Large-Scale Parallel Visualization of Particle Datasets using Point Sprites

S. Rizzi, M. Hereld, J. Insley, M.E. Papka, T. Uram, V. Vishwanath
Leadership Computing Facility, Argonne National Laboratory

MIRA

ABSTRACT

Massive particle-based datasets can be efficiently rendered using point sprites. Among other advantages, the method is efficient in its use of memory, preserves the dynamic range of the simulations, and provides excellent image quality. Preliminary results of large-scale parallel rendering of particle data sets using point sprites are presented in this poster. Performance and scalability are evaluated on a GPU-based computer cluster using datasets of up to 3200^3 particles at resolutions of up to 6144x3072 pixels.

MOTIVATION

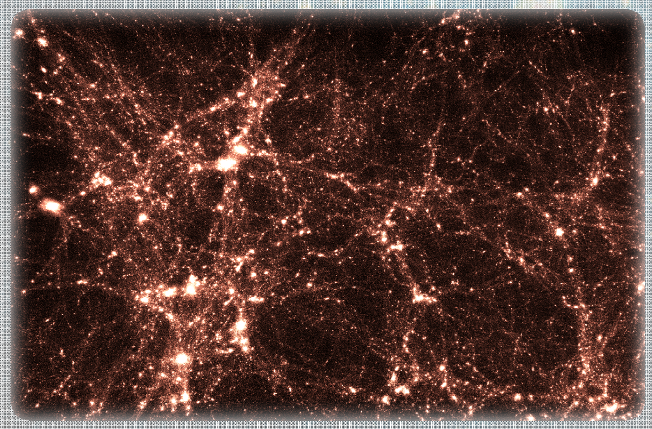
- Volume rendering results in sparse 3D grids and loss of dynamic range due to mapping the particle positions to a fixed grid
- Direct point rendering use particle positions, avoding memory sparsity and loss of dynamic range
- Hardware-accelerated point rendering is a fast and eminently parallel task

vl3 PARALLEL RENDERING FRAMEWORK

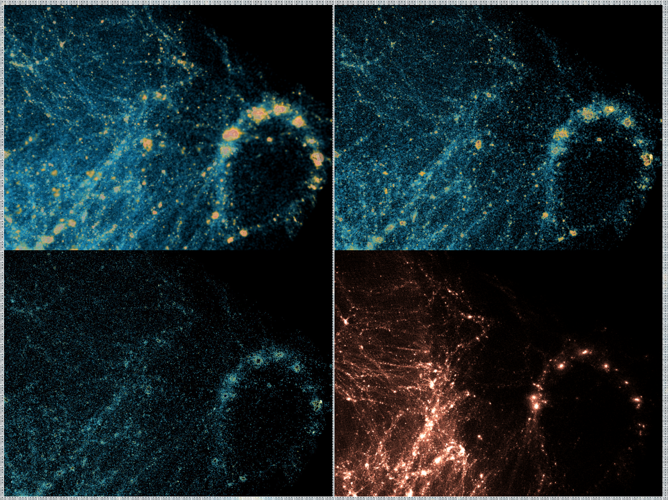
vl3 provides:
- Parallel data reading
- Different domain space partitioning strategies
- Hardware-accelerated Parallel OpenGL rendering, one MPI rank per GPU
- Multiple compositing algorithms, including serial and parallel DirectSend, and BinarySwap

IMAGE QUALITY

- Coarser grids are noticeable in volume rendering. Individual particles can not be resolved
- Point sprites clearly show higher mass density regions while preserving the individual resolution of each particle.



Volume rendering 512^3 grid | Volume rendering 1024^3 grid



Volume rendering 2048^3 grid | Point sprite rendering

PARALLEL POINT SPRITE RENDERING IN vl3

- Particles are loaded in GPU memory and rendered as OpenGL point sprites
- GLSL shaders are used to modify point size and texture color
- Color per point allows to explore velocity fields or color particles by their IDs
- Variable point size can be taken into account by vertex shader
- Exchangeable point textures are useful to identify particles with specific properties (i.e. halo particles)
- Additive alpha blending
- Multiple timesteps can be loaded for interactive spatio-temporal exploration

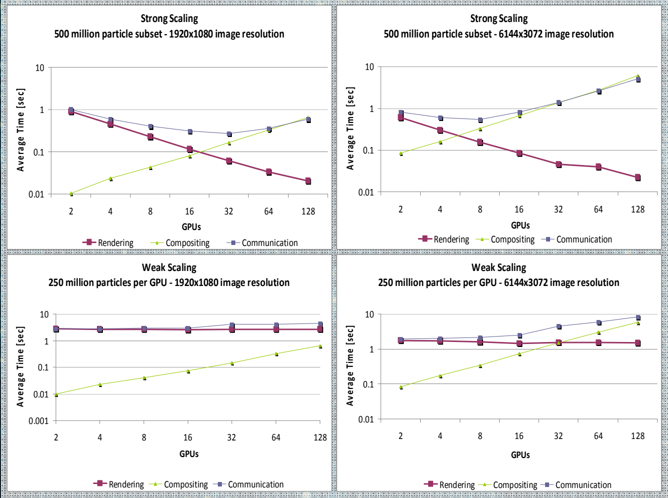
EXPERIMENTAL PLATFORM

Tukey, Argonne's visualization and analysis cluster, consisting of:
- 96 compute nodes, 16 cores, 64 GB RAM per node
- Two Nvidia Tesla M2070 GPUs per compute node
- QDR Infiniband interconnect
- Mvapich2 MPI

Image sizes are 6144x3072 pixels for Argonne's Active Mural (AM) tiled-display.

DATASETS

- Particle datasets from HACC N-body simulations
  - 1024^3 full particle dataset, 38 gigabyte
  - 3200^3 full particle dataset, 1.2 terabyte
- About 250 million particles per GPU



FUTURE WORK

- Scaling to larger datasets (8192^3 and 10240^3 particles)
- Improve interactivity
- Investigate Level-Of-Detail (LOD) and out-of-core methods to avoid GPU memory limitations
- Explore sort-first parallel rendering to mitigate compositing communication time
- Study dynamic load balancing schemes

CONCLUSION

In this work we propose parallel hardware-accelerated point sprite rendering as an alternative to other commonly used methods for rendering large-scale particle datasets. We have initially focused on <u>rendering performance and scalability</u>, scaling to a full dataset of 3200^3 particles. In terms of rendering time, the current implementation has shown excellent scalability on up to 128 GPUs.

ACKNOWLEDGMENT

Supported by the Office of Science of the U.S. Department of Energy under contract DE-AC02-06CH11357.
S.Habib, K. Heitmann, and the HACC team at Argonne are gratefully acknowledged for providing the datasets used in this study.

CONTACT

{srizzi, hereld, insley, papka, turam, venkat}@anl.gov