Finite-Time Lyapunov Exponents and Lagrangian Coherent Structures in Uncertain Unsteady Flows

Hanqi Guo, *Member, IEEE,* Wenbin He, Tom Peterka, *Member, IEEE,* Han-Wei Shen, *Member, IEEE,* Scott M. Collis, and Jonathan J. Helmus

Abstract—The objective of this paper is to understand transport behavior in uncertain time-varying flow fields by redefining the finite-time Lyapunov exponent (FTLE) and Lagrangian coherent structure (LCS) as stochastic counterparts of their traditional deterministic definitions. Three new concepts are introduced: the distribution of the FTLE (D-FTLE), the FTLE of distributions (FTLE-D), and uncertain LCS (U-LCS). The D-FTLE is the probability density function of FTLE values for every spatiotemporal location, which can be visualized with different statistical measurements. The FTLE-D extends the deterministic FTLE by measuring the divergence of particle distributions. It gives a statistical overview of how transport behaviors vary in neighborhood locations. The U-LCS, the probabilities of finding LCSs over the domain, can be extracted with stochastic ridge finding and density estimation algorithms. We show that our approach produces better results than existing variance-based methods do. Our experiments also show that the combination of D-FTLE, FTLE-D, and U-LCS can help users understand transport behaviors and find separatrices in ensemble simulations of atmospheric processes.

Index Terms—Uncertain flow visualization, stochastic particle tracing, Lagrangian coherent structures.

1 Introduction

Uncertain data is widespread in various scientific and engineering domains, such as computational fluid dynamics, aerodynamics, climate, and weather research. Instead of deterministic velocity vectors, an uncertain flow field is usually represented by distributions that are derived from experiments, algorithms, or interpolation. Although the topic of uncertainty is being extensively studied in some of the sciences named above, the visualization and analysis of uncertainty still remain grand challenges in the community.

Our focus in this paper is visualizing and analyzing transport behavior in uncertain time-varying flow fields. Although flow visualization is an established research topic, with geometry-, texture-, and topology-based methods, the visualization and analysis of uncertain datasets have not been well developed. For 2D and small-scale 3D uncertain datasets, the existing techniques encode uncertainties as additional visual channels, such as glyphs [1], [2] and textures [3]. Recently, vector field topology methods have been extended to uncertain vector fields [4], [5], but they are not directly applicable to time-varying datasets because vector field topology theory is built on streamlines, not pathlines. Uncertain time-varying datasets still are a major challenge in flow visualization.

In this study, we extend a well-established tool for unsteady flow analysis—finite-time Lyapunov exponent (FTLE)—to a probabilistic framework for analyzing uncertain data. The FTLE was proposed by Haller [6] and has become a standard tool to study transport behaviors in unsteady flow. For a certain finite-time interval, the scalar FTLE value at a given location measures the convergence or divergence rate between neighboring particles in the flow. It is defined as the maximal eigenvalue of the inner product of the gradient of flow maps. The ridges of FTLE fields can be used to derive Lagrangian coherent structures (LCSs) [7]—the boundaries between attracting or repelling particles in the flow. Thus, FTLEs and LCSs can help scientists understand flow transport behaviors.

The motivation of this work is to redefine traditional deterministic FTLE-based analysis pipelines to accommodate uncertainty. Such stochastic formulations will allow climate scientists to quantify the uncertainty of convergent and divergent transport behaviors. This behavior can help scientists understand the uncertainty of derived features such as eddies, flow segmentation, and large-scale teleconnections. We view this problem from two perspectives. One is to quantify the uncertainty of traditional FTLE values, and the other is to measure the uncertainty in convergent and divergent transport behaviors in order to define a single

- *Hanqi Guo is with the Mathematics and Computer Science Division, Argonne National Laboratory, Lemont, IL 60439, USA. E-mail: hguo@anl.gov*
- *Wenbin He is with the Department of Computer Science and Engineering, the Ohio State University, Columbus, OH 43210, USA. E-mail: he.495@buckeyemail.osu.edu*
- *Tom Peterka is with the Mathematics and Computer Science Division, Argonne National Laboratory, Lemont, IL 60439, USA. E-mail: tpeterka@mcs.anl.gov*
- *Han-Wei Shen is with the Department of Computer Science and Engineering, the Ohio State University, Columbus, OH 43210, USA. E-mail: shen.94@osu.edu*
- *Scott M. Collis is with the Environmental Science Division, Argonne National Laboratory, Lemont, IL 60439, USA. E-mail: scollis@anl.gov*
- *Jonathan J. Helmus is with the Environmental Science Division, Argonne National Laboratory, Lemont, IL 60439, USA. E-mail: jhelmus@anl.gov*

Manuscript received September 25, 2015; revised Janurary 1, 2016.

FTLE-like value that captures the underlying uncertainty. The two approaches reveal different aspects of unsteady flow uncertainties. On the one hand, the uncertainty of the FTLE can be represented by probability density functions (PDFs) for different locations, and further statistical analysis and uncertainty quantification of the LCS can be conducted. On the other hand, a generalized FTLE value can provide an overview of major transport behaviors in the uncertain datasets. Both techniques can help users understand the uncertainty of the data, and their intrinsic relationships are discussed in this paper.

Specifically, we propose two concepts: distributions of FTLE (D-FTLE) and FTLE of distributions (FTLE-D). For a given finite time interval τ, D-FTLE is represented as a *distribution field* with $2 + n$ dimensions: FTLE distribution, time, and n spatial dimensions (2 or 3 in our study). For each spatiotemporal location, the FTLE distribution is a 1D PDF of FTLE values. The D-FTLE is a statistical representation of FTLE values in uncertain unsteady flow, and it can be visualized with various statistical measurements interactively.

On the other hand, the FTLE-D, which has the same dimension as a traditional deterministic FTLE, measures differences in the advected particle distributions. It gives a statistical overview of how transport behaviors differ in neighboring locations, and it works better than existing variance-based methods [8] in our experiments.

We can further derive uncertain LCS (U-LCS), which is an n-dimensional PDF of the probability of belonging to an LCS for each spatiotemporal point in the domain. Ridges are extracted with (stochastic) ridge finding and density estimation algorithms. Because analytical solutions are not available (for anything but synthetic datasets) and the derivatives of random variables are extensively involved, we use Monte Carlo stochastic simulations to generate the D-FTLE, FTLE-D, and U-LCS.

We demonstrate the proposed methods in two real-world uncertain unsteady flow datasets from the climate and weather domain. In the first case, we use the output data from Chen et al. [9], which quantifies the uncertainties of temporal downsampling. Because time-varying datasets can be extremely large to store, a common practice is to drop time steps for further analysis; but important information can be lost in this process. Uncertainties are generated by downsampling, and we can analyze such downsampled data with our tools to reveal the uncertain transport behaviors. In the second case, the uncertainties arise from ensemble simulation runs of weather forecasts, and we further visualize the surfaces in the storm regions by extracting the U-LCS. Combined with other visualization techniques, our method can help scientists analyze the uncertainty of the simulation models.

In summary, the contribution of this paper is a novel probabilistic framework for FTLE computation and LCS extraction in uncertain time-varying flow fields that includes

- distributions of FTLE (D-FTLE),
- FTLE of distributions (FTLE-D), and
- a method of compositing ridge surfaces into uncertain LCS (U-LCS) by using a surface density estimation.

The remainder of this paper is organized as follows. Background and basics are discussed in Sections 2 and 3, respectively. The details of the D-FTLE, U-LCS, and FTLE-D are given in Sections 4, 5, and 6, respectively. In Section 7 we describe the implementation and evaluate the performance. Results are discussed in Section 8, followed by the conclusions in Section 9.

2 Background

In this section, we discuss the background concepts needed for this paper and summarize related work on uncertain flow field visualization and FTLE-based flow analysis.

2.1 Uncertain flow field visualization

Visualizing uncertain flow fields is a grand challenge in our community, which involves two major research topics: flow visualization and uncertainty visualization. Three major approaches to flow visualization exist—geometry-based [10], texture-based [11], and topology-based methods [12]. These techniques usually transform deterministic datasets into visualizations used for various analysis tasks. Uncertainty visualization has become a necessary component in this process [13], [14], and much work remains to be done to visualize uncertainty in flow fields. One may classify existing uncertain flow visualization approaches into direct methods and feature-based methods.

Direct methods include glyphs [1] and textures [3] that encode uncertainty with additional visual channels. UFLOW [15] presents a series of visual encoding schemes to visualize the uncertainties arising from different numerical integration methods in particle tracing. Flow radar glyphs [2], which visualize the change of flow directions in a spherical coordinate system, also incorporate uncertainty in the glyphs. However, direct visualization methods are usually limited to 2D or small 3D datasets; and they are not feasible for large, complex, 3D time-varying datasets. Compared with these methods, our work focuses on transforming and aggregating flow field distributions into scalar fields that can be visualized by traditional methods, irrespective of dimensionality or scale.

Feature-based methods extract important features from uncertain data. Usually, this process is done by extending methods used in deterministic flow fields to uncertain data. For example, vortex detectors such as λ_2, Q-criterion, and parallel vectors can be extended to uncertain data [16]. In such techniques, Monte Carlo simulations are typically used to trace the particles and compute the output variables. Extracted vortices are presented as a probability field instead of deterministic regions or vortex lines. The method of Petz et al. [17] is the probabilistic equivalent of local features such as critical points. Our methods also extend deterministic techniques to uncertain datasets and compute the U-LCS as a probability density field.

Recently, Otto et al. [4], [5] investigated the topology of uncertain 2D and 3D steady flow fields. However, vector field topology is not stable and thus is infeasible for time-varying datasets. One method for analyzing unsteady flow topology is FTLE, which can be further used to extract LCSs. A variance-based FTLE-like metric, the so-called FTVA, was

proposed to analyze uncertain vector fields [8], but the metric has two issues. First, it is based on principal component analysis (PCA); thus, the distribution of the advected particles is assumed to be Gaussian, which is not necessarily true. Second, the FTVA gives a single value for each location, which makes it impossible to extract the distribution of the uncertain LCS. Hummel et al. [18] extend the PCA-based variance to measure the particle divergence in ensemble flow fields, but they make the same Gaussian assumptions as in FTVA. In our study, we consider the distribution of FTLE values in uncertain flows, and we produce the U-LCS as a probability distribution.

2.2 Deterministic FTLE and LCS

The most important application of FTLE fields is to find LCSs, which are material surfaces that separate different fluid regions by particle movement behavior. LCSs are usually localized as ridges of the FTLE field [19]. In our work, we quantify the uncertainty of LCS by investigating the distribution of FTLEs.

FTLEs and LCSs are computed as follows. Given a time-varying flow field $\mathbf{v}(\mathbf{x}, t)$, we denote the end position of the pathline seeded at spatiotemporal location $(\mathbf{x}, t)$ by a point in a *flow map* $\phi(\mathbf{x}, t, \tau)$, which is the solution of the initial value problem

$$\frac{\partial \phi(\mathbf{x}, t, \tau)}{\partial \tau} = \mathbf{v}(\phi(\mathbf{x}, t, \tau), t + \tau), \text{ and } \phi(\mathbf{x}, t, 0) = \mathbf{x}, \quad (1)$$

where τ is the advection time. The definition of FTLE is based on the gradient of the flow map

$$\sigma(\mathbf{x}, t, \tau) = \frac{1}{|\tau|} \log \sqrt{\lambda_{\max}((\nabla \phi)^{\intercal} \nabla \phi)}, \quad (2)$$

where ∇ is the gradient operator with respect to $\mathbf{x}$ and $\lambda_{\max}$ computes the maximum eigenvalue of the right Cauchy-Green deformation tensor $\nabla \phi^{\intercal} \nabla \phi$. The ridges of FTLE fields, which are curves and surfaces in 2D and 3D datasets, respectively, are usually considered to be the LCS. Although FTLE ridges are not always an indication of the LCS, this method has been used in a wide range of applications [20]. We will generalize the concepts of FTLE and LCS to uncertain unsteady flow in the following sections.

The FTLE computation is extremely expensive because it requires tracing densely seeded particles in the flow field, which is costly in computation, I/O, and memory resources. Two strategies are used to accelerate FTLE computation: parallelism and approximation. Nouanesengsy et al. [21] present a parallel framework that groups parallel processes by exclusive time spans and pipelines the seeding of pathlines over time intervals, in order to reduce the I/O and synchronization overhead. Guo et al. [22] subdivide flow field data into fine-grained blocks and manage the data access with a (pre)caching parallel key-value store, thereby improving the I/O and memory efficiencies for particle tracing in FTLE computation. A graph-based seed scheduling method is proposed to compute the FTLE on desktop machines in an out-of-core manner [23]. Instead of computing a full-resolution FTLE field by brute force, an alternative way to reduce the computation cost is by approximating the FTLE. For example, an adaptive refinement approach can estimate the FTLE field with sparse samples [24]. Hlawatsch et al. [25] present a hierarchical advection scheme that provides a less accurate but faster solution. Kuhn et al. [26] evaluated various FTLE computation methods. Our work does not depend on the particular FTLE computation method, and we focus instead on deriving the uncertainty from particle tracing and FTLE results.

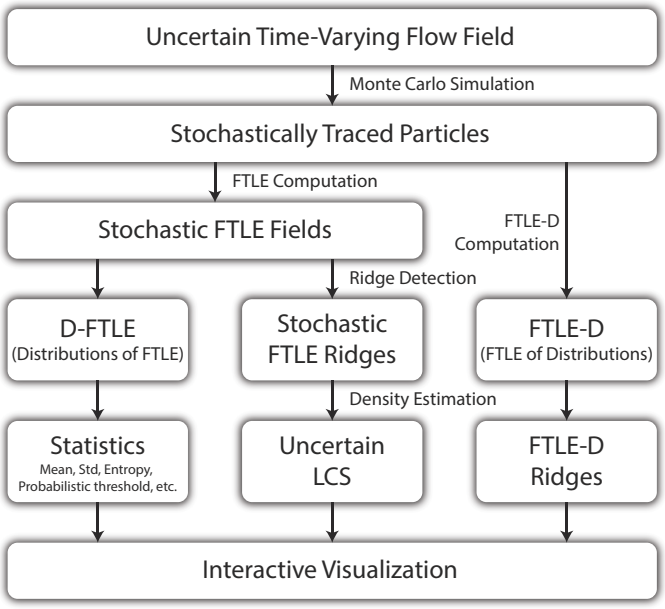


Fig. 1. Pipeline of our methods. A number of Monte Carlo simulation runs are conducted, which trace densely seed particles stochastically. Then D-FTLE, U-LCS, and FTLE-D are computed for interactive visualization.

3 Basics

Figure 1 illustrates our methods. From the input uncertain unsteady flow datasets, particles are advected stochastically (Section 3.2) with a number of Monte Carlo simulation runs. For each run and each spatiotemporal location, one single particle is traced and labeled with the run ID. Our pipeline has two major routes. One is to compute the FTLE for each run individually (so-called stochastic FTLE runs) and then generate the D-FTLE and U-LCS. The other is to compute FTLE-D values for all runs. Statistical measurements of the D-FTLE, such as the mean, standard deviation, entropy, and *statistical thresholding* (Section 4), can be calculated for interactive visualization. The U-LCSs are derived from the ridges in stochastic FTLE fields, and we composite ridges from all runs into a scalar-valued U-LCS field by curve and surface density estimation (Section 5). For the FTLE-D, we also visualize ridges similarly to deterministic LCS. Users can use different tools to visualize the uncertain unsteady flow, in order to find separatrices and understand transport behaviors.

We use a synthetic uncertain double-gyre dataset for illustration in following sections. The original deterministic double-gyre dataset[1] is a closed-form time-varying vector

1. http://mmae.iit.edu/shadden/LCS-tutorial/examples.html

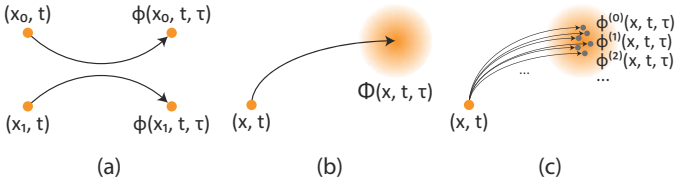


Fig. 2. Deterministic flow map ϕ (a), stochastic flow map Φ (b), and stochastically traced particles $\phi^{(i)}$ (c).

field. We arbitrarily added Gaussian noise to its u and v components of the velocity. More details about further analysis of this dataset are given in Section 8.1.

3.1 Definitions

The visualization and analysis of both the FTLE-D and D-FTLE are based on a *stochastic flow map* (SFM). We use the SFM $\Phi(\mathbf{x}, t, \tau)$ to encode the distribution of advected particles seeded from the spatiotemporal location $(\mathbf{x}, t)$. In a deterministic flow map, each spatiotemporal position contains the ending position of a particle seeded at that position. The stochastic flow map contains a distribution of ending positions instead of a single end position. The PDF of the SFM, $\rho_\Phi(\mathbf{x}, t, \tau; \mathbf{x}')$ ($\mathbf{x}' \in \mathbb{R}^n$), is a function at each position in the flow map, or a *distribution field*. As opposed to a deterministic flow map ϕ, the SFM for a given spatiotemporal location is a random variable obeying a PDF, instead of a single point (as illustrated in Figure 2).

Based on the definition of the SFM, we can extend Eq. 2 to generalize the FTLE to its stochastic version (*stochastic FTLE*):

$$\Sigma(\mathbf{x}, t, \tau) = \frac{1}{|\tau|} \log \sqrt{\lambda_{\max}((\nabla\Phi)^\intercal \nabla\Phi)}, \quad (3)$$

where ∇ is the gradient operator over the space. In this study, we visualize and analyze the distribution of the random variable Σ (the D-FTLE) as ($\rho_\Sigma(\mathbf{x}, t, \tau; \sigma)$, $\sigma \in \mathbb{R}$).

Moreover, we may generalize the definition of LCS to get its stochastic counterpart. For a given spatiotemporal location, the U-LCS value is the probability of being a ridge of a stochastic FTLE:

$$L(\mathbf{x}, t, \tau) = Pr(R(\Sigma(\mathbf{x}, t, \tau)) = 1), \quad (4)$$

where R is the ridge detection operator. In this study, we use C-Ridges [27] definitions for R, because C-Ridges are usually used to extract LCS in previous studies [28].

In addition to the D-FTLE and U-LCS, we compute the FTLE-D, which takes the same form as a deterministic FTLE. Analogous to the FTLE, which is defined on the gradient of the flow map, the FTLE-D characterizes the "gradient" of SFM distributions by measuring the differences of PDFs in neighboring regions. Formally, we define the FTLE-D as

$$\hat{\sigma}(\mathbf{x}, t, \tau) = \frac{1}{|\tau|} \log \sqrt{\lambda_{\max}(E[\nabla\Phi]^\intercal E[\nabla\Phi])}, \quad (5)$$

where $E[\cdot]$ is the expectation operator. We also visualize the ridges of $\hat{\sigma}$ compared with the deterministic LCS. The concepts and computation of the D-FTLE, U-LCS, and FTLE-D are further explained and detailed in the remainder of this paper.

Because analytical solutions of the SFM and its derivatives are unavailable in practice, we conduct a number of stochastic Monte Carlo simulations to generate numerical solutions. The FTLE-D, U-LCS, and D-FTLE are directly estimated from the stochastically traced particles. In the rest of this section, we briefly describe the stochastic particle tracing process.

3.2 Monte Carlo particle tracing

Both the FTLE-D and D-FTLE are based on stochastic particle tracing results. A number of Monte Carlo simulation runs are conducted, and then the traced particles are labeled with the run ID for further use. Formally, we denote the uncertain flow field as $\mathbf{V}(\mathbf{x}, t)$, where $(\mathbf{x}, t)$ is the spatiotemporal location. We wish to know the end location of the particle seeded at $(\mathbf{x}, t)$ after a period of time τ. This process can be turned into a stochastic differential equation system:

$$d\Phi(\mathbf{x}, t, \tau) = \mathbf{V}(\Phi(\mathbf{x}, t, \tau), t + \tau)d\tau + \mathbf{B}(\mathbf{x}, t + \tau)d\xi_\tau, \quad (6)$$

where $\mathbf{B}$ is the disturbance. If $\mathbf{V}$ obeys a Gaussian distribution, Euler-Maruyama methods or stochastic Runge-Kutta methods can be used to solve this system. Because $\mathbf{V}$ is usually non-Gaussian in real-world applications, we use Monte Carlo simulation to estimate the flow map instead:

$$\phi^{(i)}(\mathbf{x}, t, (j + 1)\Delta t) = v^{(i)}(\phi^{(i)}(\mathbf{x}, t, j\Delta t), t + j\Delta t)\Delta t, \quad (7)$$

where i is the Monte Carlo run ID, $\phi^{(i)}$ is the particle position of jth integral step of the ith run, and $\mathbf{v}^{(i)}$ is a random sample of $\mathbf{V}$. Δt is the time for each integral step. The number of runs is adaptively determined in an iterative manner. In each iteration, we conduct a number of runs, and the iteration stops if the output D-FTLE field does not statistically significantly change anymore. Although the Monte Carlo simulation is expensive, the performance could be boosted with general-purpose graphics and other accelerator hardware. We use CUDA and Nvidia GPUs in our implementation.

4 Distributions of FTLE (D-FTLE)

As we defined in Section 3.1, the D-FTLE is a distribution field of scalar FTLE values. For given spatiotemporal location $(\mathbf{x}, t)$ and advection time τ, the D-FTLE is the PDF of the stochastic FTLE Σ. In practice, directly visualizing the D-FTLE $\rho_\Sigma(\mathbf{x}, t, \tau; \sigma)$ is difficult because it is a high-dimensional scalar function defined on $\mathbb{R}^{n+3}$. Visualizing distribution fields has been studied for 2D datasets [29], but it is still challenging to visualize 3D D-FTLE data with two time dimensions t and τ. Instead, we visualize D-FTLE statistics with our tool. Users can also query the distributions at specific points by brushing.

The computation of the D-FTLE is achieved by Monte Carlo simulations. We first compute the FTLE field $\sigma^{(i)}(\mathbf{x}, t, \tau)$ for each individual Monte Carlo run in the stochastic particle tracing process and then bin the FTLE

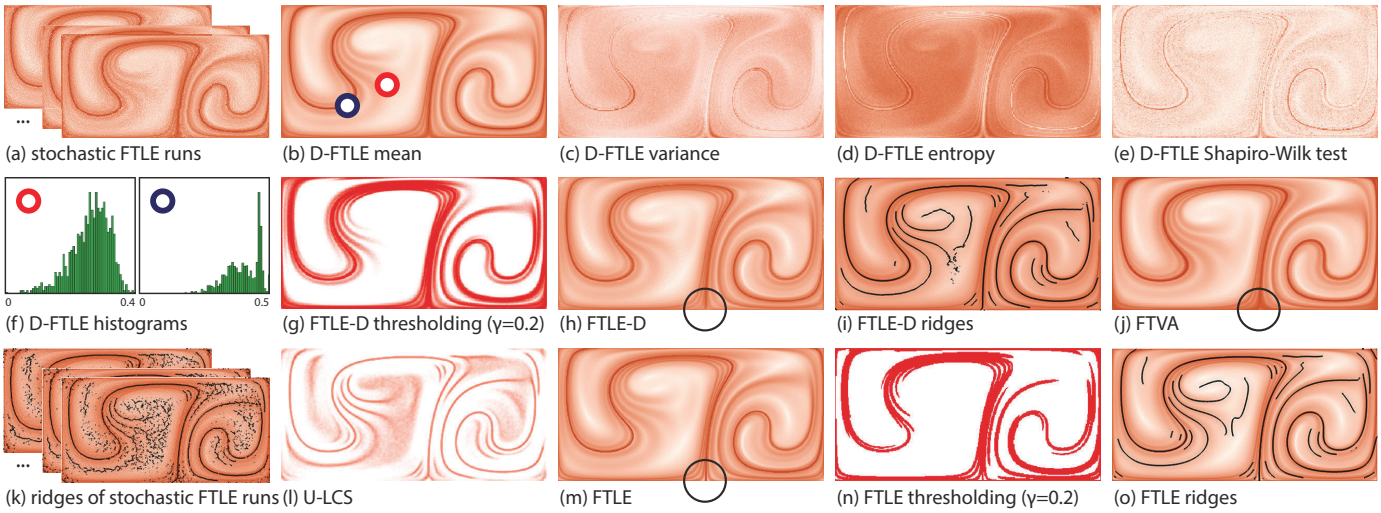


Fig. 3. Visualization of synthetic uncertain (a-m) and deterministic (n-p) double-gyre data with various methods. The time t and advection time τ are 0 and 15, respectively. The darker color indicates the larger values in the images, except that (e) is inversed.

values for each spatiotemporal location into a 1D histogram. The D-FTLE is stored as a high-dimensional array for further interactive visualization.

We provide several statistical measures for visualizing the D-FTLE, including mean, standard deviation, information entropy, normality test (Shapiro-Wilk p-value), and *statistical thresholding* (described below). Among these statistical measures, mean and standard deviation provide basic properties of the distribution; information entropy quantifies the complexity of the distribution. Shapiro-Wilk p-value measures how much the distribution is Gaussian. Figures 3(b)-(e) demonstrate these metrics with synthetic data, and we observe that regions with richer flow features usually have higher entropies and Shapiro-Wilk p-values. Based on the visualizations, users can probe the histogram for specific locations.

Statistical thresholding is another tool for measuring the likelihood of an FTLE value greater than a given threshold. In FTLE-based analysis, higher FTLE values indicate the emergence of LCS. The statistical thresholding generates comparable results to thresholding deterministic FTLE fields. Formally, statistical thresholding is defined as

$$T(\mathbf{x},t,\tau) = Pr(\Sigma|\Sigma(\mathbf{x},t,\tau) \geq \gamma) = 1 - \int_{-\infty}^{\gamma} \rho(\mathbf{x},t,\tau;\sigma)d\sigma, \tag{8}$$

where γ is the given threshold. For a discrete D-FTLE, the integral can be calculated by summing the histogram bins whose values are smaller than γ. Figures 3(g) and (h) present statistical thresholding results for the uncertain synthetic double-gyre data with different thresholds. This result is comparable with the FTLE thresholding from the deterministic data in Figure 3(o). The statistical thresholding visualizes the probability of being high FTLE values, which incorporates the data uncertainty.

5 Uncertain LCS (U-LCS) Extraction

LCSs, surfaces that separate attracting or repelling particles in unsteady flow are usually indicated by FTLE ridges. Depending on the dimensionality, the ridges are curves and surfaces in 2D and 3D datasets, respectively. An uncertain probabilistic version of the LCS was defined in Section 3.1, and we detail the U-LCS computation in this section. In an uncertain setting, the output of LCS-finding algorithms is a probability density field. The values in this field represent the probability of sitting on a ridge curve or surface. Because the gradient of stochastic FTLE fields is involved in the ridge detection, this process is also estimated by a stochastic process. As shown in the pipeline in Figure 1, after getting an ensemble of stochastic FTLE runs, ridges are detected for each Monte Carlo run, and then they are composited into a U-LCS field by density estimation.

The ridge detection algorithm is applied to each Monte Carlo run. The inputs are the individual FTLE runs $\sigma^{(i)}$, and the outputs are the ridges of the scalar fields $R(\sigma^{(i)})$. We follow the previous scale-space methods [28] to extract C-Ridges [27] in the FTLE fields. The minor eigenvectors of Cauchy-Green deformation tensors $\nabla\phi^{(i)\intercal}\nabla\phi^{(i)}$ are used to determine the transverse directions in the ridge detection. We iterate every cell in the grid and then connect the line segments (triangles for surface) into curves (surfaces) with the marching ridges algorithm [30]. Figure 3(l) and Figure 4(a) show the ridge detection results for several FTLE runs.

The ridges from all FTLE runs are then composited into the U-LCS field, which is achieved by curve (surface) density estimation. To do so, we generalize kernel density estimation from discrete points to curves and surfaces in the space. For 1D curves, this problem has been studied by Lampe and Hauser [31] with approximations. However, it is more challenging to estimate densities from curves in the 2D plane and surfaces in 3D space, because convolutions between the kernel functions and curve (surface) patches are usually not analytical. Instead, we propose an alternative solution based on smoothed particle hydrodynamics (SPH) [32] detailed below. In Figure 3(m) and Figure 4(b) show the density estimation results for curves and surfaces, respectively. Although the ridges extracted from single runs contain some artifacts due to randomness of the Monte

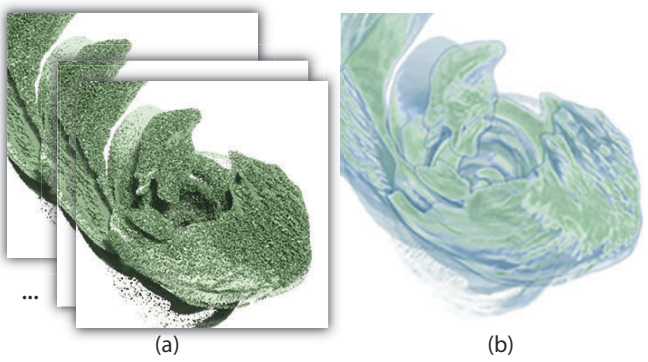


Fig. 4. U-LCS extraction by curve/surface density estimation: (a) ridges extracted from single Monte Carlo runs; (b) U-LCS, surface density estimation from (a).

Carlo process, the density estimation results are smooth. The artifacts are suppressed because the results are the composition of a number of stochastic runs.

A generalized SPH model is used to estimate the densities of ridge curves (surfaces). In the traditional SPH model for discrete points, the density of an arbitrary location $\mathbf{r}$ is the convolution of an arbitrary kernel function with discrete particles inside a sphere centered on $\mathbf{r}$:

$$\rho(\mathbf{r}) = \sum_{j=0}^{N_b-1} M_j \omega(|\mathbf{r} - \mathbf{r}_j|, h), \tag{9}$$

where N_b is the number of points in the sphere, M_j is the mass of each particle, h is the radius of the sphere, and ω is the value of the kernel function. Notice that the particle masses are identical in this study. The extension to line (surface) density estimation can be written as

$$\rho(\mathbf{r}) = \sum_{j=0}^{N_b-1} \int_{D_j} \omega(|\mathbf{r} - \mathbf{r}_j|, h) d\mathbf{s}, \tag{10}$$

where D_j is the jth line or surface intersected with the sphere and $d\mathbf{s}$ is the infinitesimal piece of D_j. However, the integral usually does not have an analytical solution for common kernel functions, such as a Gaussian kernel and $1/d^2$ (while the line integral for a $1/d^2$ kernel function has an analytical solution for curves, it does not for surfaces). In this study, we use a simple and commonly used kernel function for the density estimation:

$$\omega(d, h) = \begin{cases} 1/\pi h^2 \text{ or } 3/4\pi h^3, & \text{if } d \le h \\ 0, & \text{otherwise.} \end{cases} \tag{11}$$

As shown in Figure 5, with this kernel function, the density estimation result is essentially the length of lines in the sphere and the area of surfaces in the sphere for 2D and 3D data, respectively. The normalization factors $1/\pi h^2$ and $3/4\pi h^3$ are the inversions of area/volume of 2D/3D datasets. The cost of this method is low, and it produces robust and smooth results. Users need to specify a proper kernel size h as a parameter; we use $h = 2$ to generate the results in our experiments.

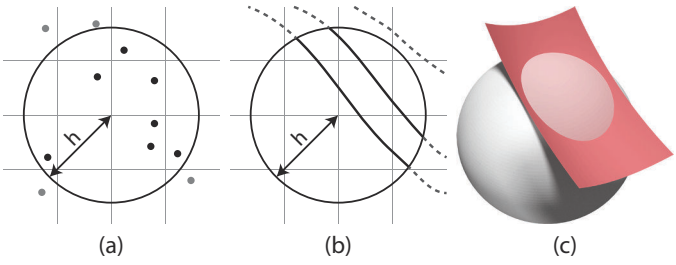


Fig. 5. SPH-based density estimation for points (a), curves (b), and surfaces (c).

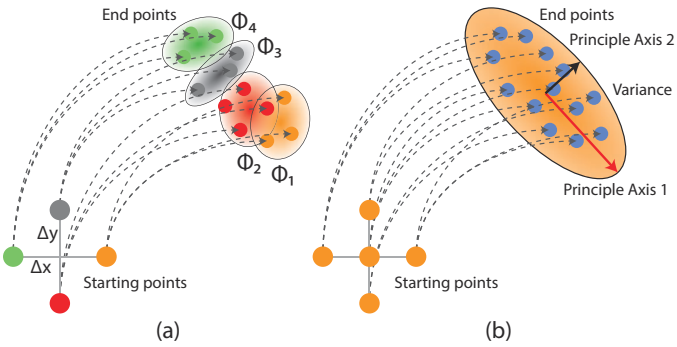


Fig. 6. Comparison between FTLE-D (a) and FTVA (b).

6 FTLE of Distributions (FTLE-D)

Unlike the D-FTLE, the FTLE-D produces a scalar field of the same form as a conventional FTLE. As defined in Eq. 5, for a given advection time τ, the FTLE-D characterizes the "averaged" difference of SFM distributions in neighboring regions. We use the expectation of the flow map gradient $E(\nabla\Phi)$ to measure the difference.

Because the gradient relies on the random variable Φ, the computation of FTLE-D is also based on Monte Carlo simulations. The expectation of $\nabla\Phi$ can be estimated by averaging the gradient of flow maps in each run:

$$E[\nabla\Phi] \approx \frac{1}{m} \sum_{i=0}^{m-1} \nabla\phi^{(i)}, \tag{12}$$

where $\nabla\phi^{(i)}$ can be estimated by the central difference method for each individual run. The 2D case is shown in Figure 6(a). The x-component of $E[\nabla\Phi]$ is $E[\Phi_1 - \Phi_4]/2\Delta x$, where $\Phi_1 - \Phi_4$ can be approximated by the stochastically traced particles $\frac{1}{m}\sum_{i=0}^{m-1} \phi_1^{(i)} - \phi_4^{(i)}$. Likewise, we can compute the other components of $E[\nabla\Phi]$ similarly.

The FTVA [8], which is a variance-based FTLE-like metric, is an alternative method for analyzing uncertain unsteady flow. The FTLE-D and FTVA are fundamentally different. First, FTLE-D is a direct generalization of deterministic FTLE, but the FTVA is not. As shown in Figure 6, the FTLE-D is based on the gradient estimation of SFMs, while FTVA is derived from the first principal component of the end points traced from a small parcel, which is not a gradient at all.

We believe that our method is conceptually and computationally closer to the traditional deterministic FTLE. First, when the "ground truth" of the uncertain data is available

in our experiments (double-gyre and Isabel data), the FTLE-D values are similar to those of the deterministic FTLE. For uncertain and deterministic double-gyre data, the root-mean-square deviation (RMSD) and peak signal-to-noise ratio (PSNR) of the FTLE-D and FTLE are 0.013 and 29.42 dB, respectively. However, the scale of the FTVA output values is different from that of the FTLE-D and FTLE, which makes it impossible to conduct a quantitative comparison between the FTVA and FTLE or FTLE-D. Second, the FTVA inherently assumes that the distribution of end points is Gaussian (due to the PCA), whereas the FTLE-D does not make any such assumptions. SFMs are often non-Gaussian, especially in divergent flow regions; the FTVA blurs features in such regions. As shown in Figure 3, the FTLE-D (i) is closer to the deterministic FTLE (n) compared with the FTVA (k) in the circled region. We observe similar results using uncertain Isabel data (all in the next section).

7 Implementation and Performance

We use GPUs to accelerate the most computation-intensive algorithms in our framework, including the stochastic particle tracing, FTLE-D, D-FTLE, and U-LCS computation. In our current implementation, we parallelize over seeds in the stochastic particle tracing. For every Monte Carlo run, each thread computes the final position of the seed, which is stored on the GPU memory for further use. For each seed location, the D-FTLE value is immediately computed after the particle tracing. The FTLE values for each run are calculated and summed to FTLE-D histograms by using atomic operations on the shared memory. U-LCS computation consists two parts: ridge detection and density estimation. We filter each cell in the domain by checking if it intersects a ridge line/surface; then, the line segments/surface patches are generated and stored on the GPU memory. The densities on each node are then calculated by per pixel/voxel operations. All the algorithms are highly parallel, and we do not need to copy intermediate data back and forth during the computation. In order to handle large time-varying data, an out-of-core strategy is used when the total data size is greater than the GPU memory. Parallelism on supercomputers will be our future solution to handle even larger datasets.

The benchmark performances of the algorithms are listed in Table 1. The prototype system is implemented with C++ and CUDA. The benchmark platform is a workstation equipped with two Intel Xeon E5620 CPUs (2.40 GHz), 12 GB RAM, and an Nvidia Tesla K40c GPU. The GPU contains 2,880 CUDA cores and 12 GB memory. Compared with the CPU performance, the GPU implantation typically runs 100 times faster in our experiments.

8 Results

We apply the proposed methods to three uncertain unsteady flow datasets: synthetic double-gyre data, Hurricane Isabel data, and and ensemble WRF simulation data. The uncertain double-gyre data are generated by adding Gaussian noise to an existing and well-known dataset. The double-gyre results are already shown in previous sections to describe the algorithms, and we further conduct sensitivity analysis on this data in this section. Uncertainty in the Hurricane Isabel data stems from temporal down-sampling errors [9]. The ensemble WRF data is from a real-world weather simulation, and the uncertainties result from the ensemble member variances.

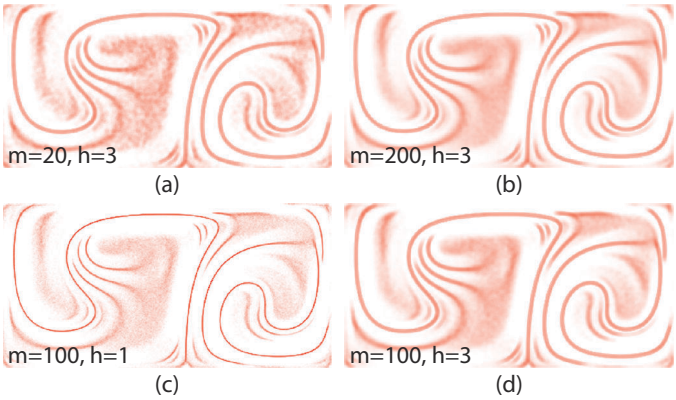


Fig. 7. U-LCS results of synthetic uncertain double-gyre data with different Monte Carlo run numbers and kernel sizes.

8.1 Synthetic Uncertain Double-Gyre Data

We synthesize the uncertain double-gyre data for illustrating and benchmarking the algorithms in this paper. The original double-gyre dataset is a closed-form time-varying 2D vector field defined on $[0, 0] \times [2, 1]$. In our experiments, we arbitrarily inject Gaussian noises ($N(0, 0.02^2)$) into the u and v components of the vector field. Particles are seeded on a 401×201 Cartesian grid, and they are stochastically traced to generate further results.

Figure 3 shows the visualization results for both the uncertain and deterministic double-gyre data. Double-gyre results also appear in previous sections; in this section, we further analyze the sensitivity to different parameters used to generate the U-LCS field. Figures 7(a) and (b) are generated with two different numbers of Monte Carlo runs. Together with Figure 3(m), we can see that more runs lead to smoother results. Of course, more resources and time are required with increasing number of runs. In our algorithms, we adaptively increase the numbers of runs until the output D-FTLE does not significantly change.

We also compare different kernel sizes used for density estimation in Figures 7(b), (c), and (d). In general, larger kernel size leads to smoother U-LCS, but fine details may be hidden in neighboring features. In our experiments, we use $h = 2$ pixels (voxels) to generate our results. Currently, users manually choose the kernel size h; we do not investigate the automatic selection of h in this paper.

8.2 Temporally Down-Sampled Hurricane Isabel Data

A common practice to alleviate the high storage cost in scientific simulation is to skip time steps and store only a small portion of the output. However, important information can be lost in the discarded data, and temporal downsampling can lead to data uncertainty. Chen et al. [9] proposed a method using quadratic Bezier curves to interpolate the unstored data; the uncertainties are modeled as Gaussian interpolation errors. In this experiment, we view the original

TABLE 1
Data specifications and timings: t_p, t_f, t_l, and t_d are timings (in seconds) for stochastic particle tracing, stochastic FTLE computation, uncertain LCS computation (density estimation), and FTLE-D computation, respectively.

Dataset	Uncertainty Source	Resolution	Performance (CPU)				Performance (GPU)			
			t_p	t_f	t_l	t_d	t_p	t_f	t_l	t_d
Double gyre	Noise	N/A (analytical)	4.60k	18.7	60.0	0.28	10.0	0.05	1.00	0.06
Isabel	Down-sampling	$500 \times 500 \times 100 \times 4$	447k	17.2k	6.30k	1.61k	4.76k	25.0	0.56k	25.8
WRF	Ensembles	$1799 \times 1059 \times 40 \times 15$	308k	13.1k	8.36k	1.21k	3.5k	22.0	0.47k	19.7

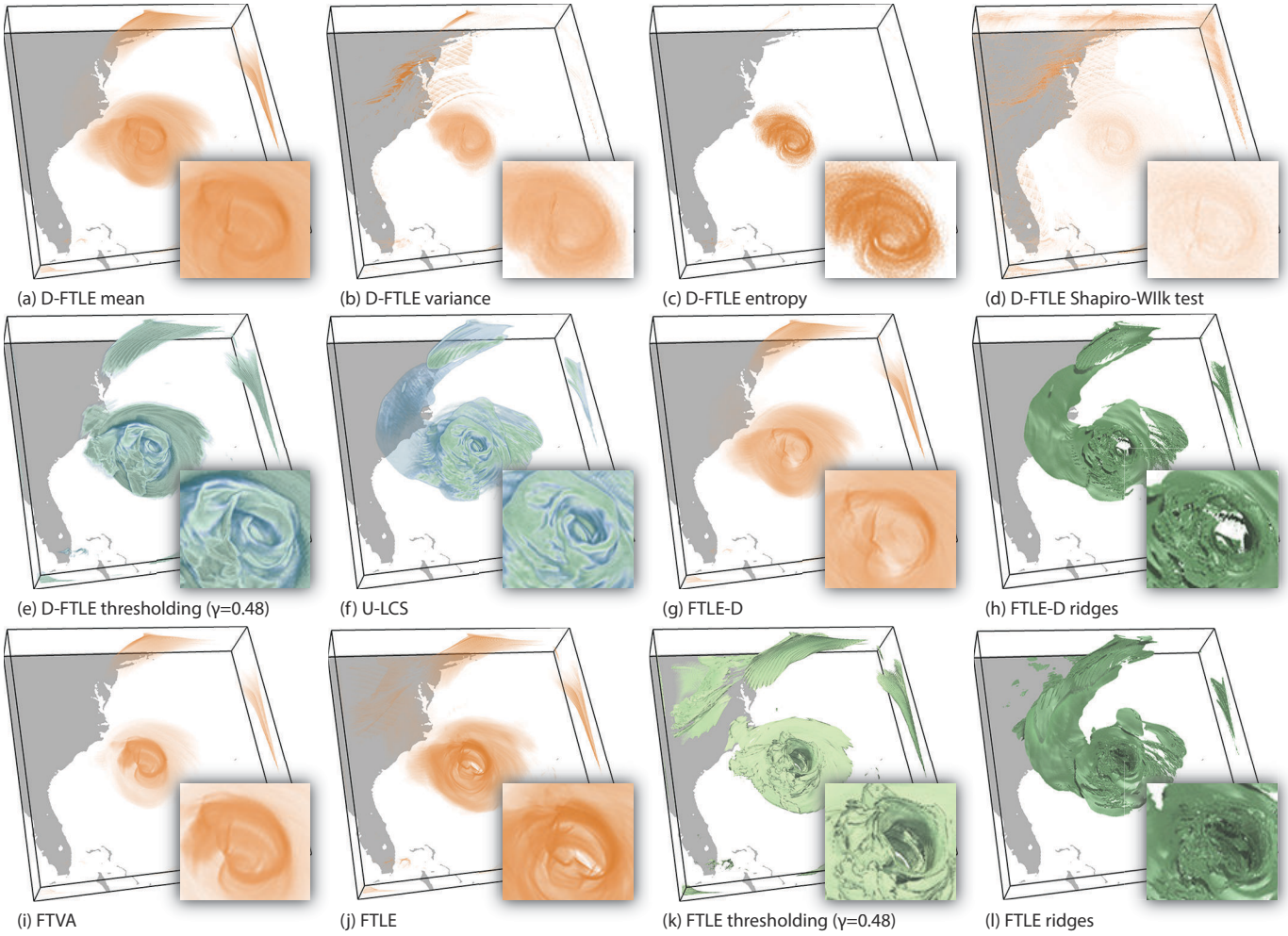


Fig. 8. Visualization of uncertain (a-j) and deterministic (k-l) Hurricane Isabel data with various methods. The darker color indicates the larger values in the images, except that (d) is inversed. The orange hues are used to visualize FTLE-like metrics (D-FTLE statistics, FTLE-D, and FTVA, etc.), and green hues are used to visualize LCS-like metrics (FTLE ridges, FTLE thresholds, and U-LCS, etc.)

Hurricane Isabel dataset as the "ground truth" from simulation output, and we use this method to obtain the down-sampled version for uncertain FTLE and LCS analysis.

The original and deterministic Isabel dataset is courtesy of the IEEE Visualization Contest 2004. The spatial resolution is $500 \times 500 \times 100$, and there are 48 time steps (hourly average) stored in separate files. Three wind field vector components U, V, and W are used in this experiment. The down-sampled Isabel dataset aggregates every 12 time steps into one. In each down-sampled frame, the quadratic Bezier curve parameters and the error distributions are stored.

Figures 8(a)-(i) shows the visualization results of down-sampled data, and we also show the results for the original data in Figures 8(j)-(l) for reference. The time t and advection time τ are 24 and 6 (in hours), respectively. From the statistics of D-FTLE in (a-e), we can see that the FTLE values, as well as the uncertainty of FTLE (measured by standard deviation) are higher near the hurricane eye. From the entropy and Shapiro-Wilk test, we also observe that the D-FTLE values are highly non-Gaussian. The U-LCS is shown in (f), and we can see the distribution of separatrices and their uncertainty.

In addition, we compare the FTLE-D and FTVA with the FTLE field derived from the original data. The FTLE-D gives an overview of the uncertain unsteady flow, which is directly comparable with traditional and deterministic FTLE fields. We can see that the FTLE-D (g) and the deterministic FTLE (j) are similar, as are their ridges in (h) and (l). Some details in the FTLE-D are blurred because of the data down-sampling, but we can still distinguish the main structures of

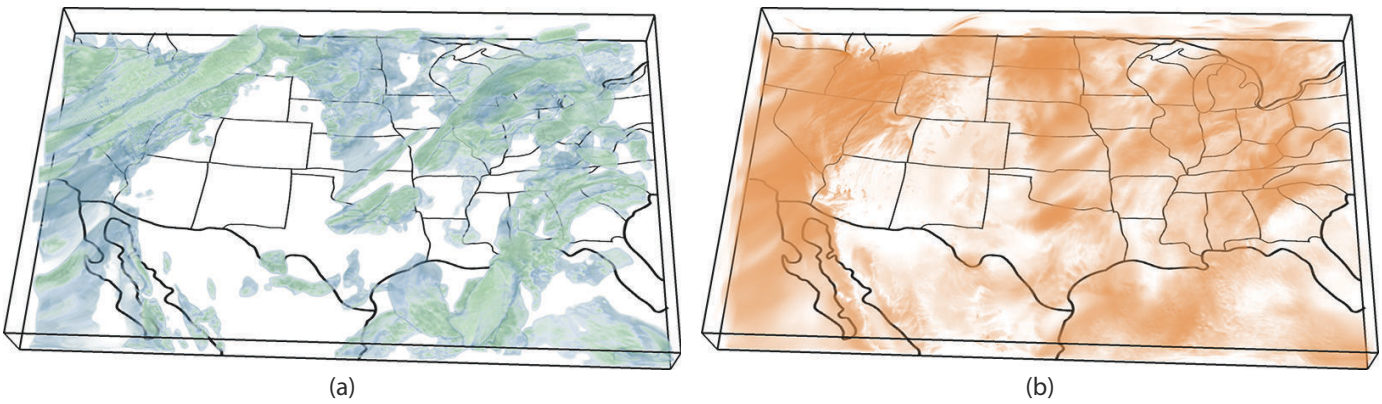


Fig. 9. Visualization of ensemble WRF simulation data: (a) FTLE-D, (b) U-LCS.

the data. The RMSD and PSNR of the FTLE-D and deterministic FTLE are 0.057 and 23.3 dB in this experiment. Compared with the FTVA, the FTLE-D captures the hurricane wall details more authentically. As discussed in Section 6, the FTVA and FTLE are not numerically comparable; hence, we can make only visual qualitative observations about the FTVA without being able to quantify its difference compared with the FTLE and FTLE-D.

Meteorologists with whom we discussed the visualization results confirmed that all visualizations, based only on wind components, show the convective bands of Isabel remarkably well. Vertical motions within the spiral arm, which extends up the east coast, separate some pathlines to the top of the atmosphere while leaving others near this oringinal level. Because of the uncertainty of updraft and downdraft features, small changes in initial conditions create an uncertain separatrix around the edge of updraft and downdraft cores, as shown in the U-LCS, and in the ridges of the FTLE-D show good spatial coherency.

8.3 Ensemble WRF Simulation Data

The National Weather Service runs a version of the Weather Research and Forecasting (WRF) model called the High Resolution Rapid Refresh (HRRR) model [33]. HRRR combines a well-tested configuration of WRF with a gridpoint statistical interpolation scheme for assimilating NOAA and other observations. HRRR is run every hour and produces a forecast out to 16 hours. It is available to the public via Unidata's THREDDS Data Server.

We use an ensemble of simulation output for analysis, and we model the uncertainty of the wind field by their averages and standard deviations on each grid point across the ensemble members. The resolution of the grid is $1799 \times 1059 \times 40$, and we use 15 hourly average data and 10 ensemble members for the experiment. The particles are seeded on a half-resolution grid ($900 \times 530 \times 20$). The starting time of our analysis is 00:00:00 UTC, August 27, 2015, and the advection time τ is 5 hours. The U-LCS and FTLE-D visualization results are shown in Figure 9. As in the previous section the visualizations highlight the edges of areas with vertical motion.

Feedback from the scientist confirms that the U-LCS and FTLE-D are greatest in areas of upward and downward motion. This is driven by the topography and variability in the land surface of the continental U.S. as well as the scale of synoptic weather patterns. In this case there are four distinct zones: on-shore flow from the Pacific being pushed over the Cascade mountains, a baroclinic zone (cold front) stretching from Oklahoma up into the Dakotas, and two unstable trough regions over the midwest and east. Given that the only inputs were wind vectors, the visualizations highlight unstable areas, and the techniques give a quantitative and mathematically robust way to show three-dimensional uncertain flow in an easily understood manner.

9 Conclusions and Future Work

In this paper, we generalize and redefine the concepts of the FTLE and LCS to visualize and analyze uncertain time-varying fields, in order to better understand uncertain transport behaviors. Three tools are presented—D-FTLE, FTLE-D, and U-LCS. The D-FTLE, which analyzes the uncertainty of FTLE values, is visualized with various statistical measurements. The FTLE-D aggregates the traced particle divergence and gives a statistical overview of uncertain transport behaviors. The U-LCS further quantifies the uncertainties of finding LCSs. Experiments show that the proposed tools can help users understand transport behaviors and find separatrices in real-world atmospheric simulations.

The two concepts of D-FTLE and FTLE-D are related but differ from each other. We believe that both are useful. The D-FTLE is a distribution field that stores histograms for every location, while the FTLE-D is a scalar field that can be compared with the FTLE. The D-FTLE cannot be directly visualized because of its high dimensionality, so we provide various statistical measurements for interactive exploration. Conversely, the FTLE-D can be directly visualized with pseudocolors or volume rendering, and it provides better results than existing variance-based methods do, as discussed in Section 6. In addition, the U-LCS, which is the probabilistic field of finding the LCS, can further help users understand transport behaviors and find separatrices.

Future work will entail reducing computation time and extending to large-scale datasets. Adaptive sampling techniques could be used to reduce the amount of particle tracing, but further efforts are needed to understand additional uncertainty that would result. Our method can also be extended to parallel environments, in order to visualize and analyze very large datasets.

ACKNOWLEDGMENTS

This material is based upon work supported by the U.S. Department of Energy, Office of Science, under contract number DE-AC02-06CH11357. This work is also supported by the U.S. Department of Energy, Office of Advanced Scientific Computing Research, Scientific Discovery through Advanced Computing (SciDAC) program. The work of Collis and Helmus has been supported by the OBER of the DOE as part of the ARM Program. This project took advantage of netCDF software developed by UCAR/Unidata.

REFERENCES

[1] C. M. Wittenbrink, A. Pang, and S. K. Lodha, "Glyphs for visualizing uncertainty in vector fields," *IEEE Trans. Vis. Comput. Graph.*, vol. 2, no. 3, pp. 266–279, 1996.

[2] M. Hlawatsch, P. Leube, W. Nowak, and D. Weiskopf, "Flow radar glyphs - static visualization of unsteady flow with uncertainty," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 12, pp. 1949–1958, 2011.

[3] R. P. Botchen, D. Weiskopf, and T. Ertl, "Texture-based visualization of uncertainty in flow fields," in *Proceedings of IEEE Visualization 2005*, 2005, pp. 647–654.

[4] M. Otto, T. Germer, H.-C. Hege, and H. Theisel, "Uncertain 2D vector field topology," *Comput. Graph. Forum*, vol. 29, no. 2, pp. 347–356, 2010.

[5] M. Otto, T. Germer, and H. Theisel, "Uncertain topology of 3D vector fields," in *Proceedings of IEEE Pacific Visualization 2011*, 2011, pp. 67–74.

[6] G. Haller, "Distinguished material surfaces and coherent structures in three-dimensional fluid flows," *Physica D: Nonlinear Phenomena*, vol. 149, no. 4, pp. 248–277, 2001.

[7] ——, "A variational theory of hyperbolic Lagrangian coherent structures," *Physica D: Nonlinear Phenomena*, vol. 240, no. 7, pp. 547–598, 2011.

[8] D. Schneider, J. Fuhrmann, W. Reich, and G. Scheuermann, "A variance based FTLE-like method for unsteady uncertain vector fields," in *Topological Methods in Data Analysis and Visualization II*, ser. Mathematics and Visualization, R. Peikert, H. Hauser, H. Carr, and R. Fuchs, Eds. Springer, 2011, pp. 255–268.

[9] C.-M. Chen, A. Biswas, and H.-W. Shen, "Uncertainty modeling and error reduction for pathline computation in time-varying flow fields," in *Proceedings of IEEE Pacific Visualization 2015*, 2015, pp. 215–222.

[10] R. S. Laramee, H. Hauser, H. Doleisch, B. Vrolijk, F. H. Post, and D. Weiskopf, "The state of the art in flow visualization: Dense and texture-based techniques," *Comput. Graph. Forum*, vol. 23, no. 2, pp. 203–222, 2004.

[11] F. H. Post, B. Vrolijk, H. Hauser, R. S. Laramee, and H. Doleisch, "The state of the art in flow visualization: Feature extraction and tracking," *Comput. Graph. Forum*, vol. 22, no. 4, pp. 1–17, 2003.

[12] A. Pobitzer, R. Peikert, R. Fuchs, B. Schindler, A. Kuhn, H. Theisel, K. Matkovic, and H. Hauser, "The state of the art in topology-based visualization of unsteady flow," *Comput. Graph. Forum*, vol. 30, no. 6, pp. 1789–1811, 2011.

[13] C. R. Johnson and A. R. Sanderson, "A next step: Visualizing errors and uncertainty," *IEEE Comput. Graph. Appl.*, vol. 23, no. 5, pp. 6–10, 2003.

[14] K. Brodlie, R. AllendesOsorio, and A. Lopes, "A review of uncertainty in data visualization," in *Expanding the Frontiers of Visual Analytics and Visualization*, J. Dill, R. Earnshaw, D. Kasik, J. Vince, and P. C. Wong, Eds. Springer London, 2012, pp. 81–109.

[15] S. K. Lodha, A. Pang, R. E. Sheehan, and C. M. Wittenbrink, "UFLOW: Visualizing uncertainty in fluid flow," in *Proceedings of IEEE Visualization 1996*, 1996, pp. 249–254.

[16] M. Otto and H. Theisel, "Vortex analysis in uncertain vector fields," *Comput. Graph. Forum*, vol. 31, no. 3, pp. 1035–1044, 2012.

[17] C. Petz, K. Pöthkow, and H.-C. Hege, "Probabilistic local features in uncertain vector fields with spatial correlation," *Comput. Graph. Forum*, vol. 31, no. 3, pp. 1045–1054, 2012.

[18] M. Hummel, H. Obermaier, C. Garth, and K. I. Joy, "Comparative visual analysis of Lagrangian transport in CFD ensembles," *IEEE Trans. Vis. Comput. Graphs.*, vol. 19, no. 12, pp. 2743–2752, 2013.

[19] S. C. Shadden, F. Lekien, and J. E. Marsden, "Definition and properties of Lagrangian coherent structures from finite-time Lyapunov exponents in two-dimensional aperiodic flows," *Physica D: Nonlinear Phenomena*, vol. 212, no. 3-4, pp. 271–304, 2005.

[20] G. Haller, "Lagrangian coherent structures from approximate velocity data," *Phys. Fluids*, vol. 14, no. 6, pp. 1851–1861, 2002.

[21] B. Nouanesengsy, T.-Y. Lee, K. Lu, H.-W. Shen, and T. Peterka, "Parallel particle advection and FTLE computation for time-varying flow fields," in *SC12: Proc. ACM/IEEE Conference on Supercomputing*, 2012, pp. 61:1–61:11.

[22] H. Guo, J. Zhang, R. Liu, L. Liu, X. Yuan, J. Huang, X. Meng, and J. Pan, "Advection-based sparse data management for visualizing unsteady flow," *IEEE Trans. Vis. Comput. Graph.*, vol. 20, no. 12, pp. 2555–2564, 2014.

[23] C.-M. Chen and H.-W. Shen, "Graph-based seed scheduling for out-of-core FTLE and pathline computation," in *Proceedings of IEEE Symposium on Large-Scale Data Analysis and Visualization 2013*, 2013, pp. 15–23.

[24] S. S. Barakat and X. Tricoche, "Adaptive refinement of the flow map using sparse samples," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 2753–2762, 2013.

[25] M. Hlawatsch, F. Sadlo, and D. Weiskopf, "Hierarchical line integration," *IEEE Trans. Vis. Comput. Graph.*, vol. 17, no. 8, pp. 1148–1163, 2011.

[26] A. Kuhn, C. Rössl, T. Weinkauf, and H. Theisel, "A benchmark for evaluating FTLE computations," in *Proceedings of IEEE Pacific Visualization 2012*, 2012, pp. 121–128.

[27] B. Schindler, R. Peikert, R. Fuchs, and H. Theisel, "Ridge concepts for the visualization of Lagrangian coherent structures," in *Topological Methods in Data Analysis and Visualization II*, ser. Mathematics and Visualization, R. Peikert, H. Hauser, H. Carr, and R. Fuchs, Eds. Springer, 2011, pp. 221–235.

[28] R. Fuchs, B. Schindler, and R. Peikert, "Scale-space approaches to FTLE ridges," in *Topological Methods in Data Analysis and Visualization II*, ser. Mathematics and Visualization, R. Peikert, H. Hauser, H. Carr, and R. Fuchs, Eds. Springer, 2011, pp. 283–296.

[29] A. Luo, D. T. Kao, J. L. Dungan, and A. Pang, "Visualizing spatial distribution data sets," in *VisSym 03: Proceedings of Symposium on Visualization*, 2003, pp. 29–38.

[30] J. D. Furst and S. M. Pizer, "Marching ridges," in *SIP'01: Proceedings of IASTED International Conference on Signal and Image Processing*, 2001, pp. 22–26.

[31] O. D. Lampe and H. Hauser, "Curve density estimates," *Comput. Graph. Forum*, vol. 30, no. 3, pp. 633–642, 2011.

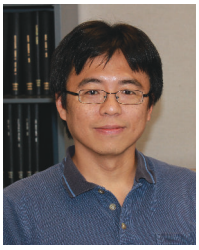
[32] J. J. Monaghan, "Smoothed particle hydrodynamics," *Ann. Reviews Astron. Astrophysics*, vol. 30, pp. 543–573, 1992.

[33] C. Alexander, S. S. Weygandt, D. C. D. S. Benjamin, T. G. Smirnova, E. P. James, M. H. P. Hofmann, J. Olson, and J. M. Brown, "The high-resolution rapid refresh: Recent model and data assimilation development towards an operational implementation in 2014," in *Proceedings of 26th Conference on Weather Analysis and Forecasting / 22nd Conference on Numerical Weather Prediction*. American Meterological Society, 2014.

Hanqi Guo is a postdoctral appointee in the Mathematics and Computer Science Division, Argonne National Laboratory. He received his Ph.D. degree in computer science from Peking University in 2014, and the B.S. degree in mathematics and applied mathematics from Beijing University of Posts and Telecommunications in 2009. His research interests are mainly in flow visualization, uncertainty visualization, and large-scale scientific data visualization.

Wenbin He is a Ph.D. student in computer science and engineering at the Ohio State University. He received his B.S. degree from the Department of Software Engineering at Beijing Institute of Technology in 2012. His research interests include analysis and visualization of large-scale scientific data, uncertainty visualization, and flow visualization.

Tom Peterka is a computer scientist at Argonne National Laboratory, fellow at the Computation Institute of the University of Chicago, adjunct assistant professor at the University of Illinois at Chicago, and fellow at the Northwestern Argonne Institute for Science and Engineering. His research interests are in large-scale parallelism for in situ analysis of scientific data. His work has led to two best paper awards and publications in ACM SIGGRAPH, IEEE VR, IEEE TVCG, and ACM/IEEE SC, among others. Peterka received his Ph.D. in computer science from the University of Illinois at Chicago, and he currently works actively in several DOE- and NSF-funded projects.



Han-Wei Shen is a full professor at the Ohio State University. He received his B.S. degree from Department of Computer Science and Information Engineering at National Taiwan University in 1988, the M.S. degree in computer science from the State University of New York at Stony Brook in 1992, and the Ph.D. degree in computer science from the University of Utah in 1998. From 1996 to 1999, he was a research scientist at NASA Ames Research Center in Mountain View California. His primary research interests are scientific visualization and computer graphics. He is a winner of the National Science Foundation's CAREER award and U.S. Department of Energy's Early Career Principal Investigator Award. He also won the Outstanding Teaching award twice in the Department of Computer Science and Engineering at the Ohio State University.

Scott M. Collis is a specialist in using remote sensing data to extract geophysical insight. He leads the radar products team at Argonne National Laboratory and is the science lead on the Python-ARM Radar Toolkit and an expert in the remote sensing of precipitating cloud systems. He is a senior fellow at the Argonne/University of Chicago Computation Institute. He is also the Translator for the ARM Climate Facility centimeter wavelength radars. In this role, he works with both instrument mentors and data users such as climate and fine scale modelers in order to provide the best possible measurements to improve the representation of cloud systems in global climate models. His research covers weather phenomena from Darwin to Oklahoma to the Arctic, with a particular focus on the interplay between large-scale forcing and local-scale impacts.

Jonathan J. Helmus is a scientist and advanced algorithms engineer at Argonne National Laboratory, where he works closely with the Atmospheric Radiation Measurement (ARM) climate research facility. His research focuses on the analysis of weather radar data, much of which is done within the open source Python ARM radar toolkit, Py-ART, for which he is the lead developer. He completed a postdoc at the University of Connecticut Health Center after receiving his Ph.D. in Chemical Physics from the Ohio State University.