where \mathbf{J} is the electric current density vector (A/m²), $\mathbf{E}(x, y, z)$ is the electric field vector, and σ is the electricity conductivity distribution, Siemens/m.

Simulation data

The FDTD simulation grid is divided into 30 x 30 x 30 cells along the x-y-z axis, with the cell size $\Delta x = \Delta y = \Delta z = 6.7$ mm (which corresponds to $\sim \lambda/25$ at the maximum frequency of 1.8 GHz, and time step of $\Delta t = 1.9 \text{ s}^{-12}$). The latter parameters were obtained using standard FDTD calculations, which are applicable to the range of



(a) 3-dimensional view



(b) Cross-sectional view

Fig. 9. Induced current density distribution in spinal cord for plane wave incidence at 1800 MHz.

frequencies used in this simulation study. The simulations were performed for a spinal cord length of 20 cm, and a peak incident plane wave amplitude of 1 V/m. Figure 7 shows the current density distribution induced in the spinal cord for plane wave incidence at 100 MHz. Figure 7a illustrates the three-dimensional view, with the peak current density distribution along the central conducting axis of the spinal cord structure. Figure 7b gives the cross-sectional view of the current density.

Similarly, Figs 8a and 8b show the 3-dimensional view and cross-sectional view, respectively, of the induced current density distribution for plane wave incidence frequency of 700 MHz, and Figs 9a and 9b show the same results for plane

wave incidence frequency of 1800 MHz. The results showed interestingly that the peak current density decreased with increase in incident wave frequency.

CONCLUSIONS

Study and visualization of electromagnetic fields is an important aspect of graduate curriculum in electrical or biomedical engineering. This paper presents a user-friendly finite difference time domain (FDTD) algorithm that utilizes the Graphical Property Editor in MATLAB to study and model electromagnetic fields and media. The algorithm creates a graphical user interface (GUI)

that allows the very convenient import of the simulation object data into the main FDTD algorithm. The running time of the algorithm is also significantly reduced by the vector-based structure of the main FDTD program.

Simulation results are presented on the case study of electromagnetic field (EMF) interaction with the human spinal cord. A concentric model of

the human spinal cord was created using the GUI program, and the induced current density was estimated at different commercially used frequencies. The results show an inverse dependency of induced current density with the incident wave frequency.

Acknowledgement—The authors would like to thank the reviewers for their helpful suggestions.

REFERENCES

1. J. Thury, *Microwaves: Industrial, Scientific and Medical Applications*, Artech House, Boston (1992).
2. K. S. Kunz and R. J. Lubbers, *The FDTD method for Electromagnetics*, CRS Press (1996).
3. A. F. Peterson, S. L. Ray and R. Mittra, *Computational Methods for Electromagnetics*, IEEE Press (1992).
4. J. Lehmann, A. Guy, V. Johnston, G. Brunner and J. Bell, Comparison of relative heating patterns produced in tissues by exposure to microwave energy at frequencies of 2450 and 900 Mc, *Arch. Phys. Med. Rehabil.*, **43**, 1962, pp. 69–76.
5. J. Wang and O. Fujiwara, FDTD computation of temperature rise in human head for telephones, *IEEE Trans. Microwave Theory and Techniques*, **47**(8), August 1999, pp. 1528–1534.
6. J. Llauro and A. Sances, *Biological & Clinical Effects of Low-Frequency Magnetic & Electric Fields*, Springfield, IL: Chas. C. Thomas Publishing Co. (1974).
7. S. Balaguru, M. Hashemkhani, B. P. Kumar and G. R. Branner, FDTD Analysis of spinal cord response to plane wave incidence, *Proc. IEEE Antennas and Propagation Society Int. Symp.*, July 2001, Boston, Vol. 3, pp. 332–335.
8. W. A. Wesselink, J. Holsheimer and H. B. K. Boom, Analysis of current density and related parameters in spinal cord simulation, *IEEE Trans. Rehabilitation Engineering*, **6**(2), June 1998, pp. 201.
9. MATLAB 6.0 User's Manual, The Mathworks, MA (2001).

Sachin Singh was born in 1975 in India. He completed his Bachelor of Engineering degree in 1998 from the Maulana Azad College of Technology, India, and the Master of Science (Electrical Engineering) degree in 2001 from the California State University, Sacramento. He has been working as a design engineer with Intel Corporation, Folsom, California, since August 2001.

Suketu Bhatt was born in 1977 in India. He completed his Bachelor of Engineering degree in 1998 from the Gujarat University, India, and the Master of Science (Electrical Engineering) degree in 2001 from the California State University, Sacramento. He has been working as a Product Development Engineer with Intel Corporation, Folsom, California, since August 2001.

Sarita Kapoor was born in 1976 in India. She completed her BS (Electrical Engineering) degree in 1997 from the Guru Nanak Dev Engineering College, Punjab, India, the MS (Electrical Power Systems) degree in 1998 from the Punjab Engineering College, Chandigarh, India, and the MS in Electrical Engineering in 2001 from the California State University, Sacramento.

Ambarish Bhavsar was born in the business capital city of India, Mumbai (aka Bombay). He completed his B.S. in Electronics and Telecommunications Engineering from Bharathi Vidyapeeth's College of Engineering, University of Mumbai in 1998. Thereafter he worked with a well-known cranes manufacturing company named WMI Cranes Ltd. as Assistant manager of Information Technology. Being in a constant touch with computers, he decided to get MS in electronics engineering specializing in the field of Chip Design, which he completed at the California State University, Sacramento in 2002. During his masters program he worked with Intel Corporation, Sacramento, as part of his curricular practical training. He worked on different kinds of chip design such as chipsets, flash memories and network integrated chips.

B. Preetham Kumar was born in Chennai, India in 1960. He obtained his Bachelor of Engineering (BE Hons., Electronics and Communication Engineering) in 1982 and Master of Engineering (ME Communication Systems) in 1984 from the College of Engineering, Guindy, Chennai, India. He holds a Ph.D. degree with specialization in Antenna Design from the Indian Institute of Technology, Chennai in 1993. He has worked as a Lecturer/Researcher in the Department of Electrical and Computer Engineering at the University of California, Davis, and concurrently as a part-time faculty in the Department of Electrical and Electronic Engineering at the California State University, Sacramento (CSUS) from 1993–1999. He joined CSUS on full-time basis from August 1999. Dr. Kumar's area of specialization is near-field applications of antenna arrays in microwave treatment of cancer, and the design of miniaturized RF components for wireless communication applications. He has published more than 50 papers in international journals and conferences, and has supervised 60 Master's thesis projects at CSUS. He received the Tau Beta Pi Outstanding Faculty Award in 1999, and the Outstanding Teacher Award from the College of Engineering and Computer Science at CSUS in 2000.