FACTA UNIVERSITATIS
Series: **Electronics and Energetics** Vol. 26, N^o 2, August 2013, pp. 121 - 132
DOI: 10.2298/FUEE1302121N

IMPLEMENTATION OF FDTD BASED SIMULATION ENVIRONMENT*

Bojana Nikolić, Bojan Dimitrijević, Nebojša Raičević, Slavoljub Aleksić

University of Niš, Faculty of Electronic Engineering, Serbia

Abstract. *In this paper we present an optimized implementation of finite difference time domain (FDTD) based simulation environment for electromagnetic (EM) analysis. The implementation is central processing unit (CPU) based. The optimisation is performed by introducing data structures, by multiple employment of the previously calculated elements and by intensive use of indexing tables. Comparison with conventional FDTD algorithm shows that a significant time saving can be realized. Also, the main features of graphical interface that are so far implemented in this simulation environment will be presented.*

Key words: *convolutional perfectly matched layer, finite-difference time-domain method, Maxwell's equations*

1. INTRODUCTION

Along with rapid development of computer technology today, computational electromagnetics (CEM) becomes the primary available tool for the design of antenna and microwave circuit components, electromagnetic compatibility (EMC) analysis, and the prediction of radio propagation [1]. Currently there is a large offer of commercial CEM software packages available. Some of the most popular of them are based on the finite element method (FEM) and the finite integration techniques (FIT).

The finite element method (FEM) is used to find approximate solution of partial differential equations (PDE) and integral equations [2], [3]. The solution approach is based on local approximation of original PDE by set of either algebraic equations or equivalent ordinary differential equations. A global system of equations is then generated from the element equations through a transformation of coordinates from the subdomains' local nodes to the domain's global nodes.

Received February 10, 2013
Corresponding author: Bojana Nikolić
University of Niš, Faculty of Electronic Engineering, Niš, Serbia
E-mail: bojana.nikolic@elfak.ni.ac.rs
Acknowledgement: This work is supported in part by the Ministry of education, science and technological development of Serbia within the Projects TR-32051 and TR-33008.

The finite integration technique (FIT) is a spatial discretization scheme to numerically solve electromagnetic field problems in time and frequency domain [3]. It preserves basic topological properties of the continuous equations such as conservation of charge and energy. This method covers the full range of electromagnetics (from static up to high frequency) and optic applications and is the basis for commercial simulation tools.

The finite difference time domain (FDTD) method is a full wave time domain differential equation based technique that received huge attention in the literature recently. It is a versatile method that was proposed by Yee [4] originally for two dimensional problems with metal boundaries. Initially the FDTD method was applied to scattering problems and subsequently has become one of the most popular methods used to simulate and analyze problems in electromagnetics, ranging from antennas, microwave wave circuits, electromagnetic compatibility (EMC) issues, bioelectromagnetics, electromagnetic scattering to novel materials and nanophotonics [5], [6]. Since it is a time-domain method, solutions can cover a wide frequency range with a single simulation run, the provided time step is small enough to satisfy the Nyquist–Shannon sampling theorem for the desired highest frequency. Basic advantage of this method is its simplicity and guaranteed convergence with proper choice of parameters. The disadvantages of the FDTD are high memory and computational requirements. However, this drawback becomes less and less significant, since the computational power will continue to grow exponentially in the future. Mathematical theorems for the FDTD formulation, concerning issues such as accuracy, convergence, dispersion, computational complexity and stability are available in [7].

The perfectly matched layer (PML) absorbing media [8] has proven to be the most robust and efficient technique for the termination of FDTD computational domain [9]. Its implementation, referred to as the convolutional PML (CPML) proofs to be superior over the other implementations of the PML, offering a number of advantages [10]. CPML is based on the stretched coordinate form [11] and the use of complex frequency shift (CFS) of PML parameters [12]. The application of the CPML is completely independent of the host medium. Thus, no modifications are necessary when applying it to inhomogeneous, lossy, anisotropic, dispersive or non-linear media.

In this paper we present an optimized implementation of FDTD based simulation environment for CEM analysis. The implementation is central processing unit (CPU) based. The optimisation is performed by introducing data structures, by multiple employment of the previously calculated elements and by intensive use of indexing tables. This significantly reduces execution speed, as it will be demonstrated through simulations, carried out for various thicknesses of CPML and sizes of total space. Also, the main features of graphical interface that are so far implemented in this simulation environment will be presented.

2. FDTD METHOD

2.1. Three dimensional FDTD formulation

The FDTD is a numerical technique used to solve Maxwell equations. In the FDTD method, Maxwell's curl equations are converted to their corresponding scalar partial differential equations (PDE). This is followed by the discretization of space and time domain. Central difference approximations are applied to the scalar PDE with respect to

the discretized time and space domain. This will result in discrete equations for each field component, referred to as update equations or time – stepping equations, which can be used to evaluate these field components.

2.2. Discretization of space and time domain and FDTD update equations

Spatial discretization is done by dividing the computational space into cuboidal elementary cells called Yee cells. The Yee cell together with the *E* and *H* field components displacement is shown in Fig. 1. As one can see, every *E* component is surrounded by four circulating *H* components, and every *H* component is surrounded by four circulating *E* components.

The update equations for *H* and *E* field components in free space are (for the brevity only equations for H_x and E_x field components are presented)

$$H^{n+1/2}_{x(i,j+1/2,k+1/2)} = \\ = H^{n-1/2}_{x(i,j+1/2,k+1/2)} + \frac{\Delta t}{\mu}\frac{E^{n}_{y(i,j+1/2,k+1)} - E^{n}_{y(i,j+1/2,k)}}{\Delta z} - \frac{\Delta t}{\mu}\frac{E^{n}_{z(i,j+1,k+1/2)} - E^{n}_{z(i,j,k+1/2)}}{\Delta y}, \tag{1}$$

$$E^{n+1}_{x(i+1/2,j,k)} = \\ = E^{n}_{x(i+1/2,j,k)} + \frac{\Delta t}{\varepsilon_0}\frac{H^{n+1/2}_{z(i+1/2,j+1/2,k)} - H^{n+1/2}_{z(i+1/2,j-1/2,k)}}{\Delta y} - \frac{\Delta t}{\varepsilon_0}\frac{H^{n+1/2}_{y(i+1/2,j,k+1/2)} - H^{n+1/2}_{y(i+1/2,j,k-1/2)}}{\Delta z}, \tag{2}$$

respectively. Similar update equations can be written for *y* and *z* components of *E* and *H* field.

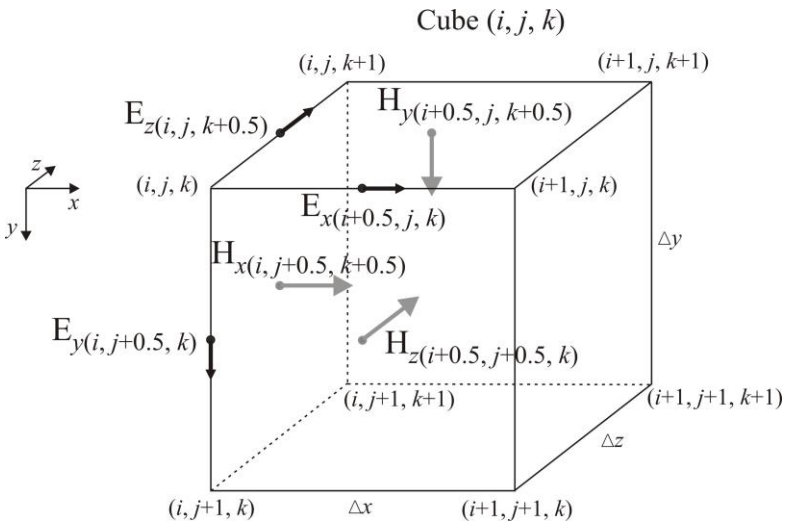

Fig. 1 Electric and magnetic field vector components in a Yee cell

It can be noticed that the temporal location of the *E* and *H* field components differs by a half time step. The *H* field components are first to be executed.

3. CPML ALGORITHM

For the simplicity, the implementation is presented on the example of free space. However, the described optimisation can be applied on arbitrary host medium.

In order to implement CPML to truncate computational domain, the stretched coordinate variables have the complex-frequency shifted (CFS) form [9] [10]

$$s_i = k_i + \frac{\sigma_i}{\alpha_i + j\omega\varepsilon_0}, \; i \in \{x, y, z\}, \tag{3}$$

where $k_i \geq 1$ is stretching coefficient, $\sigma_i \geq 0$ is medium conductivity and $\alpha_i \geq 0$ is complex frequency shift parameter. For $\alpha_i = 0$ CPML is reduced to a regular PML [8]. For $\alpha_i > 0$, the CPML is highly absorptive of evanescent and low-frequency waves [10]. However, CPML with high α_i may have problems with the absorption of propagating waves, [13], [14]. The solution to this is to put larger α_i near the air–PML boundary to allow evanescent waves to penetrate into the PML without reflection, and smaller α_i near the outer boundary of PML to improve the absorption of propagating waves.

In order to reduce numerical reflections due to FDTD space discretization, the coordinate stretching variables and thus parameters k_i, σ_i and α_i inside the CPML are graded

$$\begin{aligned} \alpha_i &= \left(\frac{D-\xi_i}{D}\right)^{m_\alpha} \alpha_{\max}, \quad k_i = 1 + \left(\frac{\xi_i}{D}\right)^{m} (k_{\max} - 1), \\ \sigma_i &= \left(\frac{\xi_i}{D}\right)^{m} \sigma_{\max}, \quad i \in \{x, y, z\}, \end{aligned} \tag{4}$$

where $\alpha_{\max}$ is the value on the air–PML interface and $k_{\max}$ and $\sigma_{\max}$ at the outer boundary. The width of the PML is denoted by D. ξ_i is actual depth in CPML, while m and m_α are CPML coefficients. Since these coefficients are one-dimensional functions, they are conveniently computed prior to the field computation and efficiently stored in one-dimensional vector arrays.

Conventional update equations for *E* and *H* field components inside the CPML are given in the following form [10]

$$\begin{aligned} &H^{n+1/2}_{x(i,j+1/2,k+1/2)} = \\ &= H^{n-1/2}_{x(i,j+1/2,k+1/2)} + \frac{\Delta t}{\mu_0}\left(\frac{E^n_{y(i,j+1/2,k+1)} - E^n_{y(i,j+1/2,k)}}{k_z \Delta z} - \frac{E^n_{z(i,j+1,k+1/2)} - E^n_{z(i,j,k+1/2)}}{k_y \Delta y}\right) \\ &+ \Psi^n_{Eyz(i,j+1/2,k+1/2)} - \Psi^n_{Ezy(i,j+1/2,k+1/2)} \end{aligned} \tag{5}$$

$$E_{x(i+1/2,j,k)}^{n+1} =$$

$$= E_{x(i+1/2,j,k)}^{n} + \frac{\Delta t}{\varepsilon_0}\left(\frac{H_{z(i+1/2,j+1/2,k)}^{n+1/2} - H_{z(i+1/2,j-1/2,k)}^{n+1/2}}{k_y \Delta y} - \frac{H_{y(i+1/2,j,k+1/2)}^{n+1/2} - H_{y(i+1/2,j,k-1/2)}^{n+1/2}}{k_z \Delta z} \right) \tag{6}$$

$$+ \Psi_{Hzy(i+1/2,j,k)}^{n+1/2} - \Psi_{Hyz(i+1/2,j,k)}^{n+1/2}$$

where auxiliary variables are defined as

$$\Psi_{Eyz(i,j,k)}^{n+1} = p_z \Psi_{Eyz(i,j,k)}^{n} + q_z \frac{\Delta t}{\mu_0} \frac{E_{y(i,j+1/2,k+1)}^{n+1} - E_{y(i,j+1/2,k)}^{n+1}}{\Delta z} \tag{7}$$

$$\Psi_{Ezy(i,j,k)}^{n+1} = p_y \Psi_{Ezy(i,j,k)}^{n} + q_y \frac{\Delta t}{\mu_0} \frac{E_{z(i,j+1,k+1/2)}^{n+1} - E_{z(i,j,k+1/2)}^{n+1}}{\Delta y} \tag{8}$$

$$\Psi_{Hyz(i+1/2,j,k)}^{n+1/2} = p_z \Psi_{Hyz(i+1/2,j,k)}^{n-1/2} + q_z \frac{\Delta t}{\varepsilon_0} \frac{H_{y(i+1/2,j,k+1/2)}^{n+1/2} - H_{y(i+1/2,j,k-1/2)}^{n+1/2}}{\Delta z} \tag{9}$$

$$\Psi_{Hzy(i+1/2,j,k)}^{n+1/2} = p_y \Psi_{Hzy(i+1/2,j,k)}^{n-1/2} + q_y \frac{\Delta t}{\varepsilon_0} \frac{H_{z(i+1/2,j+1/2,k)}^{n+1/2} - H_{z(i+1/2,j-1/2,k)}^{n+1/2}}{\Delta y}. \tag{10}$$

The update equations and corresponding auxiliary variables are presented only for H_x and E_x field components. Similar update equations and corresponding auxiliary variables can be written for y and z components of E and H field.

Coefficients in Eqs. (7), (8), (9) and (10) are

$$q_u = \frac{\sigma_u}{\sigma_u k_u + k_u^2 \alpha_u} \left(\exp\left(-\frac{\Delta t}{\varepsilon_0} \left(\frac{\sigma_u}{k_u} + \alpha_u \right) \right) - 1 \right) \tag{11}$$

$$p_u = \exp\left(-\frac{\Delta t}{\varepsilon_0} \left(\frac{\sigma_u}{k_u} + \alpha_u \right) \right), \; u \in \{x, y, z\}. \tag{12}$$

4. Implementation of Numerically Optimized FDTD Algorithm

When implementing a simulation environment there are two basic approaches that can be applied in the case of FDTD as well. The first one includes the solving of a particular in advance known problem that will be defined in the code. Although this usually guaranties fast execution and efficient use of the resources, codes like this don't seem to be very practical, since they are inflexible and it is necessary to develop a new one for every new problem. Another approach is to create a development environment with integrated graphical interface that will allow arbitrary design through the graphic editor and the possibility of extension ad upgrading by the user. Disadvantages of such an approach are the complexity of the algorithm, a significant decrease of execution speed with increase of offered options and possibilities and less efficient use of system resources. This is spe-

cially emphasized when the common FDTD method is straightforwardly implemented. However, by including a number of optimisations through the indexing and introducing of data structures, the execution speed can be significantly increased and more efficient use of resources can be achieved.

The presented implementation of FDTD algorithm retains all the advantages of development environment with integrated graphical interface. Still, the main drawbacks of this approach are compensated through optimisations in initialization part of the algorithm (*Simulation initialization* unit) that lead to the optimal run of space and time loops in main simulation body (*Simulation loop in space and time* unit).

4.1. Simulation initialization unit

- Resetting of buffer for time channels, corresponding indexes and states for each channel.
- Resetting of EF and HF structures (electric and magnetic field structures) for complete space that will be used.
- Resetting of ECF structure (32-bit field structure that describes the content of every elementary cell in space). Namely, bits 16-31 containing information necessary for acceleration are being reset, while bits 0-15 that characterize the particular design itself (geometry defined in graphic editor) stay the same.
- Resetting of auxiliary elements PE.*a*, PE.*b*, PH.*a*, PH.*b*, which store additional convolution variables Ψ_{Huv}, Ψ_{Euv} for every elementary cell in CPML.
- Additionally, one 32-bit control value is assigned to every set of PE.*a*, PE.*b*, PH.*a*, PH.*b* elements and stored in the structure named PCF. This value contains information on actual depth in CPML zone along *x, y* and *z* direction at which the particular corresponding elementary cell is placed.
- In ECF structure the presence of CPML zone along *x*, *y* and *z* direction is defined for every particular corresponding elementary cell.
- Setting of flags that point at the presence of lumped element(s) in the particular elementary cell for each axis separately. Processing of eventually present lumped element is performed separately and its functionality is defined in time in additional algorithm. In this way the following elements are defined:
  - generators (with different responses in time)
  - discrete elements (resistor, capacitor, inductor)
  - possible further extensions
- In the case of PEC (perfect electric conductor) elements, the setting of adequate flags is performed to indicate that corresponding E field components should stay equal to zero over simulation time.
- Setting of flags that point at the presence of discontinuity in material at the particular position. They determine the necessity of more complex algorithm applied for E and H field components calculation.

4.2. Simulation loop in space and time unit

The procedure that repeats in every time step consists of the following:

- Initiation of local variables that are used for the particular time step and do not change in space loops.

  - Initiation of auxiliary coefficients for field adaptation according to the medium index (up to 256 different media can be used)
  - Initiation of auxiliary coefficients for CPML calculations depending of the depth of CPML zone along x, y and z direction (0-63)
- Calculation of new H field components in every elementary cell.

Bits 16-31 in ECF structure (previously prepared in simulation initialization stage) are intended for acceleration of the simulation execution. Based on their values, the selection of one of 6 following options for FDTD algorithm realisation is performed:

1. Free space
2. Free space with CPML
3. Simple medium
4. Simple medium with CPML
5. Complex medium
6. Complex medium with CPML

In the case of options 1, 3 and 5, when CPML is not present, equations (1) and (2) are used for update of *x* field components. For calculation of *y* and *z* field components equations equivalent to (1) and (2) are applied. When CPML is present (options 2, 4 and 6) relations (5) and (6) are used for *x* field components calculation and equivalent equations for *y* and *z* field components.

When *Free space* medium is chosen (options 1 and 2) known values of the coefficients (corresponding to parameters ε_0, μ_0 and σ_0 are used in field calculation relations.

Simple medium (options 3 and 4) includes the fact that neighbouring elementary cells have the same properties (the same ε, μ and σ) like the observed cell, so the necessary coefficients, previously calculated, are taken from the corresponding tables.

In *Complex medium* (options 5 and 6) neighbouring elementary cells don't have the same properties like the observed cell, so it is necessary to calculate equivalent parameters ε, μ and σ first, as well as the coefficients necessary for field calculation.

- Calculation of new E field components in every elementary cell.
- In this case there are also 6 options:
  1. Free space
  2. Free space with CPML
  3. Simple medium
  4. Simple medium with CPML
  5. Complex medium
  6. Complex medium with CPML

  These options are equivalent to the ones for H field calculation.
- Introduction of lumped elements through the extension

The introduction of the lumped elements is separated from the standard processing in such a way that relation (2) is not used for calculation of new E field component on the position where the lumped element is placed. This is defined by the corresponding bit in ECF structure. On this position complex relations that account for the existence of the particular lumped element are additionally applied.

5. PERFORMANCE OF THE OPTIMIZED FDTD ALGORITHM

Having in mind that free space in calculations is mostly present in majority of designs and the fact that field components are fastest calculated in that case (since the default coefficients' values are used), its presence significantly increases the speed of the applied algorithm. Therefore, this algorithm gives optimal operating speeds regardless of the complexity of the particular analysed design.

In Table I the comparison between the performance of the conventional (A_1) FDTD algorithm and the optimized (A_2) one is presented for different number of elementary cells in total space (M) and different number of CPML layers (N). The results are expressed in time (ms) needed for execution of one simulation time step Δt. The simulations are carried out under WindowsXP operating system, on AMD Phenom 9750 Quad-Core 2.4GHz processor and with 4GB of RAM. Algorithms A1 and A2 are executing the same simulation problem (patch antenna on 2.4GHz). In all cases it can be noticed that the optimized algorithm operates approximately five times faster to recalculate E and H fields in whole domain in every time step than the conventional approach. This is especially important when the simulation execution is generally demanding (large number of CPML layers or large number of elementary cells in the total space).

Table 1 Performance of the conventional (A_1) and the optimized (A_2) FDTD algorithm for different number of elementary cells in total space (M) and different number of CPML layers (N).

Execution time of Δt [ms]	N=0		N=2		N=4		N=8	
	A1	A2	A1	A2	A1	A2	A1	A2
$M=10^6$	204.08	40.81	220.59	45.35	240	49.54	277.78	57.2
$M=2\cdot10^6$	408.16	81.19	437.95	88.1	468.75	94.49	535.71	109.49
$M=4\cdot10^6$	821.92	159.15	857.14	173.41	923.07	186.91	1034.48	211.27
$M=8\cdot10^6$	1621.32	319.15	1714.28	340.91	1818.18	370.37	2000	422.53

6. FEATURES OF FDTD BASED SIMULATION ENVIRONMENT

Since commercially available software does not offer a possibility of further upgrading and does not include all the effects that can be required to be taken into account, a development of an own simulation environment for EM analysis, based on FDTD method has been started at the Faculty of Electronic Engineering in Nis. The implementation is central processing unit (CPU) based because of the maximal accuracy and the computational domain is truncated by CPML.

6.1. Graphical editor

A graphical editor enables user to graphically enter a desired design of the device (Fig. 2). In the available menu one can choose the particular material and its properties that will be used for certain layers. Also, a list of the supported lumped elements and source excitations is offered. Beside this, a number or functionalities that help user to enter the design are present: magnification, view of the referent layer with desired transparency of the current one, different plot perspectives.

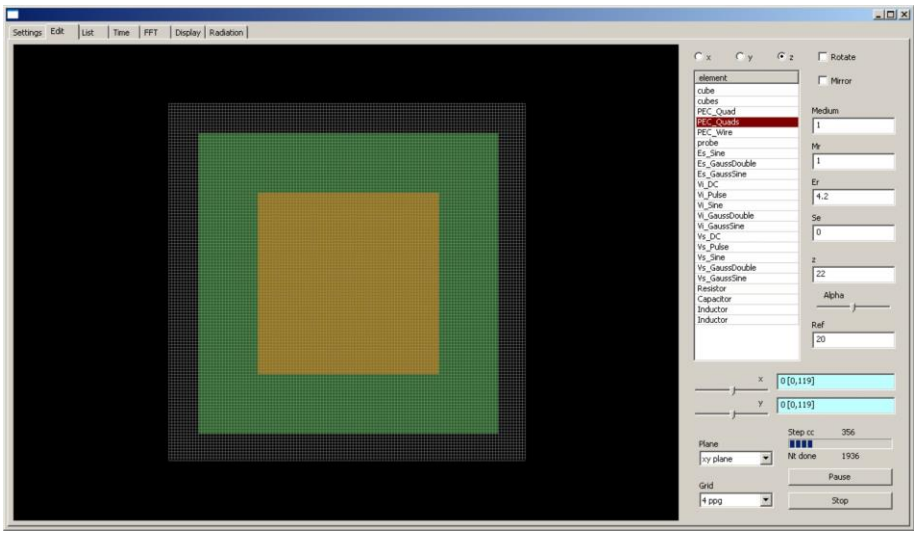

Fig. 2 Graphical editor with input design

6.2. 3D result presentation

Features of the simulation environment include 3D representations of E and H field components. All six field components can be monitored in time from different plot perspectives with or without plot of the device. In Fig.3 and Fig.4 E field components are shown in *xy* plot perspective with and without plot of the patch antenna, respectively. The intensity of the fields is not only indicated by the colour, but also by the embossing. 3D Radiation pattern feature is illustrated for the well-known case of a dipole antenna (Fig.5).

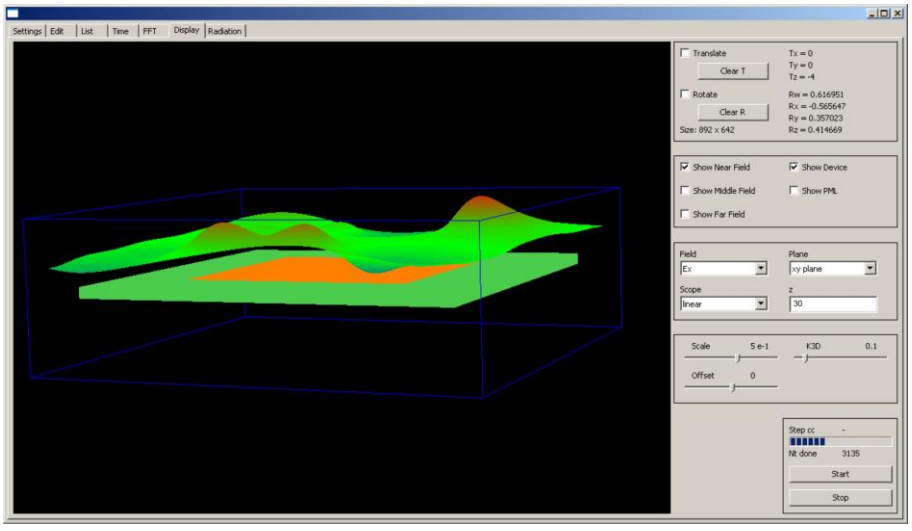

Fig. 3 E_x field component of patch antenna for *xy* plot perspective

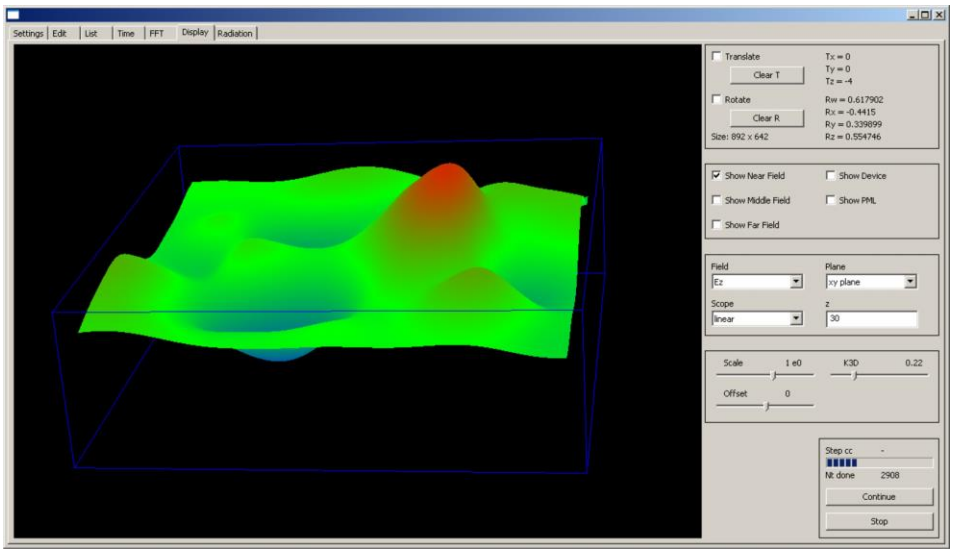

Fig. 4 E_z field component of patch antenna for *xy* plot perspective

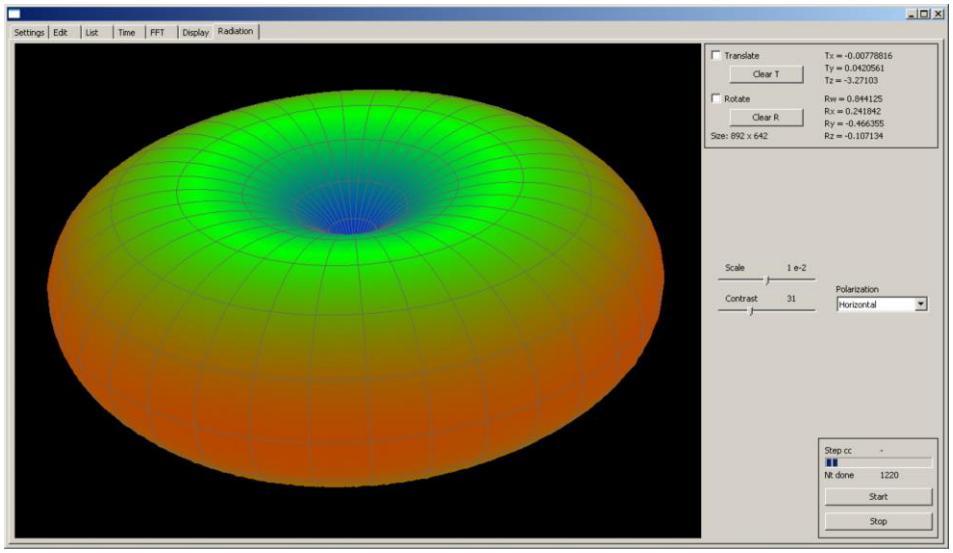

Fig. 5 Radiation pattern for dipole antenna

6.3. Result presentation in time domain

Graphical interface supports simultaneous monitoring of 4 different channels in time domain. It is also possible to monitor a difference between any two signals in analysed design. In Fig.6 direct, reflected and total signal at the position of excitation of rectangular patch antenna are presented.

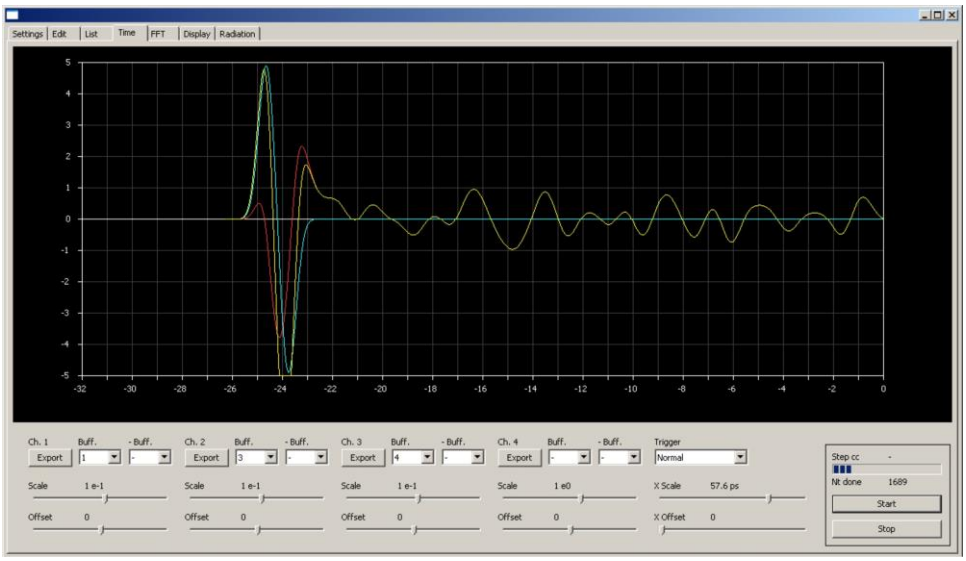

Fig. 6 Direct, reflected and total signal in time domain

6.4. Result presentation in frequency domain

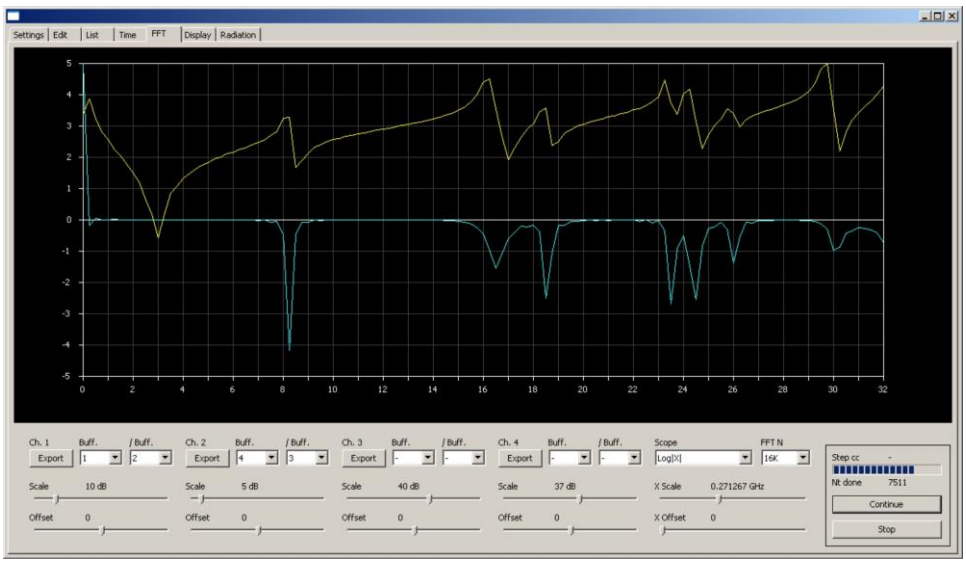

Fig. 7 $|s_{11}|$ and $|z_{11}|$ parameters

Graphical interface supports simultaneous monitoring of 4 different channels in frequency domain. The possibility of setting a ratio of any two signals in frequency domain to be viewed in the channel allows one to monitor a wide range of system parameters. For the purpose of illustration, parameters s_{11} and z_{11} of rectangular patch antenna are displayed in Fig. 7.

7. CONCLUSION

In this paper we present an optimized implementation of FDTD based simulation environment for EM analysis. The optimisation is performed by introducing data structures, multiple employment of the previously calculated elements and intensive use of indexing tables. This reduces the execution speed approximately five times in comparison with the conventional FDTD implementation. Also, the complex graphical interface with many features and possibility of simple extension and further improvement is developed.

A further development of this simulation environment should enable the analysis of beamforming, adaptive antennas, active antennas, diversity systems and RF system with antennas integrated therein.

REFERENCES

[1] W. Yu, X. Yang, Y. Liu, R. Mittra and A. Muto, *Advanced FDTD Methods: Parallelization, Acceleration, and Engineering Applications*, Norwood, USA: Artech house, 2011.

[2] S. Humphries, *Finite-element Methods for Electromagnetics*, Boca Raton, USA: CRC Press, 1997.

[3] M. N. O. Sadiku, *Numerical Techniques in Electromagnetics*, 2nd ed., Boca Raton, USA: CRC Press, 2001.

[4] K. S. Yee, "Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media", *IEEE Trans. Antennas and Propagation*, vol. AP-14, pp. 302–307, May 1966.

[5] A. Taflove and S. C. Hagness, *Computational electrodynamics: The finite-difference time-domain method*, 3rd ed., Norwood, USA: Artech House, 2005.

[6] S. Sandeep, "Broadband analysis of microstrip patch antenna using 3D FDTD – UPML", University of Colorado at Boulder, USA, ECEN 5134 - Term paper, 2008.

[7] J. C. Strikwerda, *Finite difference schemes and partial differential equations*, Monterey, USA: Wadsworth & Brooks/Cole Matematics Series, 1989.

[8] J.-P. Berenger: "A perfectly matched layer for the absorption of electromagnetic waves", *Journal of Computational Physics*, vol. 114, pp. 195–200, 1994.

[9] S.D. Gedney, "The Perfectly Matched Layer Absorbing Medium", in "*Advances in computational electrodynamics: The finite difference time domain*", A. Taflove, 1st edition, Artech House, 1998, pp. 263–340, Boston.

[10] J. Alan Roden and Stephen D. Gedney, "Convolution PML (CPML): An efficient FDTD implementation of the CFS–PML for arbitrary media", *Microwave and Optical Technology Letters*, vol. 27, no. 5, pp. 334–339, 2000.

[11] WC. Chew and WH. Weedon, "A 3-D perfectly matched medium from modified Maxwell's equation with stretched coordinates", *Microw. Opt. Technol. Lett.*, vol. 7, pp. 599–604, 1994.

[12] M. Kuzuoglu and R. Mittra, "Frequency dependence of the constitutive parameters of causal perfectly matched anisotropic absorbers", *IEEE Microw. Guid. Wave Lett.*, vol. 6, pp. 447–449, 1996.

[13] J.P. Berenger, "Numerical reflection from FDTD-PMLs: A comparison of the split PML with the unsplit and CFS PMLs", *IEEE Trans. Antennas Propag*., vol. 50, pp. 258–265, 2002.

[14] D. Correia and JM. Jin, "Performance of regular PML, CFS-PML, and second-order PML for waveguide problems", *Microw. Opt. Technol. Lett*., vol. 48, pp. 2121 – 2126, 2006.