Commun. Comput. Phys.
doi: 10.4208/cicp.OA-2018-0149

Vol. **26**, No. 4, pp. 1143-1177
October 2019

Estimating the Finite Time Lyapunov Exponent from Sparse Lagrangian Trajectories

Yu-Keung Ng[1] and Shingyu Leung[1,*]

[1] *Department of Mathematics, The Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong.*

Received 5 June 2018; Accepted (in revised version) 15 November 2018

Abstract. We propose a simple numerical algorithm to estimate the finite time Lyapunov exponent (FTLE) in dynamical systems from only a sparse number of Lagrangian particle trajectories. The method first reconstructs the flow field using the radial basis function (RBF) and then uses either the Lagrangian or the Eulerian approach to determine the corresponding flow map. We also develop a simple algorithm based on the Schur complement for updating, rather than recomputing, the reconstruction in the RBF when new trajectory data is made available in applications. We will demonstrate the effectiveness of the proposed method using examples from autonomous and aperiodic flows, and also measurements from real-life data.

AMS subject classifications: 37M05, 37M25, 37M99, 65L05

Key words: Dynamical system, visualization, finite time Lyapunov exponent, numerical methods for differential equations.

1 Introduction

Lagrangian coherent structure (LCS) is an important tool to visualize, understand and extract useful information in a complex dynamical system. It aims to distinguish surfaces of trajectories in a dynamical system and distinguishes regions of phase space that influence nearby trajectories over a time interval of interest [16]. The concept has been widely applied in various fields including ocean flows [20, 35], hurricane structures [32], flight path [4, 36], gravity waves [37] and some bio-inspired fluid flows [11, 25, 26]. One possible approach to extract such a flow structure is the finite-time Lyapunov exponent (FTLE) [12, 13, 17, 21, 35]. This quantity measures the magnitude of maximum rate of deformation in the distance between neighboring particles across a finite interval of time with an infinitesimal perturbation in the initial position. The method first computes the flow map which links the initial location of a particle with the arrival position based on

*Corresponding author. *Email addresses:* ykngad@ust.hk (Y.-K. Ng), masyleung@ust.hk (S. Leung)

the characteristic line. Mathematically these particles in the extended phase space satisfy the ordinary differential equation (ODE) given by $\mathbf{x}'(t)=\mathbf{u}(\mathbf{x}(t),t)$ with the initial condition $\mathbf{x}(t_0)=\mathbf{x}_0$ and a Lipschitz velocity field $\mathbf{u}:\mathbb{R}^d\times\mathbb{R}\to\mathbb{R}^d$. We can define the flow map $\Phi:\mathbb{R}^d\to\mathbb{R}^d$ as the mapping which takes the point $\mathbf{x}_0$ to the particle location at the final time $t=t_0+T$, i.e $\Phi(\mathbf{x}_0;t_0,T)=\mathbf{x}(t_0+T)$ with $\mathbf{x}(t)$ satisfying the governing ODE. The FTLE is then defined using the largest eigenvalue of the deformation matrix based on the Jacobian of this resulting flow map. Following the definition of Haller [12,14,15], one can see that a possible definition of the LCS is closely related to the *ridges* of the FTLE fields.

Numerically, on the other hand, because both LCS and FTLE have long been treated as a Lagrangian property of a continuous dynamical system, most (if not all) numerical methods are developed based on the traditional Lagrangian ray tracing method by solving the ODE system using a well-developed numerical integrator. Some other Lagrangian methods are summarized in the review article [1] and we refer any interested reader to paper and thereafter. We have recently proposed several Eulerian approaches to compute the FTLE on a fixed Cartesian mesh [22]. The idea is to incorporate the approach with the Level Set Method (LSM) [27] which allows the flow map to solve a set of Liouville equations. These hyperbolic partial differential equations (PDE) can then be solved by any well-established robust and high order accurate numerical methods. Some other Eulerian methods have also been developed in [23, 39–42].

In all the algorithms we have discussed and have developed so far, we assumed that the velocity field is given throughout the whole domain. However, in some applications where the raw data are measured in a Lagrangian fashion given by a finite number of particle trajectories, the above methods cannot be directly applied. In this work, we will first consider the problem of constructing the flow map from a finite number of particle trajectories. The technique proposed in this work will be important and will be served as a building block for the future development of any Eulerian algorithms.

Mathematically, we label each particle by i and denote the corresponding trajectory by $\mathbf{x}_i(t_k)$ for $i=1,2,\cdots,M$ and $k=0,1,\cdots,K$ such that $t_K=T$. This immediately implies that the flow map at some discrete locations is given by $\Phi_0^T(\mathbf{x}_i(0))=\mathbf{x}_i(t_K)$. Even though the flow map is defined in general only at scattered locations, we can still apply techniques such as the radial basis function (RBF) [3] to interpolate the flow map throughout the whole computational domain. Then the corresponding FTLE can be computed using any standard method. However, for a small number of particle trajectories, i.e. when M is small, we do not expect to obtain an accurate reconstruction of the flow map. One simple reason is that such method ignores all information from any intermediate time levels. The history from each individual path might provide a more complete description of the underlying flow. More importantly, with a small number of trajectories given initially, the provided information would not cover the whole phase space (the $\mathbf{x}-t$ space) but could contain most properties near the attractor of the dynamical system.

Therefore, instead of interpolating the flow map immediately from the given particle trajectories, we propose to first approximate the velocity at $(\mathbf{x}_i,t_k)$ in the $\mathbf{x}-t$ space by applying any simple finite difference method to the given data. Then, we will interpolate