Journal of Computational Physics 235 (2013) 1–13

Contents lists available at SciVerse ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Forward and adjoint sensitivity computation of chaotic dynamical systems

Qiqi Wang *

Department of Aeronautics and Astronautics, MIT, 77 Mass Ave., Cambridge, MA 02139, USA

ARTICLE INFO

Article history:
Received 23 February 2012
Accepted 5 September 2012
Available online 27 October 2012

Keywords:
Sensitivity analysis
Linear response
Adjoint equation
Unsteady adjoint
Chaos
Statistical average
Lyapunov exponent
Lyapunov covariant vector
Lorenz attractor

ABSTRACT

This paper describes a forward algorithm and an adjoint algorithm for computing sensitivity derivatives in chaotic dynamical systems, such as the Lorenz attractor. The algorithms compute the derivative of long time averaged “statistical” quantities to infinitesimal perturbations of the system parameters. The algorithms are demonstrated on the Lorenz attractor. We show that sensitivity derivatives of statistical quantities can be accurately estimated using a single, short trajectory (over a time interval of 20) on the Lorenz attractor.

1. Introduction

Computational methods for sensitivity analysis is a powerful tool in modern computational science and engineering. These methods calculate the derivatives of output quantities with respect to input parameters in computational simulations. There are two types of algorithms for computing sensitivity derivatives: the forward algorithms and the adjoint algorithms. The forward algorithms are more efficient for computing sensitivity derivatives of many output quantities to a few input parameters; the adjoint algorithms are more efficient for computing sensitivity derivatives of a few output quantities to many input parameters. Key application of computational methods for sensitivity analysis include aerodynamic shape optimization [3], adaptive grid refinement [9], and data assimilation for weather forecasting [8].

In simulations of chaotic dynamical systems, such as turbulent flows and the climate system, many output quantities of interest are “statistical averages”. Denote the state of the dynamical system as $x(t)$; for a function of the state $J(x)$, the corresponding statistical quantity $\langle J\rangle$ is defined as an average of $J(x(t))$ over an infinitely long time interval:

$$\langle J\rangle = \lim_{T\to\infty}\frac{1}{T}\int_0^T J(x(t))\,dt. \tag{1}$$

For ergodic dynamical systems, a statistical average only depends on the governing dynamical system, and does not depend on the particular choice of trajectory $x(t)$.

Many statistical averages, such as the mean aerodynamic forces in turbulent flow simulations, and the mean global temperature in climate simulations, are of great scientific and engineering interest. Computing sensitivities of these statistical quantities to input parameters can be useful in many applications.

* Tel.: +1 617 253 0921.
E-mail address: qiqi@mit.edu

0021-9991/$ - see front matter
http://dx.doi.org/10.1016/j.jcp.2012.09.007

The differentiability of these statistical averages to parameters of interest as been established through the recent developments in the Linear Response Theory for dissipative chaos [6,7]. A class of chaotic dynamical systems, known as "quasi-hyperbolic" systems, has been proven to have statistical quantities that are differentiable with respect to small perturbations. These systems include the Lorenz attractor, and possibly many systems of engineering interest, such as turbulent flows.

Despite recent advances both in Linear Response Theory [7] and in numerical methods for sensitivity computation of unsteady systems [10,4], sensitivity computation of statistical quantities in chaotic dynamical systems remains difficult. A major challenge in computing sensitivities in chaotic dynamical systems is their sensitivity to the initial condition, commonly known as the "butterfly effect". The linearized equations, used both in forward and adjoint sensitivity computations, give rise to solutions that blow up exponentially. When a statistical quantity is approximated by a finite time average, the computed sensitivity derivative of the finite time average diverges to infinity, instead of converging to the sensitivity derivative of the statistical quantity [5]. Existing methods for computing correct sensitivity derivatives of statistical quantities usually involve averaging over a large number of ensemble calculations [5,1]. The resulting high computation cost makes these methods not attractive in many applications.

This paper outlines a computational method for efficiently estimating the sensitivity derivative of time averaged statistical quantities, relying on a single trajectory over a small time interval. The key idea of our method, inversion of the "shadow" operator, is already used as a tool for proving structural stability of strange attractors [6]. The key strategy of our method, divide and conquer of the shadow operator, is inspired by recent advances in numerical computation of the Lyapunov covariant vectors [2,11].

In the rest of this paper, Section 2 describes the "shadow" operator, on which our method is based. Section 3 derives the sensitivity analysis algorithm by inverting the shadow operator. Section 4 introduces a fix to the singularity of the shadow operator. Section 5 summarizes the forward sensitivity analysis algorithm. Section 6 derives the corresponding adjoint version of the sensitivity analysis algorithm. Section 7 demonstrates both the forward and adjoint algorithms on the Lorenz attractor. Section 8 concludes this paper.

The paper uses the following mathematical notation: Vector fields in the state space (e.g. $f(x), \phi_i(x)$) are column vectors; gradient of scalar fields (e.g. $\frac{\partial a_i^x}{\partial x}$) are row vectors; gradient of vector fields (e.g. $\frac{\partial f}{\partial x}$) are matrices with each row being a dimension of f, and each column being a dimension of x. The $(\cdot)$ sign is used to identify matrix–vector products or vector-vector inner products. For a trajectory $x(t)$ satisfying $\frac{dx}{dt} = f(x)$ and a scalar or vector field $a(x)$ in the state space, we often use $\frac{da}{dt}$ to denote $\frac{da(x(t))}{dt}$. The chain rule $\frac{da}{dt} = \frac{da}{dx} \cdot \frac{dx}{dt} = \frac{da}{dx} \cdot f$ is often used without explanation.

2. The "shadow operator"

For a smooth, uniformly bounded n dimensional vector field $\delta x(x)$, defined on the n dimensional state space of x. The following transform defines a slightly "distorted" coordinates of the state space:

$$x'(x) = x + \epsilon\,\delta x(x), \tag{2}$$

where ϵ is a small real number. Note that for an infinitesimal ϵ, the following relation holds:

$$x'(x) - x = \epsilon\,\delta x(x) = \epsilon\,\delta x(x') + O(\epsilon^2). \tag{3}$$

We call the transform from x to x' as a "shadow coordinate transform". In particular, consider a trajectory $x(t)$ and the corresponding transformed trajectory $x'(t) = x'(x(t))$. For a small ϵ, the transformed trajectory $x'(t)$ would "shadow" the original trajectory $x(t)$, i.e., it stays uniformly close to $x(t)$ forever. Fig. 1 shows an example of a trajectory and its shadow.

Now consider a trajectory $x(t)$ satisfying an ordinary differential equation

$$\dot{x} = f(x) \tag{4}$$

with a smooth vector field $f(x)$ as a function of x. The same trajectory in the transformed "shadow" coordinates $x'(t)$ do not satisfy the same differential equation. Instead, from Eq. (3), we obtain

$$\dot{x'} = f(x) + \epsilon \frac{\partial \delta x}{\partial x} \cdot f(x) = f(x') - \epsilon \frac{\partial x}{\partial x} \cdot \delta x(x') + \epsilon \frac{\partial \delta x}{\partial x} \cdot f(x') + O(\epsilon^2) \tag{5}$$

In other words, the shadow trajectory $x'(t)$ satisfies a slightly perturbed equation

$$\dot{x'} = f(x') + \epsilon\,\delta f(x') + O(\epsilon^2), \tag{6}$$

where the perturbation δf is

$$\delta f(x) = -\frac{\partial f}{\partial x} \cdot \delta x(x) + \frac{\partial \delta x}{\partial x} \cdot f(x) = -\frac{\partial f}{\partial x} \cdot \delta x(x) + \frac{d\delta x}{dt} := (S_f \delta x)(x). \tag{7}$$

For a given differential equation $\dot{x} = f(x)$, Eq. (7) defines a linear operator $S_f : \delta x \Rightarrow \delta f$. We call S_f the "**shadow operator**" of f. For any smooth vector field $\delta x(x)$ that defines a slightly distorted "shadow" coordinate system in the state space, S_f deter-