L-5

Computation Based on Heterogeneous Parallel Processor Architecture for Electromagnetic Wave Problems

*Masaki Hirano[1], Shinichiro Ohnuki[2]

Abstract: Cell/B.E. processor is capable of achieving high performance via latest parallel computation. In this paper, we discuss optimization of the parallel code for Cell/B.E. and examine the acceleration for electromagnetic scattering problems. Our computational result shows that computational time is 17 times faster than that for a conventional CPU.

1. Introduction

Cell/B.E. processor is one of heterogeneous parallel processors which are known as the high electric power efficiency and high computational pafomance[1][2]. In fact, six computers using Cell/B.E. are listed in the top10 of the Green500[3] which is the rank of super computers with good electric power efficiency in June 2010.

In this paper, we investigate the potential of hardware acceleration using Cell/B.E. for 3D electromagnetic problems solved by the method of moments (MoM).

2. Optimazation of codes

Cell/B.E. has one PPE of general-purpose core and eight SPE of accelerator cores. To use accelerator cores, we have to optimize the computer programs for Cell/B.E.

Electromagnetic scattering from the conducting plate shown in Figure 1 is studied. The plate is discretized by triangular patches. The electric field integral equation (EFIE) for a conducting object is given by

$$\hat{t}\cdot\int_S \bar{\mathbf{G}}(\mathbf{r},\mathbf{r}')\cdot\mathbf{J}(\mathbf{r}')dS' = \frac{4\pi i}{k\eta}\hat{t}\cdot\mathbf{E}^i(\mathbf{r}), \quad (1)$$

for $\mathbf{r}$ on the plate, where $\hat{t}$ is any unit tangent vector on S.

$$\bar{\mathbf{G}}(\mathbf{r},\mathbf{r}') = \left(\bar{\mathbf{I}} - \frac{1}{k^2}\nabla\nabla'\right)g(\mathbf{r},\mathbf{r}'), \quad (2)$$

and $\mathbf{J}(\bullet)$ is unknown current density. In the MoM, unknown current density $\mathbf{J}(\bullet)$ is expanded by appropriately chosen set of basis functions, such as

$$\mathbf{J}(\mathbf{r}) = \sum_{n=1}^{N} a_n \mathbf{j}_n(\mathbf{r}). \quad (3)$$

Here, $\mathbf{j}_n(\bullet)$ is given by RWG basis functions[4] and a_n describes unknown expansion coefficients. Using the testing function $\mathbf{t}_m(\bullet)$, the EFIE is discretized by

$$\sum_{n=1}^{N} A_{mn}a_n = F_m,\ m = 1,2,\cdots N, \quad (4)$$

where

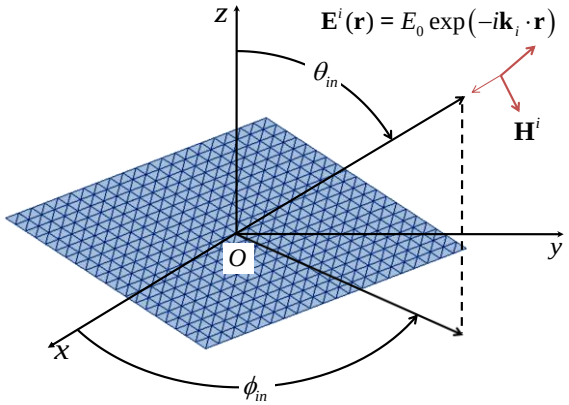

Figure 1. Geometry and coordinate system

$$F_m = \frac{4\pi i}{k\eta}\int_S dS\mathbf{t}_m(\mathbf{r})\cdot\mathbf{E}^i(\mathbf{r}), \quad (5)$$

$$A_{mn} = \int_S \mathbf{t}_m(\mathbf{r})dS\cdot\int_S \bar{\mathbf{G}}(\mathbf{r},\mathbf{r}')\cdot\mathbf{j}_n(\mathbf{r}')dS'. \quad (6)$$

We calculate this linear equation using CGNR (conjugate gradient on the normal equations) method. Equation (4) can be writen as linear simultaneous equation

$$\bar{\mathbf{A}}\mathbf{x} = \mathbf{b}. \quad (7)$$

In CGNR method, we calculate

$$\bar{\mathbf{A}}\mathbf{x} = \mathbf{b} \quad (8)$$

and

$$\bar{\mathbf{A}}^T\mathbf{x} = \mathbf{b} \quad (9)$$

iteratively. We examine optimization of equations (8) and (9) principally.

Here, we kick around the following optimization techniques.

(a) Multithreads: Jobs are equally assigned to SPEs for increasing efficiency of parallel computing.

(b) SIMD: SPE has 128 SIMD registers of 128-bit length. Therefore, scalars should be transformed into quadword (128bits) integer vectors. Using SIMD

1 : Graduate Course of Electrical Engineering, Graduate School of Science and Technology, Nihon University.

2 : Department of Electrical Engineering, College of Science and Technology, Nihon University

operation, the computation theoretically becomes four times faster.

(c) Multibuffering: DMA should be overlapped with computation by multibuffering which can hide memory latency.

3. Comptational results

Figue 2 shows the difference between the computational result obtained by Cell/B.E. and that by a conventional CPU. They are in good agreement and the relative error is less than 10^{-5}

Figure 3 shows the speedup rate of the computational tiem. Increasing the number of patchs, the speedup rate becomes higher. The maximum speedup rate is about 17 times higher, compared with a conventional CPU case. However, the rate tends to converge when the number of patches is over 4096.

Figure 4 shows the speedup rate for varying the number of SPEs compared with one SPE of Cell/B.E. for the 4096 patchs case. Increasing the number of SPEs, the rate becomes higher. To increase the number of SPEs, the rate converges to about 2.5 times. This problem is due to the transfer time between SPE and the main memory in PC.

4. Conclusions

We discuss optimization of the parallel code for Cell/B.E. and examine the acceleration for electromagnetic scattering problems. Our computational result shows that computational time is 17 times faster than that for a conventional CPU.

5. References

[1] IBM, “Cell Broadband Engine Programming Handbook Including the PowerXCell 8i Processor Version 1.12,” Cell Broadband Engine Public Information, April 3, 2009.

[2] Thomas Chen, Ram Raghavan, Jason Dale and Eiji Iwata, “Cell Broadband Engine Architecture and its first implementation - A performance view,” IBM Journal of Research and Development, Volume 51, Number 5, September 2007.

[3] THE GREEN500, The Green500 List June 2010, URL: http://www.green500.org/lists/2010/06/top/list.php

[4] S. M. Rao, D. R. Wilton, and A. W. Glisson, “Electromagnetic scattering by surfaces of arbitrary shape,” IEEE Trans. Antennas and Propagation, vol. 30, no. 3, pp. 409-418, 1982.

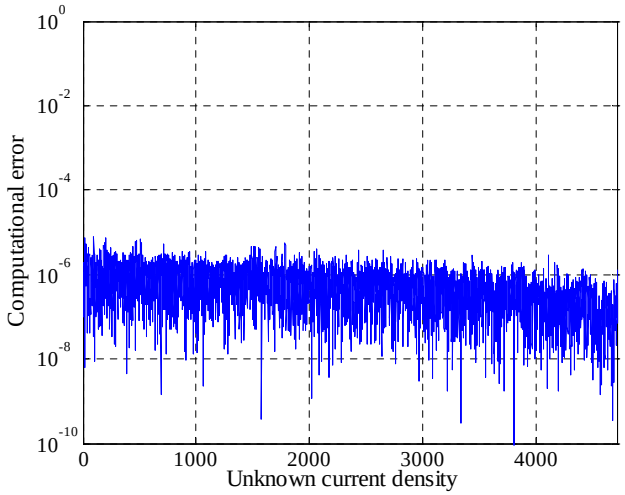

Figure 2. Computational error of Cell/B.E.

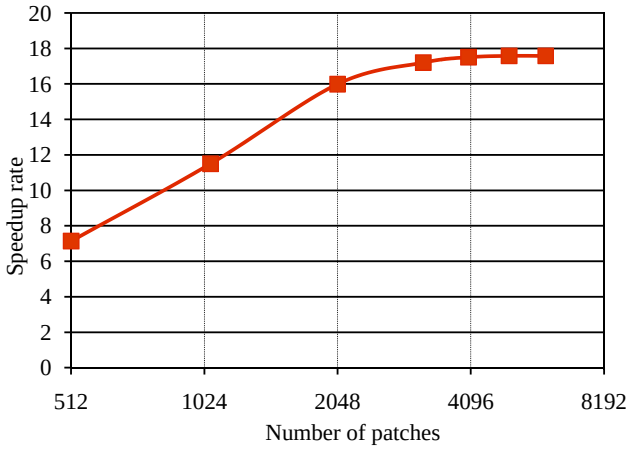

Figure 3. Speedup rate for varying the number of patchs

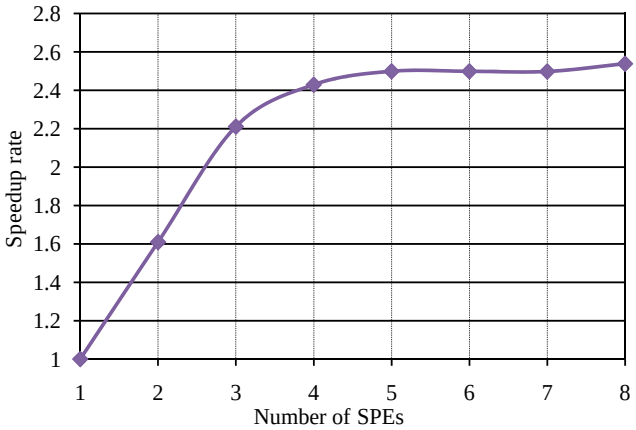

Figure 4. Speedup rate for varying the number of SPEs

[5] M. Hirano and S. Ohnuki, “Data-Parallel Programming of the Method of Moments Using Cell/B.E.”, The Papers of Technical Meeting on Electromagnetic Theory, IEE Japan, EMT-10-92, pp.169-174, July, 2010.