Electromagnetic Field Scattering of a High Speed Moving Source and Its Application

Shafrida Sahrani[1], Nor Shafrillah Isa[1], Kismet Anak Hong Ping[1], Tatsuya Akata[2] and Michiko Kuroda[2]
[1]Applied Electromagnetics Research Group, Faculty of Engineering, Universiti Malaysia Sarawak.
[2]Tokyo University of Technology, Japan.
sshafrida@unimas.my

***Abstract*—This paper presents the electromagnetic (EM) field scattering of a high speed moving source and a moving target by using Finite-Difference Time-Domain (FDTD) with Overset Grid Generation (OGG) method. The analysis is conducted for 750MHz band at the street intersection with OpenMP parallel processing technique. The performance of this proposed method is verified with theoretical results. The simulation results have shown comparatively good agreement in moving and stationary case. The proposed simulation study is of great importance to ground transportation in Intelligent Transportation System (ITS) applications.**

***Index Terms*—Direct Scattering; Doppler Effect; Intelligent Transportation System; Moving Source; Overlapped Grid**

I. Introduction

Finite-Difference Time-Domain (FDTD) combined with Overset Grid Generation (OGG) method has been previously proposed to investigate the characteristics of EM field from a moving and rotating body at high velocities [1]. In this paper, we report more comprehensive data and detailed discussion of the FDTD with OGG method for EM field scattering of a high speed moving source and its application. Lorentz transformation [2] is integrated in this proposed method to analyze the movement of the input source and moving target. The target, for example, moving vehicle at the street intersection is modelled by using the OGG method. This method consists of two parts; the main mesh and several sub-meshes. The meshes are overlapped with each other and can be calculated as one computational grid. This will reduce the complexity of the computational grid. A preliminary version of this paper appeared in [3].

In this paper, the analysis is done with OpenMP parallel processing technique. This parallel implementation is capable to increase the maximum solvable problem size and reduce the computational time [4]. The grid size ratio is first studied as a benchmark. Then, the EM field scattering from a moving source and to a moving vehicle in 750MHz band at the street intersection are analyzed. The analysis results show comparatively good agreement with stationary case. The proposed FDTD with OGG method has the potentials for epochal developments in nanoelectronics devices, particularly in the ITS applications [5]-[7].

II. FDTD Method In Parallel Processing Technique

The parallelism of FDTD method is implemented with OpenMP library using Fork-Join model on a shared memory system [8]. Figure 1 shows the illustration of the Fork-Join model of OpenMP. The OpenMP programs are initiated with master thread. Then, the parallel region I and II create a team of threads for electric field and magnetic field respectively by using OMP parallel coding *#pragma omp parallel*. The parallel threads will be processed in a single program when each region has completely constructed. The computational time between with and without openMP library is compared to evaluate its efficiency. A computation volume for main mesh is 1000×1000 grids while the overlapped single sub-mesh is 50×50 grids. The number of time iterations is 10000 steps. In this analysis, the sub-mesh moves with velocity, $v = 0.01c$ where c is the speed of light. The parallel FDTD method is analyzed by using operating system (OS) Ubuntu 14.04 LTS (64 bit) on CPU Intel Core i7-3960X with $3.3GHz$ memory. Table 1 shows the comparison of the FDTD method performance with and without OpenMP library in a single processor. Here, it is shown that FDTD method with OpenMP is 2.87 times faster compared to the FDTD method without OpenMP.

Table 1
Performance of FDTD Method

FDTD Method	Computational Time
With OpenMP	129 sec
Without OpenMP	370 sec

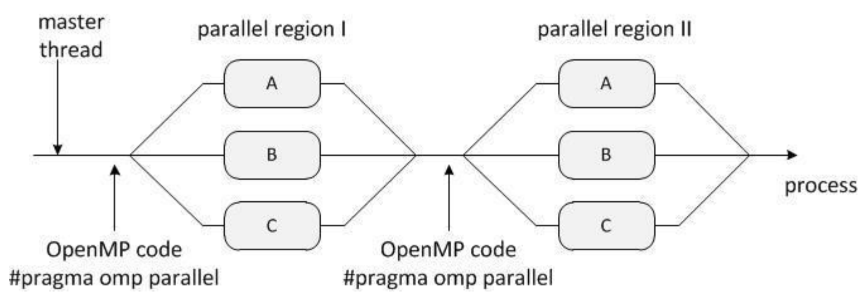

Figure 1: Illustration of Fork-Join Model of OpenMP