Acceleration of Large Electromagnetic Simulation Including Non-orthogonally Aligned Thin Structures by Using Multi-GPU HIE/C-FDTD Method

Yuta Inoue and Hideki Asai
Research Institute of Electronics
Shizuoka University
Hamamatsu, Japan
Email: 1 inoue@tzasai7.sys.eng.shizuoka.ac.jp, 2 asai.hideki@shizuoka.ac.jp

Abstract—In this paper, multi-graphics processing units (GPU) hybrid implicit-explicit/conformal finite-difference time-domain (HIE/C-FDTD) method is proposed for the large efficient electromagnetic simulations. The HIE/C-FDTD method is constructed by combination of the HIE-FDTD method and the C-FDTD method. The HIE/C-FDTD method can adopt a larger time step size than that for the conventional FDTD method and can use the large cells. In addition, the HIE/C-FDTD method is suitable for parallel computing such as GPU Computing. First, the HIE/C-FDTD method is reviewed briefly. Next, the proposed method is described. Finally, the efficiency of the proposed method is verified by some numerical results.

I. INTRODUCTION

The finite-difference time-domain (FDTD) method [1] is a powerful numerical simulation technique for solving Maxwell’s equations. The FDTD method is based on the explicit leapfrog scheme, the Courant-Friedrichs-Lewy (CFL) condition must be satisfied when this method is used. Therefore, the maximum time step size is limited by minimum cell size in a computational domain. In the FDTD method, the objects are discretized by Yee’s cells, which are the orthogonal cells. Therefore, it is difficult to generate cells exactly for the non-orthogonally aligned thin structure. Generally, the non-orthogonally aligned structure is discretized with the staircase approximation, so that the cell sizes of the surface for the non-orthogonally aligned structure becomes fine. As a result, the time step size becomes extremely small and it can make the FDTD simulation a huge time consuming task. In addition, the staircase approximation leads frequently inaccurate simulation results. Hence, the efficient and accurate electromagnetic simulation technique is strongly demanded.

In order to alleviate the CFL condition and overcome inaccuracy, the hybrid implicit-explicit/conformal finite-difference time-domain (HIE/C-FDTD) method has been proposed for the efficient electromagnetic simulations [2]. The HIE/C-FDTD method is constructed by combination of the HIE-FDTD method [3], [4] and the C-FDTD method [5]–[8]. The C-FDTD method employs the subcell scheme, which is illustrated in Fig. 1, instead of the Yee’s cells scheme for discretization of the surface of the non-orthogonally aligned structure. The C-FDTD method can reduce the number of cells without degrading accuracy. On the other hand, the HIE-FDTD method is one of the weakly conditionally stable algorithms which is based on a hybrid implicit and explicit technique. The HIE-FDTD method can use the larger time step size than that for the FDTD method. Therefore, the HIE-FDTD method can be faster than the conventional FDTD method with thin structures. Consequently, the HIE/C-FDTD method is much faster than the conventional ones. However, in order to analyze the large scale problem, the HIE/C-FDTD method is required to be further accelerated. The updating procedure of the HIE/C-FDTD method is same as the HIE-FDTD method. Thus, the HIE/C-FDTD method can employ the parallel computing technique as well as the HIE-FDTD method [9]–[11].

In this paper, the multi graphics processing units (GPU) HIE/C-FDTD method is proposed and implemented for the solution of large-scale electromagnetic problems including the non-orthogonally aligned structures. First, the HIE/C-FDTD method is shown briefly. Next, the Multi-GPU HIE/C-FDTD method is described. Finally, the efficiency of the proposed method is evaluated by several simulations.

II. HIE/C-FDTD METHOD [2]

Printed circuit boards often have non-orthogonally aligned thin structure even in the cases of interconnects and power and ground layers. Here, it is assumed that the fine dimension
should be along the $z$ direction and the non-orthogonally aligned perfectly electric conductor (PEC) is allocated in the $x, y$ directions. The updating formulas and procedures are described briefly. Here, the updating formula of the HIE/C-FDTD method is given in a matrix-vector form:

$$K_1 u^{n+1} = K_2 u^n$$  (1)

where $n$ is a time step index, and $K_1$, $K_2$, and $u$ are defined at the top of next page, $\Delta t$ is the time step size, $L_\alpha, S_\alpha (\alpha = x, y, z)$ are the diagonal submatrices which contain $l_\alpha$ and $S_\alpha$ in $x, y, z$ directions, $l_\alpha$ and $S_\alpha$ are the length of the PEC-free edge and the PEC-free area, $\hat{X}, \hat{Y}, \hat{Z}$ are the submatrices containing the grid spacings in $x, y, z$ directions. $E_\alpha, H_\alpha (\alpha = x, y, z)$ are electric and magnetic components, $\mu, \epsilon$ and $\sigma$ are permeability, permittivity and conductivity, respectively.

The updating procedure of the HIE/C-FDTD method is same as the HIE-FDTD method. $E_z$ and $H_z$ are explicitly updated by simple substitutions. After that, $E_x$ and $E_y$ are updated by solving the equation with the tridiagonal matrix by using direct matrix solver. $H_x$ and $H_y$ are updated without direct matrix solver.

### III. Multi-GPU HIE/C-FDTD Method

In this section, domain decomposition techniques and data communications are described for the proposed method. Here, CUDA is employed for GPU computing. In CUDA, the functions are called kernels. The kernel is executed on a GPU and that is invoked from the CPU with a grid. The grid is composed of arbitrary number of blocks. The block is corresponding to a streaming multiprocessor in the GPU, and is composed of arbitrary number of threads. The thread is the smallest element of the process. It is controlled by the CUDA core.

#### A. Domain decomposition

The proposed method is employed to 2 dimensional domain decomposition method, which is applied to $x, y$ direction, to divide the original spatial domain into the several subdomains. Note that neighbouring subdomains overlap at the boundary cells, which are employed for data communications between the neighboring subdomains. In addition, each subdomain is added wasted cells so that the total number of cells in $x-y$ plane is equal to the multiple of 64. The wasted cells are used for the efficient memory access. Actually, the electromagnetic components at the wasted cells are not updated.

Furthermore, each subdomain is partitioned to arbitrary number of blocks in the proposed method. In the HIE/C-FDTD method, $E_z, H_x, H_y$, and $H_z$ are obtained by simple substitutions in the updating procedure. In the updating procedure of those components, subdomains can be divided arbitrary. On the other hand, $E_x$ and $E_y$ are updated by solving the simultaneous linear equations. In this paper, the LU factorization method is used for solving the simultaneous linear equations, so that the domain decomposition method is not applicable to the $z$ direction. In the parallel direct solver, the additional operations and the data communications emerge as the overheads. Generally, in order to achieve the performance of the parallel computing, sequential processing, data communication, latency time of the data communication function, and synchronization, must be minimized in a parallel algorithm. For this reason, the domain decomposition technique is assigned to $x$ and $y$ directions in the updating procedures of $E_z$ and $E_y$. Fig.2a illustrates an example that the 3 dimensional domain decomposition is applied to $E_z, H_x, H_y$, and $H_z$. Fig.2b shows an example that the 2 dimensional domain decomposition is applied to $E_x$ and $E_y$. Here, $NX, NY, and NZ$ are the numbers of cells in $x, y, z$ directions, respectively. In the $E_z, H_x, H_y$, and $H_z$ updating procedures, a thread is assigned to a variable. In the $E_x$ and $E_y$ updating procedures, each thread is allocated to $NZ$ variables. Each block is composed of the (64, 1, 1) threads in both cases.

#### B. Overlapping computation and data communication

In general, the overhead must be minimized in the parallel computing. In the proposed method, the computation and the data communication overlaps for better performance of the parallel computing. Fig.3 is flowchart of the proposed method. Here, the data communication is performed by the non-blocking data communication function. The non-blocking data communication function returns before completion of data communication, so that the electromagnetic components can be updated during the data communications. From Fig.3,
the boundary part of magnetic components are communicated between the neighboring subdomains and the updating procedure of the electromagnetic components are divided into two parts. The computation of the electromagnetic components at boundary part are regarded as one of the overheads in the proposed method. Nevertheless, the network communication speed is slower than the computation speed. The computational costs of boundary part are vanishingly small in the updating procedures.

In order to verify the validity of the original HIE/C-FDTD method and the proposed method, several examples were simulated. In all of the simulations, Intel Xeon E5-2650 2GHz was used as CPU and Tesla C2075 was used as GPU. Fig.4 shows a computational domain including an example PCB. The example PCB is composed of the four angled transmission lines, and both ends of each line are connected to the ground through the resistance (50Ω). The input voltage source is a trapezoidal pulse of which an initial value is 0 V, a pulse

\[
K_1 = \begin{bmatrix}
\left(\frac{1}{\Delta t} + \frac{\sigma}{2\pi}\right) I - \frac{\Delta t}{\mu} Z S_{y}^{-1} L_{x} & 0 & \frac{\Delta t}{\mu} Z S_{x}^{-1} L_{x} & 0 & 0 & \frac{1}{2} Y \\
0 & \left(\frac{1}{\Delta t} + \frac{\sigma}{2\pi}\right) I - \frac{\Delta t}{\mu} Z S_{x}^{-1} L_{y} & 0 & \frac{\Delta t}{\mu} Z S_{y}^{-1} L_{y} & 0 & 0 & \frac{1}{2} X \\
0 & 0 & \frac{1}{\mu} S_{y}^{-1} L_{x} & 0 & 0 & 0 & \frac{1}{2 Y} \\
0 & 0 & 0 & \frac{1}{\mu} S_{x}^{-1} L_{y} & 0 & 0 & \frac{1}{2 X} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix},
\]

\[
K_2 = \begin{bmatrix}
\left(\frac{1}{\Delta t} + \frac{\sigma}{2\pi}\right) I & 0 & 0 & 0 & 0 & \frac{1}{2} Z & 0 \\
0 & \left(\frac{1}{\Delta t} + \frac{\sigma}{2\pi}\right) I & 0 & 0 & 0 & \frac{1}{2} Y & 0 \\
0 & 0 & \frac{1}{\mu} S_{x}^{-1} L_{x} & 0 & 0 & 0 & \frac{1}{2 X} \\
0 & 0 & 0 & \frac{1}{\mu} S_{y}^{-1} L_{y} & 0 & 0 & \frac{1}{2 Y} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix},
\]

Fig. 4. Computational domain including an example PCB.

![Flowchart of the proposed method](image)

**Fig. 3.** The flowchart of the proposed method.

**Fig. 5.** The waveform results of near end of line 2.
value is 3.3 V, a delay time is 0 sec, a rise and a fall times are 0.5 nsec, and a pulse wide is 2 nsec. The source is connected at the near end of line 1. First, we verified the efficiency and accuracy of the HIE/C-FDTD method. In this example, we use two kinds of sets of cell sizes. One of the sets constructs the fine grid (320 × 320 × 50 cells) which can accurately represent the angled pattern lengthened along the horizontal (x-y) plane. The cell sizes of the fine grid are \( \Delta x = \Delta y = 0.2(\text{mm}) \) and

\[
\Delta z = \begin{align*}
1(\text{mm}) & (k = 1 - 9) \\
0.74(\text{mm}) & (k = 10) \\
0.64 - 0.01(\text{mm}) & (k = 11 - 17) \Delta z(k) = \Delta z(k-1)/2 \\
0.02 - 0.16(\text{mm}) & (k = 18 - 21) \Delta z(k) = \Delta z(k-1)/2 \\
0.18(\text{mm}) & (k = 22) \\
0.16 - 0.01(\text{mm}) & (k = 23 - 27) \Delta z(k) = \Delta z(k-1)/2 \\
0.02 - 0.64(\text{mm}) & (k = 28 - 33) \Delta z(k) = \Delta z(k-1)/2 \\
0.74(\text{mm}) & (k = 34) \\
1(\text{mm}) & (k = 35 - 50)
\end{align*}
\]

The other set constructs the coarse grid (80 × 80 × 50) of which spatial resolution is insufficient to represent the contour of the pattern. We adopt the fine grid to the FDTD method and the HIE-FDTD method, of which the simulation results are referred as exact solution. On the other hand, the coarse grid is used for the HIE-FDTD method and the HIE/C-FDTD method. Those cell sizes are used for verification of the accuracy of the waveform results. Fig.5 illustrates the waveform results and Table I shows the computational times of those methods. The HIE/C-FDTD method does not lead inaccuracies and is about 700 times faster than the FDTD method in the case of using the coarse grids. In addition, the proposed method can get the same waveform of the HIE/C-FDTD method.

Next, we evaluated comparison of the proposed method with the HIE/C-FDTD method. Here, the example computational domain extended to about \(1.28 \times 10^8(1600 \times 1600 \times 50)\) cells. The cell sizes of this problem is same as the case of the coarse grid. This problem requires about 33GB for the simulation with double precision floating point. Here, the proposed method used 8 GPUs. From Table II, the proposed method is about 48.9 times faster than the HIE/C-FDTD method. As a result, it has been confirmed that the proposed method is much faster than the FDTD method.

The HIE/C-FDTD method does not lead inaccuracies and is referred as exact solution. On the other hand, the coarse grid is the HIE-FDTD method, of which the simulation results are the pattern. We adopt the fine grid to the FDTD method and the HIE-FDTD method is much faster than the FDTD method. As a result, it has been confirmed that the proposed method is about 48.9 times faster than the HIE/C-FDTD method, that is to say, the proposed method is much faster than the conventional FDTD method.

### V. Conclusion

This paper described the multi-GPU HIE/C-FDTD method for the solution of the large-scale electromagnetic problems including non-orthogonally aligned structure. Numerical result by the proposed method has been compared to the results of the HIE/C-FDTD method. From the numerical results, it has been confirmed that the proposed method is about 48.9 times faster than the HIE/C-FDTD method, that is to say, the proposed method is much faster than the conventional FDTD method.

### References


### Table I

<table>
<thead>
<tr>
<th>Number of Cells</th>
<th>Execution time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fine grid (320 × 320 × 50 cells)</td>
<td>78430.00</td>
</tr>
<tr>
<td>Coarse grid (80 × 80 × 50 cells)</td>
<td>Not available</td>
</tr>
</tbody>
</table>

### Table II

<table>
<thead>
<tr>
<th>Method</th>
<th>Execution time</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIE/C-FDTD method</td>
<td>57188.34</td>
</tr>
<tr>
<td>Proposed method</td>
<td>1167.07</td>
</tr>
</tbody>
</table>