Vision, Modeling, and Visualization (2011)
Peter Eisert, Konrad Polthier, and Joachim Hornegger (Eds.)

Extracting Flow Structures Using Sparse Particles

Alexy Agranovsky, Christoph Garth, and Kenneth I. Joy

Abstract

In recent years, Lagrangian Coherent Structures (LCS) have been characterized using the Finite-Time Lyapunov Exponent, following the advection of a dense set of particles into a corresponding flow field. The large amount of particles needed to sufficiently map a flow field has been a non-trivial computational burden in the application of LCS. By seeding a minimal amount of particles into the flow field, Moving Least Squares, combined with FTLE, will extrapolate the important feature locations at which further refinement is desired. Following the refinement procedure, MLS produces a continuous function reconstruction allowing the characterization of Lagrangian Coherent Structures with a lower number of particles. Through multiple data sets, we show that given a sparse and refined sampling, MLS will reproduce FTLE fields exhibiting a nominal error while maintaining a performance increase when compared to the standard, dense finite difference approach.

Categories and Subject Descriptors (according to ACM CCS): Image Processing and Computer Vision [I.4.7]: Feature Measurement—Simulation And Modeling [I.6.6]: Simulation Output Analysis—Physical Sciences and Engineering [J.2]: Engineering—.

1. Introduction

Within both the academic and professional communities, the study of fluid flow remains essential in scientific, medical, and engineering sectors. A core understanding of fluid flow behavior is often required during post-process analysis and final results. Examples include automotive design, turbomachinery, machine cooling, medical imaging, and aeronautics. Investigation of fluid flow within such regions of study yield large and numerically complex data sets that require further analysis.

We will specifically target the ability to characterize, extract and visualize salient structures within a turbulent flow. Various approaches, ranging from topological to feature-based, have been applied to this problem with the same goal, coherent structure identification. In this work, focus is set on locating Lagrangian Coherent Structures (LCS) using the Finite-Time Lyapunov Exponent (FTLE). The notion of FTLE rests on a strong theoretical framework, characterizing laminar and turbulent flow in terms of repelling and attracting material surfaces. Unfortunately, this physically intuitive and mathematically sound representation is hindered by the computational cost associated with advecting a dense set of particles across the spatio-temporal flow domain that are needed to perform the quantitive assessment of FTLE. While adaptive computation approaches have been presented in the past, they are usually tied to specific refinement schemes and require an auxiliary refinement structure.

The goal of this work is to visualize Lagrangian Coherent Structures in flow data with less effort. To this purpose, we make use of the power of the Moving Least Squares (MLS) framework for interpolation and approximation. The MLS framework allows us to do flexible local refinement without the need for laborious mesh construction and maintenance. Through its use, we are able to seed a strongly reduced number of particles to robustly and efficiently evaluate Finite-Time Lyapunov Exponents. The resulting fields are easily kept smooth, which has beneficial numerical consequences for the extraction of Lagrangian Coherent Structures. Overall, we show that Moving Least Squares has the principle properties to serve as the foundation for mesh-free LCS-based flow visualization.

The paper is structured as follows: Section 2 builds a framework, delving deeper into the numerical foundation on which FTLE is built, followed by an in-depth explanation of MLS and its adaptive features. Section 3 examines previous works on FTLE and applications benefiting from the use of MLS. Next, in Section 4, we discuss incremental refinement to provide a more robust approximation and in Section 5, cover the full MLS reconstruction and LCS extraction. Finally, we present our results and experiments in Section 6 and conclude our work in Section 7.

DOI: 10.2312/PE/VMV/VMV11/153-160

2. Background

In this paper, we propose using Moving Least Squares to calculate the Finite-Time Lyapunov Exponent from scattered flow map samples. Based on concepts from the theory of dynamical systems, FTLE is a geometric tool used to define and extract Lagrangian Coherent Structures in transient flows studied in a Lagrangian framework. The Finite-Time Lyapunov Exponent is a measure of the separation rate between infinitesimally close trajectories, which when applied in the context of finite flow fields, defines asymptotically stable and unstable Lagrangian Coherent Structures.

Previous works have completed FTLE calculations and visualized the Lagrangian Coherent Structures on (nested) regular grids, as we will discuss in Section 3. We propose using Moving Least Squares to minimize the number of seeded particles used in the computation of the Finite-Time Lyapunov Exponent. Our reasoning for choosing MLS to perform the sparse data reconstruction lies at the heart of the FTLE calculation, described further in the next section. Because FTLE measures a separation rate, we require a differentiable reconstruction and Moving Least Squares provides a local and continuous reconstruction method, making for a smooth image. Furthermore, MLS produces the two derivatives needed when extracting LCS.

2.1. Finite Time Lyapunov Exponent

To provide an in-depth explanation of FTLE calculation during the corresponding MLS reconstruction, we introduce some useful notations. Even though our work extends to three dimensions, for a concise elaboration, we consider a time-dependent two-dimensional vector field $\mathbf{v}$ defined over a finite Euclidean domain $U \subset \mathbb{R}^2$ and a finite temporal domain $I \subset \mathbb{R}$. Starting at time t_0, a particle $\mathbf{x}$ is advected to time t, where $t = t_0 + \tau$, formulated as the map $\mathbf{x}(t;t_0,\mathbf{x}_0)$. For coherency, the mapping must satisfy $\mathbf{x}(t_0;t_0,\mathbf{x}_0) = \mathbf{x}_0$ and $\dot{x}(t;t_0,\mathbf{x}_0) = \mathbf{v}(t,\mathbf{x}(t;t_0,\mathbf{x}_0))$, where the dot denotes the derivative with respect to the first parameter. A linearization of the local variation of the map $\mathbf{x}(t;t_0,\cdot)$ around the seed position $\mathbf{x}_0$ is obtained by considering its spatial gradient $\mathbf{J}_x(t,t_0,\mathbf{x}_0) := \nabla_{\mathbf{x}_0}\mathbf{x}(t;t_0,\mathbf{x}_0)$ at $\mathbf{x}_0$. The maximal dispersion after time τ of particles in a neighborhood of $\mathbf{x}_0$ can be calculated using this spatial gradient as a function of the direction $\mathbf{d}_0$ along which we move away from $\mathbf{x}_0 : \mathbf{d}_t = \mathbf{J}_x(t,t_0,\mathbf{x}_0)\mathbf{d}_{t_0}$. Maximizing the norm $\mid \mathbf{d}_t \mid$ over all possible unit directions $\mathbf{d}_{t_0}$ corresponds to computing the maximum eigenvalue of the spectral norm of $\mathbf{J}_x(t,t_0,\mathbf{x}_0)$

$$\sigma(t_0,\mathbf{x}_0) = \sqrt{\lambda_{max}(\mathbf{J}_x(t,t_0,\mathbf{x}_0)^T\mathbf{J}_x(t,t_0,\mathbf{x}_0))} \qquad (1)$$

The logarithm is then applied to obtain the average exponential separation rate $\lambda(t,t_0,\mathbf{x}_0)$ and the result is normalized by the advection time τ

$$\lambda(t,t_0,\mathbf{x}_0) = \frac{1}{\mid \tau \mid}\log\sqrt{\lambda_{max}(\mathbf{J}_x(t,t_0,\mathbf{x}_0)^T\mathbf{J}_x(t,t_0,\mathbf{x}_0))} \quad (2)$$

resulting in a rate commonly referred to as the Finite-Time Lyapunov Exponent.

2.2. Moving Least Squares

To recreate accurate FTLE fields from a field of sparse particles, we desire continuity among FTLE value reconstruction. From a structured grid, finite differences can be used to calculate the Jacobian matrix $\mathbf{J}_\mathbf{x}$ necessary for the calculation of the largest eigenvalue, and finally the Finite-Time Lyapunov Exponent. Unable to follow a similar procedure for mesh-free data, we require continuous derivatives for a suitable reconstruction. The Moving Least Squares method, offers a robust approximation over scattered data which preserves function continuity as well as smooth, well-defined derivatives. With MLS, local continuity is maintained, allowing us to retain the inherent flow of the particles during least squares approximation. In the following section, we provide an extensive explanation of the Moving Least Squares method with a focus on numerical stability.

2.2.1. MLS Approximation

At the base of the problem lies a flow map with corresponding vector values. It is within this turbulent flow that we wish to extract salient structures through the use of the Finite-Time Lyapunov Exponents. A set of particles is seeded into the flow field and advected for a finite time step. Using the final locations of the advected scattered particles and MLS, we are able to form a continuous flow map of particles at t_{i+1} as if we had advected a complete field of particles from t_i to t_{i+1}.

When calculating FTLE, particles are advected from t_i to t_{i+1}, where $t_{i+1} = t_i + \tau$. It is through this time period, that we are able to formulate the divergence of particles through FTLE. Based on these advections, MLS is used to obtain an approximation function to $f(\mathbf{x})$, where ultimately $f(\mathbf{x})$ refers to the spatial coordinates of any point at t_{i+1}. The approximation is developed in the least squares sense, using N points with coordinates x_i to develop the local approximation of f at $\mathbf{x}$. The solution poses the following minimization problem:

$$min_{f\in\prod_m^d}\sum_i \mathbf{w}(\rho)\|f(x_i)-f_i\|^2 \qquad (3)$$

where m is the degree of the basis polynomial and d is the spatial dimension with f minimized accordingly. $\mathbf{w}(\rho)$ represents the weighting function taking as parameter $\rho = \frac{\|\mathbf{x}-x_i\|}{\gamma}$, the Euclidean distance between $\mathbf{x}$ and the position of the data points x_i with respect to the domain of influence radius, γ. A more detailed discussion of the weight function will be given in Section 2.2.2. The solution of $f(\mathbf{x})$ can also be represented as a linear combination of basis functions and rewritten as the following:

$$f(\mathbf{x}) = \mathbf{b}^T(\mathbf{x})\mathbf{c}(\mathbf{x}) \qquad (4)$$

where $\mathbf{b}^T(\mathbf{x}) = [b_1(\mathbf{x}), \cdots, b_k(\mathbf{x})]$ is a complete polynomial basis of k terms and $\mathbf{c}(\mathbf{x}) = [c_1, \cdots, c_k]$ is a vector of unknown coefficients that are to be minimized, where $k = \frac{(m+d)!}{m!d!}$. By taking partial derivatives of (3) with respect to the coefficients of $\mathbf{c}(\mathbf{x})$ and solving least squares by use of the normal equation, the calculation of $\mathbf{c}(\mathbf{x})$ can be reformulated as

$$\mathbf{B}^T\mathbf{W}(\rho)\mathbf{B}\mathbf{c}(\mathbf{x}) = \mathbf{B}^T\mathbf{W}(\rho)\mathbf{f} \quad (5)$$

where $\mathbf{f}$ is a vector of coordinate values at time t_{i+1} and $\mathbf{W}(\rho)$ is the diagonal weight matrix of size $N \times N$, storing the numerical influence of points x_i on fit point $\mathbf{x}$. $\mathbf{B}$ is defined to be a matrix of polynomial basis functions $\mathbf{b}(x_i)$ for the N points associated with the least squares fit

Substituting the solution of $\mathbf{c}(\mathbf{x})$ into Equation (4), the function solution $f(\mathbf{x})$ is concluded with the explicit formulation,

$$f(\mathbf{x}) = \mathbf{b}^T(\mathbf{x})(\mathbf{B}^T\mathbf{W}(\rho)\mathbf{B})^{-1}\mathbf{B}^T\mathbf{W}(\rho)\mathbf{f} \quad (6)$$

For the calculation of the FTLE, the gradient of $f(\mathbf{x})$ must be extracted, thus we must calculate $f'(\mathbf{x})$ where the apostrophe denotes the first derivative. Levin [Lev98] and others have recommended using a simple, one term derivative:

$$f'(\mathbf{x}) = \frac{\partial f_x(\mathbf{x})}{\partial x_s} \approx \frac{\partial \mathbf{b}(\mathbf{x})^T}{\partial x_s}\mathbf{c}(\mathbf{x}) \quad (7)$$

which requires only the differentiation of the basis function $\mathbf{b}^T(\mathbf{x})$. By taking the partial derivative of the function basis for the resampled coordinate in terms of $\mathbf{x}_s$, the Jacobian matrix needed for the computation of the FTLE is formed.

2.2.2. Weight Function

The purpose of the weight function is to give points closer to $\mathbf{x}$ more emphasis for the approximation than points further away. The weight function is applied to the n nearest neighbors of $\mathbf{x}$, and the outcome stored in the $\mathbf{W}(\rho)$ diagonal matrix of size $n \times n$. After analyzing multiple weight functions and the resulting reconstruction error, we have chosen to use the following spline weight function [BMXK06] for our MLS fit,

$$\mathbf{W}(\rho) = 1 - 6\rho^2 + 8\rho^3 - 3\rho^4 \quad (8)$$

To ensure that all points are taken into consideration, υ is added to r, the Euclidean distance between $\mathbf{x}$ and neighbor x_i, for the weight function, where $\upsilon \ll 1$. Now, the weight function assures that all points within the neighborhood will contribute to the fit regardless of their proximity to $\mathbf{x}$. Even when interpolating over one of the scattered data points, continuity throughout the linear system is maintained, resulting in a smoother, more accurate reconstruction.

3. Related Work

The use of the Finite-Time Lyapunov Exponents to characterize Lagrangian Coherent Structures in transient flows was first pioneered by Haller [Hal01a]. This work followed previous papers by the same author, investigating the eigenvectors of the Jacobian of the flow velocity along pathlines to determine the location of LCS in the two-dimensional setting [Hal00, HY00]. Within his seminal paper [Hal01a], Haller presented FTLE as a geometric approach while simultaneously contrasting it with an analytic criterion. Despite the contrast, the notion of preservation of a certain stability type of the velocity gradient along the path of a particle was exhibited in both approaches. An increased level of interest in the fluid dynamics community was sparked with this initial research, both from a theoretical and from a practical viewpoint. Haller further proposed a study of the robustness of the coherent structures characterized by FTLE under approximation errors in the velocity field [Hal02]. Within the same paper, he suggests that LCS should show up as ridges in the FTLE field. Shadden et al. [SLM05] provided a formal discussion of the theory of FTLE and Lagrangian coherent structure. One major contribution of their paper was to offer an estimate of the flow across the ridge lines of FTLE and to show that it is small and typically negligible. An extension of FTLE to arbitrary dimensions is discussed in [LSM07]. These tools have been applied to the study of turbulent flows [Hal01b, MHP*07]. With respect to work done to extract height ridges in FTLE flow fields, a variety of methods have been explored in literature. Sadlo et al. [SP09] extracted LCS as height ridges and Peikert et al. [PS08] used raw feature detection, avoiding explicit eigenvalue calculation while Schultz et al. [STS10] dealt directly with the degeneracies of the Hessian. Building on FTLE and extracting LCS, analysis has been done with flight data [TCH10] as well as the splitting of the Antarctic polar vortex [LR10], exemplifying the versatility of FTLE. Acceleration of FTLE computations for two-dimensional and three-dimensional flows was previously achieved by Garth et al. [GLT*07] by mapping pathline integration to the GPU. Following this work, Garth et al. [GGTH07] further accelerated FTLE computations by reducing the number of particle paths using a refinement algorithm limited to a regular grid. Sadlo et al. have also accelerated the computation of FTLE by using adaptive mesh refinement [SP07] during the extraction of LCS. LFTLE or localized FTLE [KPH*09] was introduced by Kasten et al. and calculates FTLE by advecting particles which are infinitesimally close to the data point though may suffer from aliasing issues and these particles have the tendency to move around ridges instead of encompassing them.

Moving Least Squares was first introduced to the scientific community by Shepard [She68] in 1968 and more recently, applied in two dimensional contexts such as linear elasticity [BMXK06]. Embraced by the visualization community, Moving Least Squares has been used as a means of powerful graphic reconstruction as exemplified by Levin [Lev98]. MLS has been used to calculate gradient values from unstructured meshes to support volume rendering [LGM*08, CSO09]. Further gradient computation from MLS has been used to assist in the extraction of sharp fea-

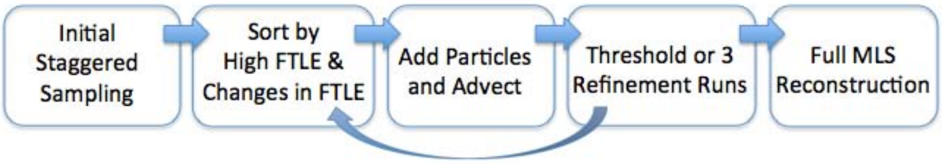

Initial Staggered Sampling
Sort by High FTLE & Changes in FTLE
Add Particles and Advect
Threshold or 3 Refinement Runs
Full MLS Reconstruction

Figure 1: *A flowchart showing steps from initial sampling to full reconstruction, displaying the multi-step refinement procedure*

tures during reconstruction [FCOS05] and the estimation of curvature information of point-sampled surfaces [MFP05]. MLS has also been proven suitable for computing higher order derivatives [CFCN*07] and for the representation of potential energy surfaces [TWMT07].

Aiming to reduce the number of particle paths as well, we propose using MLS to also offer continuity within the FTLE field reconstruction. In the context of works mentioned, particle placement using MLS is flexible, not being bound to a grid, and the continuous property of MLS allows for a smooth reconstruction regardless of sample size. In the process of the reconstruction, we refine the sampling by advecting points around LCS to obtain clearer LCS structures. In short, we are able to perform LCS refinement without LCS extraction while maintaining a low number of particle advections, the expensive portion of FTLE computation.

4. Approximation Algorithm

MLS will reconstruct the full data set from the sparse data throughout the image. However, we are most interested in accurately reconstructing Lagrangian Coherent Structures within the flow field. These structures are defined by locally maximal FTLE values. With the following refinement steps, we aim to reconstruct these areas of interest with finer precision. These steps are shown in Figure 1 and detailed in the following sections.

4.1. Initial Sampling

It is well known that the accuracy of a MLS reconstruction relies heavily on the neighborhood over which the fit is made. A "good" neighborhood has sample points uniformly distributed around $\mathbf{x}$. If the neighborhood is skewed where the majority of neighboring points lie on one side of $\mathbf{x}$, the approximation can be inaccurate. To try and eliminate the possibility of a skewed neighborhood, we employ an initial staggered uniform sampling over the flow map, shown in Figure 2. Sampling begins by taking a uniform distribution of points along all coordinate axis, separated by the distance, h. Alternate rows are then adjusted, shifting points by half the distance of the uniform sampling, $\frac{h}{2}$, creating a staggered sampling. User-specified sampling is not a necessity for the use of MLS and there are a variety of other samplings that could be utilized, though we have found that the staggered sampling maximizes the effectiveness of a single advected particle on the points around it during the least squares fit.

4.2. Pre-MLS Refinement

Prior to the full MLS reconstruction, it would be useful to know which regions of the data set would benefit most from a finer sampling around the desired fit point $\mathbf{x}$. The MLS algorithm has a minimum restriction on the number of neighbors needed for the fit; the number of neighbors must be greater than the number of terms in the basis $\mathbf{b}^T(\mathbf{x})$. Therefore, a linear basis requires a minimum three neighbors, six neighbors for a quadratic basis and ten neighbors for a cubic basis. MLS then fits the desired function over the entire neighborhood. If strong variations in values occur within the neighborhood, the fit is then subject to over-smoothing, where the function maintains continuity but loses accuracy. The sections of the flow map that have a high tendency to over-smooth are the sections where a more dense sampling is desired. These are also the same sections that have high fluctuations of FTLE values. Uniformly sampling the initial FTLE field throughout the data set, we are able to determine the desired locations in need of dense sampling.

The staggered uniform sampling from which the MLS reconstruction will be performed provides the uniform sampling locations for the FTLE computation. In calculating the

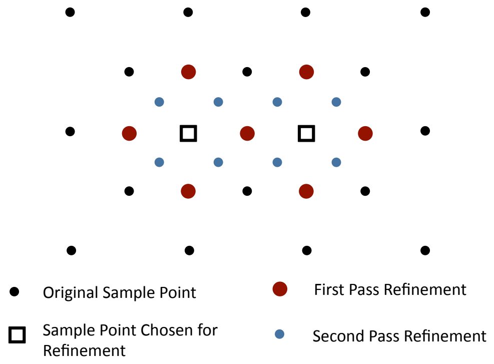


Figure 2: *An illustration of the Pre-MLS refinement procedure showing the addition of particles for each refinement pass*

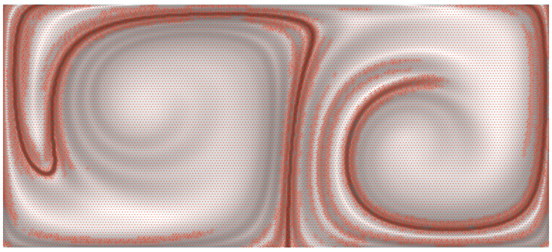


Figure 3: *An illustration of the Pre-MLS refinement step of the double gyre data set. The red dots show the points used for MLS reconstruction, with areas of interest covered by dense sampling*

FTLE of a given point, we follow the same procedure as the final MLS fit. The neighborhood size is equivalent, as is the weight function. To achieve a speed-efficient refinement step, the linear basis is used for the FTLE computation since ideal continuity will be achieved during the full reconstruction.

When choosing the appropriate locations for finer sampling, a quick second iteration is made through the original scattered data points. Once more, a neighborhood sampling is taken around each point, this time focusing on the FTLE values of the neighbors. Again, n is kept constant to retain the correspondence between the refinement and reconstruction stage. The locations requiring additional sample points are those that exhibit the greatest change in FTLE. By selecting neighborhoods based on the largest gradient of the Finite-Time Lyapunov Exponent, we are able to determine which neighborhoods would benefit most from a dense sampling. Large changes in FTLE are exhibited near sharp ridges and valleys of the reconstruction, i.e. close to salient LCS. Because we are interested in exposing the latter, all sampled points are first sorted according to FTLE value from lowest to highest, isolating areas of the flow map with high separation property. Looking at the top 50%, the number of neighborhoods to be refined are quantified by our algorithm which isolates the top quarter of the sorted fit points which exhibit the largest changes in FTLE within their domain of influence. Empirically, choosing 25% of the total fit points subject to our refinement criteria has proven to locate the main areas of interest in multiple data sets, as shown in Figure 3. Once regions of interest are located, additional sample points are added and advected through the flow field. To assure that the added points will have the greatest chance of increasing the accuracy of the MLS reconstruction, the structure of the staggered sampling field is utilized. Points are added around the center of the neighborhood, a distance that is half the uniform sampling distance, $\frac{h}{2}$, creating a finer grid sampling. Four points are added in two dimensions, six in three dimensions, allowing two points for refinement per axis. If points selected for refinement are in close proximity to one another, selection of new seed points may coincide and a number of points may not need to be advected. Using this allotted budget of extra seed points, a second refinement pass is made on the top end of the sorted fit points, as exemplified in Figure 2. This scheme forms a finer grid in the domain of influence, optimally filling empty spaces closest to the center. The algorithm supports any desired number of refinement passes according to a user-defined threshold that we set to $\frac{4}{5}$ of the neighborhood size of the initial MLS pass. We have also found that 3 passes are more than sufficient.

With the addition of points in the areas of interest, we attempt to optimize the efficiency of scattered data reconstruction. Following the guidelines of our work's goal to produce an accurate FTLE field with a low number of particles, our refinement steps allows a significant fraction of advected points to reside in regions of importance. Thus, we are able to achieve an accurate reconstruction with fewer particles.

5. Full Reconstruction and LCS

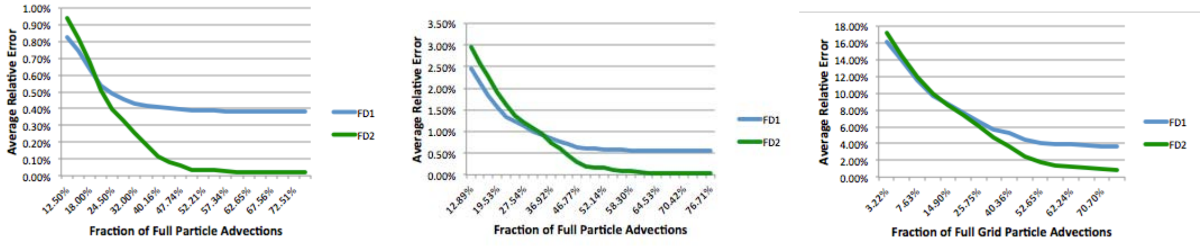
Post refinement, a full Moving Least Squares reconstruction step is taken to complete the FTLE field. Out of the least squares fit, flow map information can be recovered from MLS basis coefficients. Recall that the first four terms of any basis are $\mathbf{b}(\mathbf{x}_i) = [\,1, x, y, z, \ldots]$. Because the calculation of MLS was translated to the origin for local approximation, the first term of the solution holds the flow field value, the second coefficient holds the derivative with respect to x and the third, the derivative with respect to y. After solving the system of normal equations for both the x and y coordinate of the flow field at the fit point, the Jacobian required for FTLE computation simply drops out in the second and third terms of the fit function, retaining the continuity provided by MLS. Standard Finite-Time Lyapunov Exponent calculation follows and an FTLE field is created.

By detecting local extrema within this field, Lagrangian Coherent Structures may be obtained. LCS are extracted as height ridges in the local sense, categorized as local maxima and more specifically, locations where the scalar field of FTLE values has a local maximum in at least one direction.

The criteria for extracting local ridges are formed by using the gradient $\mathbf{g}$ and Hessian $\mathbf{H}$ of the FTLE values. Let $\mathbf{e}_i$ be the eigenvectors of the Hessian and λ_i be the eigenvalues. Height ridges are d-dimensional manifolds within n-dimensional space where $n > d \geq 0$. To be classified as a height ridge, the derivatives in the orthogonal direction must be zero and the associated eigenvalues must be negative. This leads to the two conditions that $\mathbf{e}_i \cdot \mathbf{g} = 0$ and $\lambda_i < 0$ for $i = d+1, \ldots, n$ to constrain height ridge extraction over orthogonal eigenvectors.

5.1. Height Ridge Filtering

While the two ridge criteria are guaranteed to extract height ridges, because eigenvalues and eigenvectors are taken from



(a) Double Gyre (b) Seven Squares (c) Plume

Figure 4: *Average relative error plots with FD1 and FD2 representing low-order and high-order finite-difference approximation respectively*

the Hessian, noise may appear while calculating the second derivative. To attempt to reduce this noise, the very same Moving Least Squares mechanism can be used to create a least squares fit, taking advantage of the continuous nature of MLS, producing both the Hessian and gradient from the FTLE field. As previously discussed, the second and third coefficients of the solution hold the gradient information. To extract Hessian information, MLS must use at least a quadratic basis, $\mathbf{b}(\mathbf{x}_i) =[\ 1, x, y, x^2, xy, y^2\ ]$. With the quadratic basis, terms four through six of the solution hold Hessian information. This is also true for higher order solutions. False positives, or weak ridges, are filtered out by setting a minimum cap on the FTLE field. A second technique is to filter out flat ridges, or ridges that do not have sharp curvature. This criterion is established upon the negative eigenvalue and states that $\lambda_i \leq \lambda_{bound}$ where λ_{bound} is chosen to be a cut-off point for curvature via the second derivative.

6. Results

Within this section, we describe our findings using MLS to reconstruct the divergence rates of particles seeded within a transient flow. To demonstrate the effectiveness of our method, we exemplify three time-dependent data sets of increasing complexity. The first data set, named the double gyre, is an analytic function representing two counter rotating gyres. Second, we use the seven square dataset, which reveals a flow through seven square-shaped, immovable objects. The final test data set is a three-dimensional plume with fairly complex behavior.

- Double Gyre: grid size of 800×600 with 150 advection steps per particle with a stepsize of .1
- Seven Square: grid size of 1024×512 with 100 advection steps per particle with a stepsize of .0001
- Plume: grid size of 128×256×128 with 100 advection steps per particle with a stepsize of .001

Numerical Error Analysis To compare against our MLS scheme, error analysis is performed against a ground truth obtained by using finite differences for the Jacobian calculation needed for FTLE. The graphs in Figure 4 show the relative error between our method and the finite-difference approach, based on FTLE values. For the calculation of the ground truth, each data set is sampled on a regular grid with a user defined minimum allotted spacing between data points, thereby setting a maximum number of advected particles. The same user-defined resolution is kept when inserting new particles during the adaptive stage of our method. By limiting the spacing of the two methods to the same distance between data points, we are able to use one as ground truth.

At first, we used a low-order finite-difference approximation to obtain the Jacobian required for FTLE.

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} \tag{9}$$

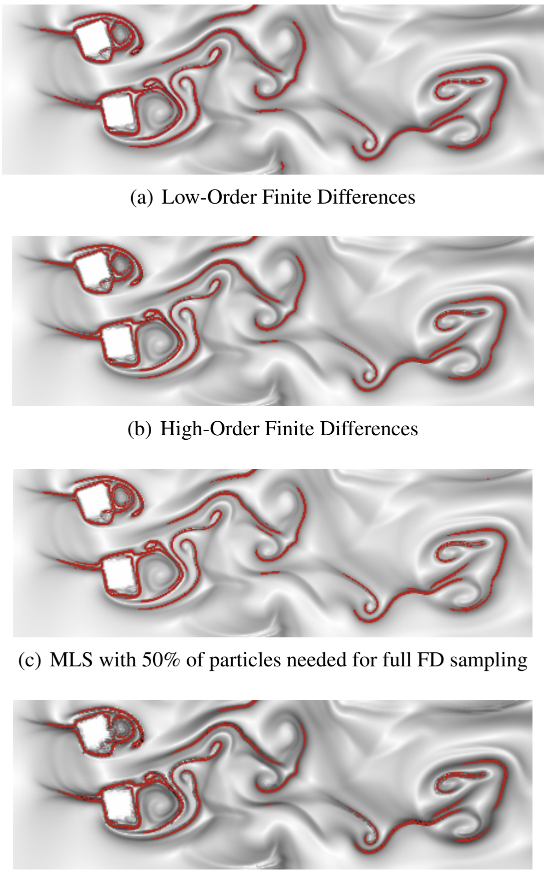
When calculating FTLE using (9), there is a horizontal asymptote that begins at slightly over 50% of full particle advection and prevents the error from reaching zero, as seen by the blue line in Figure 4. Our interpretation was that the low-order FD approach was not accurate enough and we opted for a higher order finite-difference approximation

$$f'(x) = \frac{-f(x+2h) + 8f(x+h) - 8f(x-h) + f(x-2h)}{12h} \tag{10}$$

Where (10) provides a more precise approximation to the derivative with the use of a 5-point stencil. The green line in Figure 4 shows the average relative error of our method compared against the ground truth using the higher order finite-difference equation. Examining the use of (10), the error begins to zero out as the number of particle advections reaches the fully sampled amount. Acknowledging both runs, we show that obtaining the Jacobian matrix from the least squares fit provides more accurate values than using (9) and values comparable to the use of (10). But higher-order finite-difference approximation requires more particles at specific locations while MLS can achieve comparable approximation accuracy from arbitrarily placed particles.

Visualization Results The error analysis graphs in figure 4 are able to give us a measure of the number of advections needed to produce a substantial image. In Figure 5, we have a feature level comparison of LCS extracted from a fully sampled regular grid using finite differences, to our method which uses 50% and 25% of the particles advected for the FD approach. We conclude that slightly over half the particles are needed, when using our method, to produce an image comparable to a fully sampled image produced using (9) with a finite differences calculation.

With no restriction on particle placement when using MLS to find the FTLE, our seeding scheme is intended to maximize the influence of every advected particle. Therefore, MLS is able to give the intrinsic form of the FTLE field with only a small fraction of particles. Figure 6 contains MLS reconstructions at 1.5% and .2% of the particles required for a full sampling using finite differences. Though missing the finer details, these figures convey the rise of the plume, covering the major areas of flow divergence and convergence. Comparing the 1.5% resolution of MLS to FD, MLS forms a smooth image while FD is subject to discretization artifacts. By lowering the number of initial sample points, our method provides an overview of the FTLE field while still using the refinement scheme to better convey areas of interest.


(a) Low-Order Finite Differences

(b) High-Order Finite Differences

(c) MLS with 50% of particles needed for full FD sampling

(d) MLS with 25% of particles needed for full FD sampling

Figure 5: *Ridge extraction and comparison of LCS in the Seven Square data set extracted using finite differences and MLS.*

7. Conclusion & Future Work

We have shown that the use of the Moving Least Squares framework allowed us to extract FTLE ridges and visualize Lagrangian Coherent Structures with less computational effort. The continuous properties of the framework provided numerical advantages during LCS extraction and the flexibility alloted by MLS enabled local refinement without mesh restrictions. Moreover, MLS offers competitive results to the standard FTLE calculation by finite difference, while strongly reducing the number of particles required to efficiently evaluate Finite-Time Lyapunov Exponents. In closing, we have shown how multiple areas of LCS-based flow computation and visualization has benefited from the MLS framework.

We conclude with a look at future extensions of our work. There are many avenues for consideration,

- While we did not perform ridge extraction in 3d for this paper, the corresponding technique appears straightforward and will be realized in future work.
- To achieve faster reconstruction, the refinement and fit of the data can be implemented on the GPU.

References

[BMXK06] Bodin A., Ma J., Xin X., Krishnaswami P.: A meshless integral method based on regularized boundary integral equation. *Computer Methods in Applied Mechanics and Engineering 195*, 44-47 (2006), 6258 – 6286. 3

[CFCN*07] Cueto-Felgueroso L., Colominas I., Nogueira X. A., Navarrina F., Casteleiro M.: Finite volume solvers and moving least-squares approximations for the compressible navier-stokes equations on unstructured grids. *Computer Methods in Applied Mechanics and Engineering 196*, 45-48 (2007), 4712 – 4736. 4

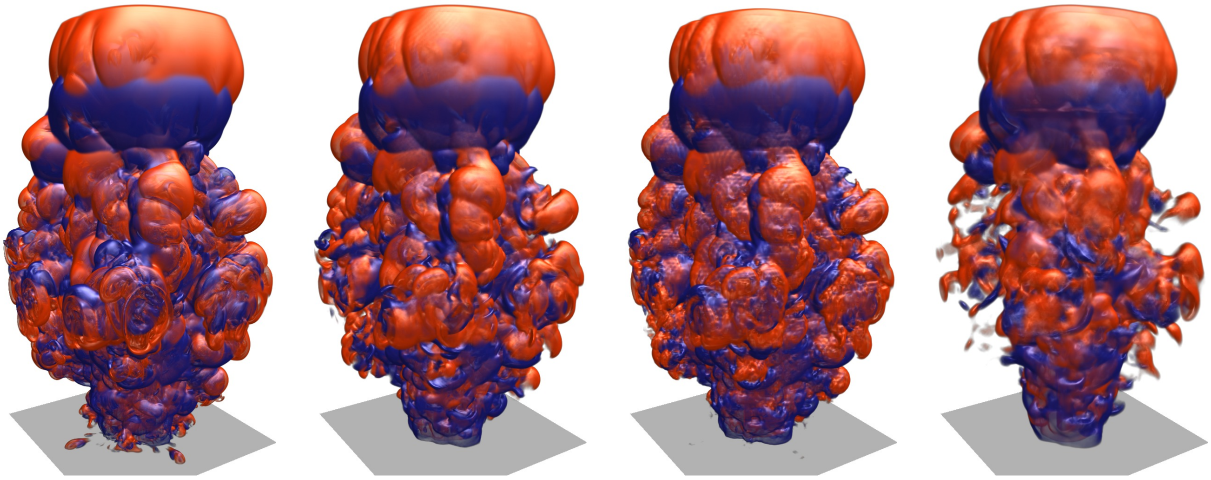
[CSO09] Chenoweth S. K., Soria J., Ooi A.: A singularity-avoiding moving least squares scheme for two-dimensional unstructured meshes. *Journal of Computational Physics 228*, 15 (2009), 5592 – 5619. 3

[FCOS05] Fleishman S., Cohen-Or D., Silva C. T.: Robust moving least-squares fitting with sharp features. *ACM Trans. Graph. 24*, 3 (2005), 544–552. 4

[GGTH07] Garth C., Gerhardt F., Tricoche X., Hagen H.: Efficient computation and visualization of coherent structures in fluid flow applications. *IEEE Transactions on Visualization and Computer Graphics 13*, 6 (2007), 1464–1471. 3

[GLT*07] Garth C., Li G., Tricoche X., Hansen C. D., Hans H.: Visualization of coherent structures in transient 2d flows. *In Topology-Based Methods in Visualization Proceedings of the 2007 Workshop* (2007), page to appear. 3

[Hal00] Haller G.: Finding finite-time invariant manifolds in two-dimensional velocity fields. *Chaos: An Interdisciplinary Journal of Nonlinear Science 10*, 1 (2000), 99–108. 3



FD, 100% samples MLS, 1.5% samples FD, 1.5% samples MLS, 0.2% samples

Figure 6: *A comparison of finite-difference (FD) and MLS reconstruction of FTLE fields in the plume data set where red is forward FTLE and blue is backward FTLE.*

[Hal01a] Haller G.: Distinguished material surfaces and coherent structures in three-dimensional fluid flows. *Physica D: Nonlinear Phenomena 149*, 4 (2001), 248 – 277. 3

[Hal01b] Haller G.: Lagrangian structures and the rate of strain in a partition of two-dimensional turbulence. *Physics of Fluids 13*, 11 (2001), 3365–3385. 3

[Hal02] Haller G.: Lagrangian coherent structures from approximate velocity data. *Physics of Fluids 14*, 6 (2002), 1851–1861. 3

[HY00] Haller G., Yuan G.: Lagrangian coherent structures and mixing in two-dimensional turbulence. *Physica D: Nonlinear Phenomena 147*, 3-4 (2000), 352 – 370. 3

[KPH*09] Kasten J., Petz C., Hotz I., Noack B. R., Christian Hege H.: Localized finite-time lyapunov exponent for unsteady flow analysis. *Vision Modeling and Visualization* (2009), 265–274. 3

[Lev98] Levin D.: The approximation power of moving least-squares. *Mathematics of Computation 67* (1998), 1517–1531. 3

[LGM*08] Ledergerber C., Guennebaud G., Meyer M., Bacher M., Pfister H.: Volume mls ray casting. *IEEE Transactions on Visualization and Computer Graphics 14(6)* (2008), 1539–1546. 3

[LR10] Lekien F., Ross S. D.: The computation of finite-time lyapunov exponents on unstructured meshes and for non-euclidean manifolds. *Chaos: An Interdisciplinary Journal of Nonlinear Science 20*, 1 (2010), 017505. 3

[LSM07] Lekien F., Shadden S. C., Marsden J. E.: Lagrangian coherent structures in n-dimensional systems. *Journal of Mathematical Physics 48*, 6 (2007), 065404. 3

[MFP05] Miao1 Y., Feng J., Peng Q.: Curvature estimation of point-sampled surfaces and its applications. *Computational Science and Its Applications Volume 3482/2005* (2005), 1023–1032. 4

[MHP*07] Mathur M., Haller G., Peacock T., Ruppert-Felsot J. E., Swinney H. L.: Uncovering the lagrangian skeleton of turbulence. *Phys. Rev. Lett. 98*, 14 (Apr 2007), 144502. 3

[PS08] Peikert R., Sadlo F.: Height ridge computation and filtering for visualization. In *Visualization Symposium, 2008. PacificVIS '08. IEEE Pacific* (2008), pp. 119 –126. 3

[She68] Shepard D.: A two-dimensional interpolation function for irregularly-spaced data. In *ACM '68: Proceedings of the 1968 23rd ACM national conference* (New York, NY, USA, 1968), ACM, pp. 517–524. 3

[SLM05] Shadden S., Lekien F., Marsden J.: Definition and properties of lagrangian coherent structures from finite-time lyapunov exponents in two-dimensional aperiodic flows. *Physica D* (2005), 212:271–304. 3

[SP07] Sadlo F., Peikert R.: Efficient visualization of lagrangian coherent structures by filtered amr ridge extraction. *Visualization and Computer Graphics, IEEE Transactions on 13*, 6 (2007), 1456 –1463. 3

[SP09] Sadlo F., Peikert R.: Visualizing lagrangian coherent structures and comparison to vector field topology. In *Topology-Based Methods in Visualization II*, Hege H.-C., Polthier K., Scheuermann G., (Eds.), Mathematics and Visualization. Springer Berlin Heidelberg, 2009, pp. 15–29. 3

[STS10] Schultz T., Theisel H., Seidel H.-P.: Crease surfaces: From theory to extraction and application to diffusion tensor mri. *IEEE Transactions on Visualization and Computer Graphics 16* (2010), 109–119. 3

[TCH10] Tang W., Chan P. W., Haller G.: Accurate extraction of lagrangian coherent structures over finite domains with application to flight data analysis over hong kong international airport. *Chaos: An Interdisciplinary Journal of Nonlinear Science 20*, 1 (2010), 017502. 3

[TWMT07] Tokmakov I. V., Wagner A. F., Minkoff M., Thompson D. L.: Gradient incorporation in one-dimensional applications of interpolating moving least-squares methods for fitting potential energy surfaces. *Theoretical Chemistry Accounts: Theory, Computation, and Modeling (Theoretica Chimica Acta) 118*, 4 (October 2007), 755–767. 4