In Situ Analysis and Visualization of Extreme-Scale Particle Simulations

Soumya Dutta[1], Dan Lipsa[2], Terece L. Turton[1], Berk Geveci[2], and James Ahrens[1]

[1] Los Alamos National Laboratory, USA
{sdutta,tlturton,ahrens}@lanl.gov
[2] Kitware Inc., USA
{berk.geveci,dan.lipsa}@kitware.com

Abstract. In situ analysis has emerged as a dominant paradigm for performing scalable visual analysis of extreme-scale computational simulation data. Compared to the traditional post hoc analysis pipeline where data is first stored into disks and then analyzed offline, in situ analysis processes data at the time its generation in the supercomputers so that the slow and expensive disk I/O is minimized. In this work, we present a new in situ visual analysis pipeline for the extreme-scale multiphase flow simulation MFiX-Exa and demonstrate how the pipeline can be used to process large particle fields in situ and produce informative visualizations of the data features. We deploy our analysis pipeline on Oak Ridge's Summit supercomputer to study its in situ applicability and usefulness.

Keywords: In Situ Analysis · Visualization · Feature Detection · High Performance Computing · Computational Science · Particle Data.

1 Introduction

With increasing computing capabilities, scientific simulations are now producing very large-scale spatio-temporal data sets, containing intricate features that need to be analyzed and visualized efficiently to further scientific discoveries. While domain scientists focus on making their simulations more accurate and efficient, they need flexible and scalable analysis capabilities to study their data. Many research studies have shown that the traditional post hoc analysis paradigm is no longer scalable as handling, managing, and analysis of extreme-scale data sets will be prohibitive [2,3,9]. This is primarily due to slow disk I/O speed compared to the rate at which data is produced coupled with the post hoc processing needs of extreme-scale data [6,11,28]. As a result, only a sparse set of time steps of the simulation can typically be stored on the disk for future analysis.

In situ analysis addresses this problem by deploying visualization algorithms directly with the simulation, i.e., while the data is produced. This powerful strategy has been shown very effective in producing high-quality visualization artifacts of the simulation data that otherwise would be significantly time-consuming to generate [3,16,19,25]. However, due to the complexity of the scientific data sets and the domain-specific features within them, it is often less effective if only the raw simulation data is visualized. An alternative approach is to first apply an appropriate data analysis algorithm in situ and then produce visual artifacts of the derived data that highlight the complex data features more clearly compared to the raw data. The informative visual artifacts generated from the derived data can be used to explore the evolution of the data features during the simulation run and application scientists can verify and/or validate various scientific hypotheses.

In this work, we present the first ParaView Catalyst-based [14] in situ analysis pipeline for the very large-scale multiphase simulation MFiX-Exa [20,21]. MFiX-Exa is currently being developed at the National Energy Technology Laboratory (NETL) and is on its way to harness the upcoming exascale supercomputers to further scientific discoveries [12] as part of the Exascale Computing Project (ECP) [13].

The primary focus of the MFiX-Exa simulation is to study the working principles of complex and large-scale chemical looping reactors. To comprehend the physics behind such rectors, MFiX-Exa developers study simulation cases where millions of particles interact with each other inside a fluidized bed. The formation of bubbles (void regions that are characterized by low particle density) in these fluidized beds is a prime phenomenon of interest for domain scientists as the evolution and characteristics of these bubbles can indicate the overall stability of the reactor. To study the bubble dynamics, the simulation needs to run for a sufficiently long duration, resulting in an extreme-scale spatio-temporal particle data set with tens of thousands of time steps. Post hoc analysis of such time-varying data is significantly time-consuming and so the experts typically run small-scale test cases as they currently lack the capability to explore full-fledged three-dimensional bubble dynamics.

Our in situ analysis pipeline addresses this issue and enables the domain experts to perform in situ analysis and visualization of their simulation data without needing to store the large-scale particle fields. We show that the Catalyst-based in situ pipeline can generate informative visualizations of the particle data and also can be used to apply data analysis algorithms so that the final visual artifacts show the bubble features clearly. Since for these large-scale particle simulations, it is impossible to see the bubbles clearly from the raw particle data, we first compute the particle density fields in situ and then produce volume-rendered images of the particle density field that clearly show the bubbles in the simulation data. We contribute a new VTK-based particle density estimation filter that users can use in their analysis pipeline to compute scalar particle density fields from particle data. Our in situ pipeline also allows storing of the in situ generated particle density fields which are significantly smaller compared to the original raw particles fields. These particle density fields can be used post hoc for further in-depth study of bubble dynamics.

2 Related Works

The need for in situ data analysis and visualization has grown significantly in recent years to address the problems arising from slow disk I/O. The visualization community has developed several high-quality libraries to enable in situ analysis and rendering of data. One of the early attempts of in situ visualization was made by Haimes [15] to visualize large unsteady data sets. For performing *in situ* analysis and visualization, Fabian et al. developed the Catalyst library [14], which uses functionalities of ParaView during in situ run. Catalyst-based in situ analysis has been widely adopted in the scientific visualization community [5,8,29]. Similarly, run-time visualization with LibSim using VisIt was proposed by Whitlock et al. [27]. In another work, Lofstead et al. added ADIOS as an in situ visualization framework [18]. Vishwanath et al. enriched simulation time data analysis by proposing GLEAN [26]. A more recent fly-weight in situ analysis infrastructure has been developed by Larsen et al. [17]. An open-source in situ visualization infrastructure called SENSEI is also being developed that allows interfacing between different in situ infrastructures with the simulation code [24]. For a more comprehensive guide to the various types of existing infrastructures, readers are referred to the following state-of-the-art report [6]. To gain detailed knowledge about the in situ relevant terminologies and standards, developed by the visualization community, please refer to [10].

3 ParaView Catalyst-based In Situ Visual Analysis Workflow

This section describes the analysis pipeline that we have developed to enable in situ analysis and visualization for the MFiX-Exa simulation. Starting with an overview of the Catalyst adapter, we describe its access to MFiX-Exa data in the in situ environment and then discuss the visualization methods and algorithms that are used to generate effective visual artifacts for MFiX-Exa data.

3.1 In Situ Catalyst Adapter Design

The first step to build an in situ analysis environment for a simulation code is to design an efficient in situ adapter that can tap into the simulation memory while the data is being generated. Making the data accessible in situ is necessary to move post hoc analyses into the simulation while it is running. Since different simulation codes have different data layouts in memory, designing a general in situ adapter can be a challenging task.

The MFiX-Exa simulation uses the AMReX [4,30] library as its internal software framework to store and process the simulated particle data. AMReX is a software framework that facilitates the development of scalable, block-based, massively parallel, and adaptive mesh refinement (AMR) applications. In this work, we have developed a ParaView (version 5.9.1) Catalyst-based (version 1) in situ adapter program that can read the particle data structures of AMReX (more

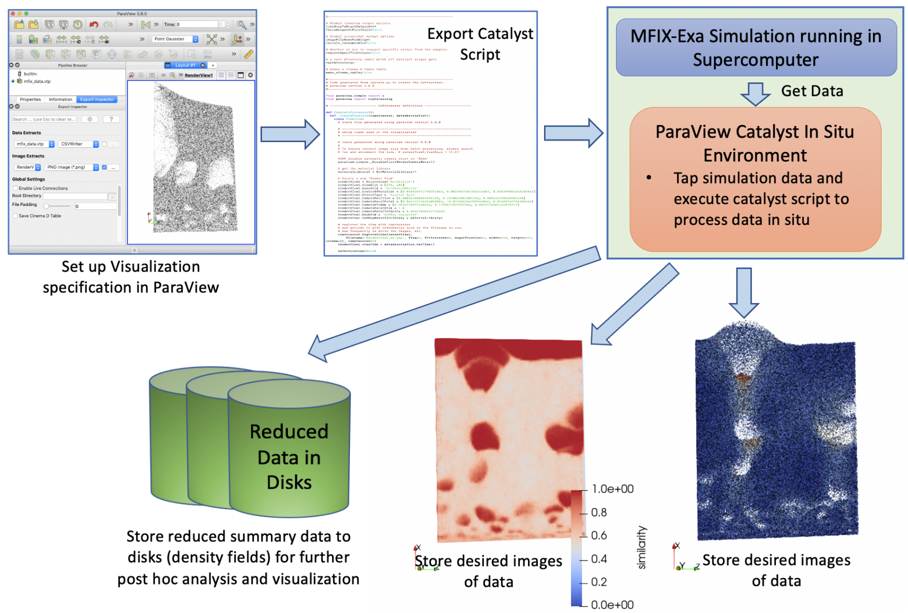


Fig. 1. Schematic of the in situ analysis and visualization pipeline showing different types of visualization and data artifact outputs.

specifically access the particle data from AMReX's ParticleContainer Class) and then convert it into a VTK-based [23] data structure and provide a handle to the user in the in situ environment. To convert AMReX -based particle data into a VTK-based data structure, currently, the data is copied out. In the future, we will move to VTK's zero-copy capabilities to pass the pointers directly. Algorithm developers can directly use this VTK data in their program to analyze or produce visualizations of the data in situ. Since MFiX-Exa produces particle data, the simulation data is represented as VTK Polydata in the in situ environment. The in situ adapter also makes the simulation's MPI communicator accessible in the in situ environment so that users can deploy data processing and visualization algorithms that require distributed communication.

One of the advantages of the Catalyst adapter is that since this adapter is developed for the AMReX's particle container, it can be generalized and reused for performing in situ analysis for other simulations that use AMReX with minimal modification. Hence, even though the focus of this work is on the MFiX-Exa simulation, the in situ adapter and visualization techniques can be easily extendable to other particle-based simulations that use AMReX for data representation.

Figure 1 shows a schematic of the in situ analysis pipeline. Users can generate a Catalyst script that contains the visual analysis pipeline to be executed during the in situ run. This Python script is generated from the ParaView application

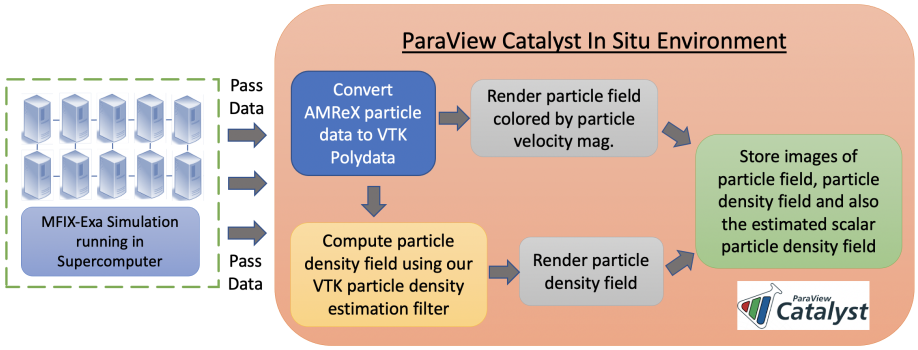
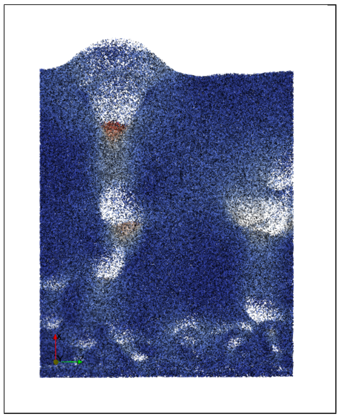


Fig. 2. Steps of in situ processing of the particle data for the proposed work.

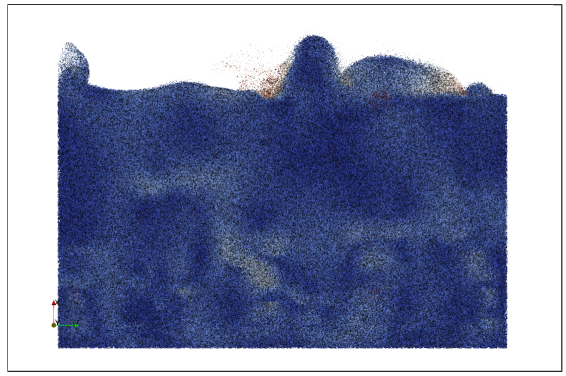
as shown. The script is deployed in situ using Catalyst's in situ infrastructure. During the in situ run, the in situ adapter makes the data available in this Python script as a VTK Polydata at each MPI process, and the user can process and analyze the data further. The pipeline uses MFiX-Exa's MPI communicator and using Catalyst's built-in fault tolerance capabilities, we ensure that even if our script is unable to process the data, the simulation does not crash. At each time step, MFiX-Exa calls a Catalyst routine and passes it data. The Catalyst routine calls the Python script that the user provides to do the analysis and visualization. So on the cluster node, we run the MFiX-Exa simulation, which periodically calls Catalyst. So the Python script is periodically called to do the visualization. In Fig. 2, we present the in situ analysis and visualization tasks that we have used in this work to explore the MFiX-Exa data set. We generate visualization outputs of the raw particle data where each particle is rendered as a sphere and colored by its velocity magnitude. The velocity magnitude is computed in situ using ParaView's Calculator function. Since one of the primary focuses of the application developers is to study the bubble features in the simulation, we also compute the particle density field and generate visualizations of this field that can show the bubbles more clearly compared to the raw particle visualization. To further analyze the particle density field and the bubble features, we also allow storing the particle density fields on disk. Note that, compared to the raw particle data, the size of this particle density scalar fields is significantly smaller and hence our method is also able to achieve sufficient data reduction. Using these reduced density fields, flexible bubble dynamics analysis can be done during post hoc analysis.

3.2 In Situ Particle Density Estimation for Effective Visualization of Data Features

Since the raw data format for MFiX-Exa is particle-based, we first add the capability to generate particle renderings at each time step. We also color each



(a) Particle Rendering for MFiX-Exa Case 1.



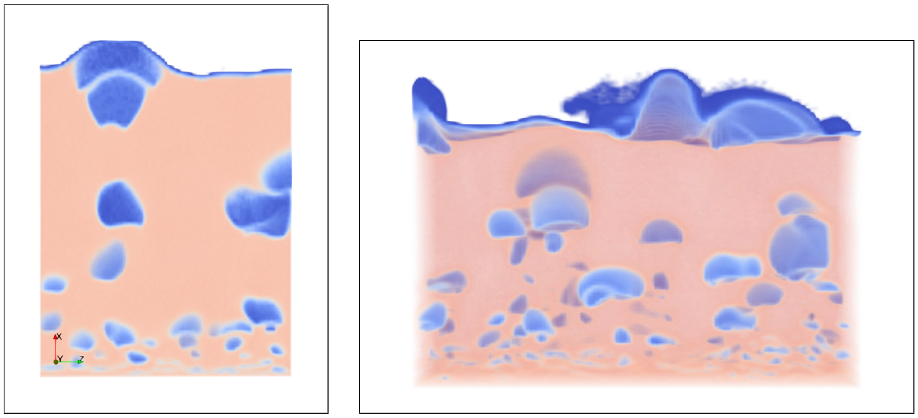
(b) Particle Rendering for MFiX-Exa Case 2.

Fig. 3. In situ generated visualization of raw particle fields for two different MFiX-Exa simulation test cases where the particles are colored by their velocity magnitude. Red particles indicate particles having higher velocity.

particle using its velocity magnitude so that the domain experts can glean additional information about the particle dynamics. In Fig. 3(a), we show the particle rendering of an MFiX-Exa simulation test case (MFiX-Exa Case 1), which contains around 4 million particles. The low-density particle regions, *bubbles*, can be seen in this figure. We also observe that particles underneath a bubble have high velocity. This visualization is similar to a post hoc visualization workflow. Potential issues with this visualization include that the actual bubble features are not seen and that smaller bubbles are difficult to visualize. These issues become much more severe as the number of particles increases in the simulation domain. In Fig. 3(b), we present particle rendering of a much larger MFiX-Exa simulation test case (MFiX-Exa Case 2), containing around 54 million particles. As can be seen, even when the size of each particle radius is quite small, we barely see any bubble feature in the data. It appears that this simulation does not have any bubbles produced. Thus, the raw particle visualizations are not suitable when the experts want to study the bubbles in their data.

To address the shortcomings of the raw particle-based in situ visualizations, we use a particle density field-based visualization that clearly shows the bubble features in the data set. The resultant visualizations are much more informative and can be used to study bubble dynamics. Density estimation is often regarded as a fundamental step necessary for sampling particle fields into a structured continuous representation [22].

We have used a spatial histogram-based technique to group particles into non-overlapping bins and then a density field is finally constructed. As the particles are distributed across multiple compute nodes, we compute the histogram in the same distributed setting. A local histogram is first constructed at each processing



(a) Particle density rendering for MFiX-Exa Case 1.

(b) Particle density rendering for MFiX-Exa Case 2.

Fig. 4. In situ generated visualization of particle density fields for two different MFiX-Exa simulation test cases where the bubble features (blue regions with low density), are clearly seen. Note the clear delineation of bubble features and the ability to see the small bubbles, even for the large number of particles in MFiX-Exa Case 2.

unit by binning the 3D locations of all particles available to each processor. A 3D histogram is required since we are binning particle locations to estimate spatial particle density. The number of bins and bin widths on each local processing unit are the same and are estimated from the global bounds of the particles. Finally, the partial histograms are combined to construct the global density histogram by using a parallel reduction operation overall processing units. Each bin in this global spatial histogram represents particle counts in a local spatial region. The global 3D histogram is mapped into a 3D regular grid-based scalar field where each 3D bin center is mapped to a voxel in the regular grid data and the particle count for that bin is assigned as the particle density value at that voxel. Specific details about this histogram-based density estimation can be found in [7] where this technique was evaluated offline. Using a spatial histogram-based approach to convert the particle data into a density field can be efficiently performed in situ, keeping the computational cost low during in situ processing. Note that other density estimation methods can be used here to estimate the particle density field. However, we believe that the histogram-based technique is generally suitable for distributed environments as the histograms can be computed via parallel reduction operation efficiently and give good results for MFiX-Exa data.

We have implemented the density estimation function in a VTK filter form so that it can be easily deployed from the Catalyst in situ script. The original code is implemented in C++ and is first integrated into VTK as an MPI-enabled parallel filter. Then we call the density estimation VTK filter from the Catalyst script. The input to the filter is the particle data and the output is a scalar

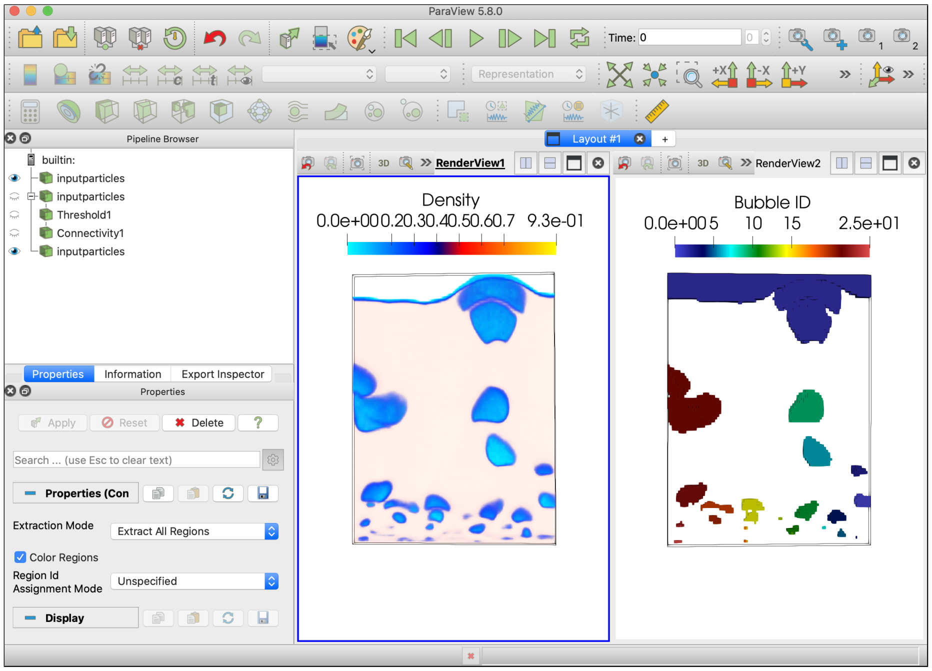


Fig. 5. Post hoc visual analysis of bubbles using in situ generated particle density fields. The left rendering window shows the density field and in the right rendering window, the bubbles are extracted using a low-density threshold value and then the connected component algorithm is applied to identify individual bubbles.

field in the form of VTK ImageData. Once this field is produced, we generate visualizations of this density field and also store the raw density field for further post hoc analysis.

In Fig. 4(a), we show the in situ rendering of the particle density field for a time step of MFiX-Exa simulation Case 1. The corresponding particle field is shown previously in Fig. 3(a). We can observe that the low-density regions in the density field, the blue regions, correspond to the bubbles in the data. The effectiveness of the density field-based visualization can be compared to the visualization of the raw particles as seen in Fig. 4(b) which shows the density field visualization for the MFiX-Exa Case 2. The corresponding particle field is depicted in Fig. 3(b). Comparing Fig. 3(b) and Fig. 4(b), one can observe that the density field shows the bubbles in the data that are hard to see from the particle-based visualization when the number of particles is large.

The in situ generated particle density fields can also be used to perform flexible post hoc bubble analysis. Since the size of the density fields is significantly smaller compared to the raw particle fields, they can be loaded into ParaView and analyzed and visualized interactively. In Fig. 5, we show one such demonstration where on the left rendering window, the density field is visualized using volume rendering. On the right window, the segmented bubble features are shown. Here, we first use a low-density value to threshold the density field and then apply the connectivity filter so that each connected segment is identified as an individual bubble feature.

4 Evaluation

We have tested the in situ pipeline on the Summit supercomputer [1], an IBM system located at the Oak Ridge Leadership Computing Facility (OLCF). Each compute node of Summit contains two IBM POWER9 processors, 512 GB of DDR4 memory, 1.6 TB of non-volatile memory, and six NVIDIA Tesla V100 GPUs. We performed an initial evaluation of our in situ pipeline by running the pipeline with two different test cases of MFiX-Exa. The first test case contains around 4 million particles, which we call MFiX-Exa Case 1, and the second case is a larger test case containing around 54 million particles. We denote the second test case as MFiX-Exa Case 2. For each of these cases, we performed particle rendering where the particles are colored with velocity magnitudes computed in situ and also volume rendering of the particle density field. The density field is first computed using a spatial histogram-based method as discussed before. In Table 1, we provide the computational timings taken by the simulation and the in situ methods. The renderings were done on GPUs and each MPI process was assigned with 1 GPU. As these timings reflect the total time for the catalyst script, they include the overhead due to data copying from AMReX to VTK data structure and the communication time. Since the simulation data evolves slowly over consecutive time steps and successive time steps are typically very similar, we performed in situ analysis at every 5^{th} time step. Note that, we are reporting the initial performance of our in situ pipeline and we plan to run our workflow on a much bigger case of MFiX-Exa, containing hundreds of millions of particles, to conduct a full-fledged performance study in the future and further optimize our code. We also plan to implement our density estimation filter as a VTKm filter so that we can execute the code with GPU acceleration in the upcoming exascale machines.

5 Conclusions

We have presented a ParaView Catalyst-based in situ analysis pipeline infrastructure for the ECP application MFiX-Exa. We demonstrate how the users can use our in situ pipeline to perform in situ analysis and produce various types of visualization artifacts. We believe that our in situ interface, which is able to read AMReX particle data structure, is an important capability for the domain scientists who can analyze and produce visualization of their data for extreme-scale simulation test cases with minimal effort to verify and validate their simulation and further improve it. In the future, we plan to deploy this in situ analysis pipeline in the upcoming exascale supercomputers to analyze

Table 1. In situ timings compared to the simulation timings for two different MFiX-Exa simulation test cases.

	Configuration	Avg. simulation time per time step (secs)	Avg. particle rendering time per time step (secs)	Avg. density estimation and rendering time per time step (secs)
MFiX-Exa Case 1 (∼4M particles)	256 MPI processes with 1 GPU per process	2.240	0.179	1.057
MFiX-Exa Case 2 (∼54M particles)	3072 MPI processes with 1 GPU per process	5.678	1.160	1.649

and visualize extreme-scale MFiX-Exa simulation data and also develop more sophisticated in situ bubble detection algorithms.

Acknowledgements The authors would like to thank the Department of Energy and Los Alamos National Laboratory for the funding and support in carrying out this research. This research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration. This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725. We thank our many ECP collaborators especially Jordan Musser, Ann Almgren, and Patrick O'Leary. This research is released under LA-UR-22-21278.

References

1. Summit supercomputer. https://docs.olcf.ornl.gov/systems/summit_user_guide.html (accessed May 24, 2022)
2. Ahern, S., Shoshani, A., Ma, K., Choudhary, A.: Scientific discovery at the exascale. Report from the DOE ASCR 2011 Workshop on Exascale Data Management. Analysis, and Visualization (February) (2011)
3. Ahrens, J., Jourdain, S., OLeary, P., Patchett, J., Rogers, D.H., Petersen, M.: An image-based approach to extreme scale in situ visualization and analysis. In: SC14: International Conference for High Performance Computing, Networking, Storage and Analysis. pp. 424–434 (2014). https://doi.org/10.1109/SC.2014.40
4. AMReX: A software framework for massively parallel, block-structured adaptive mesh refinement (AMR) applications (2021 (accessed April 7, 2021)), https://amrex-codes.github.io/amrex/index.html
5. Atzori, M., Köpp, W., Chien, S.W.D., Massaro, D., Mallor, F., Peplinski, A., Rezaei, M., Jansson, N., Markidis, S., Vinuesa, R., Laure, E., Schlatter, P., Weinkauf, T.: In-situ visualization of large-scale turbulence simulations in nek5000 with paraview catalyst . https://doi.org/10.1007/s11227-021-03990-3
6. Bauer, A.C., Abbasi, H., Ahrens, J., Childs, H., Geveci, B., Klasky, S., Moreland, K., O'Leary, P., Vishwanath, V., Whitlock, B., Bethel, E.W.: In situ methods,

infrastructures, and applications on high performance computing platforms. Computer Graphics Forum **35**(3), 577–597 (2016). https://doi.org/10.1111/cgf.12930

7. Biswas, A., Ahrens, J.P., Dutta, S., Musser, J.M., Almgren, A.S., Turton, T.L.: Feature analysis, tracking, and data reduction: An application to multiphase reactor simulation mfix-exa for in-situ use case. Computing in Science & Engineering **23**(01), 75–82 (jan 2021). https://doi.org/10.1109/MCSE.2020.3016927
8. Camata, J.J., Silva, V., Valduriez, P., Mattoso, M., Coutinho, A.L.: In situ visualization and data analysis for turbidity currents simulation. Computers & Geosciences **110**, 23–31 (2018). https://doi.org/https://doi.org/10.1016/j.cageo.2017.09.013
9. Childs, H.: Data exploration at the exascale. Supercomputing frontiers and innovations **2**(3) (2015), http://superfri.org/superfri/article/view/78
10. Childs, H., Ahern, S.D., Ahrens, J., Bauer, A.C., Bennett, J., Bethel, E.W., Bremer, P.T., Brugger, E., Cottam, J., Dorier, M., Dutta, S., Favre, J.M., Fogal, T., Frey, S., Garth, C., Geveci, B., Godoy, W.F., Hansen, C.D., Harrison, C., Hentschel, B., Insley, J., Johnson, C.R., Klasky, S., Knoll, A., Kress, J., Larsen, M., Lofstead, J., Ma, K.L., Malakar, P., Meredith, J., Moreland, K., Navrátil, P., O'Leary, P., Parashar, M., Pascucci, V., Patchett, J., Peterka, T., Petruzza, S., Podhorszki, N., Pugmire, D., Rasquin, M., Rizzi, S., Rogers, D.H., Sane, S., Sauer, F., Sisneros, R., Shen, H.W., Usher, W., Vickery, R., Vishwanath, V., Wald, I., Wang, R., Weber, G.H., Whitlock, B., Wolf, M., Yu, H., Ziegeler, S.B.: A terminology for in situ visualization and analysis systems. The International Journal of High Performance Computing Applications **34**(6), 676–691 (2020). https://doi.org/10.1177/1094342020935991
11. Dutta, S., Chen, C., Heinlein, G., Shen, H.W., Chen, J.: In situ distribution guided analysis and visualization of transonic jet engine simulations. IEEE Transactions on Visualization and Computer Graphics **23**(1), 811–820 (2017)
12. Optimizing a new technology to reduce power plant carbon dioxide emissions. https://www.exascaleproject.org/optimizing-a-new-technology-to-reduce-power-plant-carbon-dioxide-emissions/ (2022 (accessed Feb 3, 2022))
13. Exascale Computing Project (2022 (accessed Feb 12, 2022)), https://www.exascaleproject.org/
14. Fabian, N., Moreland, K., Thompson, D., Bauer, A.C., Marion, P., Gevecik, B., Rasquin, M., Jansen, K.E.: The ParaView coprocessing library: A scalable, general purpose in situ visualization library. In: 2011 IEEE Symposium on Large Data Analysis and Visualization (LDAV). pp. 89–96 (2011). https://doi.org/10.1109/LDAV.2011.6092322
15. Haimes, R.: pv3: A distributed system for large-scale unsteady cfd visualization. In: AIAA paper. pp. 94–0321 (1994)
16. He, W., Wang, J., Guo, H., Wang, K.C., Shen, H.W., Raj, M., Nashed, Y.S.G., Peterka, T.: Insitunet: Deep image synthesis for parameter space exploration of ensemble simulations. IEEE Transactions on Visualization and Computer Graphics **26**(1), 23–33 (2020). https://doi.org/10.1109/TVCG.2019.2934312
17. Larsen, M., Ahrens, J., Ayachit, U., Brugger, E., Childs, H., Geveci, B., Harrison, C.: The alpine in situ infrastructure: Ascending from the ashes of strawman. In: Proceedings of the In Situ Infrastructures on Enabling Extreme-Scale Analysis and Visualization. p. 42–46. ISAV'17, Association for Computing Machinery, New York, NY, USA (2017). https://doi.org/10.1145/3144769.3144778

18. Lofstead, J.F., Klasky, S., Schwan, K., Podhorszki, N., Jin, C.: Flexible IO and Integration for Scientific Codes Through the Adaptable IO System (ADIOS). In: Proceedings of the 6th International Workshop on Challenges of Large Applications in Distributed Environments. pp. 15–24. CLADE '08, ACM (2008). https://doi.org/10.1145/1383529.1383533
19. Lukasczyk, J., Garth, C., Larsen, M., Engelke, W., Hotz, I., Rogers, D., Ahrens, J., Maciejewski, R.: Cinema darkroom: A deferred rendering framework for large-scale datasets. In: 2020 IEEE 10th Symposium on Large Data Analysis and Visualization (LDAV). pp. 37–41 (2020). https://doi.org/10.1109/LDAV51489.2020.00011
20. MFIX-Exa (2022 (accessed Feb 3, 2022)), https://amrex-codes.github.io/MFIX-Exa/docs_html/
21. Musser, J., Almgren, A.S., Fullmer, W.D., Antepara, O., Bell, J.B., Blaschke, J., Gott, K., Myers, A., Porcu, R., Rangarajan, D., Rosso, M., Zhang, W., Syamlal, M.: MFIX-Exa: A path toward exascale CFD-DEM simulations. International Journal of High Performance Computing Applications (2021). https://doi.org/10.1177/10943420211009293
22. Peterka, T., Croubois, H., Li, N., Rangel, S., Cappello, F.: Self-Adaptive Density Estimation of Particle Data. SIAM Journal on Scientific Computing SISC Special Edition on CSE'15: Software and Big Data (2016)
23. Schroeder, W., Martin, K., Lorensen, B.: The Visualization Toolkit: An Object Oriented Approach to 3D Graphics. Kitware Inc., fourth edn. (2004), iSBN 1-930934-19-X
24. SENSEI:Scalable in situ analysis and visualization (2021 (accessed Feb 12, 2022)), https://sensei-insitu.org/
25. Tikhonova, A., Correa, C., Ma, K.L.: Explorable images for visualizing volume data. In: 2010 IEEE Pacific Visualization Symposium (PacificVis). pp. 177–184 (2010). https://doi.org/10.1109/PACIFICVIS.2010.5429595
26. Vishwanath, V., Hereld, M., Papka, M.E.: Toward simulation-time data analysis and i/o acceleration on leadership-class systems. In: 2011 IEEE Symposium on Large Data Analysis and Visualization (LDAV). pp. 9–14 (2011). https://doi.org/10.1109/LDAV.2011.6092178
27. Whitlock, B., Favre, J.M., Meredith, J.S.: Parallel in situ coupling of simulation with a fully featured visualization system. In: Proceedings of the 11th Eurographics Conference on Parallel Graphics and Visualization. pp. 101–109. EGPGV '11, Eurographics Association (2011). https://doi.org/10.2312/EGPGV/EGPGV11/101-109
28. Woodring, J., Petersen, M., Schmeiβer, A., Patchett, J., Ahrens, J., Hagen, H.: In situ eddy analysis in a high-resolution ocean climate model. IEEE Trans. on Vis. and Comp. Graphics **22**(1), 857–866 (2016). https://doi.org/10.1109/TVCG.2015.2467411
29. Yi, H., Rasquin, M., Fang, J., Bolotnov, I.A.: In-situ visualization and computational steering for large-scale simulation of turbulent flows in complex geometries. In: 2014 IEEE International Conference on Big Data (Big Data). pp. 567–572 (2014). https://doi.org/10.1109/BigData.2014.7004275
30. Zhang, W., Myers, A., Gott, K., Almgren, A., Bell, J.: Amrex: Block-structured adaptive mesh refinement for multiphysics applications. The International Journal of High Performance Computing Applications **35**(6), 508–526 (2021). https://doi.org/10.1177/10943420211022811