Binning Based Data Reduction for Vector Field Data of a Particle-In-Cell Fusion Simulation

James Kress[1,2], Jong Choi[1], Scott Klasky[1], Michael Churchill[3], Hank Childs[2], and David Pugmire[1]

[1] Oak Ridge National Laboratory, Oak Ridge TN 37831, USA
[2] University of Oregon, Eugene, OR 97403, USA
[3] Princeton Plasma Physics Laboratory, Princeton NJ 08543, USA.

Abstract. With this work, we explore the feasibility of using in situ data binning techniques to achieve significant data reductions for particle data, and study the associated errors for several post-hoc analysis techniques. We perform an application study in collaboration with fusion simulation scientists on data sets up to 489 GB per time step. We consider multiple ways to carry out the binning, and determine which techniques work the best for this simulation. With the best techniques we demonstrate reduction factors as large as 109x with low error percentage.

Keywords: in situ · data reduction · visualization

1 Introduction

As leading-edge supercomputers get increasingly powerful, scientific simulations running on these machines are generating ever larger volumes of data. However, the increasing cost of data movement, in particular moving data to disk, limits the ability to process, analyze, and fully comprehend simulation results [1]. Specifically, while I/O bandwidths regularly increase with each new supercomputer, they are well below corresponding increases in computational ability and data generated. Further, this trend is predicted to persist for the foreseeable future.

Given this reality, many large-scale simulation codes are attempting to bypass the I/O bottleneck by using in situ visualization and analysis, i.e., processing simulation data when it is generated. In situ processing can be difficult however, as it is generally not known a priori all of the analysis tasks that will be done on simulation output. One method supporting unanticipated analysis, or exploration, is to produce a reduced representation which can be written to disk. Examples of data reductions include compression techniques (both lossy and lossless), reduced precision representations, data subsets and extracts, spatial resampling, and summary data. Here care must be taken in order to preserve the information content in the data while at the same time, minimizing the size.

With this research, we consider the model where in situ processing is used to produce an information preserving reduced data representation of the simulation

data that can be saved to disk. This reduced representation data can then be used for post processing analysis, visualization, and exploration. Additionally, we evaluate the errors introduced when doing a variety of analysis and visualization operations on these reduced data representations.

In this paper we apply this in situ model to XGC1 [3], a plasma fusion simulation code that runs at scale on supercomputers. XGC1 is a gyrokinetic particle-in-cell code that is used for modeling the physics of plasmas in fusion tokamak devices. XGC1 uses a large number of particles to represent the kinetic behavior of the plasma. Summarizations of these particles are imposed upon an unstructured grid, which is small enough that it can be saved to disk. Currently, the particles, which are the representation of the plasma, are too large to save out at each time step.

In this paper we study and apply multiple data binning techniques to these particles to extract a vector field representation from the individual particle trajectories. The resulting vector field, which is orders of magnitude smaller than the entire set of particles, can be easily saved to disk and used for post processing analysis and visualization. To evaluate this method, we use streamlines and Poincaré analysis using particle advection on the vector field representations for the full and reduced representation data, and examine the errors associated with different binning techniques. One of the best binning techniques operates on 500 GB of data per time step, achieving a reduction of 109x with an average error of 1.15% in under 140 seconds.

In the remainder of this paper, we discuss related works in Section 2, describe the binning of fusion particle data in Section 3, describe our in situ workflow in Section 4, and present the results from our analyses in Section 5.

2 Related Work

We present the related work for information preserving in situ data reduction and visualization of particles in three sub-categories: (1) in situ visualization, (2) XGC1 visualization, and (3) large-scale particle visualizations.

2.1 In Situ Visualization

Visualization algorithms are particularly sensitive to I/O bandwidth [4, 5], causing the community to turn to in situ techniques to alleviate this growing problem. There has been significant work and successes with the in situ visualization paradigm. For instance, ParaView Catalyst coprocessing[6] and VisIt LibSim [22] are frameworks that are tightly-coupled to the simulation, *i.e*, the visualization runs at scale with the simulation. Alternatively, visualization and analytics can be performed during the transport of the simulation data to the I/O layer. Three examples of this loosely coupled approach are Nessie [16], GLEAN [21], and ADIOS[11]. For a more thorough overview of the three loosely-coupled in situ visualization frameworks, we refer the reader to [2].

2.2 XGC1 Visualization

Early work on production visualization for XGC1 mainly focused on addressing the immediate data needs of scientists during the course of a simulation run. One example of this was an online dashboard that was developed for XGC1 simulation monitoring called eSimon [20]. This dashboard was launched in conjunction with each simulation run, and was responsible for performing common visualization and analysis tasks in XGC1.

More recent work has focused on expanding the visualization capabilities and opportunities for XGC1 through the utilization of in situ methods. For example, they utilized the features of ADIOS and EAVL [17], and demonstrated the effectiveness of loosely coupled in situ visualization for large scale simulation codes using a workflow with dedicated data staging nodes. In that work, they focused on the performance, scalability, and ease of use of visualization plugins that were used on the output of the XGC1 simulation code. One component of this study looked at optimizing the parallel rendering pipeline in situ, and gave insight into getting high performing renderings in continuing studies.

Recent research with XGC1 has cataloged their common visualization and analysis tasks, analyzed their workflow, and captured common data sizes produced by the simulation [8], in an effort to prepare for their challenges at exascale. Further research has also done preliminary work at identifying areas of interest and some of the challenges associated with information preserving data reductions within XGC1 [7]. This work points out how data reductions must be done carefully in order to preserve the integrity of the underlying data when post-hoc analysis will need to generate derived quantities from reduced data. These derivations can end up with completely inaccurate information if data reductions are not done carefully.

2.3 Large-scale Particle Visualization

There have been several works in recent years that focus on visualizing large-scale particle data sets. The first set of works that will be described tackle this problem as a post-hoc task. The first work looked at the visualization of trillion particle data sets, and utilized a multi-scale rendering architecture which enabled hierarchical views of the data [19]. This approach enabled interactive speeds for users exploring and querying the dataset. A second example utilized bandlimited OLAP cubes which were based on kernel density plots [18]. This approach created an artifact free visualization at interactive speeds. The defining characteristics of these two techniques however, are that they are post-hoc, requiring the full particle data set to be saved to disk, so they did not address our need for in situ data reduction.

Another approach is to consider particle visualization as an in situ task. One approach in this area was to use spatially organized 2D histograms for viewing large data set in situ [15]. This work looked at several different application domains and was able to demonstrate that features within the data could be visualized by creating many different histograms for different regions of the data.

This approach is similar to ours in that it creates a velocity representation. That said, it does not describe data reduction results, or if the histograms could be used for more than just visualization, such as post-hoc analysis. A second approach performs two different steps in situ to first create probability distribution functions and then a second step to specially reorganize particles for faster post-hoc access [23]. The difference between this work and our current work is that they are still saving subsets of particles. We are utilizing all of the simulation's particles in creating our data representations, which reduces the data that we have to save. Further, this technique currently lacks an analysis of the errors introduced through their subsampling process on post-hoc visualization and analysis routines.

All of these techniques are interesting reduction operators. That said, our domain scientist collaborators are interested in binning since the code already uses a finite element mesh internally for some calculations, and so our study focused on tradeoffs within binning. Comparisons with other techniques is an interesting topic for future work

3 Binning of Fusion Data

Fusion scientists for the XGC1 fusion simulation are interested in looking at particle data coming from the simulation at a finer temporal fidelity than is currency possible in a post-hoc workflow. In the post-hoc workflow particle data is only saved out at each of the simulation checkpoints, which only occur between every 100 and 1,000 time steps [8], leaving a large temporal gap. This gap reduces the fidelity of their analysis techniques and leaves the very real possibility that something interesting in the data will be lost due to the turbulent nature of the data. Historically however, this temporal gap was necessary due to the shear size of the particle data at each time step, which is up to 20 TB.

To address the issue of the large temporal gap, some sort of in situ data reduction technique is required for this workflow due to both the size of the data generated per time step, and the short amount of time between individual time steps. To overcome this issue we created a workflow for the in situ application of a data binning technique to the particle data from the simulation. This binning technique generates vectors in each of the bins that represents the speed and direction of particles within that cell's region. Furthermore, this technique allows the scientists to tune the binning code for both specific accuracy and speed requirements by modifying the size of the binning grid, and the number of particles used to extract the vector field.

3.1 Generating the Binned Data

In creating this vector field, there are two different control points for tuning the accuracy of the vector representation: (1) the size of the binning mesh, and (2) the number of particles sampled from the simulation to extract the vector field. In this paper we looked at four different mesh sizes and three different particle sampling counts for each of the four mesh representations.

Mesh Representations An unstructured mesh representation was used for this study. Figure 1 shows an example of the representation. The reduced representations of the unstructured mesh were generated by performing a quadratic clustering decimation on the unstructured mesh from the XGC1 simulation itself, and setting the number of divisions argument for the clustering to be equal for height and width.

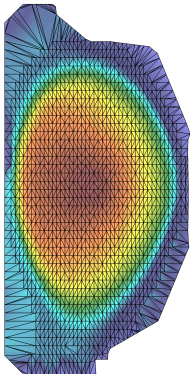

Fig. 1. Example of the unstructured mesh used. The unstructured mesh is a decimated version of the simulation mesh.

The five different mesh sizes used are shown in Table 1. In this study we use the original mesh as our *Ground Truth Mesh* comparator against the four reduced representations. We chose these reduced mesh sizes in order to have a spectrum of reduced mesh cell sizes which reduces the amount of work required to generate the vector representations, saving time during the in situ calculation.

Table 1. A Breakdown of the four reduced mesh sizes used in the study compared to the original simulation mesh. The reduced meshes were generated using quadratic clustering. The dimensions used by quadratic clustering are listed in the left-hand column.

		Number of Cells	Percent Reduction
Unstructured Mesh	**45x45**	2,903	98.6%
	90x90	11,154	94.7%
	200x200	52,707	74.9%
	450x450	186,609	1.1%
	Original	209,576	0.0%

Sampling Particles Due to the size of the particle data generated by the simulation at each time step, it was important to perform a subsample operation on the simulation particle data to reduce the number of particles transfered from the simulation to our binning routines. We used three different particle counts in our tests in order to get a sampling of the spectrum in terms of vector representation accuracy and speed. The three particle counts we used in our subselection operation were: 3 million, 30 million, and 300 million. Each of these values represents an order of magnitude reduction from the total particle count at each simulation time step, which is 3.3 billion.

The particle subsampling was performed by uniformly selecting particles from the original XGC1 mesh. Table 2 shows a breakdown of the statistics for each of these three particle selections.

Table 2. A breakdown of each of the different factors of merit for the three reduced particle counts we tested. The final output of our method is the "Binned Data" and its size is always the same, as the binning grid was fixed.

Total Particles	Particle Reduction Amount	Reduced Particle Size (GB)	Particle Subselection Time (sec)	Binning Time (sec)	Binned Data Size (MB)
3M	1092x	0.609	15	67	89
30M	109x	2.0	25	115	89
300M	11x	18.0	40	345	89

Two of the most important values from this table is the amount of data reduction, and the binning time. The amount of data reduction represents the data reduction size on disk of the subsampled particles versus the size of all of the particles on disk. The binning time is the amount of time that it took to create our binned data representation after the data subsampling took place. This number can be fine tuned during a simulation run to be lower or higher based on the vector accuracy needs, and the amount of time available to perform the binning transformation. Even when only performing a one order of magnitude reduction on the number of particles and subselecting 300 million particles it only took a maximum of 345 seconds to create the binned representation, which lies within the average time per time step for XGC1 at full scale, which lies between 5 to 10 minutes [8].

4 Experimental Overview

This section presents an overview description of the in situ workflow, explains the evaluation metrics for the vector data, and how the accuracy from those metrics was evaluated.

4.1 Workflow Description

The workflow consists of three primary elements: (1) the simulation code; (2) a data transfer system to move data from the simulation to the visualization nodes; and (3) an efficient parallel visualization library. The workflow is launched as three separate binaries, with resources for each partitioned as follows: (1) 1024 XGC1 nodes; (2) 12 Staging nodes; and (3) 32 visualization nodes. A diagram of the workflow is shown in Figure 2. This study was conducted on the Rhea

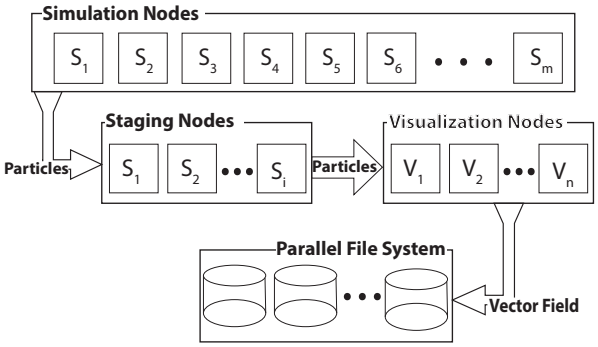

Fig. 2. Experimental setup and data flow between each of the three separate workflow components.

cluster at the Oak Ridge Leadership Computing Facility. Rhea consists of 512 nodes each with 128 GB of ram and dual Xeon processors with 16 cores each.

Each of the three elements of the workflow are described below, followed by a description of how the components of the workflow interact.

XGC1 Simulation Code The simulation code, XGC1, is a 5D gyrokinetic ion-electron particle in cell (PIC) code used to study fusion of magnetically confined burning plasmas. XGC1 is used in particular to study turbulence in the outer region of the plasma called the *edge*. The simulation proceeds by computing the interactions of a very large number of particles (ions and electrons) at each simulation time step.

For this study we are interested in using the particle data from the simulation. All of the simulation's particles can be very large, generally ranging from 400 GB to nearly 20 TB per time step. To get good particle velocity vector fields as the particles move around the tokamak, we were required to access the particles at each time step of the simulation.

ADIOS Data Staging The Adaptable I/O System (ADIOS) [9], is a componentization of the I/O layer that is accessible via a posix-style interface. ADIOS enables a loosely coupled paradigm for a clean interface and separation from XGC1 that provides ease of use, and fault tolerance. The ability to control the concurrency of the visualization tasks independent of the concurrency of XGC1 is important for ensuring good scalability on the visualization nodes

The particle subsampling operation occurred on the simulation side, with particles being uniformly subselected from the simulation's total particles before they were sent over the network with ADIOS. In our study we were only interested in examining ions from the simulation, so only they were selected and transmitted.

Visualization Library We designed our visualization routines based on an emerging community standard, VTK-m [14], which is a project building upon the success of three existing visualization frameworks, Dax [13], PISTON [10], and EAVL [12]. The VTK-m framework is targeted to emerging computational systems where parallelism and the use of accelerators are dramatically increasing, and memory per core is decreasing.

The vector field output from this portion of the workflow is written to disk for post hoc analysis. It is important to note that this step is a major data reduction. By performing the vector computation in situ, we are reducing the amount of data written to disk by between 5,494 and 11,926 times, see Table 2.

4.2 Evaluating Accuracy

We ran four different categories of tests to evaluate the accuracy of the particle vector representation. The four test types we ran were: (1) Poincaré contours, (2) Poincaré centers of mass, (3) streamlines, and (4) pathlines. Due to the turbulent nature of the XGC1 code, we further broke each of these tests down into three different regions, based on the turbulence in the code. The three regions we used are shown in Figure 3, and are (1) the core of the tokamak, (2) the core-edge interface, and (3) the edge of the tokamak. The core region is generally less turbulent than the core-edge interface, which is generally less turbulent than the edge. This breakdown allowed us to quantify what error we were getting in each of these three regions, which may be important if we are going to do an analysis that specifically cares about a given region of the plasma. Each of these tests was conducted on each of the different mesh sizes, with four different particle counts each.

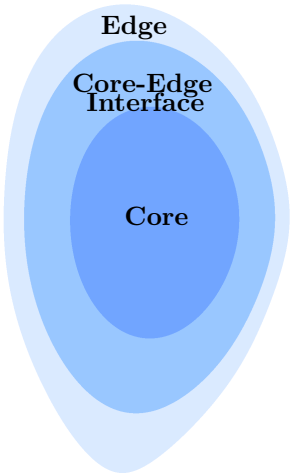

Fig. 3. The three test regions projected onto a slice of the XGC1 tokamak.

We compared the results from each of the tests against what the test produced when performed on *Ground Truth* data. We defined ground truth data to be the vector field that was created when all of the particles from the simulation were used to construct the vector field using the original unstructured XGC1 mesh. We take this vector representation to be the best possible representation when using a vector representation. An overview of each of these test types is given below.

Poincaré Tests We conducted the Poincaré tests based on the guidance of our fusion partners, as the motion of particles is something that they often study, so it has to be accurate. The Poincaré test on XGC1 data is particularly informative about the accuracy of a reduction method, because Poincaré plots can be used to track the magnetic field lines within the tokamak. The field lines are the

magnetic surfaces that particles generally follow around the toroidal shape of the tokamak. Therefore, if a reduced data set has a similar Poincaré outline and center of mass, the reduction method was successful at preserving the magnetic surfaces.

To conduct this test we advected 50 particles from random locations in the core, 25 particles from random locations in the core-edge interface, and 25 particles from random locations on the edge. We then collected the puncture locations for each test and created a contour line that connected all of the punctures to form a closed shape. Using these contours then we were able to calculate the area in the contour as well as the center of each contour.

For each test we compared the contour and contour center we got from the reduced vector representation against that of the ground truth representation. This comparison gave the difference in the area of the contour and difference in location of the centers of the contours. We then used this value as the error associated with that test.

Streamline/Pathline Tests The second series of tests we conducted were streamlines and pathlines. The streamline and pathline tests also utilized the three different test regions that the Poincaré tests used, giving three different errors for each individual test.

The streamline tests advected particles a random distance around the tokamak, returning the end position of each streamline. The end positions obtained from the ground truth data sets were compared against the reduced representations, giving a difference in end point location as our error metric. For this test we advected 2,000 particles from random locations in the core, 1,000 particles from random locations in the core-edge interface, and 1,000 particles from random locations on the edge.

The pathline tests worked the same as the streamline tests, except particles were advected over time. For this test we advected 200 particles from random locations in the core, 100 particles from random locations in the core-edge interface, and 100 particles from random locations on the edge.

5 Results

The results are organized into three subsections. Section 5.1 does an initial summary and general discussion of results. Section 5.2 examines the Poincaré test results in detail, and Section 5.3 examines the streamline and pathline results in detail.

5.1 Test Result Summary

Many different tests were run and analyzed with this work in order to find an optimal configuration for XGC1. In order to make this evaluation intuitive, we created a *lookup table* that can be used in selecting a binning configuration

based on the amount of error that is acceptable in each of the three regions of the tokamak.

Table 3 is our *lookup table* and shows the errors of all four of our tests together, presented as percent errors. This table is showing the average percentage error from each of our test configurations compared against *Ground Truth*. The table is colored according to the percent error, where the darkest blues represent low errors.

One of the most notable observations is the occasional large variation in the performance of the tests from the core to the core-edge interface to the edge of the plasma. This is an interesting feature of this reduced data representation, and can actually be exploited in certain circumstances depending on the location and type of the post-hoc visualization operation we want to run on the reduced data. For example, it is possible to use a 1092x reduction in particles on the 90x90 mesh, and maintain an error on the edge of the plasma for a Poincaré analysis under 2%.

Table 3. Percentage errors in the core, interface and edge regions, for each of the four evaluation metrics used. These percentages represent the average error for each test for each mesh size and particle count compared against the *Ground Truth* data. The color scale highlights the areas where good test configurations are found.

Reduced Mesh Size	Particle Reduction Factor	Poincare Contour Error			Poincare Center Error			Streamline End Point Error			Pathline End Point Error		
		Core	Interface	Edge	Core	Interface	Edge	Core	Interface	Edge	Core	Interface	edge
45x45 Mesh	1092x	3231.60%	11.80%	3.47%	1.30%	1.01%	1.00%	4.08%	4.23%	15.09%	5.82%	13.50%	59.18%
	109x	60.20%	7.70%	3.04%	1.06%	0.89%	1.22%	3.75%	3.31%	17.66%	7.47%	12.41%	38.07%
	11x	2132.95%	11.09%	3.09%	1.19%	1.00%	1.02%	3.85%	3.17%	20.27%	9.84%	5.72%	48.63%
	All Par.	10066.79%	8.58%	2.51%	1.38%	0.94%	0.87%	4.18%	3.18%	12.93%	10.48%	14.19%	60.32%
90x90 Mesh	1092x	81.10%	4.10%	1.88%	0.89%	0.57%	1.03%	1.88%	2.66%	14.38%	8.22%	12.28%	36.90%
	109x	30.61%	2.17%	2.02%	0.61%	0.59%	0.99%	1.44%	1.54%	6.18%	2.44%	4.47%	18.88%
	11x	26.14%	2.47%	1.67%	0.55%	0.41%	1.08%	1.52%	1.90%	3.48%	4.50%	3.41%	16.43%
	All Par.	41.26%	2.80%	2.07%	0.63%	0.57%	1.01%	0.98%	1.32%	3.29%	1.79%	3.14%	28.10%
200x200 Mesh	1092x	1591.39%	11.54%	6.47%	3.13%	2.25%	3.86%	19.96%	29.19%	50.08%	50.09%	79.29%	120.35%
	109x	4.78%	1.39%	1.17%	0.45%	0.32%	0.67%	1.05%	0.96%	5.85%	13.29%	18.73%	9.17%
	11x	2.52%	1.06%	1.06%	0.35%	0.37%	0.66%	0.50%	0.60%	1.81%	1.05%	1.44%	3.73%
	All Par.	3.49%	0.75%	1.63%	0.44%	0.29%	0.72%	0.35%	0.34%	1.95%	0.60%	0.72%	2.31%
450x450 Mesh	1092x	2572.48%	33.95%	3044.93%	5.93%	2.40%	15.25%	166.58%	169.19%	167.28%	117.68%	164.07%	142.75%
	109x	126.75%	1.98%	1.29%	2.73%	2.55%	2.39%	4.10%	2.03%	13.81%	19.93%	27.30%	19.18%
	11x	2.71%	1.08%	0.95%	0.40%	0.35%	0.73%	0.98%	0.46%	3.07%	1.86%	1.85%	9.56%
	All Par.	2.37%	0.87%	0.65%	0.27%	0.29%	0.52%	0.11%	0.10%	3.07%	0.26%	0.40%	7.14%

5.2 Poincaré Test Results

Two separate features were tacked in the Poincaré tests:

- Difference in area of the ground truth data versus the binned representations.
- Differences in the centers of mass of the two representations.

The test results from the Poincaré test for the difference in areas of the ground truth and reduced representations are presented in Figure 4. This figure presents

the results for each of the four unstructured reduced mesh representations and each of the different particle counts as boxplots in a single figure.

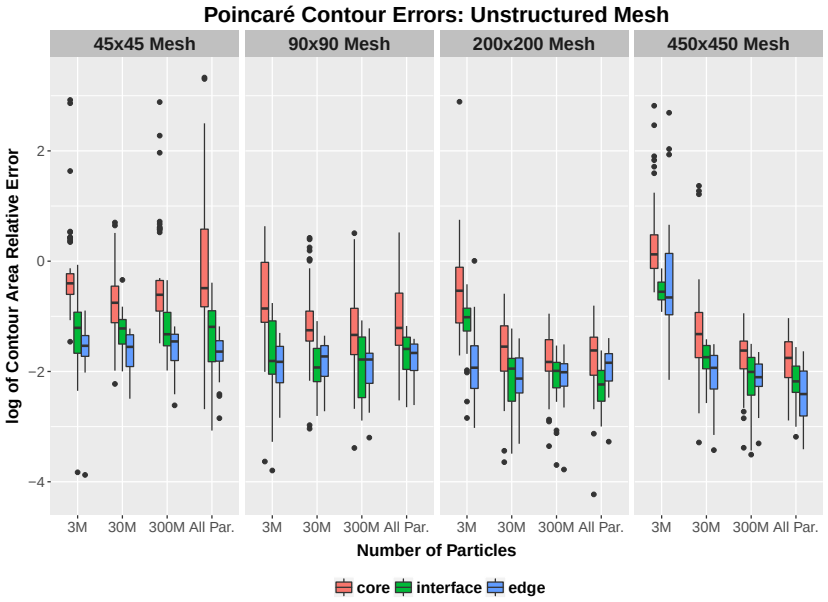

Fig. 4. Boxplots of the difference in areas of contours generated from Poincaré method on the ground truth and reduced representation data for the unstructured mesh.

The first thing to note in this figure is that some of the tests performed very well while some did not. Consider the 45x45 mesh for example. In this series of tests, the core of the plasma across all of our tests had a high median and variation in error, including the test that used the reduced 45x45 vector mesh created by averaging all particles from the simulation. From these large variations in error, we were able to figure out that the 45x45 mesh representation created too large of a decimation in the core of the plasma to create good vector averages from the particles. The core-edge interface and the edge however did perform well for all of those tests.

Moving on to the 90x90 and 200x200 mesh tests, the variation in error drops significantly, with the average core error dropping below 3% in the test case with the 200x200 mesh and 300 million particles. In the 450x450 test case the 3 million particle test jumps out, with all of the core, core-edge interface, and edge region errors jumping dramatically. This jump is due to the mesh being too refined for the number of particles that were used to generate it. In that case, it takes more than 3 million particles to create an accurate representation of the motion of the plasma. Thus, it is important to tune the number of particles to be a high enough ratio to adequately cover the number of bins in the mesh representation.

Similar results and trends were observed in the Poincaré test on the unstructured meshes for the difference in centers of mass of the ground truth and reduced representations. The main difference is that in almost all tests, the centers of the contours change very little, with the majority of the average errors being under 1%, see Table 3. The two main outlier cases are the 200x200 and 450x450 meshes with 3 million particles. These tests again show that an insufficient number of particles was used per cell to capture particle movement.

5.3 Streamline/Pathline Test Results

The test results from the streamline tests on the unstructured mesh are presented in Figure 5. This figure shows that there are a lot more outlier cases when evaluating the difference between the ground truth and reduced data representations. This is especially true for the 45x45 mesh case on both the core-edge interface and edge regions of the plasma. However, once the mesh resolution is increased, this trend decreases, and the average error of the test cases drops.

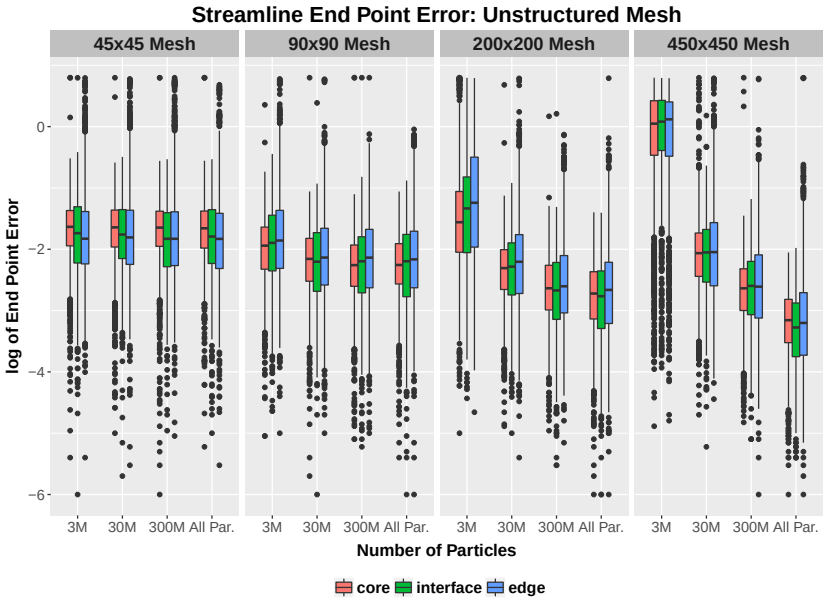

Fig. 5. Boxplots of the difference in end position of the streamlines generated from the ground truth and reduced representation data for the unstructured mesh.

One trend that persists from the Poincaré analysis are the cases where too few particles were used in the 3 million particle case for both the 200x200 and 450x450 meshes. Once the particle count was increased however, the errors dropped significantly, with errors under 1% in both the core and core-edge interface regions using 30 and 300 million particles with the 200x200 mesh.

The test results from the pathline tests on the unstructured mesh show the same general trends, with the largest difference being on the edge of the plasma, see Table 3. This result, though, is not surprising due to the very turbulent nature of the edge of the plasma in the XGC1 code, meaning that particles are bound to act erratically in that region.

6 Conclusions and Future Work

With this paper we have shown that large scale particle data can be binned in situ at every simulation time step in order to increase the temporal fidelity of the output particle data, while also reducing the amount of data transmitted over the network and subsequently saved to disk. We have further demonstrated that with domain specific knowledge it is possible to create multiple different configurations for the binning operation that enable it to be tuned for both accuracy of representation and the shortest time-to-solution. Our binning approach is capable of reducing the size of the particle data every time step from 489 GB down to 81 MB as a final output. Furthermore, we have shown through multiple post-hoc analysis operations that it is possible to generate reduced representations with known errors under 1% for certain regions of the plasma.

In future we would like to apply our binning technique to XGC1 at even larger simulation scales, and then expand to other particle codes to see if the technique remains valid and predictable. Further, while this application paper focused on in depth evaluation of our collaborators' preferred reduction technique, we think an expanded study that compared between more reduction operators would be very interesting.

References

1. Ahern, S., Shoshani, A., Ma, K.L., Choudhary, A., Critchlow, T., Klasky, S., Pascucci, V., Ahrens, J., Bethel, E., Childs, H., et al.: Scientific discovery at the exascale. Report from the DOE ASCR 2011 Workshop on Exascale Data Management (2011)
2. 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, a State-of-the-art (STAR) Report. Computer Graphics Forum, Proceedings of Eurovis 2016 **35**(3) (Jun 2016), lBNL-1005709
3. Chang, C., Ku, S., Diamond, P., Lin, Z., Parker, S., Hahm, T., Samatova, N.: Compressed ion temperature gradient turbulence in diverted tokamak edge). Physics of Plasmas (1994-present) **16**(5), 056108 (2009)
4. Childs, H., Pugmire, D., Ahern, S., Whitlock, B., Howison, M., Prabhat, Weber, G.H., Bethel, E.W.: Extreme scaling of production visualization software on diverse architectures. IEEE Comput. Graph. Appl. **30**(3), 22–31 (May 2010)
5. Childs, H., Pugmire, D., Ahern, S., Whitlock, B., Howison, M., Prabhat, Weber, G.H., Bethel, E.W.: Visualization at extreme scale concurrency. In: Bethel, E.W., Childs, H., Hansen, C. (eds.) High Performance Visualization: Enabling Extreme-Scale Scientific Insight. CRC Press, Boca Raton, FL (2012)

6. Fabian, N., Moreland, K., Thompson, D., Bauer, A., Marion, P., Geveci, B., Rasquin, M., Jansen, K.: The paraview coprocessing library: A scalable, general purpose in situ visualization library. In: Large Data Analysis and Visualization (LDAV), 2011 IEEE Symposium on. pp. 89–96. IEEE (2011)
7. Kress, J., Churchill, R.M., Klasky, S., Kim, M., Childs, H., Pugmire, D.: Preparing for in situ processing on upcoming leading-edge supercomputers. Supercomputing Frontiers and Innovations **3**(4), 49–65 (2016)
8. Kress, J., Pugmire, D., Klasky, S., Childs, H.: Visualization and analysis requirements for in situ processing for a large-scale fusion simulation code. In: Proceedings of the 2nd Workshop on In Situ Infrastructures for Enabling Extreme-scale Analysis and Visualization. pp. 45–50. IEEE Press (2016)
9. Liu, Q., Logan, J., Tian, Y., Abbasi, H., Podhorszki, N., Choi, J.Y., Klasky, S., Tchoua, R., Lofstead, J., Oldfield, R., Parashar, M., Samatova, N., Schwan, K., Shoshani, A., Wolf, M., Wu, K., Yu, W.: Hello adios: the challenges and lessons of developing leadership class i/o frameworks. Concurrency and Computation: Practice and Experience **26**(7), 1453–1473 (2014). https://doi.org/10.1002/cpe.3125, http://dx.doi.org/10.1002/cpe.3125
10. Lo, L.t., Sewell, C., Ahrens, J.P.: Piston: A portable cross-platform framework for data-parallel visualization operators. In: EGPGV. pp. 11–20 (2012)
11. 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, New York, NY, USA (2008). https://doi.org/10.1145/1383529.1383533, http://doi.acm.org/10.1145/1383529.1383533
12. Meredith, J.S., Ahern, S., Pugmire, D., Sisneros, R.: EAVL: the extreme-scale analysis and visualization library. In: Eurographics Symposium on Parallel Graphics and Visualization. pp. 21–30. The Eurographics Association (2012)
13. Moreland, K., Ayachit, U., Geveci, B., Ma, K.L.: Dax toolkit: A proposed framework for data analysis and visualization at extreme scale. In: Large Data Analysis and Visualization (LDAV), 2011 IEEE Symposium on. pp. 97–104 (Oct 2011)
14. Moreland, K., Sewell, C., Usher, W., Lo, L., Meredith, J., Pugmire, D., Kress, J., Schroots, H., Ma, K.L., Childs, H., Larsen, M., Chen, C.M., Maynard, R., Geveci, B.: VTK-m: Accelerating the Visualization Toolkit for Massively Threaded Architectures. IEEE Computer Graphics and Applications (CG&A) **36**(3), 48–58 (May/Jun 2016)
15. Neuroth, T., Sauer, F., Wang, W., Ethier, S., Ma, K.L.: Scalable visualization of discrete velocity decompositions using spatially organized histograms. In: Large Data Analysis and Visualization (LDAV), 2015 IEEE 5th Symposium on. pp. 65–72. IEEE (2015)
16. Oldfield, R.A., Widener, P., Maccabe, A.B., Ward, L., Kordenbrock, T.: Efficient data-movement for lightweight i/o. In: 2006 IEEE International Conference on Cluster Computing. pp. 1–9 (Sept 2006). https://doi.org/10.1109/CLUSTR.2006.311897
17. Pugmire, D., Kress, J., Meredith, J., Podhorszki, N., Choi, J., Klasky, S.: Towards scalable visualization plugins for data staging workflows. In: Big Data Analytics: Challenges and Opportunities (BDAC-14) Workshop at Supercomputing Conference (November 2014)
18. Reach, C., North, C.: Bandlimited olap cubes for interactive big data visualization. In: Large Data Analysis and Visualization (LDAV), 2015 IEEE 5th Symposium on. pp. 107–114. IEEE (2015)

19. Schatz, K., Müller, C., Krone, M., Schneider, J., Reina, G., Ertl, T.: Interactive visual exploration of a trillion particles. In: Large Data Analysis and Visualization (LDAV), 2016 IEEE 6th Symposium on. pp. 56–64. IEEE (2016)
20. Tchoua, R., Choi, J., Klasky, S., Liu, Q., Logan, J., Moreland, K., Mu, J., Parashar, M., Podhorszki, N., Pugmire, D., et al.: Adios visualization schema: A first step towards improving interdisciplinary collaboration in high performance computing. In: eScience (eScience), 2013 IEEE 9th International Conference on. pp. 27–34. IEEE (2013)
21. Vishwanath, V., Hereld, M., Papka, M.: Toward simulation-time data analysis and i/o acceleration on leadership-class systems. In: Large Data Analysis and Visualization (LDAV), 2011 IEEE Symposium on. pp. 9–14 (2011). https://doi.org/10.1109/LDAV.2011.6092178
22. 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. Eurographics Association (2011)
23. Ye, Y.C., Neuroth, T., Sauer, F., Ma, K.L., Borghesi, G., Konduri, A., Kolla, H., Chen, J.: In situ generated probability distribution functions for interactive post hoc visualization and analysis. In: Large Data Analysis and Visualization (LDAV), 2016 IEEE 6th Symposium on. pp. 65–74. IEEE (2016)