Revisiting Distributed Particle-Based Volume Rendering on Fugaku and Beyond

Tsukasa Osaki, Jorji Nonaka
Center for Computational Science (R-CCS)
RIKEN, Kobe, Japan
tsukasa.osaki@riken.jp

Takuma Kawamura
Center for Computational Science & e-Systems (CCSE)
Japan Atomic Energy Agency (JAEA), Kashiwa, Japan
kawamura.takuma@jaea.go.jp

***Abstract*—Modern flagship high-performance computing (HPC) systems increasingly rely on GPU-accelerated architectures, as reflected in the TOP500 supercomputer rankings. While GPUs are primarily used to accelerate simulation workloads, CPU resources may remain available during GPU-dominated computations. Previous work has explored utilizing these CPU resources for in-situ processing such as visualization. In this work, we focus on Particle-Based Volume Rendering (PBVR), which represents volumetric data as a collection of tiny opaque particles, allowing particle generation to be decoupled from the final volume rendering process. Motivated by the planned ARM-based CPU architecture (Fujitsu MONAKA-X) of FugakuNEXT, we propose a distributed PBVR framework in which particle generation is performed on the HPC system using CPUs, while interactive volume rendering is carried out on a GPU-equipped local PC. The framework demonstrated high scalability in evaluations using up to 110,592 compute cores (2,304 MPI processes with 48 OpenMP threads each) while significantly reducing the amount of visualization data transferred to the client. The proposed architecture provides a scalable visualization solution for the current Fugaku supercomputer and offers a practical model for future GPU-centric HPC systems, including FugakuNEXT.**

***Index Terms*—PBVR, CPU-based particle generation, GPU-based volume rendering, Fugaku, FugakuNEXT**

I. Introduction

Fugaku has been in operation since March 2021 as a CPU-only flagship high-performance computing (HPC) system. After more than five years of operation, it remains highly competitive, ranking 9th in the June 2026 TOP500 list and 3rd in both the HPCG benchmark and the GRAPH500 BFS and SSSP rankings. Meanwhile, NVIDIA has been announced as a co-design partner for Japan's next flagship supercomputer, FugakuNEXT, while Fujitsu is responsible for the co-design of its Arm-based CPU, known as MONAKA-X. While modern flagship HPC systems increasingly rely on GPUs to accelerate simulation workloads, CPU resources may remain underutilized during GPU-dominated computations. Exploiting these available CPU resources for in-situ processing has therefore attracted increasing attention. For example, Mateevitsi et al. [1] demonstrated the use of idle CPU resources for in-situ visualization. Building on this idea, we investigate a distributed Particle-Based Volume Rendering (PBVR) [2] framework that performs particle generation on the HPC system using CPUs while offloading interactive rendering to a GPU-equipped client (Fig. 1).

Volume rendering is a well-established visualization technique for generating two-dimensional images directly from three-dimensional scalar fields without first extracting geometric primitives [3]. Particle-Based Volume Rendering (PBVR), in contrast, represents volumetric data as a collection of stochastically generated opaque particles. The particle density is determined by the opacity assigned to each scalar value through the transfer function, with higher opacity resulting in a greater particle density. PBVR was the only visualization technique implemented in the official *Visualization Library* [4] provided by Fujitsu for the K computer, the predecessor of the Fugaku supercomputer. More recently, JAEA has continued developing PBVR as a comprehensive particle-based remote visualization framework, supporting capabilities such as in-situ processing [5], interactive steering [6], multivariate visualization, and immersive visualization. In this work, we build upon the JAEA PBVR framework to implement and evaluate a distributed PBVR workflow on Fugaku.

II. CPU-based Particle Generation

The particle-based intermediate representation decouples particle generation from the final volume rendering process, allowing these stages to be executed independently. We evaluated CPU-based particle generation on Fugaku using the OpenFOAM v2206 computational fluid dynamics (CFD) package and its icoFoam/cavity tutorial case. For in-situ particle generation, the VTK (Visualization Toolkit) and KVS (Kyoto Visualization System) libraries were integrated with OpenFOAM. The simulation code was instrumented with the PBVR Adapter through its C/C++ and Fortran APIs and recompiled using the Fujitsu compiler. Table I shows the simulation grid size used and the parallel processing parameters used in the experimental evaluations. A single MPI process was assigned to each A64FX compute node.

TABLE I
Utilized data and parallel processing parameters

Item	Description
Simulation grid size	$432 \times 432 \times 432$
MPI Processes	$72, 144, 288, 576, 1152, 2304$
OpenMP threads	48 threads

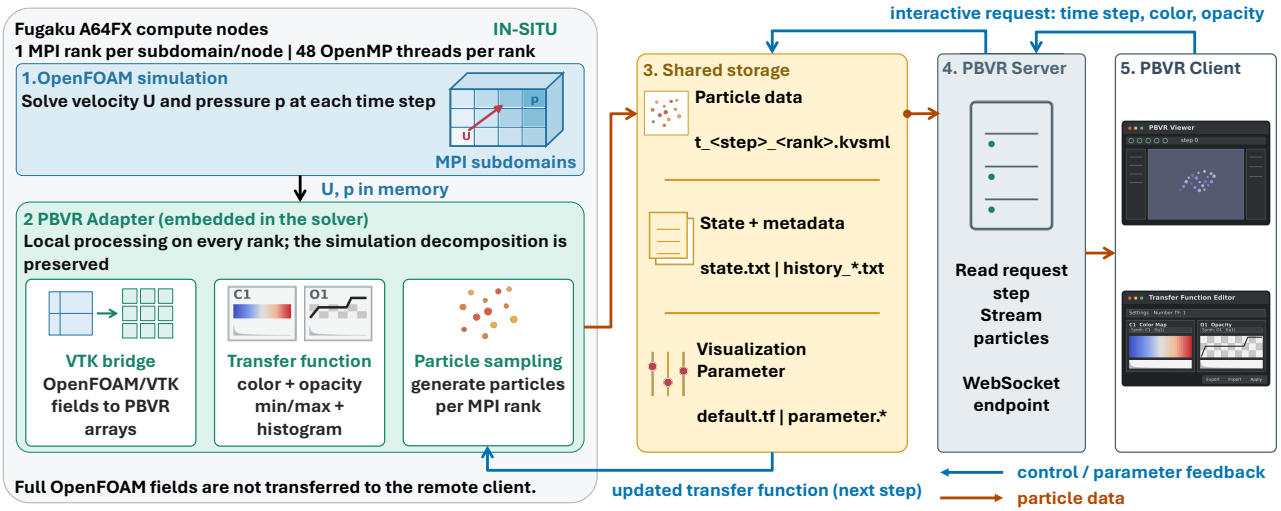


Fig. 1. Overview of the distributed Particle-Based Volume Rendering (PBVR) framework. In-situ particle generation (left) is performed on the CPUs of the HPC system, with the generated particles written to shared storage (middle). Interactive visual exploration can be performed asynchronously or during the simulation (right), enabling the transfer function to be interactively updated for subsequent particle generation.

A. *Parallel processing and Data compression*

Figure 2 shows the in-situ particle generation time for different numbers of MPI processes, each with 48 OpenMP threads. The results demonstrate good scalability up to more than 100,000 CPU cores. In addition, the generated particle data reduced the amount of data to be written by 55% to 90% compared with the original simulation output for a (120^3) grid. Although in-situ particle generation introduces additional computational overhead compared with directly writing the original simulation data, the substantial reduction in output data size can compensate for this overhead by reducing the overall I/O cost.

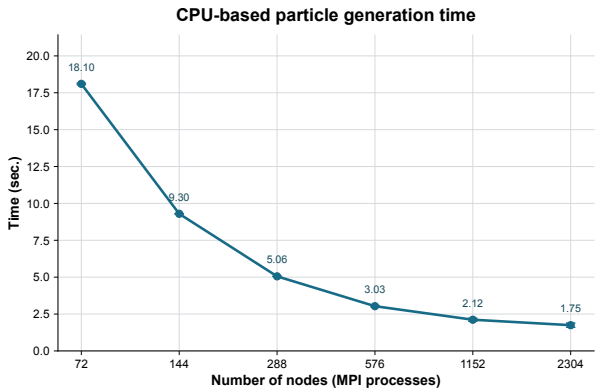


Fig. 2. Mean rank-maximum non-I/O time over 11 steps and three runs.

III. Conclusion

We evaluated the proposed distributed PBVR framework, in which particle generation is performed on the HPC system using CPUs while interactive volume rendering is carried out on a GPU-equipped local PC. The framework demonstrated high scalability, while substantially reducing the amount of visualization data transferred from the HPC system. Although particle generation introduces additional computational overhead, the resulting data reduction can reduce the overall I/O requirements compared with directly writing the original simulation data. As Fugaku is expected to remain in operation until around 2030, this distributed PBVR framework can help reduce simulation data I/O while enabling efficient large-scale visualization. Furthermore, the proposed architecture is readily applicable to emerging GPU-based HPC systems, where available CPU resources can be utilized for in-situ processing during GPU-dominated simulations.

Acknowledgment

This work used computational resources of the supercomputer Fugaku provided by the RIKEN R-CCS (Project ID: ra260007). AI tools (Grammarly and ChatGPT) were used to correct and improve the text.

References

[1] V. A. Mateevitsi, A. Sewell, J.-H. Göbbert, M. Bode, P. Fischer, J. A. Insley, I. Kavroulakis, D. Konioris, Y.-H. Lan, M. Min, D. Papageorgiou, S. Petruzza, S. Rizzi, A. Tomboulides, and M. E. Papka, "Free lunch in in situ visualization: Leveraging idle CPU resources to mitigate GPU contention," in *IEEE LDAV 2025*, 2025, pp. 12–21.

[2] N. Sakamoto, J. Nonaka, K. Koyamada, and S. Tanaka, "Particle-based Volume Rendering," in *Proceedings of the IEEE Asia-Pacific Symposium on Visualization*, 2007, pp. 129–132.

[3] C. D. Hansen and C. R. Johnson, Eds., *The Visualization Handbook*. Academic Press / Elsevier, 2005.

[4] A. Ogasa, H. Maesaka, K. Sakamoto, and S. Otagiri, "Visualization technology for the K computer," *Fujitsu Scientific and Technical Journal*, vol. 48, no. 3, pp. 348–356, 2012.

[5] T. Kawamura, T. Noda, and Y. Idomura, "In-situ visual exploration of multivariate volume data based on particle based volume rendering," in *2016 Second Workshop on In Situ Infrastructures for Enabling Extreme-Scale Analysis and Visualization (ISAV)*, 2016, pp. 18–22.

[6] T. Kawamura, Y. Hasegawa, and Y. Idomura, "Interactive steering on in situ particle-based volume rendering framework," *Journal of Visualization*, vol. 27, no. 1, p. 89–107, Sep. 2023.