

ACOUSTICAL LETTER

GPU-computing-based high-speed visualization techniques for sound wave propagation using permeable multi-cross-section contours

Naoki Kawada[1,*], Kan Okubo[1,†], Norio Tagawa[1], Takao Tsuchiya[2], and Takashi Ishizuka[3]

[1]*Faculty of System Design, Tokyo Metropolitan University, 6–6 Asahigaoka, Hino, 191–0065 Japan*
[2]*Doshisha University, 1–3 Tatara Miyakodani, Kyotanabe, 612–0321 Japan*
[3]*Shimizu Corporation, 3–4–17 Etchujima, Koto-ku, Tokyo, 135–8530 Japan*

(*Received 30 June 2014, Accepted for publication 6 August 2014*)

Keywords: High-speed visualization, GPU computing, Sound wave propagation, Interactive simulation, Personal supercomputer
PACS number: 43.28.Js, 43.55.Ka, 87.50.Kk, 43.20.El, 43.28.Fp [doi:10.1250/ast.36.268]

1. Introduction

Acoustic simulation in the time domain [1–4] is an effective technique for the estimation of time-series sound pressure data (e.g., nonlinear acoustic propagation phenomena, ultrasonic phenomena, acoustical measurements and instrumentation, acoustical imaging). For time domain acoustic simulation, the development of a visualization method is a very important technical issue.

Recently, graphic processing units (GPUs) have been used as an acceleration tool for the calculation in various study fields. This movement is called general purpose computing on GPUs (GPGPU) [5–7]. In the last few years, the performance of GPUs has continued to improve rapidly. That is, a personal computer (PC) with GPUs might become a personal supercomputer. GPU computing gives a high-performance computing environment at a lower cost than before. Therefore, the use of GPUs is contributing to a significant reduction of the calculation time.

Moreover, the use of a GPU has an advantage for the visualization of calculated fields, since the GPU was originally architecture for graphics processing. In a GPU calculation, the calculated data is always stored in the video memory (VRAM). Therefore, we can directly write drawing information to the VRAM on the video card by combining CUDA and OpenGL, where OpenGL is a multiplatform application programming interface for rendering 2D and 3D graphics [8].

Although methods based on CUDA and OpenGL are gradually becoming better known, a high-speed visualization technique on a multi-GPU platform for acoustic numerical simulation has not yet fully developed. The purpose of this study is to develop high-speed visualization techniques for sound wave propagation using GPU parallel calculation with OpenGL. In this study, we examined high-speed visualization on a multi-GPU platform and demonstrated the feasibility of an interactive simulation using a GPU parallel calculation with permeable multi-cross-section contours (PMCC) as a visualization technique.

*e-mail: kawadanaoki@gmail.com
†e-mail: kanne@sd.tmu.ac.jp

2. Time-domain sound wave simulation for high-speed visualization

2.1. FDTD (Finite-Difference Time-Domain) method

For the high-speed visualization, we employ the FDTD method [9] because it requires less computational cost than other time domain methods. The finite-difference time-domain (FDTD) method directly solves the governing equation using a finite difference with a staggered grid. The values on each grid are mutually independent. That is, the valuea in the field can be calculated independently of each other; the FDTD method is thus suitable for parallelization on a GPU.

2.2. GPU programming with CUDA and GPGPU-visualization with OpenGL

In 2007, NVIDIA introduced compute unified device architecture (CUDA), a programming language designed for GPGPU. It is an extension of the C language. This made programming for a GPU easier than before.

Next, we describe the schematic procedure of conventional visualization using the CPU. Figure 1(a) shows the conventional methos of visualization using a CPU. First, the calculation results are converted to drawing information. Then, the drawing information is transferred to VRAM for drawing on a video card. In this process by CPU computation, the calculation results are stored once and need to be transferred to the VRAM from the main memory (RAM). A PCIe bus transfer is required; therefore, this process might be a bottleneck in the visualization of calculated results.

On the other hand, in the implementation based on a GPU computing by CUDA, the data of OpenGL can be associated. In addition, the calculated data is always stored in the VRAM. Therefore, it is possible to write drawing information to the VRAM on the video card without a PCIe bus data transfer. Figure 1(b) shows the method of visualization using a single GPU. The GPU has a memory bandwidth of around 200 GB/s, while the PCIe gen.2($\times$16) link speed is 8.0 GB/s per direction. This is a major advantage for high-speed visualization.

Next, we describe the schematic procedure of visualization using the multiple GPUs. Figure 1(c) shows a method of visualization using dual GPUs. Morevoer, Fig. 1(d) shows a method of visualization using dual GPUs

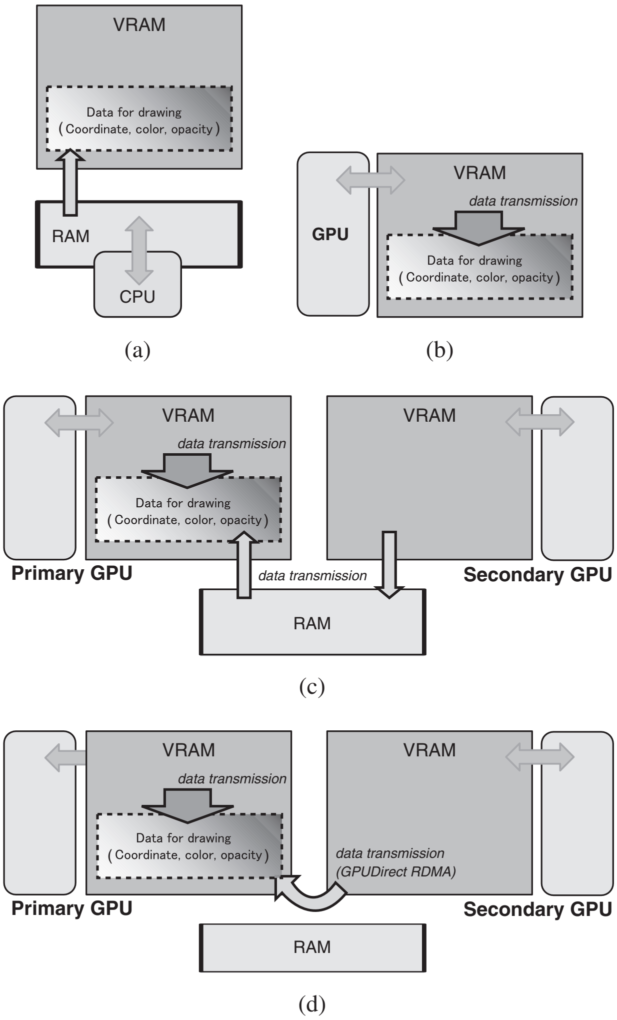

Fig. 1 Schematic procedure of CPU/GPU visualization.

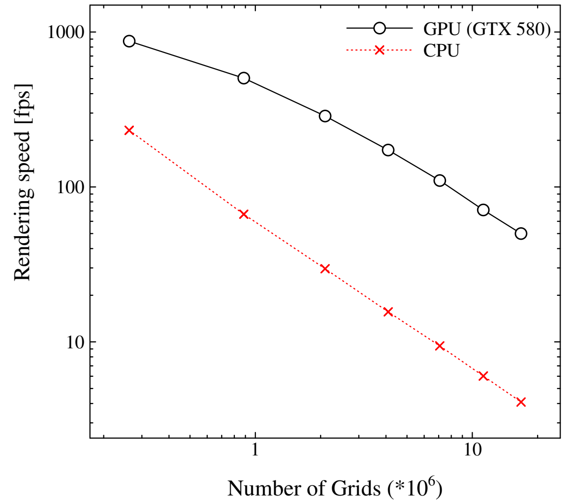

Fig. 2 Rendering speed in CPU and GPU 3D simulation.

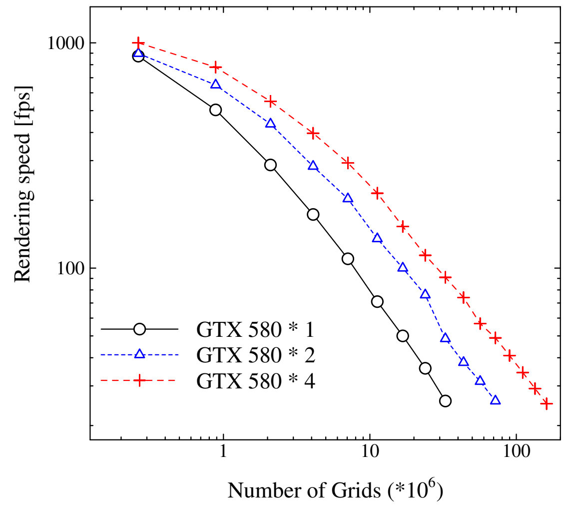

Fig. 3 Rendering speed in multi-GPU 3D simulation.

and GPUDirect remote direct memory access (RDMA) technology [10], which provides peer-to-peer direct communication between GPUs. Therefore, as shown in this figure, we can write drawing information to the VRAM on the primary GPU without the transfer of the RAM by GPUDirect technology.

3. 3D acoustic simulation and visualization method

Recently, as a visualization technique, so-called volume rendering [11] using opacity has been proposed. A feature of this method is that it uses all the information of a 3D sound field. Therefore, it can display an entire 3D region at a given time. However, this method has to handle all information in each voxel of a 3D sound field. This reduces the drawing speed to less than that of other conventional methods.

We have proposed PMCC [12] as a visualization method for 3D sound fields. PMCC is a method that can set the opacity in the multiple cross-section contours. Number and angle of the cross sections can be set arbitrarily; this provides flexibility in the computational visualization as a pseudo-3D expression.

4. Evaluation of high-speed visualization and interactive simulation using PMCC

We perform a high-speed visualization and interactive simulation that combines OpenGL and CUDA. In this study, we use GeForce GTX 580 GPU (with a memory bandwidth of 192.4 GB/s) and a Core i7 950 processor. The CPU calculation was parallelized and optimized using OpenMP. The calculations use single-precision floating-point numbers.

We compare the speed of visualization. Figure 2 shows the results of comparing the rendering speed using PMCC between the GPU and CPU against the number of grid points in a 3D acoustic field. We find that the speed of visualization using the GPU is ca. 10–12 times faster than that using the CPU.

Moreover, we examine visualization using multi-GPU computing. Figure 3 shows the results for the rendering speed in a 3D acoustic field against the number of grid points. By using quad GPUs (GTX 580 ×4) and GPUdirect 2.0, we can perform a visualization in 25 fps for a field with 512^3 grid points.

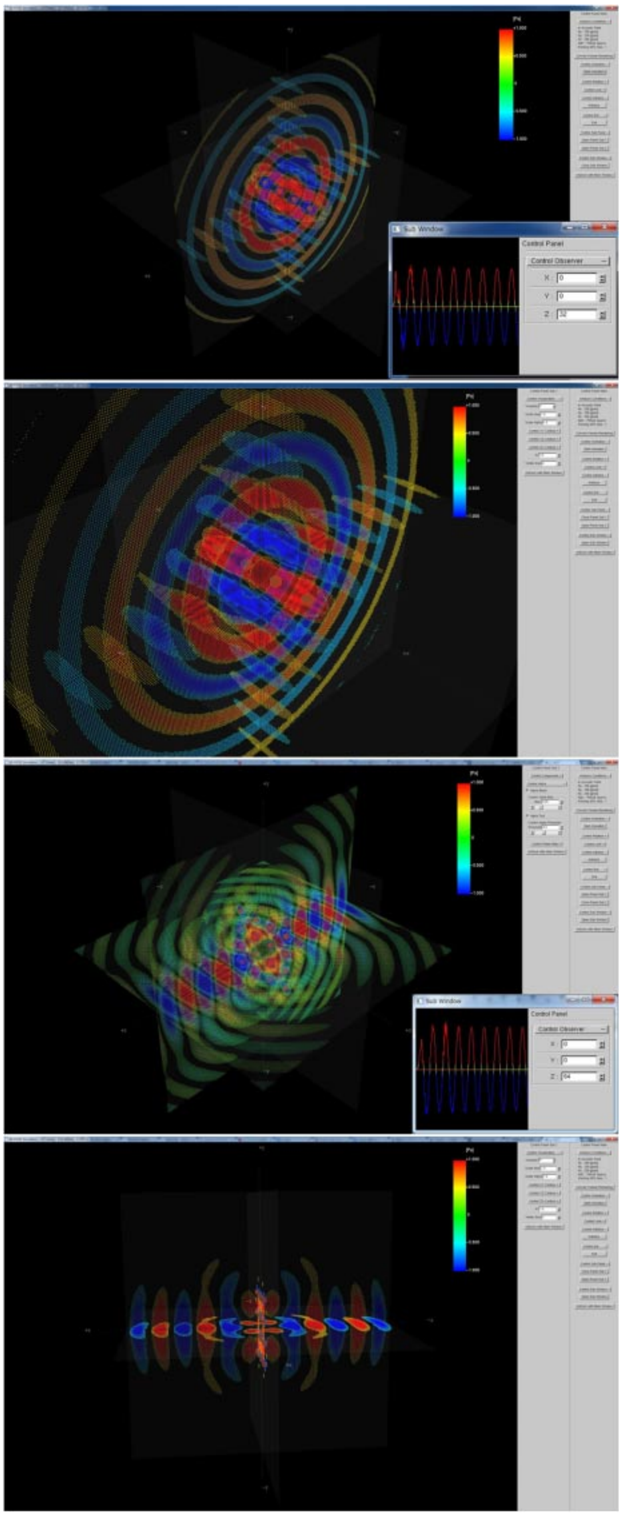

Fig. 4 Examples of visualization using PMCC.

The high-speed visualization using the GPU makes the following possible:

(1) The analysis parameters can be changed during computation and visualization.

(2) The analysis point can be changed during computation and visualization.

(3) The input signal can be changed during computation and visualization.

Hence, it is demonstrated that an interactive simulation is possible by using GPGPU-based calculation and visualization. Figure 4 shows examples of 3D sound field visualization. The plot window has a main frame that displays a 3D region using the PMCC. Additionally, as shown in the lower right of two of the images in Fig. 4, it also has sub frames to display the sound field intensity in a time series for selected points. The main window can display an adaptive visualization of the sound pressure or the particle velocity of a sound field.

5. Conclusion

In this study, we examined GPU-computing-based high-speed visualization and its feasibility for the interactive simulation of 3D sound field numerical analysis. As a visualization method of 3D sound fields, PMCC in the GPU implementation was employed. By using quad GPUs (GTX 580 $\times 4$), we could perform a visualization in 25 fps for a field of 512^3 grid points using PMCC. The GPU-computing-based interactive simulation is expected to be a new important technique of sound field analysis.

References

[1] M. Toyoda and D. Takahashi, "Prediction for architectural structure-borne sound by the finite-difference time-dimain method," *Acoust. Sci. & Tech.*, **30**, 265–276 (2009).

[2] M. Konno, K. Okubo, T. Tsuchiya and N. Tagawa, "Performances of various types of constrained interpolation profile methods for two-dimensional numerical acoustic simulation," *Jpn. J. Appl. Phys.*, **47**, 3962–3963 (2008).

[3] T. Tsuchiya and A. Kumagai, "Numerical simulation of acoustic imaging using a combination of finite difference time domain and boundary integral equation methods," *Jpn. J. Appl. Phys.*, **48**, 07GN02 (2009).

[4] K. Okubo, S. Oh, T. Tsuchiya and N. Takeuchi, "Consideration of the boundary condition between two media in acoustic field analysis using the Constrained Interpolation Profile (CIP) method," *IEICE Trans. Fundum.*, **E90-A**, 2000–2005 (2007).

[5] http://www.nvidia.co.jp/object/cuda-jp.html (accessed 2014-7-12).

[6] T. Aoki and A. Nukada, *CUDA Programming for Beginners* (Kohgakusha, Tokyo, 2009) (in Japanese).

[7] T. Tsuchiya, T, Ishii, Y. Iwaya and M. Otani, "Sound field rendering with GPU cluster and high definition auralization," *Tech. Rep. IEICE*, EA 112(171), pp. 1–6 (2012).

[8] http://www.opengl.org/ (accessed 2014-7-12).

[9] K. S. Yee, "Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media," *IEEE Trans. Antennas Propag.*, **AP-14**, 302–307 (1966).

[10] http://docs.nvidia.com/cuda/cuda-c-programming-guide/ (accessed 2014-7-12).

[11] R. A. Drebin, L. Carpenter and P. Hanrahan, "Volume rendering," *SIGGRAPH Comput. Graph.* **22**(4), pp. 65–74 (1988).

[12] N. Kawada, K. Okubo, N. Tagawa, T. Tsuchiya and T. Ishizuka, "Examination on high-speed visualization of acoustic numerical analysis using CUDA and OpenGL: Proposal of PMCC (Permeable Multi Cross-section Contours)," *IEICE Tech. Rep. Ultrasonics*, 110(419), pp. 13–18 (2011) (in Japanese).