CTWatch QUARTERLY

Emerging Visualization Technologies for Ultra-Scale Simulations

Kwan-Liu Ma, University of California at Davis

Supercomputers give scientists the power to model highly complex and detailed physical phenomena and chemical processes, leading to many advances in science and engineering. With the current growth rates of supercomputing speed and capacity, scientists are anticipated to study many problems at unprecedented complexity and fidelity and attempt many new problems for the first time. The size and complexity of the data produced by such ultra-scale simulations, however, present tremendous challenges to the subsequent data visualization and analysis tasks, creating a growing gap between scientists' ability to simulate complex physics at high resolution and their ability to extract knowledge from the resulting massive data sets. The Institute for Ultrascale Visualization [1][2], funded by the U.S. Department of Energy's SciDAC program [3], aims to close this gap by developing advanced visualization technologies that enable knowledge discovery at the peta and exa-scale. This article reveals three such enabling technologies that are critical to the future success of scientific supercomputing and discovery.

Parallel Visualization

Parallel visualization can be a useful path to understanding data at the ultra scale but is not without its own challenges especially across our diverse scientific user community. The Ultravis Institute has brought together leading experts from visualization, high-performance computing, and science application areas to make parallel visualization technology a commodity for SciDAC scientists and the broader community. One distinct effort is the development of scalable parallel visualization methods for understanding vector field data. Vector field visualization is more difficult to do than scalar field visualization because it generally requires more computing for conveying the directional information and more storage space to store the vector field.

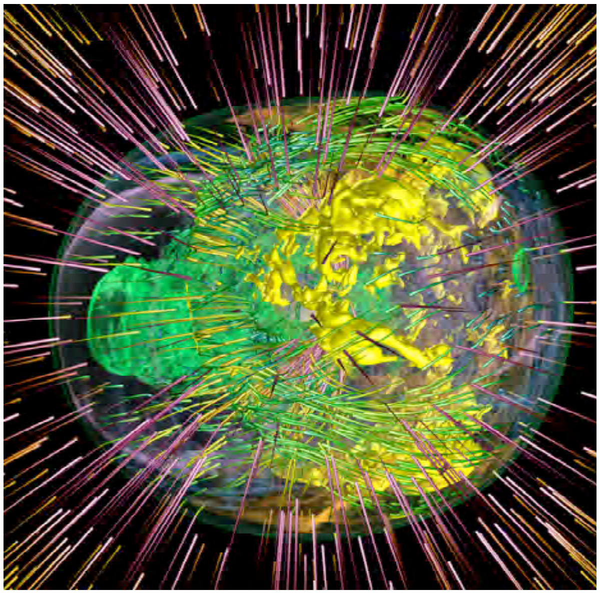

Figure 1. Simultaneous visualization of velocity and angular momentum fields obtained from a supernova simulation.

So far, more researchers have worked on the visualization of scalar field data than vector field data, regardless of the fact that vector fields in the same data sets are equally critical to the understanding of the modeled phenomena. 3D vector field visualization particularly requires more attention from the research community because most of the effective 2D vector field visualization methods incur visual clutter when directly applied to depicting 3D vector data. For large data sets, a scalable parallel visualization solution for depicting vector field is even more needed because the expanded space requirement and additional calculations needed to ensure temporal coherence for visualizing time-varying vector data. Furthermore, it is challenging to simultaneously visualize both scalar and vector fields due to the added complexity of rendering calculations and combined computing requirements. As a result, previous works in vector field visualization primarily focused on 2D, steady flow field, the associated seed/glyph placement problem, or the topological aspect of the vector fields.

Particle tracing is fundamental to portraying the structure and direction of a vector flow field. When an appropriate set of seed points are used, we can construct paths and surfaces from the traced particles to effectively characterize the flow field. Visualizing a large time-varying vector field on a parallel computer using particle tracing presents some unique challenges. Even though the tracing of each individual particle is independent of other particles, a particle may drift to anywhere in the spatial domain over time, demanding interprocessor communication. Furthermore, as particles move around, the number of particles each processor must handle varies, leading to uneven workloads. We have developed a scalable parallel particle tracing algorithm allowing us to visualize large time-varying 3D vector fields at the desired resolution and precision [4]. Figure 1 shows visualization of velocity field superimposed with volume rendering of a scalar field from a supernova simulation.

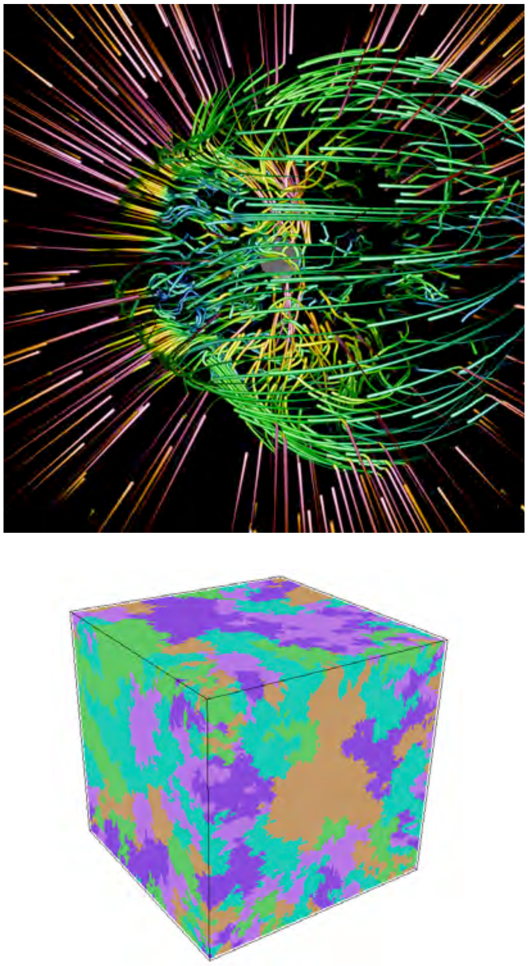

Figure 2. Pathline visualization of velocity field from a supernova simulation and the corresponding vector field partitioning.

We take a high-dimensional approach by treating time as the fourth dimension, rather than consider space and time as separate entities. In this way, a 4D volume is used to represent a time-varying 3D vector field. This unified representation enables us to make a time-accurate depiction of the flow field. More importantly, it allows us to construct pathlines by simply tracing streamlines in the 4D space. To support adaptive visualization of the data, we cluster the 4D space in a hierarchical manner. The resulting hierarchy can be used to allow visualization of the data at different levels of abstraction and interactivity. This hierarchy also facilitates data partitioning for efficient parallel pathline construction. We have achieved excellent parallel efficiency using up to 256 processors for the visualization of large flow field [4]. This new capability enables scientists to see their vector field data in unprecedented detail, at varying abstraction levels, and with higher interactivity, as shown in Figure 2.

Visualization Interfaces

Over the past twenty years, many novel visualization techniques have been invented but few have been deployed in production systems and tools. Even though some of techniques are made available in a few open-source visualization tools, scientists seem to prefer the more rudimentary tools they have been using. There are several reasons for this. First, scientists are reluctant to switch to a new tool unless the tool can seamlessly fit in their existing computing and analysis environment. Second, although the new technique may produce highly desired visualizations, it would not be widely employed if it requires a tedious process and special hardware to operate. Third and most importantly, for scientists to adopt a new tool, the tool must be very easy and intuitive to use. The past effort in the visualization research community largely focused on improving the performance and quality of visualization calculations. Only from a few years ago, the design and deployment of appropriate user interfaces for advanced visualization techniques began to receive more attention [5][8].

Interface design has played a major role in several of our visualization projects. One such visualization interface designed for exploring time-varying multivariate volume data consists of three components, which abstract the complexity of exploring in different spaces of the data and visualization parameters [6]. One important concept realized here is that the interface is also the visualization itself. As shown in Figure 3, the right-most panel displays the time histograms of the data. A time histogram shows how the distribution of data values changes over the whole time sequence and can thus help the user to identify time steps of interest and to specify time-varying features. The middle panel attempts to display the potential correlation between each pair of variables in parallel coordinates for a selected time step. By examining different pairs of variables the user can often identify features of interest based on the correlations observed. The left-most panel displays hardware accelerated volume rendering enhanced with the capability to render multiple variables into a single visualization in a user controllable fashion. Such simultaneous visualization of multiple scalar quantities allows the user to more closely explore and validate their simulations from the parallel-coordinate space to the 3D physical space. These three components are tightly cross linked to facilitate tri-space data exploration, offering scientists new power to study their time-varying volume data.

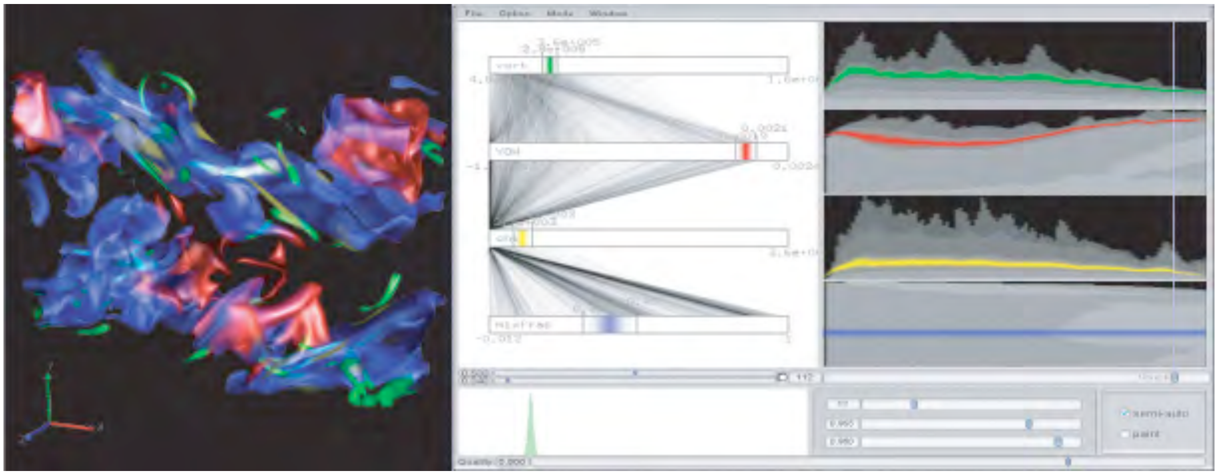

Figure 3. Interface for tri-space visual exploration of time-varying multivariate volume data [6]. From left to right, the spatial view, variable view, and temporal view of the data are given.

The other interface design effectively facilitates visualization of multidimensional particle data output from a gyrokinetic simulation [6]. Depicting the complex phenomena associated with the particle data presents a challenge due to the large quantity of particles, variables, and time steps. By utilizing two modes of interaction–physical space and variable space–our system allows scientists to explore collections of densely packed particles and discover interesting features within the data. While single variables can be easily explored through the use of a one dimensional transfer function, we again turn to the information visualization approach of parallel coordinates for interactively selecting particles in multivariate space. In this manner, particles with deeper connections can be separated from the rest of the data and then rendered using sphere glyphs and pathlines, as shown in Figure 4. With this system, scientists at Princeton Plasma Physics Laboratory are able to more easily identify features of interest, such as the location and motion of particles that become trapped in turbulent plasma flow. The combination of scientific and information visualization techniques extend our ability to analyze complex collections of particles.

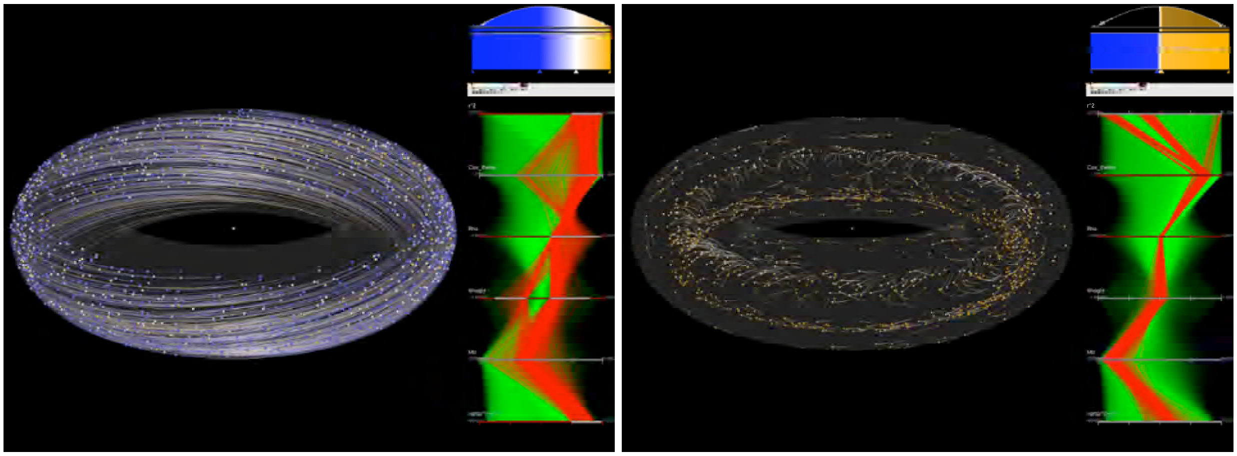

Figure 4. A parallel coordinate interface for multidimensional particle data visualization. The six axes of the parallel coordinates, from top to bottom, are: toroidal coordinate, trap particle condition, parallel velocity, statistical weight, perpendicular velocity, and distance from the center. Left: Visualization of those particles in a layer far from the center, with high parallel velocity and non-zero statistical weight. Right: Visualization of those particles changing direction frequently. This is achieved by restricting the parallel velocity values in a small range.

In addition, we have been studying how to incorporate machine learning into the process of visualization, leading to an intelligent interface for data visualization. Intelligent interfaces are anticipated to replace the current clutter of hardware-specific and algorithm-specific controls with a simple and intuitive interface supported by an invisible layer of complex intelligent algorithms [8]. Only high-level, goal-oriented decisions need to be made by the user, making cutting-edge visualization technology directly accessible to a wide range of application scientists. To make intelligent interfaces widely employed, we need to evaluate the effectiveness of the resulting interface designs using a variety of applications. These studies will pave the way to the creation of next-generation visualization technology. We believe the next generation visualization technology will be built upon further exploitation of human perception to simplify visualization, advanced hardware features to accelerate visualization calculations, and machine learning to reduce the complexity, size, and high-dimensionality of data.

In-Situ Visualization

Due to the size of data output by a large-scale simulation, visualization is almost exclusively done as a post-processing step. Even though it is desirable to monitor and validate some of the simulation stages, the cost of moving the simulation output to a visualization machine could be too high to make interactive visualization feasible. A better approach is not to move the data, or to keep the data that must be moved to a minimum. That is, both simulation and visualization calculations run on the same parallel supercomputer so the data can be shared, as shown in Figure 5. Such in-situ processing can render images directly or extract features, which are much smaller than the full raw data, to store for on-the-fly or later examination. As a result, reducing both the data transfer and storage costs early in the data analysis pipeline can optimize the overall scientific discovery process.

In practice, however, this approach was seldom adopted because of two reasons. First, most scientists were reluctant to use their supercomputer time for visualization calculations. Second, it could take a significant effort to couple a legacy parallel simulation code with an in-situ visualization code. In particular, the domain decomposition optimized for the simulation is often unsuitable for parallel visualization, resulting in the need to replicate data for speeding up the visualization calculations. Hence, the common practice for scientists has been to store only a small fraction of the data or to study the stored data at a coarser resolution, which defeats the original purpose of performing the high-resolution simulations. To enable scientists to study the full extent of the data generated by their simulations and for us to possibly realize the concept of steering simulations at extreme-scale, we ought to begin investigating the option of in-situ processing and visualization. Many scientists become convinced that simulation-time feature extraction, in particular, is a feasible solution to their large data problem. An important fact is that during the simulation time, all relevant data about the simulated field are readily available for the extraction calculations.

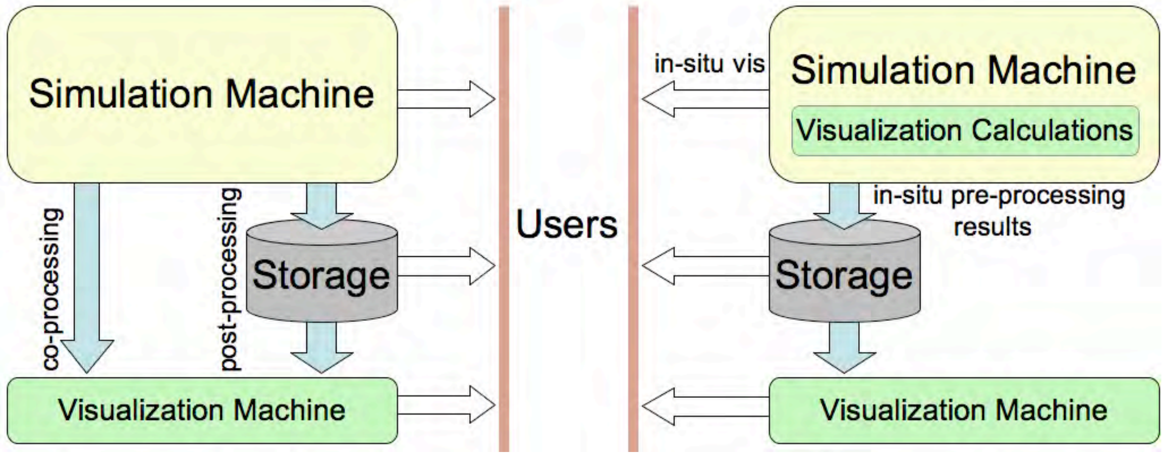

Figure 5. Left: the conventional ways to visualize a large-scale simulation running on a supercomputer. Right: In-situ processing and visualization of large-scale simulations.

In many cases, it is also desirable and feasible to render the data in-situ for monitoring and steering a simulation. Even in the case that runtime monitoring is not practical due to the length of the simulation run or the nature of the calculations, it could still be desirable to generate an animation characterizing selected parts of the simulation. This in-situ visualization capability is especially helpful when a significant amount of the data is to be discarded. Along with restart files, the animations could capture the integrity of the simulation with respect to a particularly important aspect of the modeled phenomenon.

We have been studying in-situ processing and visualization for selected applications to understand the impact of this new approach on ultra-scale simulations, subsequent visualization tasks, and how scientists do their work. Compared with a traditional visualization task that is performed in a post-processing fashion, in-situ visualization brings some unique challenges. First of all, the visualization code must interact directly with the simulation code, which requires both the scientist and the visualization specialist to commit to this integration effort. To optimize memory usage, we have to find a way for the simulation and visualization codes to share the same data structures to avoid replicating data. Second, visualization workload balancing is more difficult to achieve since the visualization has to comply with the simulation architecture and be tightly coupled with it. Unlike parallelizing visualization algorithms for standalone processing where we can partition and distribute data best suited for the visualization calculations, for in-situ visualization, the simulation code dictates data partitioning and distribution. Moving data frequently among processors is not an option for visualization processing. We need to rethink to possibly balance the visualization workload so the visualization is at least as scalable as the simulation. Finally, visualization calculations must be low cost, with decoupled I/O for delivering the rendering results while the simulation is running. Since the visualization calculations on the supercomputer cannot be hardware accelerated, we must find other ways to simplify the calculations such that adding visualization would take away only a very small fraction of the supercomputer time allocated to the scientist.

We have realized in-situ visualization for a terascale earthquake simulation [9]. This work also won the HPC Analytics Challenges of the SC 2006 Conference [10] because of the scalability and interactive volume visualization we demonstrated. Over a wide-area network, we were able to interactively change view angles, adjust sampling steps, edit color and opacity transfer function, and zoom in and out for visually monitoring the simulation running on 2048 processors of a supercomputer at the Pittsburgh Supercomputing Center. We were able to achieve high parallel efficiency exactly because we made the visualization calculations, i.e., direct volume rendering, to use the data structures used by simulation code, which removes the need to reorganize the simulation output and replicate data. Rendering is done in-situ using the same data partitioning made by the simulation, and thus no data movement is needed among processors. Similar to the traditional parallel volume rendering algorithms, our parallel in-situ rendering pipeline consists of two stages: parallel rendering and parallel image compositing. In the rendering stage, each processor renders its local data using software ray-casting. Note that this stage may not be balanced given a set of visualization parameters and the transfer function used. In the image compositing stage, a new algorithm is designed to build a communication schedule in parallel on the fly. The basic idea is to balance the overall visualization workload by carefully distributing the compositing calculations. This is possible because parallel image compositing uses only the data generated by the rendering stage and is thus completely independent of the simulation.

For implementation of in-situ visualization, no significant change is needed for the earthquake simulation code for the integration. The only requirement for the simulation is to provide APIs for the access of the simulation internal data structure, which does not require much effort in practice. Furthermore, because all the access is read operation, the simulation context is not affected by the visualization calculations. The advantage of our approach is obvious. Scientists do not need to change their code to incorporate in-situ visualization. They only need to provide an interface for the visualization code to access their data, as everything else is taken care of by the visualization part. This approach is certainly the most acceptable by scientists.

Conclusion

We are not too far from peta- and exa-scale computing. Will we have the adequate tools for possibly extracting meaning from the data sets generated by such extreme-scale simulations? The investment made by the DOE SciDAC program in ultra-scale visualization [2] is timely and ensures that challenges will be addressed. In this article, we point out the grand challenges facing extreme-scale data analysis and visualization, and present several key technologies for gaining insights in ultra-scale simulations. While we have had some success in deploying some of these technologies, further research and experimental studies are still needed to make these new technologies benefit the scientific supercomputing community at large.

Acknowledgments This work is supported in part by the DOE SciDAC program and NSF ITR program. The images displayed in this article were made by members of the Ultravis Institute and the VIDI research group at the University of California at Davis. The supernova data set was provided by Dr. John Blondin at the North Carolina State University. The turbulent combustion data set was provided by Dr. Jackie Chen at the Sandia National Laboratories.

References

1. Institute for Ultrascale Visualization, DOE SciDAC. http://ultravis.org.

2. K.-L. Ma, R. Ross, J. Huang, G. Humphreys, N. Max, K. Moreland, J. D. Owens, H.-W. Shen. “Ultra-scale visualization: research and education,” Journal of Physics, Volume 78. (also Proceedings of SciDAC 2007 Conference, 24-28 June, 2007, Boston, Massachusetts)

3. Scientific Discovery through Advanced Computing, Office of Science, Department of Energy. http://www.scidac.gov.

4. H. Yu, C. Wang, K.-L. Ma. “Parallel Hierarchical Visualization of Large 3D Time-Varying Vector Fields,” in Proceedings of the ACM/IEEE Supercomputing 2007 Conference (SC ’07).

5. K.-L. Ma. “Visualizing Visualizations: User Interfaces for Managing and Exploring Scientific Visualization Data,” IEEE Computer Graphics and Applications, Volume 20, Number 5, 2000, pp. 16-19.

6. H. Akiba and K.-L. Ma. “A Tri-Space Visualization Interface for Analyzing Time-Varying Multivariate Volume Data,” In Proceedings of Eurographics/IEEE VGTC Symposium on Visualization, 2007, pp. 115-122.

7. C. Jones, K.-L. ma, A. Sanderson, L. R. Myers Jr. “Visual Interrogation of Gyrokinetic Particle Simulation,” Journal of Physics, Volume 78. (also Proceedings of SciDAC 2007 Conference, 24-28 June, 2007, Boston, Massachusetts)

8. K.-L. Ma. “Machine Learning to Boost the Next Generation of Visualization Technology,” IEEE Computer Graphics and Applications, Volume 27, Number 5, 2007, pp. 6-9.

9. T. Tu, H. Yu, L. Ramirez-Guzman, J. Bielak, O. Ghattas, K.-L. Ma, D. R. O’Hallaron. “From mesh generation to scientific visualization: an end-to-end approach to parallel supercomputing,” in Proceedings of ACM/IEEE Supercomputing 2006 Conference (SC ’06).

10. H. Yu, T. Tu, J. Bielak, O. Ghattas, J. C. Lopez, K.-L. Ma, D. R. O’Hallaron, L. Ramirezguzman, N. Stone, R. Taborda-Rios, J. Urbanic. “Remote runtime steering of integrated terascale simulation and visualization,” HPC Analytics Challenge, ACM/IEEE Supercomputing 2006 Conference (SC ’06).