Spatio-Temporal Feature Exploration in Combined Particle/Volume Reference Frames

Franz Sauer, *Member, IEEE* and Kwan-Liu Ma, *Fellow, IEEE*

Abstract—The use of large-scale scientific simulations that can represent physical systems using both particle and volume data simultaneously is gaining popularity as each of these reference frames has an inherent set of advantages when studying different phenomena. Furthermore, being able to study the dynamic evolution of these time varying data types is an integral part of nearly all scientific endeavors. However, the techniques available to scientists generally limit them to studying each reference frame separately making it difficult to draw connections between the two. In this work we present a novel method of feature exploration that can be used to investigate spatio-temporal patterns in both data types simultaneously. More specifically, we focus on how spatio-temporal subsets can be identified from both reference frames, and develop new ways of visually presenting the embedded information to a user in an intuitive manner. We demonstrate the effectiveness of our method using case studies of real world scientific datasets and illustrate the new types of exploration and analyses that can be achieved through this technique.

Index Terms—Particle data, volume data, feature exploration, spatio-temporal analysis

1 Introduction

Scientific simulations have become an essential tool for studying physical phenomena. While simulations that represent data in either a particle-based (Lagrangian) or volume-based (Eulerian) reference frame are ubiquitous, simulations that represent and produce results in both formats simultaneously are growing in popularity. These "hybrid codes" can use the inherent advantages of each reference frame to study different aspects of a particular system and are commonly used in a variety of fields [1], [2], [3]. This becomes especially useful when investigating the time-varying properties of different phenomena. For example, the motion of discrete particles, such as ions or electrons, are best represented in the Lagrangian frame which can easily track their evolution throughout the domain, whereas other properties, such as background electromagnetic field strengths, are better represented in the Eulerian frame which can track variations at fixed spatial locations.

While a complete understanding of all aspects of a simulation is ideal, the post-processing analysis techniques available to scientists generally limit them to studying each reference frame separately, making it difficult to draw connections between them. Developing new ways in which both the Lagrangian and Eulerian representations can be simultaneously explored will lead to a better understanding of the interplay between different aspects of a physical system, especially if this exploration includes temporal analysis. In the Eulerian specification, it is beneficial to analyze time-varying fluctuations at fixed locations in the domain (a major advantage of this spatially-consistent representation). Furthermore, in order to fully take advantage of the Lagrangian frame, one can also investigate the geometric properties of trajectory shapes and capture flow patterns in an intuitive way. It is simply not feasible to project all data into one reference frame without losing the advantages inherent to the other and in many cases may be counterproductive altogether (e.g., trying to represent the trajectory of an electron as volume data). Instead, coupling these representations to support one another can offer new perspectives into datasets and enhance traditional analysis.

There has only been limited work in the visualization community to develop visual and analytical means of correlating results from each data type and little to no work on extending this joint exploration to study temporal variations as well. As a result, this paper focuses on developing a new generalized feature exploration scheme that can be used to study the time-varying properties of simulation data in both the Lagrangian and Eulerian representations simultaneously. This presents a number of visual and computational challenges and is namely due to the fundamental differences of each frame since the volume data remains spatially static while particle data is spatially dynamic. Visually representing information from both data types at once by simply superimposing the two frames is not enough as it will easily overwhelm a viewer with clutter.

In order to address these concerns, specific design decisions need to be made regarding not only how spatio-temporal subsets can be selected from both frames but also how these subsets are computed, stored, and visually presented to a user. Our technique uses an extraction result in one reference frame as a "spatio-temporal origin" and uses a temporal window in the opposing reference frame to compute accompanying information about the system in the chosen neighborhood. The ability to use such an origin

• *The authors are with the Department of Computer Science, University of California, Davis, CA 95616.*
E-mail: fasauer@ucdavis.edu, ma@cs.ucdavis.edu.

Manuscript received 5 Oct. 2016; revised 28 Jan. 2017; accepted 8 Feb. 2017. Date of publication 24 Feb. 2017; date of current version 3 May 2017.
Recommended for acceptance by T. Dwyer and Y. Wu.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TVCG.2017.2674918

helps to orient a user within the two data spaces allowing them to investigate complex patterns from a familiar starting point. Another advantage of this technique is that this accompanying information can be used to further segment and/or classify the original extraction result by propagating a classification from one reference frame in the spatio-temporal object onto the opposing reference frame. Segmentation schemes that utilize both the Eulerian and Lagrangian representations can lead to more control and detail in feature exploration.

While this work focuses mainly on analyzing flow data from scientific simulations, these techniques can be applied to a broader range of data by coupling particle-based and volume-based representations from multiple sources. For example, one can combine movement data (a Lagrangian format) with measurements taken at fixed geospatial locations (an Eulerian format). Using the spatio-temporal techniques described here can reveal new insights by combining data sources with entirely different representations.

This paper presents our technique on combining the Eulerian and Lagrangian specifications to enhance spatio-temporal analysis through the following contributions:

- The introduction of a new feature exploration scheme that represents information from both representations to study spatio-temporal patterns.
- The extension of this technique to generate segmentation results that utilize both reference frames in order to produce a more detailed data classification.
- The development of a new visual representation to intuitively represent temporal information from both reference frames simultaneously.

We also demonstrate the usefulness of our method through case studies using real world datasets and illustrate new ways this type of data can be explored.

2 Background

As previously described, limited work has been done on coupling the particle and volume data types, and even less so for use in temporal analysis. However, ways of representing temporal patterns in either representation separately have been extensively studied and provide a useful starting point for coupling the two representations.

2.1 Related Work

The limited amount of work that specifically focuses on joint Eulerian-Lagrangian techniques (outside the realm of scientific simulation design) exists primarily through specific application-based methods such as the interpolation of satellite data [4], realistic bubble representation [5], or flow trajectory computation [6]. Alternatively, both reference frames have also been used for rendering techniques such as for particle-based volume rendering [7], [8], [9]. A more generalized use of both particle and field data for post hoc analysis is more difficult to locate but can appear throughout various works in flow visualization.

There are cases in illustrative flow rendering which extend traditional visualization techniques (for particle or field data) to represent additional properties, sometimes from the opposing reference frame. In a Lagrangian sense, Stoll et al. [10] used a method of stylizing line primitives. By adjusting the color, shape, or texture of streamlines they could represent additional information to the viewer, such as the underlying properties of the vector flow field. In addition, Burger et al. [11] employ an importance driven approach to selecting particles and trajectories in flow visualization. This is then coupled with glyphs to show other parameters. Kirby et al. [12] utilized concepts from painting to visualize additional scalar information through multiple layers of color and texture in conjunction with flow representing glyphs. Similarly Urness et al. [13] used color in conjunction with LIC to represent additional properties. Both these methods have been shown to work well for showing instantaneous information in 2D flows. More examples can be found in the survey by Brambilla et al. [14].

While some of these methods can be used to visually display both particle and field-based information at instantaneous points in time, they do not focus on representing the spatio-temporal interplay between the frames. A previous work of ours [15] did use both particle and volume data as an alternate way of performing feature tracking. While there is a temporal component to the approach, correlations between the reference frames were made at discrete jumps in time to temporally connect volumetric features. This new work, on the other hand, focuses on studying variations in particle and volume data simultaneously within a detailed time window. Other notable works include that by Chandler et al. [16] who visualized temporal properties of particles through illustrative volume visualization. In addition, Salzbrunn et al. [17] utilized pathline predicates to extract and visualize unsteady flow features. Furthermore, FTLE [18] uses Lagrangian advection to generate a scalar field representing flow structures. Although these methods focuses on particles, they utilize techniques in a volume reference frame for generating visual representations.

Feature tracking is one method of analyzing time-varying volume data. However, there are other manners in which volume data can temporally analyzed. A notable work by Wang et al. [19] employs an importance driven approach to visualizing time-varying volume data. Features are classified on an importance scale based on conditional entropy characterizing local temporal behavior. As a result temporally interesting features can be extracted and highlighted to the user. Woodring and Shen [20] enhance comparative volume visualization by presenting multiple volumes generated through various operators and using specially designed contextual cues. Furthermore, Balabanian et al. [21] used a 4D multi-volume raycaster in conjunction with temporal transfer functions to render volume features and their evolution at multiple timesteps simultaneously. A summary of a few other techniques focused on time-varying volume visualization can be found in [22]. Focused more on computational efficiency, Peterka et al. [23] choose to organize datasets using 4D spatio-temporal blocks representing a coherent neighborhood to improve parallel particle tracing. Such a scheme allows for fast data access of nearby spatial locations and timesteps.

When it comes to the Lagrangian reference frame, the temporal dimension is often implied to be along a particle trajectory. Analyzing its geometric properties can allow researchers to explore a history of flow patterns as the

particle travels through the domain. Marchesin et al. [24] use the geometric properties of streamlines as well as their screen space projection for filtering purposes. This allows a system to select representative streamlines to describe the flow in a low clutter manner. Li et al. [25] focused on improving streamline similarity metrics for selection purposes using a spatially sensitive bag of features approach and extended their work further to a more sophisticated user assisted segmentation scheme [26]. Furthermore, Wei et al. [27] developed a sketch-based interface to extract trajectory shapes based on user drawn curves.

We can also look towards the extensive set of techniques outside of flow visualization aimed at developing intuitive visualizations of spatio-temporal neighborhoods. For example, spatio-temporal visualization techniques have been used in the analysis of movement data [28], [29], enhanced weather forecasting [30], or the validation of geoscientific models [31] to name a few. A broader survey can be found in [32]. While each of these methods have domain specific differences, they commonly tend to: show time and other parameters in conjunction with one another, link views to avoid clutter, and filter/cluster to reduce the data space being explored. The work in this paper draws upon these design choices for inspiration when designing a means to explore temporal neighborhoods in both the Eulerian and Lagrangian representations of scientific simulation datasets.

2.2 Correlating Particle and Volume Data

As mentioned earlier, it is simply not feasible to project all data into one reference frame without losing key advantages that the other frame provides. As a result we must draw connections between the reference frames while maintaining the benefits of each. The most direct way of correlating the particle and volume data types is through spatial proximity. Since volume data is defined on a discretized grid we can use a nearest-neighbor (Voronoi) approach to define an association between particles and a volume grid point. This nearest-neighbor region can be thought of as a volume cell surrounding the grid point (similar to the cube-like representation of a voxel of a 3D regular grid volume). These volume cells can be structured or unstructured and can physically contain zero or more particles at any point in time; each particle maps to exactly one volume cell and each volume cell maps to a unique set of particles.

In order for our spatio-temporal approach to be effective the following conditions must hold: 1) given a particle, we must be able to quickly identify its associated volume cell and 2) given a volume cell, we must be able to quickly identify its associated set of particles. While the first condition can easily be met when the volume data lies on a structured grid, both conditions can become costly in unstructured cases. Traditionally, hierarchical data structures, such as the uniform octree [33] or non-uniform k-d tree [34], [35], are used when spatially organizing entities for easy access. While these approaches are useful for locating nearby entities given any spatial location in a domain not known *a priori*, they exceed our current needs since our search will always be done at the known spatial location of particles and volume cells.

As a result, we favor using an indexing scheme to store particle and cell IDs. Each particle stores an additional attribute which indicates its associated volume cell, and each volume cell stores a list of associated particles. Given that the data is sorted based on these IDs, this guarantees a constant time complexity in identifying associated entities in the counterpart reference frame. The additional memory overhead of using this indexing approach is $2n$ integer values per timestep, where n is the total number of particles. Such an overhead tends to be small compared to the full dataset size per timestep which tends to contain many high precision floating point values for each particle and volume cell.

This indexing scheme requires a one time preprocessing step, but can also be computed in situ if desired. This would be advantageous since many simulations already tend to spatially distribute the particle and field data among several computing nodes. Not only can the preprocessing be done in parallel, but the search area for connecting particles and volume cells is also greatly decreased. While we use the indexing scheme in our current implementation, each of the two conditions could be met through an alternate technique if it happens to be more appropriate given a certain application area.

3 Methods

In our approach, the Eulerian and Lagrangian data types support one another by using an extraction result in one reference frame as a "spatio-temporal origin". Next, a temporal window in the opposing reference frame is used to compute accompanying information in a chosen neighborhood. This initial origin is meant to help guide the user in the extraction process from a familiar starting point. The manner in which these objects are constructed and visually represented is described in the following sections.

3.1 Spatio-Temporal Neighborhood Extraction

First, we define a set of objects that can be extracted and used to quantify temporal information in both the particle and volume reference frames. We begin this description using only one spatial dimension (x) and volume data that is organized on a regular Cartesian grid as it is easier to pictorially represent. However, these representations can all be generalized to higher spatial dimensions and for use in unstructured grids as well in a straightforward manner.

Let $p_i(t)$ represent a particle trajectory with index i as a function of time t. This trajectory represents both a changing position x as well as other associated Lagrangian variables (velocity, temperature, etc.), which we denote as s, that vary along the trajectory

$$p_i(t) = \{x(t), s(t)\}.$$

In the volume reference frame, let $v_j(t)$ represent a grid location with index j as a function of time t. This grid location does not change in position, but does have other time-varying Eulerian variables, which we denote as s'

$$v_j(t) = \{x', s'(t)\}.$$

Fig. 1 shows one manner in which we can use both reference frames to analyze a temporal neighborhood. As shown on the left, we identify a particular grid location (volume cell) at a particular timestep of interest, $v_j(t_0)$. Using this

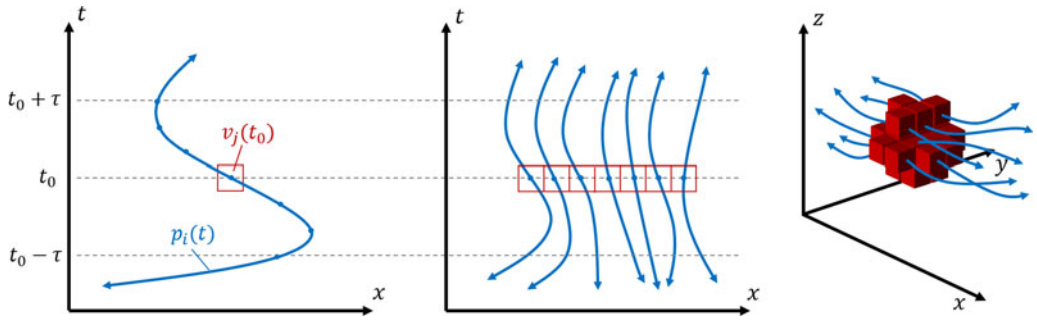

Fig. 1. *Left)* Using a volume grid cell (red) as an origin and studying the temporal neighborhood of an associated particle trajectory (blue). *Middle)* Extending this concept to a group of volume cells (e.g., an extracted volume feature) and studying the temporal neighborhood of all associated particles. *Right)* A 3D depiction of the middle case with time implied to be varying along each particle trajectory.

grid location as an origin, we can explore the temporal neighborhood of any spatially associated particles. This can be represented as either a distribution of Lagrangian values (stored as an array/vector) or as aggregated information accumulated throughout the temporal window via a user specified function, f. Since this is a temporal window from the Lagrangian reference frame, aggregated information could be derived from particle variables or from the geometric properties of its trajectory. More details regarding these functions are discussed in Section 3.1.1. We call the object representing this particular case α and it contains information regarding the volume cell origin as well as temporal information from the corresponding particles

$$\alpha(v_j(t_0)) = \{x', s'(t_0), \omega'(\tau)\},$$

where ω' can be a vector/array of values in a temporal window of size $2\tau + 1$

$$\omega'(\tau) = (p_i(t_0 - \tau), \ldots, p_i(t_0), \ldots, p_i(t_0 + \tau)),$$

or aggregated information accumulated throughout the window

$$\omega'(\tau) = \int_{t_0-\tau}^{t_0+\tau} f(p_i(t))dt.$$

Note that if there are multiple particles associated with a volume cell, we can compute a separate $\omega'(\tau)$ for each particle. If there are no associated particles, then there is no need to compute α.

This can naturally be extended to consider a group of volume cells (e.g., a feature of interest segmented from the volume data) as an origin as shown in the middle portion of the same figure. In this case, α is computed for every volume cell in the feature and the results are encompassed in a new object called $A(V)$ where V is a set of volume cells. This object contains information regarding the origin in the volume data as well as the temporal information in the particle data

$$A(V) = \begin{cases} X' & \text{a vector/array of positions} \\ S'(t) & \text{a vector/array of Eulerian variables} \\ \Omega'(\tau) & \text{temporal window information} \end{cases}.$$

Note that Ω' is now a matrix if representing the distribution of values in all the temporal windows or a vector/array when representing aggregated information from each of the temporal windows. The right side of the figure shows a depiction of such an object in 3D space; in this case, it represents a volume feature of interest (red) as well as temporal information from associated particle trajectories.

On the other hand, we can use the particle data in the opposing reference frame as our origin as represented in Fig. 2. As shown in the left side of the figure, we identify a particular particle at a particular timestep of interest $p_i(t_0)$ and use its location as an origin. We can then explore the temporal neighborhood of its associated volume cell. This can be represented as a distribution of Eulerian values (stored as a vector/array) or as aggregated information accumulated throughout the temporal window. An example of aggregated information could be a representation of the total amount of fluctuation in the variable throughout

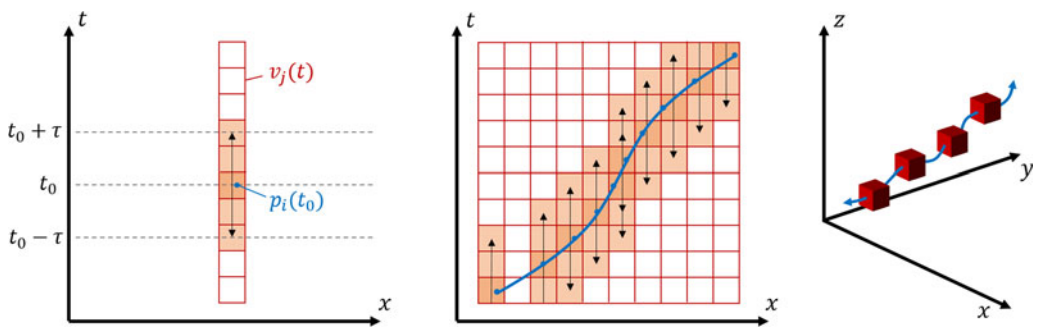

Fig. 2. *Left)* Using a particle (blue) as an origin and studying the temporal neighborhood of its associated volume grid cell (red). *Middle)* Extending this concept to a group of particles (e.g., a trajectory of points) and studying the temporal neighborhood of all associated volume cells. Note that one of the columns is empty because there is no point along the trajectory associated with the volume cell (the particle moved more than one cell to the right between subsequent timesteps). *Right)* A 3D depiction of the middle case with time not shown.

the time window. We call the object representing this particular case β and it contains information regarding the particle origin as well as temporal information from the corresponding volume cell.

$$\beta(p_i(t_0)) = \{x(t_0), s(t_0), \omega(\tau)\},$$

where ω is defined as a vector of values in a temporal window of size $2\tau + 1$

$$\omega(\tau) = (v_j(t_0 - \tau), \ldots, v_j(t_0), \ldots, v_j(t_0 + \tau)),$$

or as aggregated information accumulated throughout the window

$$\omega(\tau) = \int_{t_0-\tau}^{t_0+\tau} f(v_j(t))dt.$$

Note that since there must be exactly one volume cell associated with a particle, $\omega(\tau)$ only needs to be computed once.

Once again, we can extend this case to consider a group of particles (e.g., a set of particle positions along a trajectory) as an origin and is shown in the middle of the same figure. In this case, β is computed for each particle position and the results are encompassed in the object called $B(P)$ where P is a set of particles. This object contains all the information regarding the origin in the particle frame as well as the temporal information from the volume data

$$B(P) = \begin{cases} X(t) & \text{a vector/array of positions} \\ S(t) & \text{a vector/array of Lagrangian variables} \\ \Omega(\tau) & \text{temporal window information} \end{cases}.$$

Once more, Ω is now a matrix when representing the distribution of values in all the temporal windows or a vector/array when representing aggregated information from each of the temporal windows. Any redundancies that occur when multiple particles are associated to the same volume cell can also be eliminated and is discussed in more detail in Section 3.1.2. The right side of the figure shows a depiction of such an object in 3D space; a trajectory of interest is selected (blue) and the temporal neighborhoods of associated volume cells (red) are analyzed. Note that it is difficult to intuitively visually represent the temporal information from the Eulerian frame in a 3D view. As a result, a new visual representation needs to be developed to handle this type of feature (Section 3.3).

When operating near the temporal boundaries (near the start or end of the available timesteps), we only consider temporal information that is available, sometimes resulting in an uneven window around the chosen origin, t_0. An uneven window may also be of interest to researchers even when not operating near these boundaries. There is also the possibility of setting $\tau = 0$ in case users do not want to extract an entire temporal window and are instead interested in only the corresponding instantaneous values in the opposing reference frame.

3.1.1 Aggregation Functions

As described in the previous section, extracted temporal information from the opposing reference frame can either be stored and visualized as a distribution of values, or aggregated into a collapsed result. An advantage of aggregation is that it can summarize the underlying patterns in a temporal window and makes visual representations less cluttered. It is easier to visually depict a single aggregated value per entity rather than a full distribution of values. For example, one could color a trajectory based on aggregated values from temporal windows in the volume data, or color volume cells based on aggregated values from temporal windows in the particle data. However, the disadvantage is that certain components of information, which only a distribution can show, will always be lost. Thus we provide both options in our implementation.

When choosing to aggregate information from the temporal window, a diverse set of functions can be used to generate results according to the researcher's needs. Besides the standard statistical properties that can be computed from a distribution of values (e.g., mean, range, mode, etc.) there are a number of other potentially useful metrics. For example, a measure of volatility can be useful in quantifying the amount of variation of data values in a temporal window. The Fourier transform is another powerful tool in analyzing time-varying data series since it can extract fluctuations in data values can take on regular periodic patterns.

The above examples can be used to aggregate temporal information in both the volume and particle data. The only difference is that we use Lagrangian variables for a particular particle or Eulerian variables for a particular grid location. However, the particle data also represents a changing location in space, and as a result, forms trajectories with interesting geometric properties. We can therefore aggregate geometric information throughout the temporal neighborhood of a trajectory. For example, we can compute the number of directional changes throughout a trajectory. Studying the number of turns present in a trajectory can identify portions of the flow that have numerous changes in direction (high turbulence) or few changes in direction (zonal flows). Alternatively, we could investigate total length, displacement, vorticity, or use more sophisticated techniques as described in Section 2.1.

Another major advantage of these aggregation techniques is that they can easily be used to cluster/segment the accompanying temporal extraction results. This segmentation result can then be propagated to the opposing reference frame since they are inherently linked through the way they were extracted. Being able to segment results using information from both the Eulerian and Lagrangian representations can lead to more control and detail in feature exploration. This is discussed further in Section 3.2. The overall choice of which techniques are most useful will be dependent on the domain specific area and which parameters are being studied and therefore should be up to the user.

3.1.2 Implementation and Efficient Computation

Extracting and representing the aforementioned spatio-temporal objects ($A(V)$ and $B(P)$) is heavily based on queries of appropriate data subsets. Whether a user chooses to explore a distribution of values or an aggregated result, information from the entire desired temporal window must first be loaded from disk. As described in Section 2.2 we assume that corresponding particles and volume cells can be easily located and use an indexing scheme in our implementation. We also assume that the initial spatio-temporal

origin of interest, such as a volumetric feature or trajectory, has already been identified through the appropriate domain specific means.

Constructing $A(V)$ can be done straightforwardly by iterating over each volume cell in V and querying α. This will extract the temporal information from all corresponding particles within a desired window. However, this is practically inefficient since most scientific applications tend to store timesteps as separate files. Instead, we choose to first generate a complete list of particles associated with the volume feature and then iterate through all timesteps within the temporal window. This extracts all necessary information from each timestep/file at once and exploits memory locality.

Constructing $B(P)$ efficiently is less straightforward because of the free moving nature of the Lagrangian particle data. First, multiple particles can be associated with a particular volume cell. These redundancies should be eliminated since there is no need to query temporal information from the same volume cell more than once. Second, since P can represent a set of points along a trajectory, these redundancies may have different temporal origins (e.g., a trajectory passes through a volume cell, turns around, and passes through the same volume cell again at a later point in time). As a result, we generate a complete list of associated volume cells and eliminate any redundancies in both time and space. In other words, if the same volume cell needs to be queried more than once, but at different temporal origins, any redundant overlap in timesteps are first eliminated. We can then iterate through all necessary timesteps and once again exploit memory locality.

Once the desired spatio-temporal object has been queried, it is locally stored in memory for easy access. Since each object is loaded directly from disk, this technique is suitable for use in large-scale datasets since the total computation time is dependent solely on the size of the extracted spatio-temporal object, rather than the full data.

3.1.3 Temporal Resolution Differences

It is possible for a simulation to output particle and volume data at different temporal frequencies. When this is the case, some timesteps will be missing one of the two representations. This is not an issue as long as the chosen spatio-temporal origin (the point(s) in time through which connections between the reference frames are made) has both particle and volume data available. Any extracted temporal windows can then be constructed using the temporal resolution of the available data. However, this can become a limitation as users may want to choose a specific origin point which does not have available information in both frames.

As a result, we resort to interpolation schemes to fill in any missing components. In the case where volume data is missing, Eulerian information is interpolated into the missing temporal region before extracting the spatio-temporal object. While we use linear interpolation in our implementation, other higher order techniques are certainly a possibility. In the case where particle data is missing, Lagrangian information, including the position of each particle, is interpolated into the missing temporal region. We leave it up to the users in their domain specific field to determine if the errors associated with interpolation exceed the value of choosing a spatio-temporal origin at a specific point in time.

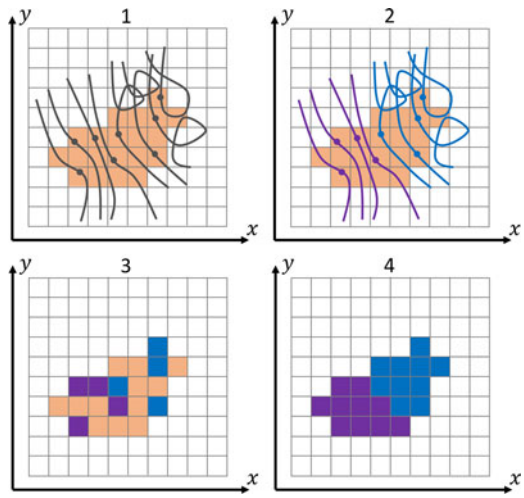

Fig. 3. Segmenting a volume feature according to which volume cells correspond to a particular trajectory type. 1) A feature of interest in beige in the Eulerian representation with a set of associated trajectories in the Lagrangian representation. The dots show points along a trajectory that match the timestep of the volume feature. 2) The trajectories are clustered into groups based on their geometric properties. 3) Clusters are propagated to volume cells at the spatio-temporal origin. 4) Region growing techniques fill in any gaps.

3.2 Spatio-Temporal Segmentation Techniques

We can use the computed information from temporal windows in either reference frames to enhance segmentation results in the opposing frame. Fig. 3 shows an example of using the geometric properties of associated trajectories to segment a volumetric feature. Users would first identify a particular volume feature of interest and extract temporal information from any associated particle in the form of trajectories. Next, the trajectories are analyzed namely by using aggregated information from the temporal window. In this case, the trajectories are clustered according to whether it contains a loop. The trajectory clusters can then be propagated to the opposing reference frame through the spatio-temporal origin (the point at which the particles and volume cells have matching timesteps).

If more than one particle is associated with a particular volume cell, the cell acquires the cluster matching the majority of associated particles. If there is an even split, one of the available clusters are assigned randomly. If there are no particles associated with a particular volume cell, it is left initially without an assigned cluster, resulting in unclassified gaps in the volume feature. We therefore employ breadth-first region growing to fill in the unclassified regions. Cluster types are iteratively propagated to unclassified neighboring cells based on a simple majority. If an unclassified cell has an equal number of differently classified neighbors, it is left unclassified for this iteration. This process iterates until all unclassified cells have been classified or if it does not converge, we force any leftover unassigned cells to choose a classification from a random neighbor. This results in a complete segmentation of our original volume feature based on corresponding trajectory properties.

This type of segmentation scheme can also be done in reverse, in which temporal properties in the Eulerian reference frame can be used to classify and segment clusters of particles or trajectories in the Lagrangian reference frame.

For example, one could use a group of particles (along a single trajectory) as a spatio-temporal origin and extract Eulerian information from a temporal window. Each volume cell can be clustered according to aggregated information from the window and then propagate its cluster type onto the trajectory. In the same way that the volume feature was segmented according to time-varying Lagrangian properties in the previous example, this trajectory is segmented according to time-varying Eulerian properties. Note that there is no need to perform the region growing step in the Lagrangian reference frame since every particle (point along the trajectory) will correspond to exactly one volume cell and will therefore always receive a classification.

The ability to refine the original spatio-temporal origin into a more detailed segmentation result can be very advantageous because it not only allows users to generate a classification based on information from both reference frames but also can take temporal variations into account. This dual segmentation scheme gives researchers an extra level of control in identifying the extent of specific simulation properties that was previously unavailable. This is especially true when employing a segmentation of volume features based on corresponding trajectory shapes, since it would be very challenging to project the geometric properties of trajectories into the Eulerian frame otherwise. Furthermore, the temporal window through which aggregated results are computed can be adaptively modified. This can change the segmentation results giving yet another investigative parameter to the user.

3.3 Visual Representations

Users can interact with these spatio-temporal objects primarily through their construction and modification. Adjusting key parameters, such as the spatio-temporal origin, time windows, or aggregation functions will highlight different aspects of the data. Interactive exploration with real time visual feedback is necessary to fully understand the interplay between both reference frames. As a result, each of our spatio-temporal objects needs an effective visual representation which depicts both the spatio-temporal origin and accompanying temporal information in the opposing frame.

Visually representing temporal patterns of trajectories associated with a volume feature can be done straightforwardly because the temporal component of the Lagrangian space varies along a trajectory that can be drawn in 3D space. As a result, we determine that simply using color to render the volumetric feature and trajectories in the same 3D space is sufficient to visually comprehend this spatio-temporal object. We include the option to toggle each data type on/off so they can be viewed separately, reducing occlusion.

On the other hand, visually representing temporal patterns of volume cells associated with a trajectory requires a new design because it is extremely difficult to add a temporal dimension to the stationary volume cells in a 3D space. In our design, we use two linked views to intuitively represent all aspects of this type of spatio-temporal object as shown in Fig. 4. A trajectory of interest (the spatio-temporal origin) and its geometric shape can be explored in a 3D view on top. On the bottom, a separate view projects the trajectory flat along a horizontal axis.

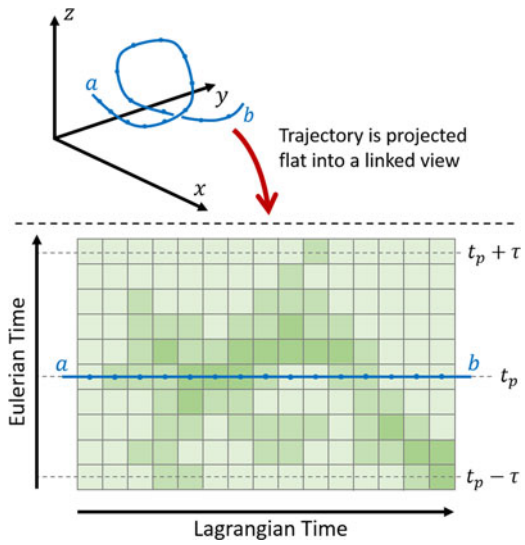

Fig. 4. *Top)* A 3D view of a selected trajectory increasing in time from point A to point B. *Bottom)* A flat projection of the same trajectory with an associated Eulerian temporal neighborhood visible as a "heatmap" in green. Each column in the heatmap represents a different volume cell with time varying vertically.

A temporal neighborhood of associated volume cells in the Eulerian representation can be seen as a "heatmap" in green. In this case, t_p represents the timestep of a particular particle position, which changes along the trajectory. The distribution of associated Eulerian values in their temporal windows are represented vertically with each column corresponding to a different volume cell. As a result, all cells drawn below the trajectory represent events that occur before the particle reached that location in the domain, while all cells drawn above the trajectory represent events that occur afterwards.

In this way, users can explore temporal distributions for both a trajectory and all associated volume cells simultaneously in an intuitive and low clutter manner. Note that we choose not to interpolate values in the heatmap space and draw them as discrete colored squares. This is because it is possible for a fast moving particle (coupled with a low time resolution) to move several volume cells away in one consecutive timestep. Interpolating between non-neighboring cells in the heatmap may lead to misleading results and may remove any visible discontinuities, which could be considered useful information. Also note that if a particle remains associated with the same volume cell between two consecutive timesteps, we still draw both columns in the heatmap as each will represent a different time window (offset by one timestep).

4 Results

We test these techniques using real world applications in the fields of combustion research and accelerator physics to justify their usefulness when analyzing datasets that contain both a particle and volume based component. Fig. 5 shows an example of the volume portion of the combustion dataset and the particle portion of the accelerator dataset. In these examples, we try to focus on specific insights into the data that could not normally be made without investigating spatio-temporal properties in both reference frames.

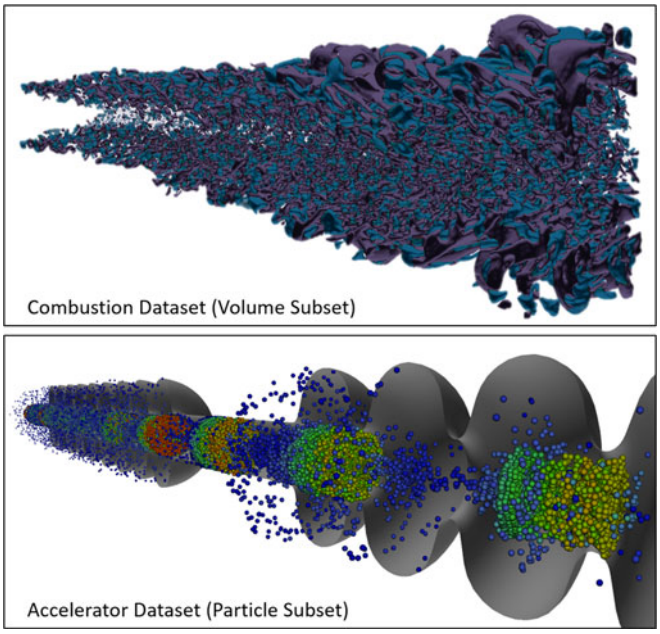

Fig. 5. An overview of each of the test datasets used in our case studies. *Top)* A volume rendering of prominent flow feature types in the combustion dataset. *Bottom)* A point-sprite rendering of particles in the accelerator dataset colored according to their momentum.

4.1 Combustion Dataset

Our first case study comes from the field of combustion and utilizes data from S3D [3], a large-scale combustion simulation developed by scientists at Sandia National Labs. Through these simulations, researchers are investigating phenomena such as autoignition and other processes involved in and leading up to combustion. Research in this field is essential in developing next generation engines of high efficiency. As a result, the temporal component of produced data is very important in this research area. In order to investigate the cause of various phenomena, a complete understanding of the system before and during the event is required.

The volume data we choose to investigate lies on a structured grid and represents a set of different flow classifications that were determined by computing a local rate-of-deformation tensor from the underlying vector field [36]. This generates a topological description of the various flow structures that make up the combustion jet. The particle data represents a set of massless tracer particles which are advected throughout the flow and can record the relative mass fractions of various chemical components as they diffuse through the domain. The rate of change of these mass fractions as well as the geometric structure of these trajectories help to describe underlying processes in the simulation. We can use our method to couple the two data types by extracting a spatio-temporal object.

A topological volume feature representing a FS/S (focusing stretching stable) classification type is identified and is used as our spatio-temporal origin in this example. The temporal information from any corresponding particles (trajectories) is extracted as well and can be seen in Fig. 6a. The volume feature is shown as a connected set of gray volume cells drawn as cubes. The trajectories are also drawn in the 3D space as lines colored according to their hydroxide (OH) content with time progressing forwards as the trajectories move towards the top-left of the image. We choose hydroxide content since this chemical is a common byproduct of combustion. By identifying increases in hydroxide content, we can also identify which trajectories are moving towards a burning region of the combustion jet. From the image, it is clear that some trajectories undergo a rapid increase in hydroxide content while others do not, even though they all originated from the same topological feature.

We can continue this exploration by clustering the trajectories into two groups based on their future hydroxide content as shown in Fig. 6b. Red trajectories indicate a high future content while blue trajectories indicate a low future content. Fig. 6c shows a phase space plot of hydroxide content over time as well as the two corresponding clusters. We can use our spatio-temporal segmentation techniques to propagate the trajectory clustering to the volume feature and segmenting it into two groups as well potentially giving further insights into the future dynamics of this particular

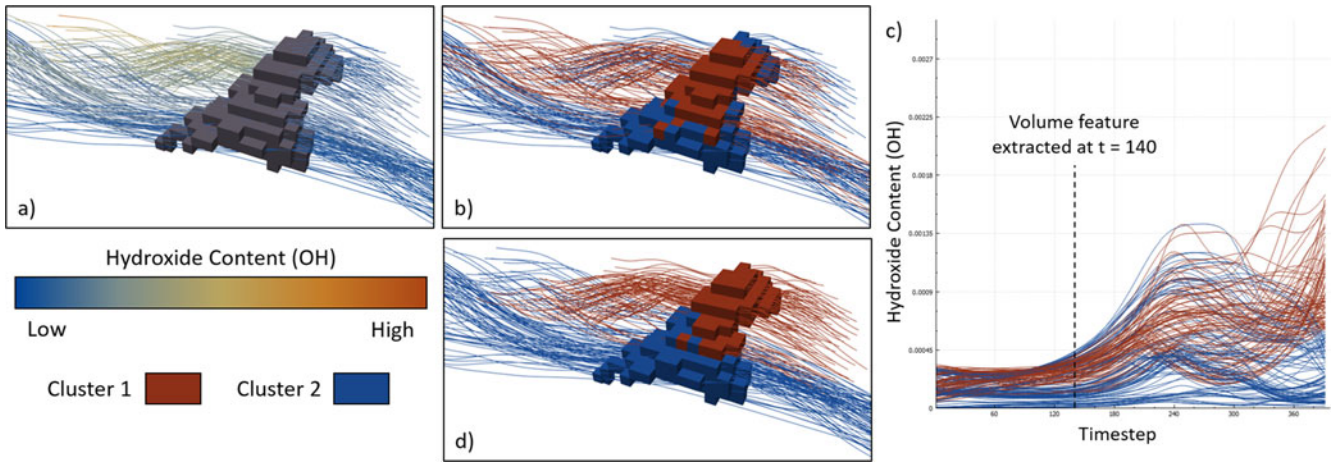

Fig. 6. a) An example of spatio-temporal feature extracted from the combustion dataset. A volume feature of type FS/S is shown as a set of gray volume cells drawn as cubes. Corresponding trajectories colored according to their hydroxide content. b) Trajectories are clustered according their future hydroxide content with red indicating a high content and blue indicating a low content. The same clustering is then propagated back to the volume feature. c) A phase space plot showing the hydroxide content of the associated trajectories over time as well as the two corresponding clusters. d) Clustering trajectories according to their overall length with red indicating short trajectories and blue indicating long trajectories. This clustering is propagated back to the volume feature and reveals a different pattern from (b).

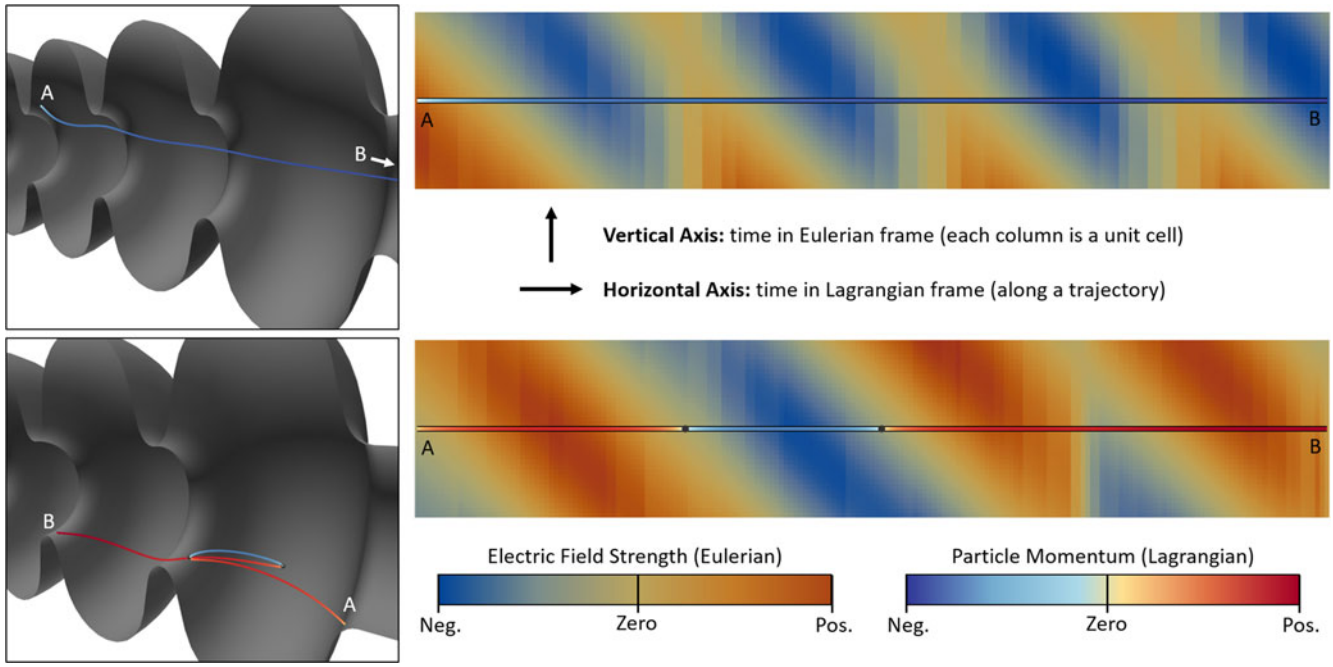

Fig. 7. Two examples of spatio-temporal features extracted from the accelerator dataset with the 3D representation of the origin trajectory to the left and the projected trajectory over the Eulerian heatmap to the right. *Top)* A particle is generated a point (A) and accelerates in the the negative direction (to the right of the image) in phase with the alternating electric field. Each time the particle enters a new cavity, the electric field in that location matches the motion of the particle allowing it to travel straight. *Bottom)* A particle is generated at point (A) and begins to accelerate in the positive direction. Since it does not reach the next cavity in time, it reverses direction when the electric field changes sign. This instability eventually sets the particle on a path towards the device wall resulting in a collision at point (B).

flow feature. Such an analysis would not normally be possible without fully investigating spatio-temporal properties in both reference frames since all the particles exhibit a low hydroxide content at the timestep corresponding to the volume feature. It is not until several timesteps later that the future dynamics of the system become clear.

Alternatively, we can cluster the trajectories using a geometric property. Fig. 6d shows trajectories clustered according to their overall length (an aggregated property over the time window) with shorter trajectories in red and longer trajectories in blue. Once again this clustering is propagated back to the volume feature itself and results in a different segmentation result from our previous example. This can indicate the underlying strength of surrounding flow patterns at more than just one instant in time. Note that we choose to use a cube like depiction of volume cells to make it clear which cells become classified into their respective clusters. Other visual methods such as using a smooth boundary or an isosurface are a possibility as well, and may be preferable with larger features.

4.2 Accelerator Dataset

Our second case study comes from the field of accelerator physics and utilizes data from ACE3P [2], which is used to simulate electromagnetic dynamics in particle accelerators. In this example, we study the phenomenon of dark current in cryomodule devices, which are a set of resonating cavities driven by an RF antenna. Electromagnetic field fluctuations build in the cavities and can be used to accelerate charged particles. Researchers are interested in the phenomenon of dark current in which charged particles are emitted from cavity surfaces. These particles interact with the oscillating electromagnetic field and, depending on when and where they are emitted, can build momentum and collide with the device walls causing damage. A complete understanding of the motion of these particles as well as the fluctuating magnetic field in both space and time is necessary to study this phenomenon.

The volume data in this case lies on an unstructured mesh resembling the cryomodule device. In the Eulerian space, we investigate vector field data resembling the electric field strength and focus on the component that points along the length of the accelerator device. The particle data represents charged particles that are emitted from cavity surfaces at key points throughout the simulation. In the Lagrangian space, we investigate another vector quantity, the momentum of the particle, and also focus on the component that points along the accelerator device. By correlating patterns between these two data types in both time and space we can gain a deeper understanding of the role dark current plays in these devices.

In this case study, we look at the other type of spatio-temporal feature that can be extracted. Trajectories are used as our spatio-temporal origin and were chosen based on how many times the particle changed its direction of motion. Next, temporal information for corresponding volume cells is extracted to help explain their behavior. For this type of spatio-temporal feature we utilize our new visual representation described previously in Section 3.3 and can be seen in Fig. 7. One view shows the 3D shape of the trajectory in the physical domain colored according to a Lagrangian variable (momentum). The other view shows a flat projection of the trajectory ($\sim$200 timesteps) horizontally over a heatmap representing a corresponding Eulerian variable (electric field strength). Each column represents

variations in a volume cell over a time window (~30 timesteps) centered at the time the particle passed through it.

The top of Fig. 7 shows an example of a trajectory with no directional changes and represents a particle emitted at point (A). At the time/location it is emitted, the electric field is just beginning to flip to a negative value, causing the particle to accelerate in a negative direction (towards the right of the image). When the particle enters the next cavity, the electric field there also begins to flip to a negative value, causing the particle to continue gaining momentum in that direction. From the visualization, one can see that this particle travels in phase with the alternating electric field allowing it to safely enter and join the accelerator beam. Each time the particle enters a new cavity, the electric field flips sign and matches the direction of the particle motion causing it to accelerate in a straight line. This is a desirable effect in the cryomodule device.

The bottom of Fig. 7 shows an example of an undesirable effect. A particle is emitted at point (A) at a time and location where the electric field is just beginning to flip to a positive value. This causes the particle to accelerate in a positive direction (towards the left of the image). Before the particle moves onto the next cavity, the electric field then flips to a negative value causing the particle to stop and change direction (as indicated by a black dot). Once the electric field flips to a positive value again, the particle changes direction once more and once again travels in the positive direction. This time the particle is able to move into the next cavity in time and enters just as the electric field there switches to a positive value. This causes the particle to travel in phase in the positive direction and begin gaining momentum. However, the particle soon collides with the surface of the device at point (B) and is destroyed. This is likely due to the initial instability caused when it reversed direction which set it on a angled course towards the wall.

Overall, such a visualization allows one to simultaneously visualize both the change of the electric field as well as the motion of the particle. This makes it easy to not only visualize how these two attributes are closely synchronized with one another but also make predictions about altered initial conditions. For example, in the top image, it is clear that if the particle was emitted a few timesteps earlier, it would have encountered a still positive electric field value preventing it from accelerating quickly enough to match the phase of the alternating field in each cavity. This could lead to a similar fate as the one experienced by the particle in the bottom image. Insights like these would be very difficult to identify using traditional means of exploring each of the reference fames separately.

5 Discussion

The above results demonstrate the usefulness of being able to extract and explore the properties of spatio-temporal features in both a particle and volume reference frame. By investigating the interplay between these two data types and how they evolve over time, researchers can study their data from new perspectives, facilitating scientific insights. The techniques presented here aim to generalize the way these features can be extracted and presented so that they can apply to a large variety of applications. As with all research endeavors, there are a number of ways in which these ideas can be modified or expanded to broaden their scope.

5.1 Alternate Types of Spatio-Temporal Objects

In the examples presented, we focused on two types of spatio-temporal objects: one which includes a volume feature as an origin with associated temporal variations in the particle data and another which includes a trajectory as an origin with associated temporal variations in the volume data. We choose these because they best demonstrate the immediate benefits of our approach. However, this does not necessarily encompass all the types of spatio-temporal objects that our generalized technique is able to explore.

When using the volume reference frame as a spatio-temporal origin, we focused on a set of volume cells that make up a connected volume feature. However, groups of disconnected volume cells exhibiting interesting properties could be used instead, allowing users to investigate correlations between the both data types through the domain globally. Moreover, one could select volume cells to represent abstract shapes, such as within a cube or sphere, transforming this technique into a spatial trajectory selection tool.

When using the particle reference frame as a spatio-temporal origin, we focused on a set of particles that defined a single trajectory. Alternatively, one could use any group of particles to extract a spatio-temporal object. This could include sets of particles that occupy a specific region of the domain or particles that all exhibit similar properties in their Lagrangian variables. The one disadvantage to selecting particles in this manner is that it is more difficult to visually interpret the time varying results of the corresponding Eulerian data. It is still possible to use the heatmap representation but since the particles have no predefined ordering (like they did as a trajectory) the sorting of each of the heatmap columns becomes arbitrary. It may be possible to order the particles/corresponding volume cells based on either their similarity or 3D proximity to one another, but that remains a topic for future research.

Lastly, there is the possibility of using these techniques to study data types outside the realm of scientific simulations. This can be done by coupling point and field-based representations that are experimentally collected in the real world. Movement data is a common example of a time varying point-based representation as the entities in question (e.g., vehicles, marine wildlife, etc.) can independently maneuver in a Lagrangian manner. This can be coupled with a set of Eulerian-like measurements taken at fixed geospatial locations (e.g., air pollution levels, ocean temperatures, etc.).

5.2 Limitations and Future Plans

Besides expanding upon the Eulerian heatmap, there are some other limitations to this work which we plan to address through future research. First, this method assumes that the spatio-temporal origin (either a group of volume cells or a group of particles) has already been identified through some existing means. We find that this assumption is initially acceptable since there are numerous approaches that scientists already use in their regular workflow to select

data subsets for analysis and exploration. However, this raises the question of how we can better integrate these existing selection methods with our technique. Instead of using two steps to generate the spatio-temporal feature, the process can become more streamlined by developing a brand new selection scheme that manages both reference frames at the same time from the very beginning.

Next, while this method does extract a spatio-temporal object which represents information from both reference frames simultaneously, it is sometimes limited to representing a single point in time in one reference frame while representing an entire temporal window in the other. For example, using a trajectory as a spatio-temporal origin will extract a feature in which time varies in both an Eulerian and Lagrangian sense, whereas using a volume feature as the origin will only extract time varying properties in the particle reference frame. In other words, one of the data types is sometimes missing a temporal component in the final visualization. The current implementation of this technique can still address this limitation as long as multiple spatio-temporal features are extracted and visually presented separately. In the future, we plan to extend our general description of spatio-temporal objects and work towards a method to intuitively comprehend patterns in multiple temporal windows in both spaces simultaneously. Achieving this presents challenges not only in the extraction process but also in how to visually present the results in an understandable manner.

Lastly, future work will focus on studying the evolution of these spatio-temporal features in both reference frames. Currently, these features are extracted with specific temporal windows in mind, and any analysis is targeted towards studying patterns within these temporal windows. However, "sliding" the center of these windows throughout the entire simulation time could potentially reveal an interesting set of dynamics buried within simulation datasets. In order words, how can we explore the spatio-temporal evolution of our extracted spatio-temporal features? Due to the already complex nature of visually representing information embedded within a "static" spatio-temporal feature, such a technique would likely need to employ an animation-based mechanism to describe the evolution of these features. By recognizing how components within a spatio-temporal feature evolve in conjunction with one another, researchers can better understand the interplay between the two formats.

6 Conclusion

Overall, this work focuses on enhancing the exploration of temporal neighborhoods in both the Eulerian and Lagrangian reference frames. As more simulations utilize the advantages of each representation, more sophisticated visualization tools that can explore the spatial and temporal interplay between these data types become essential. By allowing each representation to support one another in both space and time, we can provide the means for researchers to explore their datasets in new visual and analytical ways. Furthermore, we extend these concepts to support a more elaborate segmentation technique that is now able to use information from both reference frames, leading to more control and detail in feature exploration. Case studies from real world scientific datasets demonstrate the immediate practicality of our design which supports new forms of exploration in a large variety of scientific and other fields.

Acknowledgments

We would like to thank Sandia National Laboratory for the combustion dataset and SLAC National Laboratory for the accelerator dataset. This research has been sponsored in part by the U.S. Department of Energy via grants DE-SC0007443 and DE-SC0012610, and the National Science Foundation via grant IIS-1320229.

References

[1] M. Adams, S.-H. Ku, P. Worley, E. D'Azevedo, J. Cummings, and C.-S. Chang, "Scaling to 150K cores: Recent algorithm and performance engineering developments enabling XGC1 to run at scale," *J. Phys. Conf. Series*, vol. 180, no. 1, 2009, Art. no. 012036.

[2] O. Kononenko, L. Ge, K. Ko, Z. Li, C. K. Ng, and L. Xiao, "Progress on the multiphysics capabilities of the parallel electromagnetic ACE3P simulation suite," in *Proc. 31st Int. Rev. Progress Appl. Comput. Electromagn.*, Mar. 2015, pp. 1–2.

[3] C. S. Yoo, E. Richardson, R. Sankaran, and J. Chen, "A DNS study on the stabilization mechanism of a turbulent lifted ethylene jet flame in highly-heated coflow," *Proc. Combustion Inst.*, vol. 33, no. 1, pp. 1619–1627, Oct. 2011.

[4] B. Jonsson, J. Salisbury, and A. Mahadevan, "Extending the use and interpretation of ocean satellite data using lagrangian modelling," *Int. J. Remote Sens.*, vol. 30, no. 13, pp. 3332–3341, Jul. 2009.

[5] S. Patkar, M. Aanjaneya, D. Karpman, and R. Fedkiw, "A hybrid lagrangian-eulerian formulation for bubble generation and dynamics," in *Proc. ACM SIGGRAPH/Eurographics Symp. Comput. Animation*, 2013, pp. 105–114.

[6] A. Agranovsky, D. Camp, C. Garth, E. W. Bethel, K. I. Joy, and H. Childs, "Improved post hoc flow analysis via lagrangian representations," in *Proc. Large Data Anal. Vis. Symp.*, Nov. 2014, pp. 67–75.

[7] N. Sakamoto, J. Nonaka, K. Koyamada, and S. Tanaka, "Particle-based volume rendering," in *Proc. Asia-Pacific Symp. Vis.*, Feb. 2007, pp. 129–132.

[8] R. van Pelt, A. V. i Bartroli, and H. van de Wetering, "Illustrative volume visualization using GPU-based particle systems," *IEEE Trans. Vis. Comput. Graph.*, vol. 16, no. 4, pp. 571–582, Jul./Aug. 2010.

[9] F. Yang, Q. Li, D. Xiang, Y. Cao, and J. Tian, "A versatile optical model for hybrid rendering of volume data," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 6, pp. 925–937, Jun. 2012.

[10] C. Stoll, S. Gumhold, and H. P. Seidel, "Visualization with stylized line primitives," in *Proc. IEEE Vis.*, Oct. 2005, pp. 695–702.

[11] K. Burger, P. Kondratieva, J. Kruger, and R. Westermann, "Importance-driven particle techniques for flow visualization," in *Proc. IEEE Pacific Vis. Symp.*, Mar. 2008, pp. 71–78.

[12] R. M. Kirby, H. Marmanis, and D. H. Laidlaw, "Visualizing multivalued data from 2D incompressible flows using concepts from painting," in *Proc. Conf. Vis.*, 1999, pp. 333–340.

[13] T. Urness, V. Interrante, I. Marusic, E. Longmire, and B. Ganapathisubramani, "Effectively visualizing multi-valued flow data using color and texture," in *Proc. IEEE Vis.*, 2003, pp. 115–121.

[14] A. Brambilla, R. Carnecky, R. Peikert, I. Viola, and H. Hauser, "Illustrative flow visualization: State of the art, trends and challenges," in *Proc. Eurographics State Art Rep.*, 2012, pp. 75–94.

[15] F. Sauer, H. Yu, and K. L. Ma, "Trajectory-based flow feature tracking in joint particle/volume datasets," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 12, pp. 2565–2574, Dec. 2014.

[16] J. Chandler, H. Obermaier, and K. I. Joy, "Illustrative rendering of particle systems," in *Proc. Vis. Model. Vis.*, 2013, pp. 177–185.

[17] T. Salzbrunn, C. Garth, G. Scheuermann, and J. Meyer, "Pathline predicates and unsteady flow structures," *Visual Comput.*, vol. 24, no. 12, pp. 1039–1051, 2008.

[18] S. Barakat, C. Garth, and X. Tricoche, "Interactive computation and rendering of finite-time Lyapunov exponent fields," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 8, pp. 1368–1380, Aug. 2012.

[19] C. Wang, H. Yu, and K.-L. Ma, "Importance-driven time-varying data visualization," *IEEE Trans. Vis. Comput. Graph.*, vol. 14, no. 6, pp. 1547–1554, Nov. 2008.
[20] J. Woodring and H.-W. Shen, "Multi-variate, time varying, and comparative visualization with contextual cues," *IEEE Trans. Vis. Comput. Graph.*, vol. 12, no. 5, pp. 909–916, Sep. 2006.
[21] J.-P. Balabanian, I. Viola, T. Moller, and E. Groller, "Temporal styles for time-varying volume data," in *Proc. 4th Int. Symp. 3D Data Process. Vis. Trans.*, Jun. 2008, pp. 81–89.
[22] K.-L. Ma, "Visualizing time-varying volume data," *Comput. Sci. Eng.*, vol. 5, Mar. 2003, Art. no. 34.
[23] T. Peterka, et al., "A study of parallel particle tracing for steady-state and time-varying flow fields," in *Proc. Parallel Distrib. Process. Symp.*, May 2011, pp. 580–591.
[24] S. Marchesin, C.-K. Chen, C. Ho, and K.-L. Ma, "View-dependent streamlines for 3D vector fields," *IEEE Trans. Vis. Comput. Graph.*, vol. 16, no. 6, pp. 1578–1586, Nov. 2010.
[25] Y. Li, C. Wang, and C.-K. Shene, "Streamline similarity analysis using bag-of-features," *Proc. SPIE*, vol. 9017, pp. 90 170N–90 170N–12, 2013.
[26] Y. Li, C. Wang, and C.-K. Shene, "Extracting flow features via supervised streamline segmentation," *Comput. Graph.*, vol. 52, pp. 79–92, 2015.
[27] J. Wei, C. Wang, H. Yu, and K.-L. Ma, "A sketch-based interface for classifying and visualizing vector fields," in *Proc. IEEE Pacific Vis. Symp.*, Mar. 2010, pp. 129–136.
[28] G. Andrienko and N. Andrienko, "Spatio-temporal aggregation for visual analysis of movements," in *Proc. IEEE Symp. Visual Anal. Sci. Technol.*, Oct. 2008, pp. 51–58.
[29] Z. Wang, et al., "Visual exploration of sparse traffic trajectory data," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 12, pp. 1813–1822, Dec. 2014.
[30] A. Diehl, et al., "Visual analysis of spatio-temporal data: Applications in weather forecasting," *Comput. Graph. Forum*, vol. 34, no. 3, pp. 381–390, 2015.
[31] A. Unger, S. Schulte, V. Klemann, and D. Dransch, "A visual analysis concept for the validation of geoscientific simulation models," *IEEE Trans. Vis. Comput. Graph.*, vol. 18, no. 12, pp. 2216–2225, Dec. 2012.
[32] N. Andrienko, G. Andrienko, and P. Gatalsky, "Exploratory spatio-temporal visualization: An analytical review," *J. Visual Languages Comput.*, vol. 14, no. 6, pp. 503–541, 2003.
[33] R. Xu, L. Kang, and H. Tian, "A G-Octree based fast collision detection for large-scale particle systems," in *Proc. Int. Conf. Comput. Sci. Electron. Eng.*, 2012, pp. 269–273.
[34] C. Garth and K. I. Joy, "Fast, memory-efficient cell location in unstructured grids for visualization," *IEEE Trans. Vis. Comput. Graph.*, vol. 16, no. 6, pp. 1541–1550, Nov. 2010.
[35] C. Wächter and A. Keller, "Instant ray tracing: The bounding interval hierarchy," in *Proc. 17th Eurographics Conf. Rendering Tech.*, 2006, pp. 139–149.
[36] M. S. Chong, A. E. Perry, and B. J. Cantwell, "A general classification of three dimensional flow fields," *Phys. Fluids*, vol. 2, no. 5, pp. 765–77, 1990.

Franz Sauer received the BS degree in physics from the California Institute of Technology. He is a fifth-year graduate student with the University of California, Davis, studying computer science and scientific visualization under Kwan-Liu Ma. His research interests include data visualization, large-scale scientific simulations, computer graphics, and physics. He is a member of the IEEE.

Kwan-Liu Ma received the PhD degree in computer science from the University of Utah. He is a professor of computer science and the chair of the Graduate Group in Computer Science (GGCS), University of California, Davis, where he leads the VIDi research group and directs the UC Davis Center for Visualization. His research interests include visualization, high-performance computing, and user interface design. He is a fellow of the IEEE.