Investigating In Situ Reduction via Lagrangian Representations for Cosmology and Seismology Applications

Sudhanshu Sane[1], Chris R. Johnson[1], and Hank Childs[2]

[1] SCI Institute at University of Utah, USA
[2] University of Oregon, USA

Abstract. Although many types of computational simulations produce time-varying vector fields, subsequent analysis is often limited to single time slices due to excessive costs. Fortunately, a new approach using a Lagrangian representation can enable time-varying vector field analysis while mitigating these costs. With this approach, a Lagrangian representation is calculated while the simulation code is running, and the result is explored after the simulation. Importantly, the effectiveness of this approach varies based on the nature of the vector field, requiring in-depth investigation for each application area. With this study, we evaluate the effectiveness for previously unexplored cosmology and seismology applications. We do this by considering encumbrance (on the simulation) and accuracy (of the reconstructed result). To inform encumbrance, we integrated in situ infrastructure with two simulation codes, and evaluated on representative HPC environments, performing Lagrangian in situ reduction using GPUs as well as CPUs. To inform accuracy, our study conducted a statistical analysis across a range of spatiotemporal configurations as well as a qualitative evaluation. In all, we demonstrate effectiveness for both cosmology and seismology — time-varying vector fields from these domains can be reduced to less than 1% of the total data via Lagrangian representations, while maintaining accurate reconstruction and requiring under 10% of total execution time in over 80% of our experiments.

Keywords: Lagrangian analysis · in situ processing · vector data

1 Introduction

High-performance computing resources play a key role in advancing computational science by enabling modeling of scientific phenomena at high spatiotemporal resolutions. A challenge with regard to studying the output of a simulation is the prohibitively large size of the total data generated. Compromise in the form of storing a subset of the data can impact the extent and accuracy of subsequent post hoc exploratory analysis and visualization. In particular, for accurate time-varying vector field analysis and visualization, access to the full spatiotemporal resolution is required. Since storing the entire simulation output is expensive, scientists resort to temporal subsampling or lossy compression, and often limit analysis to individual time slices. An emerging paradigm to address large data challenges is the use of in situ processing to perform runtime analysis/visualization or data reduction to support exploratory post hoc analysis.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77961-0_36

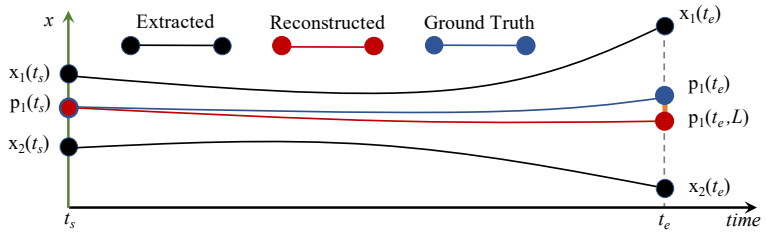


Fig. 1: Notional space-time visualization of Lagrangian representations for a time-varying 1D flow. The black trajectories are computed in situ and encode the behavior of the vector field between start time t_s and end time t_e. In a post hoc setting, a Lagrangian-based advection scheme L, i.e., a technique to interpolate the extracted data, is used to calculate the trajectory of a new particle p_1 . The red trajectory is the trajectory reconstructed post hoc and the blue trajectory is the ground truth. The end location of the red trajectory deviates by a margin of error from the ground truth.

Lagrangian analysis is a powerful tool to study time-varying vector fields and is widely employed for ocean modeling applications [28]. The notion of calculating a Lagrangian representation or *flow map*, i.e., sets of particle trajectories, "online" (in situ) for "offline" (post hoc) exploration was first proposed by Vries et al. [29] for an ocean modeling simulation. Figure 1 illustrates the approach. More recently, multiple works have advanced Lagrangian research along axes such as strategies for in situ extraction of reduced Lagrangian representations [1][19][22], post hoc reconstruction [6][21][10], and theoretical error analysis [4][5][9].

An open challenge for time-varying vector field exploration is predicting the uncertainty and variability in accuracy for different analysis techniques. Although the effectiveness of Lagrangian representations for any possible time-varying vector field that can be produced by a scientific simulation remains an open question, prior theoretical demonstration of Lagrangian techniques [1][6][4][5][9][20][21][19][10] on analytical, SPH, climate and ocean modeling data, and practical application in ocean activity analysis [23], has provided encouraging results. Using Lagrangian representations, the quality of post hoc reconstruction depends on the vector field itself, as well as configuration specifics such as sampling strategy and frequency of storage. Thus, to investigate the potential benefits of Lagrangian representations for a broader range of applications and to gauge its viability in practice, we leverage the recent developments of runtime in situ infrastructure that enable the straightforward extraction via APIs to study Lagrangian representations for cosmology and seisomology applications.

In this paper, our unique contribution is an investigation of Lagrangian representations to encode self-gravitating gas dynamics of a cosmology simulation and seismic wave propagation of a seisomology simulation. We measure the effectiveness of the technique by considering in situ encumbrance and post hoc accuracy. For both applications, our experiments show that Lagrangian representations offer high data reduction, in many cases requiring less than 1% storage of the complete time-varying vector fields, for a small loss of accuracy. Further, our study shows Lagrangian representations are viable to compute in representative HPC environments, requiring under 10% of total execution time for data analysis and visualization in the majority of configurations tested.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77961-0_36

2 Background and Related Work

2.1 Frames of Reference

In fluid dynamics, there are two frames of reference to observe fluid motion: Eulerian and Lagrangian. With the Eulerian frame of reference, the observer is in a fixed position. With the Lagrangian frame of reference, the observer is attached to a fluid parcel and is moving through space and time.

Storage of a flow field in an Eulerian representation is typically done by means of its velocity field. A velocity field v is a time-dependent vector field that maps each point $x \in \mathbb{R}^d$ in space to the velocity of the flow field for a given time $t \in \mathbb{R}$

$$v : \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}^d,\ x,t \mapsto v(x,t) \tag{1}$$

In a practical setting, a flow field at a specific time/cycle is defined as vector data on a fixed, discrete mesh. Time-varying flow is represented as a collection of such data over a variety times/cycles.

Storage of a flow field in a Lagrangian representation is done by means of its flow map $F_{t_0}^t$. The flow map is comprised of the starting positions of massless particles x_0 at time t_0 and their respective trajectories that are interpolated using the time-dependent vector field. Mathematically, a flow map is defined as the mapping

$$F_{t_0}^t(x_0) : \mathbb{R} \times \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}^d,\ t \times t_0 \times x_0 \mapsto F_{t_0}^t(x_0) = x(t) \tag{2}$$

of initial values x_0 to the solutions of the ordinary differential equation

$$\frac{d}{dt}x(t) = v(x(t),t) \tag{3}$$

In a practical setting, the flow map is represented as sets of particle trajectories calculated in the time interval $[t_0,t] \subset \mathbb{R}$. The stored information, encoded in the form of known particle trajectories (i.e., a Lagrangian representation), encodes the behavior of the time-dependent vector field over an interval of time.

2.2 Lagrangian Analysis

Within the vector field analysis and visualization community, Lagrangian methods have been increasingly researched in the past decade. In this paper, we focus on the use of Lagrangian methods to store time-varying vector fields in situ and enable subsequent post hoc analysis. In sparse temporal settings, Lagrangian representations are expected to perform better than their Eulerian counterparts. The key intuition behind this expectation is that Lagrangian representations capture the behavior of the flow field over an interval of time, as opposed to the state at a single time slice. However, in addition to the frequency of temporal sampling, the nature of the vector field and spatial sampling resolution impacts the quality of reconstruction.

Agranovsky et al. [1] conducted the seminal work to evaluate the efficacy of reduced Lagrangian representations. To maintain domain coverage, the study proposed the use of uniform spatial sampling to extract sets of temporally non-overlapping basis trajectories. Sane et al. [20] studied performance across a range of spatiotemporal configurations when operating using a fixed storage budget. The experiments in these works were conducted in a theoretical in situ setting, i.e., files were loaded from disk. Most recently,

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77961-0_36

Jakob et al. [10] trained a DNN to upsample FTLE visualizations derived from reduced Lagrangian representations. To generate training data, they first computed Lagrangian representations of a 2D flow field using a tightly-coupled integration with an open-source CFD solver on HPC resources and reported computation costs. However, the grid size of 4×4 per rank used in the study is not representative of real-world applications. Thus, the current literature lacks in situ encumbrance measurements in representative settings.

Lagrangian representations of a time-varying vector field can be extracted by adopting various strategies. Sane et al. [21] explored computing trajectories of variable duration and placement. Rapp et al. [19] applied their void-and-cluster sampling technique to identify a representative set of scattered samples. Although these strategies improved accuracy, they increased computation costs and are presently limited to single node settings. To address scalability challenges of extracting a Lagrangian representation in distributed memory, Sane et al. [22] explored an accuracy-performance tradeoff and demonstrated the use of a communication-free model that stored only trajectories that remain within the rank domain during the interval of computation.

Prior works have presented research pertaining to post hoc reconstruction using Lagrangian-based interpolation schemes. Hlawatsch et al. [8] proposed a hierarchical reconstruction scheme that can improve accuracy, but relies on access to data across multiple time intervals. Chandler et al. [6] proposed a modified k-d tree as a search structure for Lagrangian data extracted from an SPH simulation. Further, Chandler et al. [5] identified correlations between Lagrangian-based interpolation error and divergence in the flow field. Bujack et al. [4] evaluated the use of parameter curves to fit interpolated pathline points to improve the aesthetic of trajectories calculated using Lagrangian data. Lastly, Hummel et al. [9] provided theoretical error bounds for error propagation that can occur when calculating trajectories using Lagrangian representations.

2.3 Time-Varying Vector Field Reduction

Although Eulerian representations have been shown to be susceptible to temporal sparsity [27][18][1][20], temporal subsampling remains the widely used solution to limit data storage. Our study adds to this body of work by using temporal subsampling for comparison. Multiple works have proposed single time step vector field reduction strategies while maintaining an Eulerian representation [13][25][26]. Although these techniques could be used to reduce and store data more frequently, they do not inherently address the challenge of increasing temporal sparsity.

In a recent large-scale tornadic supercell thunderstorm study [15], Leigh Orf modified the I/O code to use a hierarchical data format and lossy floating-point compression via ZFP [12]. ZFP provides dynamic accuracy control by allowing the user to specify a maximum amount of deviation. Orf stated that although ZFP is effective for scalar fields that do not require differentiation during post hoc analysis, only a very small value of deviation can be chosen for each component of velocity to maintain accurate time-varying vector field reconstruction. Thus, ZFP allowed a limited amount of compression to vector field data without introducing significant uncertainty to post hoc analysis. The technique provided an average reduction of 30% of total uncompressed vector field data, with regions of high gradient resulting in less compression. Overall, Orf referred to the use of lossy compression as unfortunate but necessary.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77961-0_36

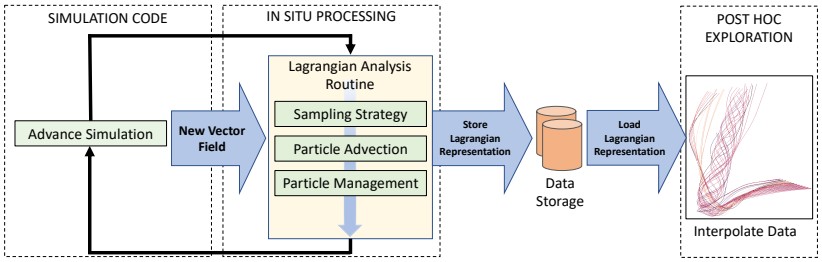


Fig. 2: Schematic of the Lagrangian in situ reduction and post hoc exploration workflow.

3 In Situ Reduction via Lagrangian Representations

This section describes the instantiation we consider for our study. Figure 2 shows a high-level description of the Lagrangian in situ reduction post hoc exploration (L-ISR-PHE) workflow. For our study, we focused on the current best practices in this space. To describe our instantiation, the remainder of this section is divided based on the two phases: in situ reduction and post hoc exploration.

In Situ Reduction Both simulations we considered partitioned space amongst ranks, with each rank owning one portion of the vector field. Our in situ routines followed this pattern, with an instance of our Lagrangian analysis routine executing for each rank, accessing its portion of the vector field. Further, for both simulations we were interested in capturing time-varying vector field behavior across the entire domain. Thus, for our in situ data reduction strategy, we prioritized domain coverage. Similar to Agranovsky et al. [1], we used uniform spatial sampling and a predetermined interval to store/reset particles. Thus, we computed sets of temporally non-overlapping basis trajectories over the duration of the simulation. Each set of basis trajectories encodes the behavior of the time-varying vector field over a specific interval of time. Our particle termination followed the local Lagrangian flow map model from Sane et al. [22], where particles are terminated once they reach the end of the interval or exit the block. Our implementation had two main knobs that control the total data storage and quality of reconstruction: number of basis trajectories, i.e., spatial sampling resolution, and frequency of storing information to disk, i.e., storage interval. The effect of these settings varies depending on the underlying vector field.

We used the Ascent [11] in situ infrastructure and VTK-m [14] library to implement L-ISR. The Ascent API can be used to perform tightly-coupled integration with an application code and access various in situ analytics capabilities. The VTK-m Lagrangian filter on each rank operated independently and maintained its own list of particles. We used the existing particle advection infrastructure available in VTK-m [17]. RK4 particle advection is implemented using VTK-m worklets (kernels) that offer performance portability by utilizing the underlying hardware accelerators. In our implementation, each Lagrangian filter stored the displacement of each particle (three double), as well as its validity (one Boolean), i.e., whether the particle remained within the domain during the interval of calculation. Overall, computing a Lagrangian representation increased the runtime memory cost on the simulation by approximately by four one-dimensional simulation "fields". Simulations often have tens to hundreds of fields defined on the simulation grid, and thus, this cost would likely be acceptable for most simulations.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77961-0_36

To compute a Lagrangian representation, the simulation invoked Ascent after every cycle it advanced. Ascent accessed the simulation vector field data and consequently invoked the Lagrangian filter. The Lagrangian filter used the vector field to advance particles, and triggered the storage of trajectories at the end of an interval. For integration, all the steps involved — creating an instance of Ascent, specifying parameters, and invoking the VTK-m Lagrangian filter — required only 23 lines of code (C++).

Post Hoc Exploration For post hoc analysis, new particle trajectories are computed to explore the time-varying vector field. To construct new particle trajectories, we first identified which basis trajectories to follow and then performed interpolation. Based on the study of accuracy of various Lagrangian-based advection schemes in [2], our study employed a Delaunay triangulation to identify the neighborhood of valid basis trajectories and second-order barycentric coordinates for interpolation. We used the CGAL [24] library to construct and search the Delaunay triangulation. After constructing new pathlines or deriving new scalar fields from the basis trajectories, we used VisIt [7] to generate visualizations.

4 Study Overview

This section provides an overview of our study. It is organized as follows: runtime environment (4.1), simulation codes (4.2), experiments (4.3), and evaluation metrics (4.4).

4.1 Runtime Environment

Our study used the Summit supercomputer at ORNL. A Summit compute node has two IBM Power9 CPUs, each with 21 cores running at 3.8 GHz and 512 GBytes of DDR4 memory. Nodes on Summit also have enhanced on-chip acceleration with each CPU connected via NVLink to 3 GPUs, for a total of 6 GPUs per node. Each GPU is an NVIDIA Tesla V100 with 5120 CUDA cores, 6.1 TeraFLOPS of double precision performance, and 16 GBytes of HBM2 memory. Lastly, it has a Mellanox EDR 100G InfiniBand, Non-blocking Fat Tree as its interconnect topology.

4.2 Simulation Codes

Nyx: In this cosmological simulation [3], baryonic matter is evolved by solving the equations of self-gravitating gas dynamics. We derived the velocity field using the fields of momentum and density of the baryonic gas. The simulation involves particles gravitating toward high-density regions to form multiple clusters across the domain. The distribution of high-density clusters and their formation is of interest to scientists. To study the distribution, scientists currently perform statistical analysis of gas particle density at a single time slice. We investigated the potential of reduced Lagrangian representation to accurately visualize the particle evolution and the distribution of high-density clusters using pathlines. The Nyx simulation we built executed as a single rank using two CPUs on a single Summit compute node.

SW4: In this seismology simulation [16], seismic wave propagation is studied using a fourth-order method. The application simulates waves radiating from the epicenter through viscoelastic media. We used the 3D time-varying displacement vector defined at each grid point as input. We investigated how accurately we can derive and visualize the

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77961-0_36

field encoding displacement over time in two regions: at the epicenter and away from the epicenter. The SW4 simulation we built executed using six ranks per compute node on Summit. Each rank was allocated a GPU for execution.

4.3 Experiments

For each application in this study, we organized our experiments to inform in situ encumbrance and post hoc accuracy. We considered four evaluation criteria (EC). To inform in situ encumbrance, we measured the execution time (EC1) and runtime memory usage (EC2) for in situ processing. To inform post hoc accuracy, we measured the size of data artifacts (EC3) and the reconstruction quality of time-varying vector field data (EC4). Next, we identified four factors that when varied produce the workloads we want to evaluate for our study:

- **Number of particles:** the spatial sampling resolution denoted using **1:X**, where X is the reduction factor. For example, a 1:8 configuration states that one basis particle is used for every 8 grid points (≈12.5% of the original data size).
- **Storage interval:** the number of cycles between saves and denoted by **I**.
- **Grid size:** the spatial resolution of the mesh.
- **Concurrency:** the scale of the execution and underlying parallelization hardware.

Rather than consider a complete cross-product of options for every workload factor, we sampled the space of possible options. Our goal was to provide coverage and allow us to see the impact of certain workload factors, all while staying within our compute budget. For Nyx, we ran 18 experiments, with 6 informing in situ encumbrance (varying **1:X**, grid size) and 12 informing post hoc accuracy (varying **1:X**, **I**). For SW4, we ran 11 experiments, with 7 informing in situ encumbrance (varying **1:X**, grid size, concurrency) and 4 informing post hoc accuracy (varying **1:X**). The specific options selected are presented along with the results in Section 5.

4.4 Evaluation Metrics

We selected our evaluation metrics based on the evaluation criteria listed in Section 4.3.

For EC1, we measured the average cost of invoking the Lagrangian VTK-m filter through Ascent every cycle, **Step**, in seconds. Additionally, we presented the percentage of simulation time spent on data analysis and visualization, or **DAV%**. We used $\mathbf{Sim}_{cycle}$ to denote the average time required for a single simulation cycle in seconds.

For EC2, we measured **InSituMem**, the runtime memory cost incurred by every compute node to maintain the state (current position) of particles at runtime in Bytes.

For EC3, we measured the total data storage (**DS**) required on the file system and report it in Bytes stored. In addition to I/O being infrequently performed, we observed that for the scale of study we conducted, Summit provided fast write times. In comparison to performing in situ processing every cycle, we found the I/O write cost to be negligible.

For EC4, we considered both a statistical and qualitative analysis. For Nyx, we derived pathlines from the basis trajectories and measured the reconstruction error by calculating the average Euclidean distance of interpolated points from the ground truth (precomputed using the complete simulation data) for each trajectory. We visualized

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77961-0_36

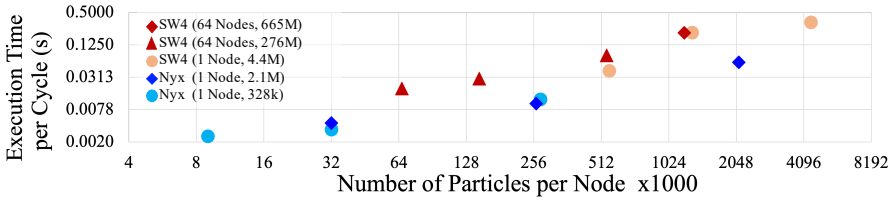


Fig. 3: Lagrangian in situ reduction cost per cycle for all in situ encumbrance experiments. The SW4 simulation executes with six ranks (each allocated one GPU) sharing memory on every node. The Nyx simulation executes on a single rank using all the cores of two CPUs on a single node. The legend includes concurrency and number of simulation grid points in parenthesis and both axes use logarithmic scales.

the distribution of pathline reconstruction error for every configuration using violin plots, and for a subset of configurations, the pathline clustering directly. For SW4, we derived a field encoding magnitude of displacement over time from the basis trajectories. In this case, we visualized and compared the derived field to the ground truth time-varying displacement field using violin plots and isosurfaces.

5 Results

Our results are organized as follows. Sections 5.1 and 5.2 present findings from our study investigating reduced Lagrangian representations for cosmology and seismology applications, respectively. Tables 1 and 2 provide information pertaining to in situ encumbrance experiments, such as concurrency information, spatial dimensions, Sim_{cycle}, number of particles per compute node, **InSituMem** per compute node, **Step**, and **DAV%**, for each application. Figure 3 shows the execution time per cycle for all the in situ encumbrance experiments. Figures 4, 5, 6, and 7 show the results of our post hoc accuracy evaluation. For each application, the figures are annotated with configuration specifics such as the **DS**, **1:X**, and **I**. Further, Lagrangian and Eulerian tests are distinguished explicitly in the captions or are labeled LT and ET, respectively, where T is the test number.

5.1 Nyx Cosmology Simulation

In Situ Encumbrance Using all the cores of two CPUs on a single compute node, we used OpenMP to parallelize the Nyx simulation and Lagrangian VTK-m filter. We tested two options for grid size - 69^3 and 129^3 - on a single rank, and three particle advection workloads (1:1, 1:8, 1:27) each. In a single compute node hour, the simulation performed approximately 300 and 38 cycles when using 69^3 and 129^3 grid sizes, respectively. An 8X increase in grid size resulted in a proportional increase in Sim_{cycle} but only a small increase in particle advection costs for the same number of particles. In practice, we would expect a single rank to operate on between 32^3 to 256^3 grid points, and thus our workloads provided a representative estimate of **DAV%**.

An encouraging finding was the low in situ encumbrance when performing L-ISR on the CPUs. Depending on the setup of various simulations and the form of integration for in situ processing, future work can consider offloading L-ISR computation to CPUs. Overall, considering the longer Sim_{cycle} times for the Nyx simulation, and parallel

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77961-0_36

Nodes	Ranks	Dimensions	**Sim**$_{cycle}$	Particles	**InSituMem**	**Step**	**DAV%**
1	1	$65 \times 65 \times 65$	10.9s	9k	0.2MB	0.0025s	0.023%
				32k	0.8MB	0.0033s	0.030%
				274k	6.8MB	0.0122s	0.0112%
		$129 \times 129 \times 129$	88.3s	78k	1.9MB	0.0044s	0.005%
				262k	6.5MB	0.0101s	0.011%
				2.1M	53.6MB	0.0596s	0.067%

Table 1: In situ encumbrance evaluation and experiment configurations for the Nyx simulation executing on CPUs.

compution coupled with low memory latency when using CPUs, the highest in situ encumbrance to extract a Lagrangian represenation was 0.1% of the simulation time or under 0.06s to compute 2.1M basis trajectories per cycle.

Post Hoc Accuracy To evaluate the usefulness of Lagrangian representations to encode time-varying self-gravitating gas dynamics, we considered a 69^3 grid over 400 cycles, three options for data reduction (1:1, 1:8, 1:27) and four options for **I** (25, 50, 100, 200). We constructed pathlines for 50,000 randomly placed particles during post hoc analysis. We visualize the distribution of reconstruction error for all tests in Figure 4.

The self-gravitating gas dynamics of this simulation produce a vector field that captures the transport of randomly distributed particles to multiple high-density clusters. Particles travel with increasing velocity as clusters increase in density. For this data, we found that Eulerian temporal subsampling performs better for small values of **I**. This result can be expected given reconstruction using an Eulerian representation and fourth-order Runge Kutta interpolation remain more accurate than second-order barycentric coordinates interpolation employed to interpolate Lagrangian representations [4][9]. However, as the value of **I** increases, the distribution of error for the Lagrangian tests indicates that a larger percentage of samples are reconstructed more accurately. In particular, this is true when a high spatial sampling resolution is used. Thus, particle

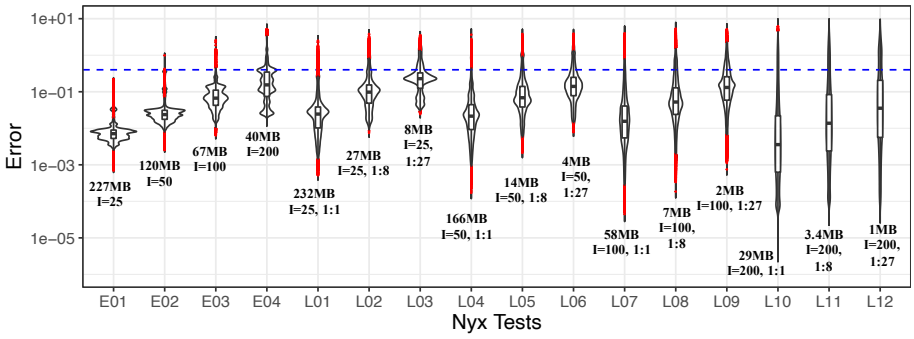


Fig. 4: Accuracy results for the Nyx experiments. Each violin plot shows the distribution of the particle reconstruction error for a specific configuration and the horizontal blue dashed line in the chart represents an error equivalent to a single grid cell side. The error axis uses a logarithmic scale. While Eulerian configurations contain greater uncertainty as the value of storage interval **I** increases, the Lagrangian representations offer the opportunity for improvements in accuracy. Additionally, we find high reconstruction accuracy relies on a high spatial sampling resolution as well.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77961-0_36

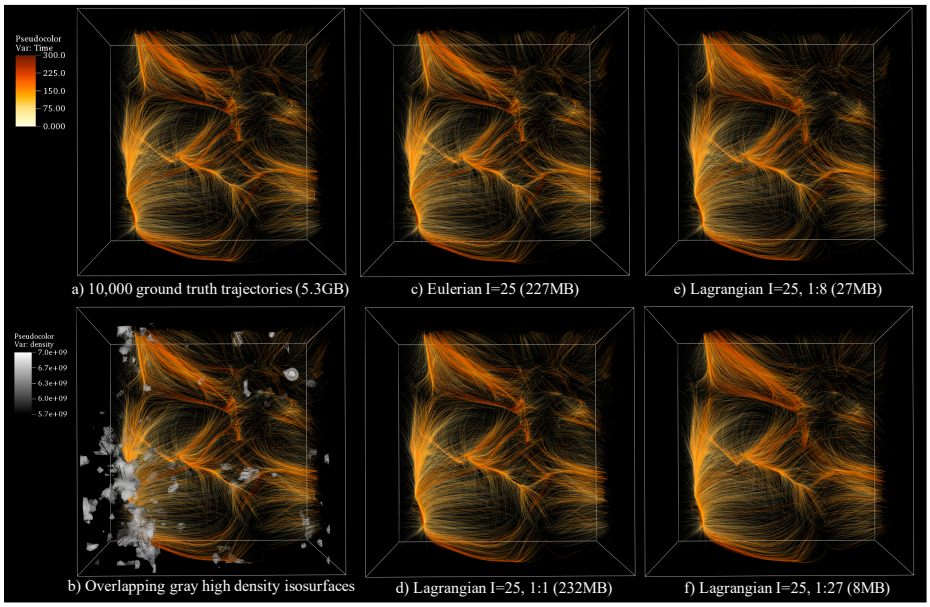


Fig. 5: Pathline visualization of baryonic particles evolution in self-gravitating gas dynamics of Nyx simulation. Using 10,000 randomly seeded particles, we visualize pathlines over 300 cycles. To focus on regions where particles cluster to form dense regions, we set opacity of the pathline to be directly proportional to time. Thus, we are able to focus on clustering as well as provide context of transport toward these regions. Lagrangian representations are able to reconstruct the ground truth trajectories and capture clustering accurately when high spatial sampling is used (1:1, 1:8). However, when using a 1:27 data reduction factor, some clusters are visualized less clearly.

evolution in this cosmology simulation can be tracked more accurately when a dense set of basis trajectories integrated for a long duration are interpolated. In contrast, Eulerian representations become less effective at reconstructing the vector field due to increased numerical approximation.

We used pathlines with manually set transfer functions to visualize the evolution and clustering of particles in regions of high density. The total size of the simulation vector field data used to compute the ground truth is 5.3GB. We visualized a random subset of 10,000 pathlines in Figure 5 for configurations with **I** set to 25. The Lagrangian representations demonstrate the ability to closely reconstruct regions where dense clusters are formed while requiring a fraction of the total simulation data size. For example, the 1:8 Lagrangian configuration enables the visualization of transport to dense clusters while requiring only 27MB, i.e., a 200X data reduction of the uncompressed vector field.

5.2 SW4 Seismology Simulation

In Situ Encumbrance For the SW4 simulation, we considered five grid sizes at varying concurrencies. In each case, we used all six GPUs available on a compute node to execute the simulation and L-ISR. For all L-ISR workloads tested, the execution time required

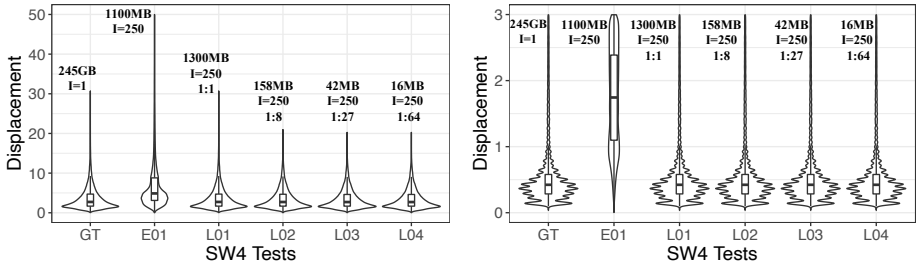
ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77961-0_36

per cycle remained under 0.5 seconds on average, and the maximum in situ memory required by a node was 112 MB to compute the trajectories for 4.4M particles. The cost for performing L-ISR was most dependent on the number of particles and only slightly impacted by increasing grid sizes. Referencing Figure 3, although the SW4 experiments used six GPUs, we found execution time to be slower than Nyx experiments due to the use of shared memory by multiple ranks (each has its own data block) and the high cost of launching kernels on the GPU for limited amounts of computation (each basis particle advances by only a single step/cycle each invocation).

Nodes	Ranks	Dimensions	**Sim**$_{cycle}$	Particles	**InSituMem**	**Step**	**DAV%**
1	6	$251 \times 251 \times 70$	0.35s	555k	13.89MB	0.0412s	11.67%
		$335 \times 335 \times 93$	2.02s	1.3M	33.16MB	0.2125s	10.48%
		$501 \times 501 \times 139$	7.58s	4.4M	111.13MB	0.3309s	4.365%
64	384	$1001 \times 1001 \times 276$	1.6s	66k	1.6MB	0.0194s	1.201%
			1.5s	146k	3.6MB	0.0295s	1.944%
			1.3s	540k	13.5MB	0.0798s	6.175%
		$1335 \times 1335 \times 368$	2.9s	1.2M	31.9MB	0.2095s	7.074%

Table 2: In situ encumbrance evaluation and experiment configurations for the SW4 simulation executing on GPUs. Particles and **InSituMem** are per compute node.

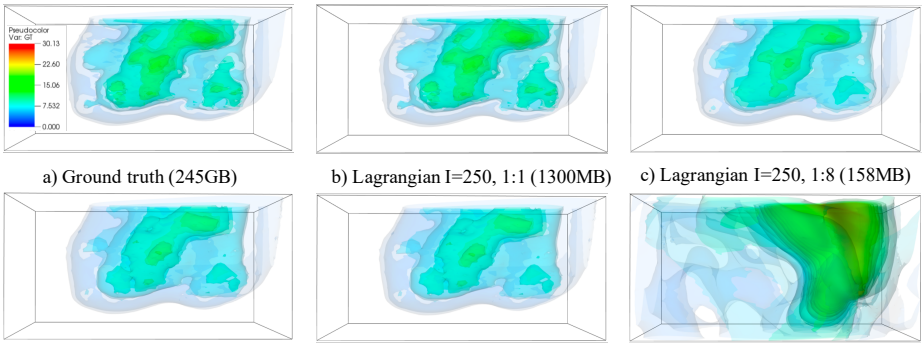
Post Hoc Accuracy We studied the reconstruction of the time-varying displacement vector field encoding wave propagation by considering four options for data reduction (1:1, 1:8, 1:27, 1:64) and one option for **I** (250). The ground truth was computed using data defined on a regular mesh containing 4.5M grid points over 2000 simulation cycles and required 245 GB. The displacement was highest near the epicenter and reduced as waves propagate further away. For each simulation run, we measured the displacement of 200,000 samples reconstructed near the epicenter (Figure 6a) and 90,000 samples reconstructed in six regions away from the epicenter (Figure 6b). Here, we directly compared against the distribution of ground truth (GT) displacement. In both cases, Lagrangian representations offered significant data reduction while maintaining high accuracy. We found that as the number of basis trajectories extracted reduces, the displacement for some samples near the epicenter can be underestimated. In contrast, using a tempo-



(a) High displacement near the epicenter. (b) Low displacement away from the epicenter.

Fig. 6: Violin plots of the distribution of particle displacement for the ground truth (GT), one Eulerian configuration and four Lagrangian configurations. The Eulerian configuration, with access to a limited number of time slices, overestimates the displacement. The Lagrangian representation captures displacement in both settings, in regions near and away from the epicenter, accurately.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77961-0_36



a) Ground truth (245GB) b) Lagrangian I=250, 1:1 (1300MB) c) Lagrangian I=250, 1:8 (158MB)

d) Lagrangian I=250, 1:27 (42MB) e) Lagrangian I=250, 1:64 (16MB) f) Eulerian I=250 (1100MB)

Fig. 7: Visualization of the displacement field derived from reduced Lagrangian representations near the epicenter using multiple isosurfaces. The ground truth is computed using 2000 cycles of the seismic wave propagation simulation. Although at higher data reduction factors regions of high displacement are underestimated, Lagrangian representations are capable of accurately reconstructing the overall feature structure.

rally subsampled Eulerian representation (E01) results in significant overestimation of displacement. This result can be expected since temporal subsampling fails to capture the transient nature of wave propagation, whereas Lagrangian representations encoding behavior over an interval of time remain accurate. Compared to Figure 6a, the ground truth in Figure 6b has smaller displacement and a multimodal distribution, which is the result of samples collected from six regions of the domain away from the epicenter.

Figure 7 visualizes field encoding displacement over time near the epicenter using multiple semi-opaque isosurfaces. Although regions of highest displacement can be underestimated as the data reduction factor increases, the overall structure is well preserved using highly compressed Lagrangian representations. In all cases, Lagrangian representations required less than 1% of the storage of the complete vector data.

6 Conclusion

Accurate exploratory analysis and visualization of time-varying vector fields is challenging. On the one hand, it can be performed accurately if the entire spatiotemporal resolution is available. However, storing all the data to disk for exploratory post hoc analysis is expensive. On the other hand, if subsets of the data are stored, predicting uncertainty and variability of accuracy for analysis techniques post hoc is difficult. In this context, Lagrangian representations computed using the full spatiotemporal resolution via in situ processing demonstrate the potential to enable accurate exploratory time-varying vector field analysis for reduced data storage costs.

For wider adoption and consideration of Lagrangian representations, an important step is characterization of effectiveness across a broad range of real-world applications. In this paper, we investigated in situ reduction via Lagrangian representations for vector fields from Nyx cosmology and SW4 seismology simulations. For the Nyx cosmology simulation, we found that Lagrangian representations are sensitive to both the spatial

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77961-0_36

and temporal sampling rate, notably providing higher reconstruction accuracy when basis trajectories are computed using a high spatial and low temporal resolution. For the SW4 seismology simulation, we found Lagrangian representations are well suited to capture the transient seismic waves and offer high data reduction options for a small loss of accuracy. For both simulations, the percentage of execution time spent on computing the Lagrangian representation in situ was under 10% in most cases. Overall, we believe the findings of this study demonstrates that two computational science simulations considered benefit from Lagrangian representations for time-varying vector field exploration. This finding also provides confidence that more computational areas can benefit, and we encourage future work in this direction.

Acknowledgment

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.

References

1. Agranovsky, A., Camp, D., Garth, C., Bethel, E.W., Joy, K.I., Childs, H.: Improved post hoc flow analysis via lagrangian representations. In: 4th IEEE Symposium on Large Data Analysis and Visualization, LDAV. pp. 67–75 (2014)
2. Agranovsky, A., Camp, D., Joy, K.I., Childs, H.: Subsampling-based compression and flow visualization. In: Visualization and Data Analysis 2015. vol. 9397, pp. 207 – 220. International Society for Optics and Photonics, SPIE (2015)
3. Almgren, A.S., Bell, J.B., Lijewski, M.J., Lukić, Z., Van Andel, E.: Nyx: A massively parallel amr code for computational cosmology. The Astrophysical Journal **765**(1), 39 (2013)
4. Bujack, R., Joy, K.I.: Lagrangian representations of flow fields with parameter curves. In: IEEE Symposium on Large Data Analysis and Visualization (LDAV). pp. 41–48 (2015)
5. Chandler, J., Bujack, R., Joy, K.I.: Analysis of error in interpolation-based pathline tracing. In: Proceedings of the Eurographics/IEEE VGTC Conference on Visualization: Short Papers. pp. 1–5. Eurographics Association (2016)
6. Chandler, J., Obermaier, H., Joy, K.I.: Interpolation-based pathline tracing in particle-based flow visualization. IEEE Transactions on Visualization and Computer Graphics **21**(1), 68–80 (2015)
7. Childs, H.: Visit: An end-user tool for visualizing and analyzing very large data (2012)
8. Hlawatsch, M., Sadlo, F., Weiskopf, D.: Hierarchical line integration. IEEE Transactions on Visualization and Computer Graphics **17**(8), 1148–1163 (2011)
9. Hummel, M., Bujack, R., Joy, K.I., Garth, C.: Error estimates for lagrangian flow field representations. In: Proceedings of the Eurographics/IEEE VGTC Conference on Visualization: Short Papers. pp. 7–11. Eurographics Association (2016)
10. Jakob, J., Gross, M., Günther, T.: A fluid flow data set for machine learning and its application to neural flow map interpolation. IEEE Transactions on Visualization and Computer Graphics (Proc. IEEE Scientific Visualization) (2020)

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77961-0_36

11. Larsen, M., Ahrens, J., Ayachit, U., Brugger, E., Childs, H., Geveci, B., Harrison, C.: The alpine in situ infrastructure. In: Proceedings of the In Situ Infrastructures on Enabling Extreme-Scale Analysis and Visualization. pp. 42–46. ACM (2017)
12. Lindstrom, P., Isenburg, M.: Fast and efficient compression of floating-point data. IEEE Transactions on Visualization and Computer Graphics **12**(5), 1245–1250 (2006)
13. Lodha, S.K., Faaland, N.M., Renteria, J.C.: Topology preserving top-down compression of 2d vector fields using bintree and triangular quadtrees. IEEE Transactions on Visualization and Computer Graphics **9**(4), 433–442 (2003)
14. Moreland, K., Sewell, C., Usher, W., Lo, L.t., Meredith, J., Pugmire, D., Kress, J., Schroots, H., Ma, K.L., Childs, H., et al.: Vtk-m: Accelerating the visualization toolkit for massively threaded architectures. IEEE Computer Graphics and Applications **36**(3), 48–58 (2016)
15. Orf, L.: A violently tornadic supercell thunderstorm simulation spanning a quarter-trillion grid volumes: Computational challenges, i/o framework, and visualizations of tornadogenesis. Atmosphere **10**(10) (2019)
16. Petersson, N.A., Sjögreen, B.: Wave propagation in anisotropic elastic materials and curvilinear coordinates using a summation-by-parts finite difference method. Journal of Computational Physics **299**, 820–841 (2015)
17. Pugmire, D., Yenpure, A., Kim, M., Kress, J., Maynard, R., Childs, H., Hentschel, B.: Performance-Portable Particle Advection with VTK-m. In: Eurographics Symposium on Parallel Graphics and Visualization. The Eurographics Association (2018)
18. Qin, X., van Sebille, E., Sen Gupta, A.: Quantification of errors induced by temporal resolution on lagrangian particles in an eddy-resolving model. Ocean Modelling **76**, 20–30 (2014)
19. Rapp, T., Peters, C., Dachsbacher, C.: Void-and-cluster sampling of large scattered data and trajectories. IEEE Transactions on Visualization and Computer Graphics **26**(1), 780–789 (2019)
20. Sane, S., Bujack, R., Childs, H.: Revisiting the evaluation of in situ lagrangian analysis. In: Eurographics Symposium on Parallel Graphics and Visualization. The Eurographics Association (2018)
21. Sane, S., Childs, H., Bujack, R.: An interpolation scheme for VDVP lagrangian basis flows. In: Eurographics Symposium on Parallel Graphics and Visualization. The Eurographics Association (2019)
22. Sane, S., Yenpure, A., Bujack, R., Larsen, M., Moreland, K., Garth, C., Childs, H.: Scalable in situ lagrangian flow map extraction: demonstrating the viability of a communication-free model. arXiv preprint arXiv:2004.02003 (2020)
23. Siegfried, L., Schmidt, M., Mohrholz, V., Pogrzeba, H., Nardini, P., Böttinger, M., Scheuermann, G.: The tropical-subtropical coupling in the southeast atlantic from the perspective of the northern benguela upwelling system. PloS one **14**(1) (2019)
24. The CGAL Project: CGAL User and Reference Manual. CGAL Editorial Board, 5.2.1 edn. (2021), `https://doc.cgal.org/5.2.1/Manual/packages.html`
25. Theisel, H., Rossl, C., Seidel, H..: Combining topological simplification and topology preserving compression for 2d vector fields. In: 11th Pacific Conference on Computer Graphics and Applications, 2003. Proceedings. pp. 419–423 (2003)
26. Tong, X., Lee, T.Y., Shen, H.W.: Salient time steps selection from large scale time-varying data sets with dynamic time warping. In: IEEE Symposium on Large Data Analysis and Visualization (LDAV). pp. 49–56. IEEE (2012)
27. Valdivieso Da Costa, M., Blanke, B.: Lagrangian methods for flow climatologies and trajectory error assessment. Ocean Modelling **6**(3), 335–358 (Jan 2004)
28. van Sebille, E., et al.: Lagrangian ocean analysis: Fundamentals and practices. Ocean Modelling **121**, 49 – 75 (2018)
29. Vries, P., Döös, K.: Calculating lagrangian trajectories using time-dependent velocity fields. Journal of Atmospheric and Oceanic Technology **18**(6), 1092–1101 (2001)

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77961-0_36