An Alternating Implicit Block Overlapped FDTD (AIBO-FDTD) Method and Its Parallel Implementation

Pornanong Pongpaibool[1], Atsushi Kamo[2], Takayuki Watanabe[3], and Hideki Asai[4]
[1] Department of Electronics and Information Engineering, Graduate School of Technology, Tokyo University of Agriculture and Technology, 2-24-16 Nakacho, Koganei-shi, Tokyo 184-8588, Japan
[2] Sony LSI Design Inc., TNC Broadcasting Hall 21F, 2-3-2 Momochihama, Sawara-ku, Fukuoka 814-0001, Japan
[3] School of Administration and Informatics, University of Shizuoka, 52-1 Yada, Shizuoka 422-8562, Japan
[4] Department of System Engineering, Faculty of Engineering, Shizuoka University, 3-5-1 Johoku, Hamamatsu 432-8561, Japan
Tel. +81-53-478-1237, Fax.: +81-53-478-1269
e-mail : pornong@cc.tuat.ac.jp, hideasai@sys.eng.shizuoka.ac.jp

Abstract: In this paper, a new algorithm for two-dimensional (2-D) finite-difference time-domain (FDTD) method is presented. By this new method, the maximum time step size can be increased over the Courant-Friedrich-Levy (CFL) condition restraint. This new algorithm is adapted from an Alternating-Direction Implicit FDTD (ADI-FDTD) method. However, unlike the ADI-FDTD algorithm, the alternation is performed with respect to the blocks of fields rather than with respect to each respective coordinate direction. Moreover, this method can be efficiently simulated with parallel computation, and it is more efficient than the conventional FDTD method in terms of CPU time. Numerical formulations are shown and simulation results are presented to demonstrate the effectiveness and efficiency of our proposed method.

1. Introduction

The finite-difference time-domain (FDTD) method [1]-[3] has been widely used for solving various types of electromagnetic problems, such as anisotropic and nonlinear problems. The FDTD method provides accurate predictions of field behaviors. Moreover, since it is a time-domain method, one single run of simulation can provide information over a large bandwidth when the excitation is chosen to be of large bandwidth. We have also proposed novel methods [4]-[5] to efficiently simulate high-speed interconnects with the model order reduction techniques.

As the traditional FDTD method is based on an explicit finite-difference algorithm, the Courant-Friedrich-Levy (CFL) condition [2] must be satisfied. A maximum time-step size is limited by the minimum cell size in a computational domain. Therefore, CPU time significantly increases if an object of analysis has a fine-scale dimension compared to the wavelength.

To avoid this limitation, the Alternating-Direction Implicit FDTD (ADI-FDTD) method [6] was proposed. The maximum time-step size then no longer depends on the CFL condition, and the CPU time decreases. However, because the alternation of the ADI-FDTD algorithm is performed with respect to each coordinate direction, the partition for parallel computation is inefficient.

In this paper, we introduce an Alternating Implicit Block Overlapped FDTD (AIBO-FDTD), which can increase the maximum time-step size of the calculation time-domain. With this algorithm, the electromagnetic problems can be efficiently partitioned and simulated using parallel computation.

2. Numerical Formulation for a 2-D TE Wave

The AIBO-FDTD method is adapted from the ADI-FDTD method in the following manners. The implicitly computed values no longer depend on direction, but depend on blocks created, as shown in Figure 1. The simulation is performed in 2 steps for one discrete time step. In each step, all the magnetic fields and the electric fields inside the implicit blocks are computed with the implicit algorithm. The rest of the electric fields are computed with the explicit algorithm. For the next step, the implicit blocks are moved to overlap the implicit blocks of the previous step, and the same computations are performed. The specified implicit blocks in this method can be efficiently computed in parallel computation.

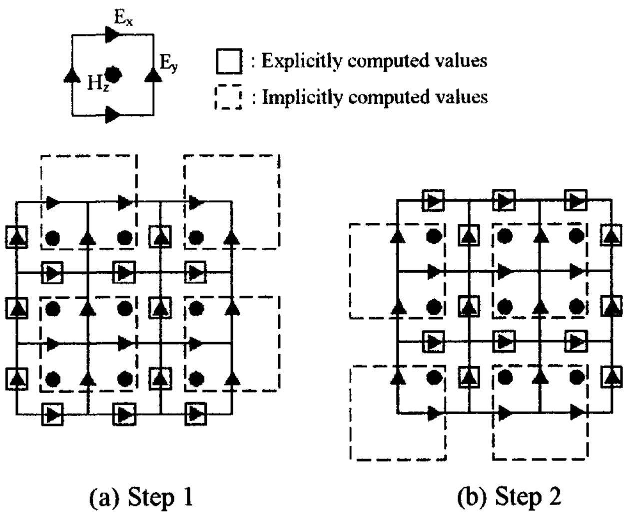


Figure 1. Calculation diagram of the AIBO-FDTD method

The numerical formulation of the AIBO-FDTD for a 2-D TE wave can be presented as in (1)-(10). We assume that the medium in which the wave propagates is a vacuum, and all cells in the computational domain have the same size. The calculation for one discrete time step is performed using 2 steps, the first-half time step (i.e., at the (n+½)*th* time step) and the second-half time step (i.e., at the (n+1)*th* time step).

Step 1

Explicit algorithm:

$$E_x^{n+\frac{1}{2}}(i+\tfrac{1}{2},j) = E_x^n(i+\tfrac{1}{2},j)+\frac{\Delta t}{2\varepsilon\Delta y} \cdot\left\{H_z^n(i+\tfrac{1}{2},j+\tfrac{1}{2})-H_z^n(i+\tfrac{1}{2},j-\tfrac{1}{2})\right\} \quad (1)$$

$$E_y^{n+\frac{1}{2}}(i,j+\tfrac{1}{2}) = E_y^n(i,j+\tfrac{1}{2})-\frac{\Delta t}{2\varepsilon\Delta x} \cdot\left\{H_z^n(i+\tfrac{1}{2},j+\tfrac{1}{2})-H_z^n(i-\tfrac{1}{2},j+\tfrac{1}{2})\right\} \quad (2)$$

Implicit algorithm:

$$E_x^{n+\frac{1}{2}}(i+\tfrac{1}{2},j+1) = E_x^n(i+\tfrac{1}{2},j+1)+\frac{\Delta t}{2\varepsilon\Delta y} \cdot\left\{H_z^{n+\frac{1}{2}}(i+\tfrac{1}{2},j+\tfrac{3}{2})-H_z^{n+\frac{1}{2}}(i+\tfrac{1}{2},j+\tfrac{1}{2})\right\} \quad (3)$$

$$E_y^{n+\frac{1}{2}}(i+1,j+\tfrac{1}{2}) = E_y^n(i+1,j+\tfrac{1}{2})-\frac{\Delta t}{2\varepsilon\Delta x} \cdot\left\{H_z^{n+\frac{1}{2}}(i+\tfrac{3}{2},j+\tfrac{1}{2})-H_z^{n+\frac{1}{2}}(i+\tfrac{1}{2},j+\tfrac{1}{2})\right\} \quad (4)$$

$$H_z^{n+\frac{1}{2}}(i+\tfrac{1}{2},j+\tfrac{1}{2}) = H_z^n(i+\tfrac{1}{2},j+\tfrac{1}{2})+\frac{\Delta t}{2\mu\Delta y} \cdot\left\{E_x^{n+\frac{1}{2}}(i+\tfrac{1}{2},j+1)-E_x^n(i+\tfrac{1}{2},j)\right\}-\frac{\Delta t}{2\mu\Delta x} \cdot\left\{E_y^{n+\frac{1}{2}}(i+1,j+\tfrac{1}{2})-E_y^n(i,j+\tfrac{1}{2})\right\} \quad (5)$$

Step 2

Explicit algorithm:

$$E_x^{n+1}(i+\tfrac{3}{2},j+1) = E_x^{n+\frac{1}{2}}(i+\tfrac{3}{2},j+1)+\frac{\Delta t}{2\varepsilon\Delta y} \cdot\left\{H_z^{n+\frac{1}{2}}(i+\tfrac{3}{2},j+\tfrac{3}{2})-H_z^{n+\frac{1}{2}}(i+\tfrac{3}{2},j+\tfrac{1}{2})\right\} \quad (6)$$

$$E_y^{n+1}(i+1,j+\tfrac{3}{2}) = E_y^{n+\frac{1}{2}}(i+1,j+\tfrac{3}{2})-\frac{\Delta t}{2\varepsilon\Delta x} \cdot\left\{H_z^{n+\frac{1}{2}}(i+\tfrac{3}{2},j+\tfrac{3}{2})-H_z^{n+\frac{1}{2}}(i+\tfrac{1}{2},j+\tfrac{3}{2})\right\} \quad (7)$$

Implicit algorithm:

$$E_x^{n+1}(i+\tfrac{3}{2},j+2) = E_x^{n+\frac{1}{2}}(i+\tfrac{3}{2},j+2)+\frac{\Delta t}{2\varepsilon\Delta y} \cdot\left\{H_z^{n+1}(i+\tfrac{3}{2},j+\tfrac{5}{2})-H_z^{n+1}(i+\tfrac{3}{2},j+\tfrac{3}{2})\right\} \quad (8)$$

$$E_y^{n+1}(i+2,j+\tfrac{3}{2}) = E_y^{n+\frac{1}{2}}(i+2,j+\tfrac{3}{2})-\frac{\Delta t}{2\varepsilon\Delta x} \cdot\left\{H_z^{n+1}(i+\tfrac{5}{2},j+\tfrac{3}{2})-H_z^{n+1}(i+\tfrac{3}{2},j+\tfrac{3}{2})\right\} \quad (9)$$

$$H_z^{n+1}(i+\tfrac{3}{2},j+\tfrac{3}{2}) = H_z^{n+\frac{1}{2}}(i+\tfrac{3}{2},j+\tfrac{3}{2})+\frac{\Delta t}{2\mu\Delta y} \cdot\left\{E_x^{n+1}(i+\tfrac{3}{2},j+2)-E_x^{n+\frac{1}{2}}(i+\tfrac{3}{2},j+1)\right\}-\frac{\Delta t}{2\mu\Delta x} \cdot\left\{E_y^{n+1}(i+2,j+\tfrac{3}{2})-E_y^{n+\frac{1}{2}}(i+1,j+\tfrac{3}{2})\right\} \quad (10)$$

Step 1 and step 2 can be further simplified for efficient computation. In step 1, by substituting the expression for $H_z^{n+1/2}(i+1/2,j+1/2)$ and $H_z^{n+1/2}(i+1/2,j+3/2)$ to (3), one can obtain

$$\left\{\left(\frac{2\sqrt{\varepsilon\mu}\Delta y}{\Delta t}\right)^2+2\right\}E_x^{n+\frac{1}{2}}(i+\tfrac{1}{2},j+1)+\left(\frac{\Delta y}{\Delta x}\right) \cdot\left\{E_y^{n+\frac{1}{2}}(i+1,j+\tfrac{3}{2})-E_y^{n+\frac{1}{2}}(i+1,j+\tfrac{1}{2})\right\}$$
$$=\left(\frac{2\sqrt{\varepsilon\mu}\Delta y}{\Delta t}\right)^2 E_x^n(i+\tfrac{1}{2},j+1)+\left(\frac{2\mu\Delta y}{\Delta t}\right) \cdot\left\{H_z^n(i+\tfrac{1}{2},j+\tfrac{3}{2})-H_z^n(i+\tfrac{1}{2},j+\tfrac{1}{2})\right\} +\left\{E_x^n(i+\tfrac{1}{2},j+2)+E_x^n(i+\tfrac{1}{2},j)\right\} +\left(\frac{\Delta y}{\Delta x}\right)\left\{E_y^n(i,j+\tfrac{3}{2})-E_y^n(i,j+\tfrac{1}{2})\right\} \quad (11)$$

All the field components on the RHS are known values at the previous time steps, while the components on the LHS are unknown values at the current time steps. The same procedure is applied to (4) and the implicit equations of step 2. Similar equations can be obtained for other components.

From (11). there is no relationship between each implicit block. From this knowledge, we can compute this method efficiently in parallel computation. The parallel computation provides a great advantage in reducing the CPU time because the implicit block of the AIBO-FDTD method is small and independent. Furthermore, the number of processors used in parallel computing does not effect the maximum time-step size and the results.

3. Numerical Results

In this section, a few numerical examples are shown to verify the performance of the proposed AIBO-FDTD method.

Example 1: The 2-D free-space model used for this simulation is shown in Figure 2. The computational domain is 321.4×321.4 cm^2, and Mur's first-order absorbing boundary condition [7] is set on the outer surface.

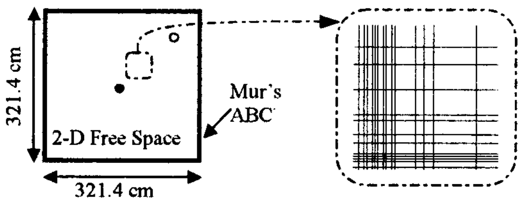


Figure 2. 2-D free-space model (• : Excitation point. ○ : Observation point)

The computational domain contains 120×120 cells. The cell size is set as follows:

$$\Delta(k)_{(mm)} = \begin{cases} 50.0 & (k = 1\text{-}26) \\ 1.2\,\Delta(k+1) & (k = 27\text{-}39) \\ 4.0 & (k = 40\text{-}81) \\ 1.2\,\Delta(k-1) & (k = 82\text{-}94) \\ 50.0 & (k = 95\text{-}120) \end{cases}$$

where $\Delta(k) = \Delta x(i)$, and $\Delta(k) = \Delta y(j)$ for x- and y-direction, respectively. The excitation is applied to the H_z-component at the central cell of the computational domain ($i = j = 60$), and the excited waveform is set as follows:

$$H_z = H_z + \sin^2(\pi t / T) \quad , \; T = 9.4 \text{ ns} \tag{12}$$

The time step size is 28.2 ps for the AIBO-FDTD method and 9.4 ps for the conventional FDTD method. The latter is decided by the limitation of the 2-D CFL condition:

$$\Delta t \le \frac{1}{c\sqrt{\left(\frac{1}{\Delta x}\right)^2 + \left(\frac{1}{\Delta y}\right)^2}} \tag{13}$$

where Δx and Δy indicate the minimum cell size, both of which are 4.0 mm for this model. The total simulation time is 42.3 ns. All simulations were run on the network of workstations listed in Table 1. The simulation with N processors were run on processor #1 to #N.

Table 1. The network of workstations

Processor	Type	CPU Clock	Memory
# 1	Enterprise 450	480 MHz	512 MB
# 2	Ultra 10	440 MHz	256 MB
# 3	Ultra 5	270 MHz	128 MB
# 4	Enterprise 4000	250 MHz	64 MB

The H_z-field at observation point ($i = j = 111$) is shown in Figure 3. The simulation results of the AIBO-FDTD and conventional FDTD method are in good agreement. The results of the AIBO-FDTD with parallel computing are also exactly the same as that of the non-parallel. The CPU time is shown in Table 2. The CPU time of the AIBO-FDTD method can be further decreased when the number of processors used for the simulation is increased.

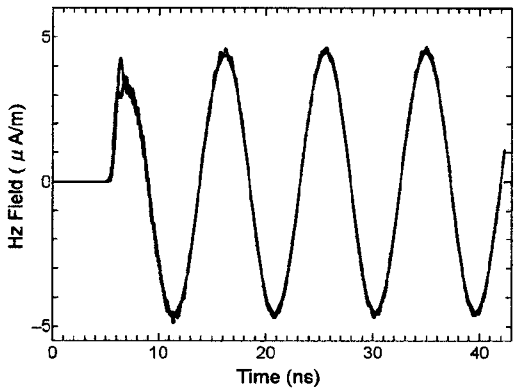


Figure 3. H_z field in time domain.

Table 2. Comparison of CPU time of example 1.

Number of Processors	CPU Time (s)		
	Conventional FDTD	AIBO-FDTD	Speed-up Factor
1	79.34	71.09	1.12
2	52.80	40.34	1.31
3	41.70	29.57	1.41
4	38.86	25.91	1.50

Example 2: The 2-D free-space model with a circle perfect electric conductor (PEC) centered within the region used in this simulation is shown in Figure 4. Mur's first-order absorbing boundary condition is set on the outer surface. The computational domain contains 192×192 cells. The cell size for all cells is set to 1.0 cm × 1.0 cm. The radius of a circle PEC is 20.0 cm.

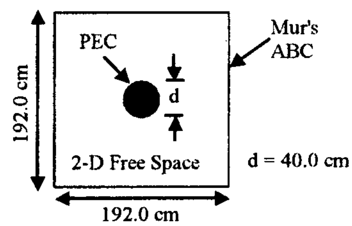


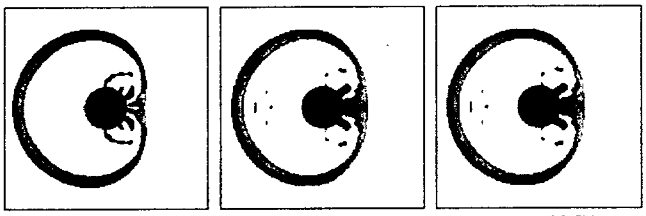
Figure 4. 2-D free-space model with a circle PEC

A Gaussian-pulse plane wave is generated at the left terminal as the incident wave. This wave propagates in $+x$-direction with full amplitude and no distortion. The waveform of this plane wave is as follows:

$$H_z = \exp\left[-\frac{(t-t_0)^2}{T^2}\right] \quad , t_0 = 100\text{ ps}, T = 125\text{ ps} \quad (14)$$

The time-step size is 47.0 ps for the AIBO-FDTD method and 23.5 ps for the conventional FDTD method. The latter is decided by the CFL condition. The total simulation time is 9.4 ns. All the simulations were run on the network of workstations listed in Table 1.

The scattered H_z-field distribution within the grid at t=4.7 ns are shown in Figure 5. In this example, the higher the time-step size of the AIBO-FDTD method is set, the slower the dispersion of the scattered wave is occurred. The comparison of CPU time between the conventional FDTD and the AIBO-FDTD method is shown in Table 3. The CPU time can be further decreased when the number of processors used for the simulation is increased.



Conventional FDTD | AIBO-FDTD (Single processor) | AIBO-FDTD (Parallel processors)

Fig. 5. Scattered H_z-field at t = 4.7 ns.

Table 3. Comparison of CPU time of example 2.

Number of Processors	CPU Time (s)		
	Conventional FDTD	AIBO-FDTD	Speed-up Factor
1	113.44	93.39	1.21
2	68.62	54.64	1.26
3	52.93	40.10	1.32
4	50.90	37.43	1.36

4. Conclusion

We introduced the AIBO-FDTD method for solving the 2-dimensional electromagnetic problems. The maximum time-step size can be increased to 2-3 times over the limitation of the CFL condition. Numerical results show that this new method is efficient especially with parallel computation. The CPU time can be further decreased through the increase of the number of the parallel processors. The results of this new method also agree very well with the conventional FDTD method.

References

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

[2] A. Taflove, *Computational Electrodynamics: The Finite-Difference Time-Domain Method.* Norwood, MA: Artech House, 1996.

[3] T. Uno, *FDTD method for electromagnetic field and antenna analysis.* Tokyo: Corona Publ. Co., 1998.

[4] T. Watanabe and H. Asai, "Synthesis of Time-Domain Models for Interconnects Having 3-D structure Based on FDTD Method," *IEEE Trans. on Circuits and Syst.-II:Analog and Digital Signal Processing,* vol.47, no.4, pp. 302-305, April 2000.

[5] M. Suzuki, H. Miyashita, A. Kamo, T. Watanabe and H. Asai, "A Synthesis Technique of Time-Domain Interconnect Models by MIMO Type of Selective Orthogonal Least Square Method," *IEEE Trans. Microwave Theory Tech.*, vol.49, no.10, pp. 1708-1714, Oct. 2001.

[6] T. Namiki, "A New FDTD Algorithm Based on Alternating-Direction Implicit Method," *IEEE Trans. Microwave Theory Tech.*, vol.47, pp. 2003-2007, Oct. 1999.

[7] G. Mur, "Absorbing boundary conditions for the finite-difference approximation of the time-domain electro-magnetic field equations," *IEEE Trans. Electromag. Compat.*, vol. EMC-23, pp. 377-382, Nov. 1981.