

# Least Squares Shadowing Sensitivity Analysis of Chaotic Flow around a Two-Dimensional Airfoil

Patrick J. Blonigan\* and Qiqi Wang<sup>†</sup>

*Massachusetts Institute of Technology, Cambridge, MA, 02139, USA*

Eric J. Nielsen<sup>‡</sup>

*NASA Langley Research Center, Hampton, VA, 23681*

Boris Diskin<sup>§</sup>

*National Institute of Aerospace, Hampton, VA, 23666*

Gradient-based sensitivity analysis has proven to be an enabling technology for many applications, including design of aerospace vehicles. However, conventional sensitivity analysis methods break down when applied to long-time averages of chaotic systems. This breakdown is a serious limitation because many aerospace applications involve physical phenomena that exhibit chaotic dynamics, most notably high-resolution large-eddy and direct numerical simulations of turbulent aerodynamic flows. A recently proposed methodology, Least Squares Shadowing (LSS), avoids this breakdown and advances the state of the art in sensitivity analysis for chaotic flows. The first application of LSS to a chaotic flow simulated with a large-scale computational fluid dynamics solver is presented. The LSS sensitivity computed for this chaotic flow is verified and shown to be accurate, but the computational cost of the current LSS implementation is high.

## Nomenclature

$\hat{\mathbf{A}}$	$mn \times mn$ Schur complement of Karush-Kuhn-Tucker (KKT) matrix
$\hat{\mathbf{B}}$	$mn \times (m+1)n$ forward time integration block in KKT matrix
$\mathbf{b}$	$n \times 1$ explicit sensitivity of instantaneous spatial residual
$\hat{\mathbf{b}}$	$mn \times 1$ vector of $\mathbf{b}$ at all time steps
$\hat{\mathbf{C}}$	$mn \times m$ time dilation block in KKT matrix
$\mathbf{D}$	$1 \times p$ vector of design variables
$D$	Drag
$\mathbf{F}, \mathbf{G}$	$n \times n$ time integration matrices
$f, F$	Objective function or instantaneous contribution to objective function
$\mathbf{g}$	$n \times 1$ linearization of instantaneous contribution to objective function
$\hat{\mathbf{g}}$	$(m+1)n \times 1$ vector of $\mathbf{g}$ at all time steps
$h$	Time dilation instantaneous contribution to sensitivity of objective function
$\hat{h}$	$m \times 1$ vector of $h$ at all time steps
$\mathbf{L}$	$3 \times 1$ vector of nonlinear sources in Lorenz equation
$L$	Lagrangian
$M$	Mach number
$m$	Number of time steps in time horizon
$m_0$	Time step index at start of objective function averaging window
$m_1$	Time step index at end of objective function averaging window

\*Doctoral Candidate, Aeronautics and Astronautics, Cambridge, MA, AIAA Member.

<sup>†</sup>Associate Professor, Aeronautics and Astronautics, Cambridge, MA, AIAA Member.

<sup>‡</sup>Research Scientist, Computational AeroSciences Branch, MS 128, AIAA Associate Fellow.

<sup>§</sup>NIA Research Fellow and Research Associate Professor, MAE Department, University of Virginia, Charlottesville, VA, AIAA Associate Fellow.

$n$	Number of grid points times number of conserved quantities
$p$	Objective function exponent
$\mathbf{Q}$	$n \times 1$ vector of conserved variables
$\mathbf{Q}'$	Shadow trajectory for unsteady flow field
$\mathbf{R}$	$n \times 1$ spatial residual vector
$T_0$	Start of time-averaging window
$t$	Time
$\mathbf{U}$	$3 \times 1$ vector of Lorenz system states $x$ , $y$ , and $z$
$\mathbf{U}'$	$3 \times 1$ vector of Lorenz system shadow trajectory
$\mathcal{V}$	$n \times n$ diagonal matrix of dual volumes at a time step
$\hat{\mathcal{V}}$	$(m+1)n \times (m+1)n$ block-diagonal matrix of dual volumes at all time steps
$\mathbf{V}$	$n \times p$ tangent instantaneous solution matrix
$\hat{\mathbf{V}}$	$(m+1)n \times 1$ vector of $\mathbf{V}$ at all time steps
$\mathbf{W}$	$n \times 1$ instantaneous Lagrange multiplier for tangent LSS
$\hat{\mathbf{W}}$	$mn \times 1$ vector of $\mathbf{W}$ at all time steps
$\mathbf{X}$	$nm \times 1$ vector to verify LSS
$\mathbf{x}, \mathbf{y}$	$n \times 1$ vectors
$x, y, z$	Lorenz system states
$\alpha$	LSS time-transformation weighting
$\Delta T$	Time-averaging window
$\Delta t$	Time step
$\varepsilon$	Design-parameter perturbation
$\zeta$	Additional shadow time
$\eta$	Time dilation
$\hat{\eta}$	$m \times 1$ vector of $\eta$ at all time steps
$\mathbf{\Lambda}$	$n \times 1$ instantaneous adjoint field
$\hat{\mathbf{\Lambda}}$	$mn \times 1$ adjoint field at all time steps
$\rho, \sigma, \beta$	Lorenz system parameters
$\sigma$	Standard deviation estimator
$\tau$	Transformed time for shadow trajectory
$\Phi$	Instantaneous adjoint for LSS
$\hat{\Phi}$	$(m+1)n \times 1$ vector of $\Phi$ at all time steps
$\psi$	Adjoint $\eta$ for LSS
$\hat{\psi}$	$m \times 1$ vector of $\psi$ at all time steps
$\omega$	Objective-function weighting parameter

#### Subscript

$f$	$n \times 1$ vector to be communicated
$j$	$j$ th column of a matrix
$\infty$	Freestream value
$\Delta T$	Time-averaged quantity with averaging window of length $\Delta T$

#### Superscript

$IF$	Initial flow field
$i$	Time step
$\wedge$	Indicates quantity related to all time levels
$-$	Indicates time-averaged quantity

## I. Introduction

Gradient-based sensitivity analysis is an important tool for high-fidelity design and optimization of complex dynamical systems. The analysis computes sensitivities of objective functions to input and design parameters. A forward (tangent) approach to computing sensitivities involves a small perturbation of one parameter at a time and a simulation of the perturbed system to compute the corresponding values of the

objective functions. This forward approach is cost effective for problems with many objective functions and few design/input parameters. Typical high-fidelity aerospace applications involve many design parameters and a high computational cost; for such applications, the forward approach becomes prohibitively expensive. A reverse (adjoint) approach to computing sensitivities involves solving a linear adjoint problem for each objective function. With the adjoint solution available, one can directly compute sensitivities of the corresponding objective function with respect to many design parameters. The total cost of the adjoint-based sensitivity analysis is similar to the cost of the baseline simulation. As a result, adjoint-based sensitivity analysis becomes an enabling methodology for high-fidelity aircraft design.<sup>1-5</sup>

The adjoint-based design and optimization methodology has been successfully applied to complex steady and unsteady aerodynamic flows, including flows governed by the Reynolds-Averaged Navier Stokes (RANS) equations.<sup>4,5</sup> Adjoint-based methods have also benefited other areas beyond high-fidelity sensitivity analysis. In particular, many aerospace computations use adjoint-based methods for mesh adaptation,<sup>6</sup> error estimation,<sup>7</sup> flow control,<sup>8,9</sup> and uncertainty quantification.<sup>10,11</sup> However, conventional gradient-based sensitivity analysis breaks down when the objective function is a long-time average of a chaotic flow solution. This is a serious limitation because practical aerospace problems involve physical phenomena that exhibit chaotic dynamics, most notably high-resolution large-eddy simulations (LES) and direct numerical simulations (DNS) of turbulent flows. Long-time averages of aerodynamic forces, such as lift or drag, or heat transfer are examples of important optimization metrics for aerospace applications. New approaches are needed for robust adjoint-based sensitivity analysis of highly-resolved turbulent flows.

The breakdown of conventional sensitivity analysis can be traced to certain fundamental properties of chaotic systems. Specifically, Lorenz's butterfly effect implies that the time evolution of a chaotic system is highly sensitive to initial conditions.<sup>12</sup> A small perturbation to initial conditions may grow exponentially over time, resulting in large differences in instantaneous solutions. This high sensitivity is observed for all chaotic systems. On the other hand, long-time averages have little sensitivity to initial conditions.

To illustrate the breakdown of conventional sensitivity analysis for chaotic systems, one can consider the Lorenz-63 model,<sup>12</sup> a three-degree-of-freedom model of the Rayleigh-Benard convection between hot and cold plates:

$$\begin{aligned}\frac{dx}{dt} - \sigma(y - x) &= 0, \\ \frac{dy}{dt} - x(\rho - z) + y &= 0, \\ \frac{dz}{dt} - xy + \beta z &= 0,\end{aligned}\tag{1}$$

where  $x$ ,  $y$ , and  $z$  are solution states, and  $t$  is time. For chaotic behavior, the parameters  $\rho$ ,  $\sigma$ , and  $\beta$  are chosen to be 28.0, 10.0, and 8/3, respectively.<sup>12</sup>

The sensitivity of the time-averaged solution state  $z$  to the parameter  $\rho$  is considered,

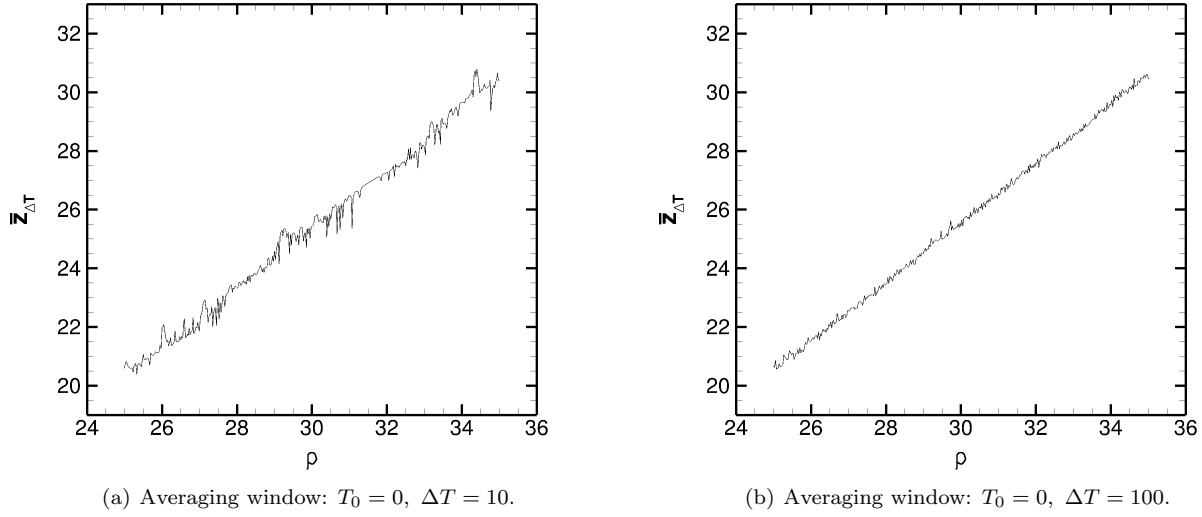
$$\bar{z}_{\Delta T}(\rho) = \frac{1}{\Delta T} \int_{T_0}^{T_0 + \Delta T} z(t, \rho) dt.\tag{2}$$

Figure 1 shows the variation of  $\bar{z}_{\Delta T}(\rho)$  versus  $\rho$  for two different time averaging windows. The variation is oscillatory. The amplitude of these oscillations decreases as the length of the averaging window,  $\Delta T$ , increases. At the same time, the frequency of these oscillations increases with  $\Delta T$ . The mean slope is close to one for both averaging windows, which is consistent with past studies.<sup>13,14</sup> Conventional sensitivity analysis computes local derivatives that are unbounded in the limit of an infinite averaging window, and are not representative of the mean slope. However, Ruelle's linear response theorem<sup>15</sup> indicates that, for many chaotic systems, parametric variations of infinite-time-averaged quantities, such as

$$\bar{z}_{\infty}(\rho) = \lim_{\Delta T \rightarrow \infty} \frac{1}{\Delta T} \int_{T_0}^{T_0 + \Delta T} z(t, \rho) dt,\tag{3}$$

are smooth, and the corresponding derivatives  $d\bar{z}_{\infty}/d\rho$  are well defined and bounded. Conventional sensitivity analysis fails to approximate such derivatives.

The fundamental reason for this breakdown is that the infinite-time limit and the infinitesimal-perturbation limit do not commute for time-averaged quantities in chaotic systems.



**Figure 1.** Parametric variation of the time-averaged solution,  $\bar{z}_{\Delta T}(\rho)$ , of Eq. (1) for the initial conditions  $x(0) = 1, y(0) = 0, z(0) = 28$ .

$$\lim_{\Delta \rho \rightarrow 0} \lim_{\Delta T \rightarrow \infty} \frac{1}{\Delta T} \int_{T_0}^{T_0 + \Delta T} \frac{z(t, \rho + \Delta \rho) - z(t, \rho)}{\Delta \rho} dt \neq \lim_{\Delta T \rightarrow \infty} \frac{1}{\Delta T} \int_{T_0}^{T_0 + \Delta T} \lim_{\Delta \rho \rightarrow 0} \frac{z(t, \rho + \Delta \rho) - z(t, \rho)}{\Delta \rho} dt \quad (4)$$

or

$$\frac{d\bar{z}_{\infty}}{d\rho} \neq \lim_{\Delta T \rightarrow \infty} \frac{d\bar{z}_{\Delta T}}{d\rho}. \quad (5)$$

A number of sensitivity analysis modifications have been proposed to avoid the breakdown faced by conventional methods. One such approach is the ensemble-adjoint method.<sup>13,16</sup> This method has been applied to an ocean circulation model with some success.<sup>17</sup> The ensemble-adjoint method involves averaging over a large number of ensemble calculations. It was found that the sample mean of sensitivities computed with the ensemble adjoint converges slower than  $N^{-0.5}$  in the cases considered, where  $N$  is the number of samples.<sup>16</sup> This method is less efficient than a brute-force Monte-Carlo approach. More recently, an adjoint approach for climate sensitivity analysis has been derived.<sup>18</sup> This approach involves finding a probability density function that satisfies a Fokker-Planck equation to model the climate. The adjoint of this equation has been used to compute derivatives with respect to long-time-averaged quantities. However, this method requires a phase-space discretization, making it computationally infeasible for systems with a high number of dimensions, such as high-fidelity aerodynamic simulations. Methods based on the fluctuation dissipation theorem have been successful in computing climate sensitivities for complex chaotic dynamical systems.<sup>19</sup> However, this approach has different variants for different types of chaotic systems, most of which have excessive computational costs.

The Least Squares Shadowing method (LSS) avoids many of these difficulties and is able to compute sensitivities of long-time averages for chaotic dynamical systems, including aerodynamic simulations. LSS relies on the shadowing lemma,<sup>20</sup> an important result from the theory of dynamical systems, and on the ergodicity assumption that the long-time averages of a chaotic solution are independent of the initial condition. The concept of shadowing has previously been applied in continuation methods to find homoclinic and heteroclinic orbits of dynamical systems<sup>21,22</sup> and for noise reduction.<sup>23</sup> Recently, LSS has been applied to sensitivity analysis of a number of chaotic systems, including the Lorenz-63 system,<sup>14</sup> a modified Kuramoto-Sivashinsky equation,<sup>24</sup> and a DNS of homogeneous isotropic turbulence.<sup>25</sup> This paper presents the first application of LSS to chaotic solutions computed by an adjoint-enabled large-scale computational fluid dynamics (CFD) solver.<sup>26</sup> As computational capabilities expand and become more accessible, LSS may enable sensitivity analysis for applications requiring the higher resolution of LES or DNS.

The test case considered in this paper is a two-dimensional airfoil flow exhibiting chaotic vortex shedding. The main goal of this study is to demonstrate that LSS can compute an accurate sensitivity of a long-time-averaged quantity of a chaotic flow. Additionally, a methodology to verify the LSS implementation is presented, and the computational cost of LSS sensitivity analysis is assessed. The paper is organized as follows. Section II presents the baseline CFD and sensitivity analysis solvers. Section III discusses the LSS methodology. The verification of the LSS implementation is shown in Section IV. The application of LSS sensitivity analysis to a flow with chaotic vortex shedding is presented in Section VI, followed by some concluding remarks in Section VII. Three appendices present detailed mathematical derivations related to LSS.

## II. Baseline Solvers

### II.A. CFD Analysis

The CFD analysis solver used in the current study solves compressible and incompressible, steady and unsteady, inviscid, laminar, and turbulent flow equations discretized on unstructured grids.<sup>26</sup> The unsteady governing equations can be written as

$$\frac{\partial \mathbf{Q}}{\partial t} + \mathbf{R}(\mathbf{Q}, \mathbf{D}) = 0, \quad (6)$$

where  $\mathbf{Q}$ ,  $\mathbf{R}$ , and  $\mathbf{D}$  are vectors of solutions, spatial residuals, and design parameters, respectively. The governing equations are discretized using a node-based finite-volume scheme in which the solutions are stored at the vertices of meshes comprised of any arbitrary combination of tetrahedral, prismatic, pyramidal, and hexahedral elements.

Convective fluxes are discretized in an upwind manner and second-order accuracy is achieved using an unstructured reconstruction procedure.<sup>27</sup> Viscous terms are formed using an approach equivalent to a Galerkin finite-element procedure.<sup>27</sup> For non-simplicial element types, these terms are augmented with edge-based gradients to improve the h-ellipticity of the operator.<sup>28</sup> Several classes of temporal discretizations are available, including conventional backwards difference formulae (BDF), as well as various multistep/multistage schemes.<sup>28</sup>

A broad range of gas dynamics models may be used, including a classical perfect gas assumption and considerably more complex models encompassing thermochemical nonequilibrium effects.<sup>29</sup> Available turbulence closures range from the one-equation model of Spalart and Allmaras<sup>30</sup> to full Reynolds stress models and hybrid RANS-LES approaches.<sup>26</sup> These models use spatial and temporal discretizations consistent with the mean flow.

Dynamic mesh computations are accommodated using either rigid or deforming-mesh paradigms, or combinations thereof. The global computational domain may consist of any number of overset component meshes. Each individual mesh may be assigned a specific mesh motion or deformation schedule; alternatively, such attributes may be driven by external models representing complementary disciplines such as aeroelasticity, six degree of freedom motion, or ablation.<sup>28</sup> To leverage massively parallel hardware architectures, domain decomposition approaches are combined with message passing techniques.<sup>26</sup>

### II.B. Sensitivity Analysis

The baseline solver offers extensive sensitivity analysis capabilities. Sensitivity analysis computes derivatives of an objective function

$$f = f(\mathbf{Q}, \mathbf{D}) \quad (7)$$

with respect to design parameters,  $\mathbf{D}$ . For reverse (or adjoint) sensitivity analysis, a discretely-consistent adjoint implementation is available.<sup>4</sup> For forward (or tangent) sensitivity analysis, the CFD solver may be converted to a complex-variable formulation using an established scripting procedure.<sup>31,32</sup> In this approach, the sensitivity of the objective function to a selected input parameter,  $\mathbf{D}_j$ , is readily computed by perturbing the imaginary part of the parameter and executing the complex-valued CFD analysis. The objective function is evaluated for the complex-valued solution, and the sensitivity is obtained as follows:

$$\frac{df}{d\mathbf{D}_j} \approx \frac{\text{Im}[f(\mathbf{Q}, \mathbf{D}_j + i\varepsilon)]}{\varepsilon}. \quad (8)$$

The advantage to this approach is that the perturbation size may be chosen arbitrarily small without concern for round-off errors that are typically present in real-valued divided differences. For the current study, a perturbation of  $\varepsilon = 10^{-50}$  is used, so that the resulting sensitivities are accurate to machine precision. Complex-variable simulations are critical for verification of real-valued sensitivity analysis implementations.

Sensitivity analysis methods can be applied to steady and unsteady flows. For steady flows, Eq. (6) may be written as

$$\mathbf{R}(\mathbf{Q}, \mathbf{D}) = 0. \quad (9)$$

To compute sensitivities, Eq. (7) is differentiated with respect to  $\mathbf{D}_j$ ,

$$\frac{df}{d\mathbf{D}_j} = \frac{\partial f}{\partial \mathbf{Q}} \mathbf{V}_j + \frac{\partial f}{\partial \mathbf{D}_j}, \quad (10)$$

where  $\mathbf{V}_j \equiv \frac{d\mathbf{Q}}{d\mathbf{D}_j}$  is the solution sensitivity vector, which satisfies the tangent equations

$$\frac{\partial \mathbf{R}}{\partial \mathbf{Q}} \mathbf{V}_j + \frac{\partial \mathbf{R}}{\partial \mathbf{D}_j} = 0. \quad (11)$$

Forward sensitivity analysis uses Eqs. (10) and (11) to compute the sensitivity of the objective function.

To derive the discretely-consistent adjoint formulation, one can consider the Lagrangian functional

$$L = f + \mathbf{\Lambda}^T \mathbf{R}, \quad (12)$$

where  $\mathbf{\Lambda}$  is a vector of Lagrange multipliers (adjoint solutions) and the superscript  $T$  denotes the transposition operation. Differentiating Eq. (12) with respect to  $\mathbf{D}$  leads to

$$\frac{df}{d\mathbf{D}} = \frac{dL}{d\mathbf{D}} = \frac{\partial f}{\partial \mathbf{D}} + \mathbf{\Lambda}^T \frac{\partial \mathbf{R}}{\partial \mathbf{D}} + \left( \frac{\partial f}{\partial \mathbf{Q}} + \mathbf{\Lambda}^T \frac{\partial \mathbf{R}}{\partial \mathbf{Q}} \right) \frac{d\mathbf{Q}}{d\mathbf{D}}. \quad (13)$$

Transposing the coefficient of  $\frac{d\mathbf{Q}}{d\mathbf{D}}$  and equating it to zero results in the adjoint equations

$$\left[ \frac{\partial \mathbf{R}}{\partial \mathbf{Q}} \right]^T \mathbf{\Lambda} + \left[ \frac{\partial f}{\partial \mathbf{Q}} \right]^T = 0. \quad (14)$$

If  $\mathbf{\Lambda}$  is the solution of Eq. (14), then the sensitivity to all design parameters  $\mathbf{D}$  is computed as

$$\frac{df}{d\mathbf{D}} = \frac{\partial f}{\partial \mathbf{D}} + \mathbf{\Lambda}^T \frac{\partial \mathbf{R}}{\partial \mathbf{D}}. \quad (15)$$

Conventional sensitivity analysis for time dependent governing equations on stationary grids,

$$\mathcal{V} \frac{\mathbf{Q}^i - \mathbf{Q}^{i-1}}{\Delta t} + \mathbf{R}^i(\mathbf{Q}^i, \mathbf{D}) = 0, \quad \mathbf{Q}^0 - \mathbf{Q}^{IF} = 0, \quad (16)$$

can be derived in a similar way. Here,  $\mathbf{Q}^i$  and  $\mathbf{R}^i$  are vectors of solutions and discretized spatial residuals at time step  $i$ , respectively,  $\mathbf{Q}^{IF}$  is an initial flow field,  $\mathcal{V}$  is a diagonal matrix with control volumes for each unknown along the main diagonal, and  $\Delta t$  is a time step. The first-order BDF1 temporal discretization is shown for simplicity. The objective function  $f$  depends on solutions  $\mathbf{Q}^i$  defined at time steps  $0 \leq i \leq m$ . The sensitivity of the objective function is computed as follows

$$\frac{df}{d\mathbf{D}_j} = \sum_{i=0}^m \frac{\partial f}{\partial \mathbf{Q}^i} \mathbf{V}_j^i + \frac{\partial f}{\partial \mathbf{D}_j}, \quad (17)$$

where  $\mathbf{V}_j^i \equiv \frac{d\mathbf{Q}^i}{d\mathbf{D}_j}$  is the sensitivity of the solution at time step  $i$  to the design parameter  $\mathbf{D}_j$ . The sensitivity  $\mathbf{V}_j^i$  satisfies the tangent equations that correspond to the forward time marching

$$\begin{aligned} \mathbf{V}_j^0 &= 0 \\ \mathcal{V} \frac{\mathbf{V}_j^i - \mathbf{V}_j^{i-1}}{\Delta t} + \frac{\partial \mathbf{R}^i}{\partial \mathbf{Q}^i} \mathbf{V}_j^i + \frac{\partial \mathbf{R}^i}{\partial \mathbf{D}_j} &= 0, \quad i > 0. \end{aligned} \quad (18)$$

The discretely-consistent adjoint formulation is derived from the Lagrangian

$$L = f + \sum_{i=1}^m \left( [\boldsymbol{\Lambda}^i]^T \left( \mathcal{V} \frac{\mathbf{Q}^i - \mathbf{Q}^{i-1}}{\Delta t} + \mathbf{R}^i(\mathbf{Q}^i, \mathbf{D}) \right) \right) + [\boldsymbol{\Lambda}^0]^T (\mathbf{Q}^0 - \mathbf{Q}^{IF}). \quad (19)$$

The adjoint solutions,  $\boldsymbol{\Lambda}^i$ , satisfy the adjoint equations corresponding to reverse time marching,

$$\begin{aligned} \boldsymbol{\Lambda}^{m+1} &= 0, \\ \mathcal{V} \frac{\boldsymbol{\Lambda}^i - \boldsymbol{\Lambda}^{i+1}}{\Delta t} + \left[ \frac{\partial \mathbf{R}^i}{\partial \mathbf{Q}^i} \right]^T \boldsymbol{\Lambda}^i + \left[ \frac{\partial f}{\partial \mathbf{Q}^i} \right]^T &= 0, \quad 1 \leq i \leq m, \\ \mathcal{V} \frac{-\boldsymbol{\Lambda}^1}{\Delta t} + \boldsymbol{\Lambda}^0 + \left[ \frac{\partial f}{\partial \mathbf{Q}^0} \right]^T &= 0. \end{aligned} \quad (20)$$

The sensitivity of the objective function to all design parameters  $\mathbf{D}$  is computed as

$$\frac{df}{d\mathbf{D}} = \frac{\partial f}{\partial \mathbf{D}} + \sum_{i=1}^m \left( [\boldsymbol{\Lambda}^i]^T \frac{\partial \mathbf{R}^i}{\partial \mathbf{D}} \right) - [\boldsymbol{\Lambda}^0]^T \frac{\partial \mathbf{Q}^{IF}}{\partial \mathbf{D}}. \quad (21)$$

The cost of conventional adjoint sensitivity analysis, Eqs. (20) and (21), is similar to the cost of CFD analysis, Eq. (16).

In the current adjoint implementation, all components of the baseline solver are manually differentiated and verified independently using the complex-variable approach outlined above. The implementation is valid for the general unsteady form of the governing equations and includes additional grid adjoint equations for the dynamic, deforming, and overset grid options.

For chaotic flows, one observes an unbounded exponential growth of the tangent solution, Eq. (18), in forward time and the adjoint solution, Eq. (20), in reverse time. This growth is another indication of the breakdown of conventional sensitivity analysis methods.

### III. Least Squares Shadowing

#### III.A. Shadow Trajectory

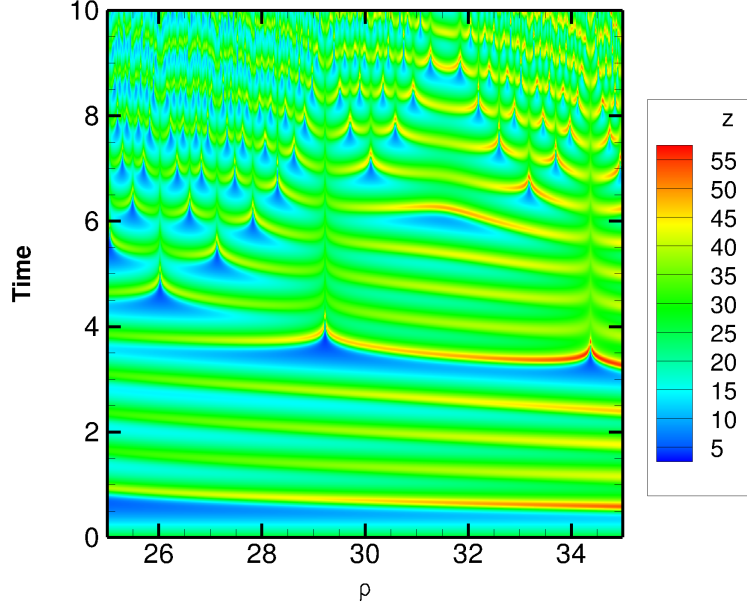
LSS seeks tangent and adjoint solutions that do not exhibit exponential growth. The existence of bounded tangent and adjoint solutions follows from the shadowing lemma.<sup>20</sup> The shadowing lemma states that a reference solution of a chaotic dynamical system has a shadow trajectory, i.e., a solution of a perturbed dynamical system that remains close to the reference solution for all time.

The importance of shadow trajectories can be illustrated using the chaotic Lorenz system, Eq. (1). For this system, the shadowing lemma states that for any  $\delta$  and solution  $\mathbf{U}(t) = (x(t), y(t), z(t))^T$  corresponding to a set of parameters  $(\rho, \sigma, \beta)$ , there exists a small number  $\varepsilon$ , a smooth time transformation  $\tau(t)$ , and a solution  $\mathbf{U}'(\tau(t)) = (x'(\tau(t)), y'(\tau(t)), z'(\tau(t)))^T$  corresponding to a set of parameters  $(\rho + \varepsilon, \sigma, \beta)$  such that

$$\|\mathbf{U}'(\tau(t)) - \mathbf{U}(t)\| < \delta \quad (22)$$

$$\left| 1 - \frac{d\tau}{dt} \right| < \delta \quad (23)$$

Here,  $\|\cdot\|$  is the  $l_2$  vector norm. Figure 2 shows contours of the solution state  $z(t, \rho)$ . The initial conditions are the same for all values of the parameter  $\rho$ . At early times ( $t < 3$ ),  $z(t, \rho)$  is smooth both in time and



**Figure 2.** Contours of  $z(t, \rho)$ . The initial conditions are fixed as  $x(0) = 1, y(0) = 0, z(0) = 28$ .

with respect to variations in  $\rho$ . However, as time increases,  $z(t, \rho)$  becomes non-smooth, exhibiting large local variations. This is a manifestation of the butterfly effect.

In contrast, Fig. 3 shows the time evolution of the solution state  $z'(\tau(t), \rho)$  for a set of shadow trajectories for the same Lorenz system, Eq. (1). The reference solution corresponds to  $\rho = 28$ . The initial conditions and the time transformation are chosen to satisfy the shadowing lemma. The solution state is smooth with respect to the parameter  $\rho$  at all times and the time-averaged solution state,  $\bar{z}'_{\Delta T}(\rho)$ , shown in Fig. 4 is smooth as well. Transformed time is indicated by the black lines in Fig. 3. The contours of  $\tau(t)$  line up with the peaks and troughs of  $z'(\tau(t), \rho)$ . Thus,  $\tau(t)$  can be considered as a generalized phase shift to align features of  $z$  in some time  $\tau$  to ensure that Eq. (22) is satisfied and the shadow  $\mathbf{U}'$  and the reference  $\mathbf{U}$  are close for all time.

To approximate the shadow trajectory, the assumption of ergodicity is used and the corresponding initial conditions are sought through a least-squares minimization problem

$$\min_{\mathbf{U}(t), \tau(t)} \frac{1}{2} \int_{T_0}^{T_0 + \Delta T} \|\mathbf{U}'(\tau(t)) - \mathbf{U}(t)\|^2 + \alpha^2 \left(1 - \frac{d\tau}{dt}\right)^2 dt \quad \text{s.t.} \quad \frac{d\mathbf{U}'}{d\tau} + \mathbf{L}(\mathbf{U}') = 0 \quad (24)$$

where  $\frac{d\mathbf{U}'}{d\tau} + \mathbf{L}(\mathbf{U}')$  is a vector representation of Eq. (1) and  $\alpha$  is a weighting parameter. It can be shown that  $\mathbf{U}'(t)$  converges to a shadow trajectory as  $\Delta T \rightarrow \infty$ .<sup>14,33</sup>

Forward and adjoint LSS sensitivity analyses for discrete flow equations are derived from a least-squares minimization problem similar to Eq. (24). The derivation of this minimization problem is shown in Appendix A.

### III.B. Tangent LSS

To derive the tangent form of LSS for the discrete governing equations, Eq. (16), a least-squares minimization formulation is used

$$\min_{\mathbf{V}_j^i, \eta_j^i} \frac{1}{2} \sum_{i=0}^m [\mathbf{V}_j^i]^T \mathbf{V}_j^i + \alpha^2 [\eta_j^i]^2 \quad \text{s.t.} \quad \mathbf{V}_j^i \frac{\mathbf{V}_j^i - \mathbf{V}_j^{i-1}}{\Delta t} + \frac{\partial \mathbf{R}^i}{\partial \mathbf{Q}^i} \mathbf{V}_j^i + \frac{\partial \mathbf{R}^i}{\partial \mathbf{D}_j} + \eta_j^i \mathbf{R}^i = 0, \quad (25)$$

where  $\eta_j^i$  is called the time dilation term. Eq. (25) is a linearly-constrained minimization problem with the following Karush-Kuhn-Tucker (KKT) equations:

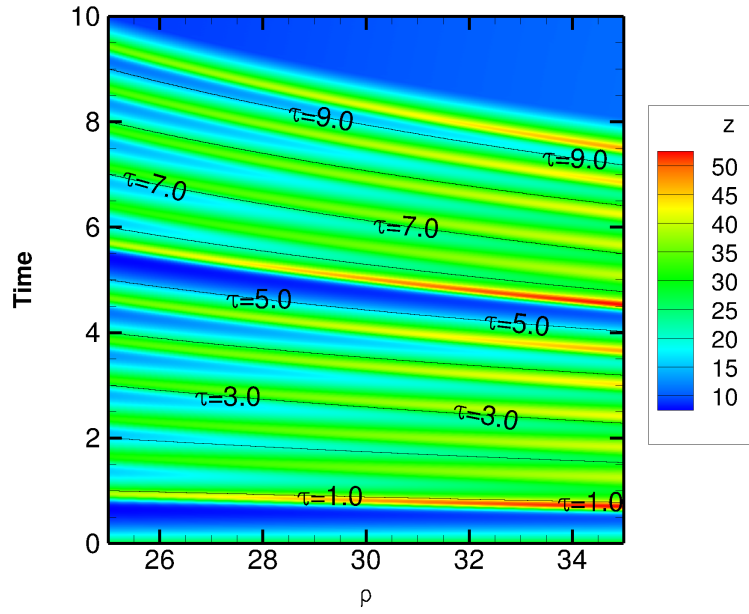


Figure 3. Contours of  $z'(\tau(t), \rho)$  for a set of shadow trajectories at different values of  $\rho$ . The time horizon for the shadow trajectories is between  $T_0 = 0$  and  $T_0 + \Delta T = 10$ . Contours of the time transformation  $\tau(t)$  are indicated by the black contour lines.

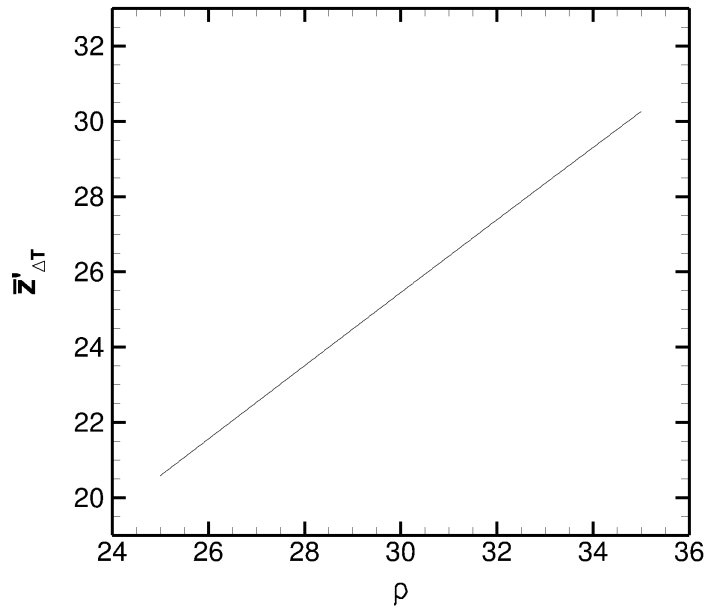


Figure 4. Variation of the time-averaged solution,  $z'_{\Delta T}(\rho)$ , for the set of shadowing trajectories shown in Fig. 3. The averaging window is from  $T_0 = 0$  to  $T_0 + \Delta T = 10$ .

$$\alpha^2 \eta_j^i + [\mathbf{R}^i]^T \mathbf{W}_j^i = 0, \quad (27)$$

$$\mathcal{V} \frac{\mathbf{V}_j^i - \mathbf{V}_j^{i-1}}{\Delta t} + \frac{\partial \mathbf{R}^i}{\partial \mathbf{Q}^i} \mathbf{V}_j^i + \frac{\partial \mathbf{R}^i}{\partial \mathbf{D}_j} + \eta_j^i \mathbf{R}^i = 0, \quad (28)$$

where  $\mathbf{W}_j^i$  is the vector of Lagrange multipliers corresponding to time level  $i$ . The system of equations Eqs. (26)-(28) forms a boundary-value problem in time. The block-matrix representation of this KKT system is the following.

$$\begin{pmatrix} \begin{array}{c|c|c} \begin{array}{ccc} \nu & & \\ & \nu & \\ & & \ddots \\ & & & \nu & \\ & & & & \nu \end{array} & \begin{array}{c} [\mathbf{F}^0]^T \\ [\mathbf{G}^1]^T \\ \\ \\ [\mathbf{G}_m]^T \end{array} & \begin{array}{ccc} [\mathbf{F}_1]^T & & \\ & [\mathbf{G}_2]^T & \\ & & \ddots \\ & & & [\mathbf{F}_{m-1}]^T \\ & & & & [\mathbf{G}_m]^T \end{array} \\ \hline \begin{array}{ccc} \alpha^2 & & \\ & \alpha^2 & \\ & & \ddots \\ & & & \alpha^2 \end{array} & \begin{array}{c} [\mathbf{R}^1]^T \\ \\ \\ [\mathbf{R}^m]^T \end{array} \\ \hline \begin{array}{cc} \mathbf{F}^0 & \mathbf{G}^1 \\ & \mathbf{F}^1 & \mathbf{G}^2 \\ & & \ddots \\ & & & \mathbf{F}^{m-1} & \mathbf{G}^m \end{array} & \begin{array}{c} \mathbf{R}^1 \\ & \mathbf{R}^2 \\ & & \ddots \\ & & & \mathbf{R}^m \end{array} \end{array} \begin{pmatrix} \mathbf{V}_j^0 \\ \mathbf{V}_j^1 \\ \vdots \\ \mathbf{V}_j^m \end{pmatrix} = - \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \mathbf{b}_j^1 \\ \mathbf{b}_j^2 \\ \vdots \\ \mathbf{b}_j^m \end{pmatrix}, \quad (29)$$

where

$$\mathbf{F}^i = -\frac{\mathcal{V}}{\Delta t}, \quad \mathbf{G}^i = \frac{\mathcal{V}}{\Delta t} + \frac{\partial \mathbf{R}^i}{\partial \mathbf{Q}^i}, \quad \mathbf{b}_j^i = \frac{\partial \mathbf{R}^i}{\partial \mathbf{D}_j}, \quad i = 0, \dots, m. \quad (30)$$

Each block ( $\mathcal{V}$ ,  $\mathbf{G}^i$ , and  $\mathbf{F}^i$ ) is a sparse  $n \times n$  matrix. The blocks highlighted in red, white, and yellow correspond to Eqs. (26)-(28), respectively. Instead of solving the entire KKT system, the Schur complement is solved.

To derive the Schur complement, a block-matrix notation for the KKT system is adopted,

$$\begin{pmatrix} \hat{\mathcal{V}} & 0 & \hat{\mathbf{B}}^T \\ 0 & \alpha^2 \mathbf{I} & \hat{\mathbf{C}}^T \\ \hat{\mathbf{B}} & \hat{\mathbf{C}} & 0 \end{pmatrix} \begin{pmatrix} \hat{\mathbf{V}}_j \\ \hat{\eta}_j \\ \hat{\mathbf{W}}_j \end{pmatrix} = - \begin{pmatrix} 0 \\ 0 \\ \hat{\mathbf{b}}_j \end{pmatrix}. \quad (31)$$

Here, the superscript accent ^ designates vectors and matrices describing all time levels collectively,  $\hat{\mathbf{B}}$  is an  $mn \times (m+1)n$  matrix,  $\hat{\mathbf{C}}$  is an  $mn \times m$  matrix,  $\hat{\mathbf{V}}$  is an  $(m+1)n \times (m+1)n$  diagonal matrix,  $\hat{\mathbf{I}}$  is the  $m \times m$  identity matrix,  $\hat{\mathbf{V}}_j$  is an  $(m+1)n \times 1$  vector,  $\hat{\eta}_j$  is an  $m \times 1$  vector,  $\hat{\mathbf{W}}_j$  is an  $mn \times 1$  vector, and  $\hat{\mathbf{b}}_j$  is an  $mn \times 1$  vector.

Conducting block Gaussian elimination, the Schur complement is found to be:

$$\hat{\mathbf{A}}\hat{\mathbf{W}}_j = \hat{\mathbf{b}}_j, \quad \mathbf{W}_j^0 = \mathbf{W}_j^{m+1} = 0, \quad \hat{\mathbf{A}} = \hat{\mathbf{B}}\hat{\mathcal{V}}^{-1}\hat{\mathbf{B}}^T + \frac{1}{\alpha^2}\hat{\mathbf{C}}\hat{\mathbf{C}}^T. \quad (32)$$

The exact structure of matrix  $\hat{\mathbf{A}}$  can be seen by reverting to the notation of Eq. (29)

$$\hat{\mathbf{A}} = \begin{pmatrix} \mathbf{F}^0 \mathcal{V}^{-1} [\mathbf{F}^0]^T + \mathbf{G}^1 \mathcal{V}^{-1} [\mathbf{G}^1]^T & \mathbf{G}^1 \mathcal{V}^{-1} [\mathbf{F}^1]^T & & \\ \mathbf{F}^1 \mathcal{V}^{-1} [\mathbf{G}^1]^T & \mathbf{F}^1 \mathcal{V}^{-1} [\mathbf{F}^1]^T + \mathbf{G}^2 \mathcal{V}^{-1} [\mathbf{G}^2]^T & \mathbf{G}^2 \mathcal{V}^{-1} [\mathbf{F}^2]^T & \\ & \ddots & \ddots & \\ & \mathbf{F}^{m-1} \mathcal{V}^{-1} [\mathbf{G}^{m-1}]^T & \mathbf{F}^{m-1} \mathcal{V}^{-1} [\mathbf{F}^{m-1}]^T + \mathbf{G}^m \mathcal{V}^{-1} [\mathbf{G}^m]^T & \end{pmatrix} + \frac{1}{\alpha^2} \begin{pmatrix} \mathbf{R}^1 [\mathbf{R}^1]^T & & & \\ & \mathbf{R}^2 [\mathbf{R}^2]^T & & \\ & & \ddots & \\ & & & \mathbf{R}^m [\mathbf{R}^m]^T \end{pmatrix}. \quad (33)$$

The Schur complement matrix  $\hat{\mathbf{A}}$  is an  $mn \times mn$  symmetric, positive-definite and block tridiagonal matrix. The objective function is often defined in the form

$$f(\mathbf{D}) = \omega \left( \frac{1}{m_1 - m_0 + 1} \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \right)^p \Delta t, \quad (34)$$

where  $\omega$  is a weighting parameter,  $\Delta t$  is a time step, and  $[m_0, m_1]$ ,  $0 \leq m_0 < m_1 \leq m$ , are the time steps bounding the averaging window. Note that  $f$  and  $F$  are both scalars. One example of  $F$  is the drag coefficient:

$$F(\mathbf{Q}^i) = C_D(\mathbf{Q}^i). \quad (35)$$

The LSS sensitivity of the objective function with respect to design parameter  $\mathbf{D}_j$  can be obtained from the following equation (see Appendix B):

$$\frac{df}{d\mathbf{D}_j} = \frac{\partial f}{\partial \mathbf{D}_j} + \sum_{i=m_0}^{m_1} ([\mathbf{g}^i]^T \mathbf{V}_j^i + h^i \eta_j^i), \quad (36)$$

where

$$[\mathbf{g}^i]^T = \frac{\partial f}{\partial \mathbf{Q}^i} = \omega \Delta t \frac{p \bar{F}^{p-1}}{m_1 - m_0 + 1} \frac{\partial F^i}{\partial \mathbf{Q}^i}, \quad (37)$$

$$h^i = \omega \Delta t \frac{p \bar{F}^{p-1}}{m_1 - m_0 + 1} (F(\mathbf{Q}^i) - \bar{F}), \quad (38)$$

$$\bar{F} = \frac{1}{m_1 - m_0 + 1} \sum_{j=m_0}^{m_1} F(\mathbf{Q}^j), \quad (39)$$

$$\frac{\partial f}{\partial \mathbf{D}_j} = \omega \Delta t \frac{p \bar{F}^{p-1}}{m_1 - m_0 + 1} \sum_{i=m_0}^{m_1} \frac{\partial F^i}{\partial \mathbf{D}_j}. \quad (40)$$

The only difference between Eq. (36) and the conventional sensitivity analysis, Eq. (17), is the term containing time dilation  $\eta_j^i$ .

### III.C. Adjoint LSS

To compute many sensitivities for a single objective function in an efficient manner, a discrete formulation adjoint to tangent LSS is derived. First, Eq. (36) is rewritten using the block-matrix notation of equation Eq. (31)

$$\begin{aligned} \frac{df}{d\mathbf{D}_j} = & \begin{pmatrix} \hat{\mathbf{g}}^T & \hat{h}^T & 0 \end{pmatrix} \begin{pmatrix} \hat{\mathbf{V}}_j \\ \hat{\eta}_j \\ \hat{\mathbf{W}}_j \end{pmatrix} + \frac{\partial f}{\partial \mathbf{D}} \\ & + \begin{pmatrix} \hat{\Phi}^T & \hat{\psi}^T & \hat{\Lambda}^T \end{pmatrix} \left( \begin{pmatrix} \hat{\mathcal{V}} & 0 & \hat{\mathbf{B}}^T \\ 0 & \alpha^2 \hat{\mathbf{I}} & \hat{\mathbf{C}}^T \\ \hat{\mathbf{B}} & \hat{\mathbf{C}} & 0 \end{pmatrix} \begin{pmatrix} \hat{\mathbf{V}}_j \\ \hat{\eta}_j \\ \hat{\mathbf{W}}_j \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \hat{\mathbf{b}}_j \end{pmatrix} \right), \end{aligned} \quad (41)$$

where  $(\hat{\Phi}, \hat{\psi}, \hat{\Lambda})$  are adjoint LSS solutions,  $\hat{\mathbf{g}}$  is an  $mn \times 1$  vector of  $\mathbf{g}^i$  at all times, and  $\hat{h}$  is an  $m \times 1$  vector of  $h^i$  at all times. Eq. (41) is rearranged to group terms dependent on the tangent LSS solution  $(\hat{\mathbf{V}}_j^T \ \hat{\eta}_j^T \ \hat{\mathbf{W}}_j^T)$ ,

$$\begin{aligned} \frac{df}{d\mathbf{D}_j} = & \begin{pmatrix} 0 & 0 & \hat{\mathbf{b}}_j^T \end{pmatrix} \begin{pmatrix} \hat{\Phi} \\ \hat{\psi} \\ \hat{\Lambda} \end{pmatrix} + \frac{\partial f}{\partial \mathbf{D}_j} \\ & + \begin{pmatrix} \hat{\mathbf{V}}_j^T & \hat{\eta}_j^T & \hat{\mathbf{W}}_j^T \end{pmatrix} \left( \begin{pmatrix} \hat{\mathcal{V}} & 0 & \hat{\mathbf{B}}^T \\ 0 & \alpha^2 \hat{\mathbf{I}} & \hat{\mathbf{C}}^T \\ \hat{\mathbf{B}} & \hat{\mathbf{C}} & 0 \end{pmatrix} \begin{pmatrix} \hat{\Phi} \\ \hat{\psi} \\ \hat{\Lambda} \end{pmatrix} + \begin{pmatrix} \hat{\mathbf{g}} \\ \hat{h} \\ 0 \end{pmatrix} \right). \end{aligned} \quad (42)$$

To eliminate any dependence of  $\frac{df}{d\mathbf{D}_j}$  on  $(\hat{\mathbf{V}}_j^T \ \hat{\eta}_j^T \ \hat{\mathbf{W}}_j^T)$ , the following system of adjoint LSS equations must be satisfied

$$\begin{pmatrix} \hat{\mathcal{V}} & 0 & \hat{\mathbf{B}}^T \\ 0 & \alpha^2 \hat{\mathbf{I}} & \hat{\mathbf{C}}^T \\ \hat{\mathbf{B}} & \hat{\mathbf{C}} & 0 \end{pmatrix} \begin{pmatrix} \hat{\Phi} \\ \hat{\psi} \\ \hat{\Lambda} \end{pmatrix} = - \begin{pmatrix} \hat{\mathbf{g}} \\ \hat{h} \\ 0 \end{pmatrix}. \quad (43)$$

The adjoint LSS equations can also be written in the same form as Eqs. (26)-(28)

$$\mathcal{V} \frac{\Lambda^i - \Lambda_j^{i+1}}{\Delta t} + \left[ \frac{\partial \mathbf{R}_i}{\partial \mathbf{Q}_i} \right]^T \Lambda^i + \mathcal{V} \Phi^i + \left[ \frac{\partial f}{\partial \mathbf{Q}^i} \right]^T = 0, \quad \Lambda^0 = \Lambda^{m+1} = 0, \quad (44)$$

$$\alpha^2 \psi^i + [\mathbf{R}^i]^T \Lambda^i + h^i = 0, \quad (45)$$

$$\mathcal{V} \frac{\Phi^i - \Phi^{i-1}}{\Delta t} + \frac{\partial \mathbf{R}^i}{\partial \mathbf{Q}^i} \Phi^i + \psi^i \mathbf{R}^i = 0. \quad (46)$$

Since the KKT system, Eq. (43), is symmetric, the matrix for adjoint LSS is the same as the matrix for tangent LSS, Eq. (31), but the right-hand side is different. The Schur complement of the KKT system Eq. (43) is solved for the adjoint solution  $\hat{\Lambda}$ ,

$$\hat{\mathbf{A}} \hat{\Lambda} = -\hat{\mathbf{B}} \hat{\mathcal{V}}^{-1} \hat{\mathbf{g}} - \frac{1}{\alpha^2} \hat{\mathbf{C}} \hat{h}, \quad \Lambda^0 = \Lambda^{m+1} = 0. \quad (47)$$

Finally, the sensitivities are computed as follows

$$\frac{df}{d\mathbf{D}_j} = \hat{\mathbf{b}}_j^T \hat{\Lambda} + \frac{\partial f}{\partial \mathbf{D}} = \sum_{i=1}^m [\mathbf{b}^i]^T \Lambda^i + \frac{\partial f}{\partial \mathbf{D}_j}. \quad (48)$$

All sensitivities of the objective function  $f$  can be computed using the same adjoint solution  $\hat{\Lambda}$ ,

$$\frac{df}{d\mathbf{D}} = \frac{\partial f}{\partial \mathbf{D}} + \sum_{i=1}^m [\Lambda^i]^T \frac{\partial \mathbf{R}^i}{\partial \mathbf{D}}. \quad (49)$$

Note that this expression is similar to Eq. (21) that is used for conventional adjoint-based sensitivity analysis. The only difference is that Eq. (49) has no dependence on the initial adjoint field  $\Lambda^0$ . This is consistent with the assumption of ergodicity.

### III.D. Least Squares Shadowing Implementation

The linear system of equations given by Eq. (43) represents a global space-time system that is considerably larger than a time-marching CFD problem. The overall dimension of the system is the number of time steps in the LSS window multiplied by the number of spatial grid points multiplied by the number of governing equations. To assemble and solve this system, the required data is generated using the CFD and sensitivity analysis solvers described in Section II. The flow field is first determined by integrating the governing equations forward in time. The cell volumes  $\mathcal{V}$  are stored to disk, as well as the nonlinear residual vector  $\mathbf{R}^i$  at each physical time step. The conventional adjoint solver is then used to progress backwards in time to construct the linearization of the residual vector and objective function at each physical time step. Here, the conventional adjoint solver is used solely to generate and store the required linearization matrices to disk; the solution procedure itself is not carried out. Note that the amount of data produced by this process can be tremendously large, even for small simulations typically carried out on a single workstation. Seemingly trivial two-dimensional simulations with  $\mathcal{O}(10^4)$  mesh points and a moderate number of time steps can easily yield data sets consisting of several terabytes.

The entire data set is then read into a separate application, which assembles and solves the adjoint LSS system, Eq. (47). To leverage massively parallel hardware yet avoid cumbersome infrastructure associated with traditional spatial decompositions, the current implementation decomposes the problem in time rather than space, with an entire solution at a single time step assigned to each processor. This approach assumes that the data required to assemble contributions to Eq. (47) from a single time step do not exceed the memory available to each processor. This constraint is acceptable for the small simulations considered here, where the goal is solely to evaluate the ability of the LSS approach to produce meaningful sensitivity information for simple applications. This decomposition lends itself to a straightforward communication pattern. Since a first-order BDF scheme is used, processor  $i$  needs to only communicate with processors  $i - 1$  and  $i + 1$ . More elaborate implementations are relegated to future work.

To solve Eq. (47), the preconditioned Generalized Minimal Residual (GMRES)<sup>34</sup> method provided in the SPARSKIT package<sup>35</sup> is used. The matrix-vector products required by this approach are computed on a term-by-term basis. Contributions due to off-diagonal blocks occurring in Eq. (33) are formed by communicating necessary information between processors responsible for adjacent time planes. Note that the size of these communication messages is different from typical halo message sizes associated with a spatial decomposition. The messages include entire vectors from a given time step with dimension equal to the size of the spatial grid multiplied by the number of governing equations.

The algorithm to compute a matrix-vector product  $\mathbf{y} = \hat{\mathbf{A}}\mathbf{x}$ , where  $\hat{\mathbf{A}}$  is the matrix defined in Eq. (33) and  $\mathbf{x}$  is an arbitrary vector of dimension  $mn \times 1$  is described below:

1.  $\mathbf{y}^i = 0, \varphi = 0$
2. Send  $\mathbf{x}^i$  to processor  $i - 1$ ; receive  $\mathbf{x}^{i+1}$  from processor  $i + 1$ .
3. Loop through the control volumes; compute  $\mathbf{y}_f = \mathbf{F}^i \mathcal{V}^{-1} [\mathbf{G}^i]^T \mathbf{x}^i$  and send it to processor  $i + 1$ .
4. Loop through the control volumes; compute:
  - i  $\varphi = \varphi + \frac{1}{\alpha^2} [\mathbf{R}^i]^T \mathbf{x}^i$
  - ii For each control volume perform the following:
    - a Loop through its neighbors to compute corresponding rows of  $[\mathbf{G}^i]^T \mathbf{x}^i$  and  $[\mathbf{F}^i]^T \mathbf{x}^{i+1}$ .
    - b Divide by the current control volume  $\mathcal{V}^{-1} [\mathbf{G}^i]^T \mathbf{x}^i$  and  $\mathcal{V}^{-1} [\mathbf{F}^i]^T \mathbf{x}^{i+1}$ .
    - c Loop through its neighbors again, distribute the contribution of the current control volume to compute  $\mathbf{y}^i = \mathbf{y}^i + \mathbf{G}^i \mathcal{V}^{-1} ([\mathbf{G}^i]^T \mathbf{x}^i + [\mathbf{F}^i]^T \mathbf{x}^{i+1}) + \mathbf{F}^i \mathcal{V}^{-1} [\mathbf{F}^i]^T \mathbf{x}^i$ .
  - iii  $\mathbf{y}^i = \mathbf{y}^i + \mathbf{y}_f$
5. Loop through the control volumes;  $\mathbf{y}^i = \mathbf{y}^i + \varphi \mathbf{R}^i$

The blocks of  $[\mathbf{G}^i]^T$  are stored in Compressed-Sparse-Row format, with each row corresponding to the linearization of the equations with respect to solutions defined at the current and neighboring grid points.

Right GMRES preconditioning is applied using a local block-ILU(0) technique within a single time step. The coefficient matrix given by the left hand side of Eq. (47) is used for this purpose, with the

exception of the  $\hat{\mathbf{C}}\hat{\mathbf{C}}^T$  term, which is dense and prohibitively expensive to store. In the current approach, the overall storage requirements on each processor are dominated by the linearization matrices  $\partial\mathbf{R}^i/\partial\mathbf{Q}^i$  and the explicitly-stored matrix used for preconditioning.

In general, the solution approach outlined here was found to be inefficient for solving the adjoint LSS system. The number of processors required by the implementation is determined by the number of time steps contained in the LSS solution window, and several hours of wall clock time were generally required to reduce the residual norm by 6 to 7 orders of magnitude. Beyond this point, the convergence rate of the system deteriorated rapidly. Alternative strategies such as temporally-overlapped preconditioning and multigrid in time were investigated but did not yield appreciable performance improvements. As noted above, computationally efficient approaches to solving Eq. (47) for realistic applications will be the focus of future studies.

## IV. LSS Verification

The adjoint LSS implementation is verified using tangent LSS. The procedure has two stages. First, the KKT system without time dilation terms is verified. Next, a similar set of tests is carried out with the time dilation terms included. The tests and the order in which they have been performed for verification of the current LSS implementation are described below.

1. Run the baseline solvers to generate and store linearization matrices  $\partial\mathbf{R}^i/\partial\mathbf{Q}^i$  and  $\partial\mathbf{R}^i/\partial\mathbf{D}_j$  for one design variable  $\mathbf{D}_j$ .
2. Run tangent LSS (Eq. (33)) with  $1/\alpha^2 = 0$ , output  $\mathbf{V}_j^0 = -(\mathbf{I}/\Delta t)\mathbf{W}_j^1$  and  $\mathbf{V}_j^1$  through  $\mathbf{V}_j^m$ .
3. Run complex-variable CFD analysis, with the following imaginary perturbations:
  - Initial flow field:  $\mathbf{Q}^0 + i\epsilon\mathbf{V}_j^0$
  - Design variable:  $\mathbf{D}_j + i\epsilon$
4. Compare the LSS instantaneous tangent solution ( $\mathbf{V}_j^i$ ) and the imaginary part of the complex-valued solution ( $\mathbf{Q}^i$ ); compare the LSS gradient (Eq. (36) with  $\eta_j^i = 0$ ) with the imaginary part of the objective function (Eq. (34)) evaluated with the complex-valued solution. The verification test passes if the corresponding quantities match to machine precision.
5. If any comparison in the previous step fails to match to machine precision, compute vector  $\mathbf{X}$ , where

$$\mathbf{X} = \begin{pmatrix} \mathbf{F}^0 & \mathbf{G}^1 & & & \\ & \mathbf{F}^1 & \mathbf{G}^2 & & \\ & & \ddots & \ddots & \\ & & & \mathbf{F}^{m-1} & \mathbf{G}^m \end{pmatrix} \begin{pmatrix} \mathbf{V}_j^0 \\ \mathbf{V}_j^1 \\ \vdots \\ \mathbf{V}_j^m \end{pmatrix} + \begin{pmatrix} \mathbf{b}_j^1 \\ \mathbf{b}_j^2 \\ \vdots \\ \mathbf{b}_j^m \end{pmatrix}, \quad (50)$$

and  $\mathbf{F}^i$ ,  $\mathbf{G}^i$ , and  $\mathbf{b}_j^i$  are as defined in Eq. (29).

- i If  $\mathbf{X}$  is not zero, then the matrix product  $\hat{\mathbf{B}}\hat{\mathcal{V}}^{-1}\hat{\mathbf{B}}^T$  in Eq. (32) is not correct.
- ii If  $\mathbf{X}$  is zero, but  $\mathbf{V}_j^i$  does not match the imaginary component of  $\mathbf{Q}^i$ , then  $\frac{\partial\mathbf{R}^i}{\partial\mathbf{D}_j}$  is not correct.
- iii If  $\mathbf{X}$  is zero,  $\mathbf{V}_j^i$  matches the imaginary component of  $\mathbf{Q}^i$ , but the LSS gradient does not match the imaginary part of the objective function, then the vector  $\hat{\mathbf{B}}\hat{\mathcal{V}}^{-1}\hat{\mathbf{g}}^T$  in Eq. (47) is not correct.

Next, steps 2-5 are repeated, with some modifications, to verify the implementation with the time dilation terms:

2. Run tangent LSS (Eq. (33)) with  $1/\alpha^2 = 1$ , output  $\mathbf{V}_j^0 = -(\mathbf{I}/\Delta t)\mathbf{W}_j^1$ ,  $\mathbf{V}_j^1$  through  $\mathbf{V}_j^m$ , and  $\eta_j^i = \frac{1}{\alpha^2}[\mathbf{R}^i]^T\mathbf{W}_j^i$ .

3. Run complex-variable CFD analysis, with the following imaginary perturbations:

- Initial flow field:  $\mathbf{Q}^0 + i\epsilon \mathbf{V}_j^0$
- Design variable:  $\mathbf{D}_j + i\epsilon$

The equations for the complex-variable CFD analysis are modified as

$$\nu \frac{\mathbf{Q}^i - \mathbf{Q}^{i-1}}{\Delta t} + \mathbf{R}^i(\mathbf{Q}^i, \mathbf{D}) + i\epsilon \eta_j^i \text{Re}[\mathbf{R}^i] = 0, \quad \mathbf{Q}^0 - \mathbf{Q}^{IF} = 0, \quad (51)$$

4. Compare the LSS instantaneous tangent solution ( $\mathbf{V}_j^i$ ) and the imaginary part of the complex-valued solution ( $\mathbf{Q}^i$ ); compare the LSS gradient (Eq. (36)) with the imaginary part of the objective function (Eq. (34)) evaluated with the complex-valued solution. The verification test passes if the corresponding quantities match to machine precision.
5. If any comparison in the previous step fails to match to machine precision, compute vector  $\mathbf{X}$  from Eq. (50).

- i If  $\mathbf{X}$  is not zero, then the matrix  $\frac{1}{\alpha^2} \hat{\mathbf{C}} \hat{\mathbf{C}}^T$  in Eq. (32) is not correct.
- ii If  $\mathbf{X}$  is zero,  $\mathbf{V}_j^i$  matches the imaginary component of  $\mathbf{Q}^i$ , but the gradient does not match the imaginary part of the objective function, then the vector  $\frac{1}{\alpha^2} \hat{\mathbf{C}} \hat{\mathbf{h}}$  in Eq. (47) is not correct.

The current implementation has passed all verification tests.

## V. Application of LSS to a Steady Flow

Another verification for the LSS implementation can be performed by applying the algorithm to a steady flow. A steady-state solution can be computed directly from the steady-state equations or by time marching a solution of the unsteady equations. In the former approach, a steady-state adjoint formulation is suitable, whereas in the latter case, an unsteady adjoint formulation can be applied. A steady-flow verification case should satisfy two conditions. First, the solutions computed directly from the steady-state equations and by time marching should be the same. Second, the conventional adjoint solution for the time marching approach should decay to zero in reverse time. This latter condition implies diminishing sensitivities of remote instantaneous solutions to the initial conditions. If these two conditions are satisfied, then for a long enough time horizon, the sensitivity of a steady solution computed with the conventional unsteady adjoint approach is expected to match the sensitivity computed with the steady-state adjoint approach.

Steady flows do not require time dilation, so  $1/\alpha^2 = 0$ , and the adjoint LSS equations are reduced to

$$\mathbf{\Lambda}^{m+1} = 0, \quad \mathbf{\Lambda}^0 = 0, \quad (52)$$

$$\nu \frac{\mathbf{\Lambda}^i - \mathbf{\Lambda}^{i+1}}{\Delta t} + \left[ \frac{\partial \mathbf{R}^i}{\partial \mathbf{Q}^i} \right]^T \mathbf{\Lambda}^i + \nu \mathbf{\Phi}^i + \left[ \frac{\partial f}{\partial \mathbf{Q}^i} \right]^T = 0, \quad 1 \leq i \leq m, \quad (53)$$

$$\nu \frac{\mathbf{\Phi}^i - \mathbf{\Phi}^{i-1}}{\Delta t} + \frac{\partial \mathbf{R}^i}{\partial \mathbf{Q}^i} \mathbf{\Phi}^i, \quad 1 \leq i \leq m. \quad (54)$$

With a sufficiently long LSS window, Eqs. (52)-(54) have the solution  $[\mathbf{\Lambda}^i, \mathbf{\Phi}^i]$  satisfying the conventional unsteady adjoint equations, Eq. (20),

$$\nu \frac{\mathbf{\Lambda}^i - \mathbf{\Lambda}^{i+1}}{\Delta t} + \left[ \frac{\partial \mathbf{R}^i}{\partial \mathbf{Q}^i} \right]^T \mathbf{\Lambda}^i + \left[ \frac{\partial f}{\partial \mathbf{Q}^i} \right]^T = 0 \quad \text{and} \quad \mathbf{\Phi}^i = 0. \quad (55)$$

Thus, for a steady-flow verification test, the sensitivities computed with the steady, conventional unsteady, and LSS adjoint approaches are expected to match to machine precision.

An inviscid flow around a two-dimensional NACA 0012 airfoil at one degree angle of attack is considered as a steady-flow verification test. The freestream Mach number is  $M_\infty = 0.25$ . The grid, shown in Fig. 5,

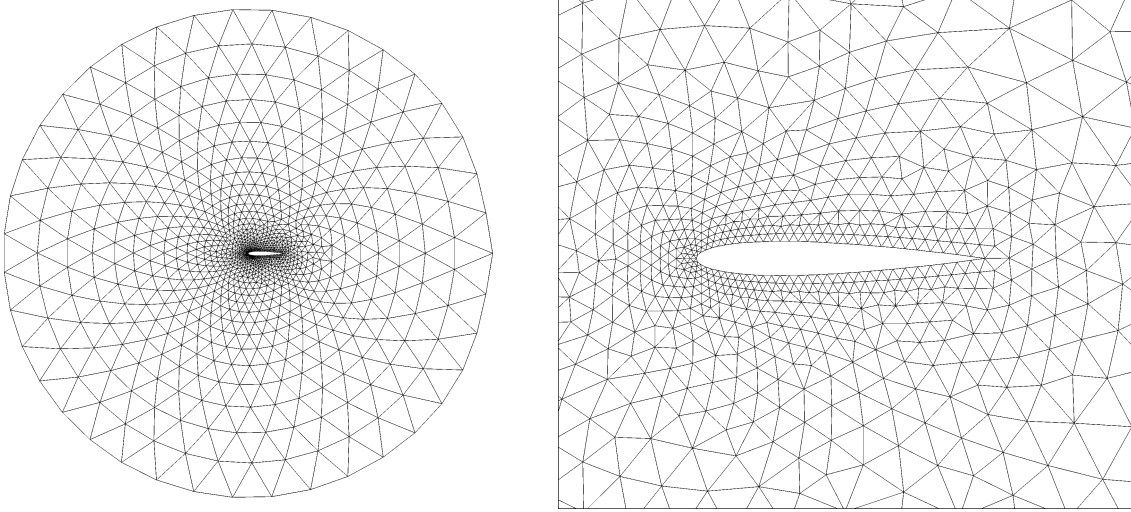


Figure 5. Farfield and nearfield views of the NACA 0012 mesh.

consists of a single layer of 2,117 prismatic elements in the spanwise direction and 2,218 nodes. The farfield boundary is located seven chord lengths from the leading edge.

The objective function is defined as

$$f = \frac{1}{m_1 - m_0 + 1} \sum_{i=m_0}^{m_1} F^i, \quad (56)$$

where

$$F^i = \frac{D^i}{\frac{1}{2} \rho_\infty U_\infty^2 c}. \quad (57)$$

Here,  $D^i$  is the drag on the airfoil at time step  $i$ ,  $\rho_\infty$  is the freestream density,  $U_\infty$  is the freestream velocity, and  $c$  is the chord length. The sensitivity of the objective function with respect to freestream Mach number is evaluated. For clarity, the explicit dependence on the freestream Mach number is not considered. The adjoint-based sensitivity is computed as  $\sum_{i=1}^m [\mathbf{\Lambda}^i]^T \frac{\partial \mathbf{R}^i}{\partial \mathbf{D}}$ ; the complex-variable sensitivity evaluation neglects the imaginary part of the freestream velocity in the denominator of Eq. (57). A time horizon of 5,100 time steps is considered, with the averaging window defined over the last 100 steps.

Table 1 shows sensitivities computed with complex-variable steady and unsteady simulations as well as with steady, conventional unsteady, and LSS adjoint methods. The complex-variable sensitivities and the steady and conventional unsteady adjoint sensitivities match to 14 decimal places. The LSS adjoint sensitivity differs from the sensitivities computed using the other methods by  $2.14 \times 10^{-12}$ . This difference is slightly greater than machine precision. This mismatch is due to some small contributions from the LSS adjoint solution at times beyond time step 5,100. The conventional unsteady adjoint solution is zero beyond time step 5,100. To illustrate the magnitude of these differences, Fig. 6 shows the time evolution of the  $l_2$  norms of the conventional unsteady and LSS adjoint solutions. The magnitude of the conventional unsteady adjoint solution has decayed to  $10^{-22}$  in reverse time by the initial time level. It reduces below  $10^{-15}$  by time step 1,800. This observation motivated the current choice of the LSS window that starts at time step 1,801 and includes 6,500 time steps, namely, the averaging window defined from step 5,001 to step 5,100 and 3,200 steps before and after the averaging window.

The norm of the LSS adjoint solution matches that of the conventional unsteady adjoint solution for time steps from 1,801 to 5,100. The norm of the LSS adjoint solution exhibits a sharp drop beyond the averaging window, but does not vanish entirely. The norm varies between  $10^{-13}$  and  $10^{-17}$  for the time steps from 5,100 to 8,300. This latter portion of the LSS window produces small contributions to the sensitivity that explain the observed mismatch. Note that the LSS adjoint solution is approaching the conventional adjoint solution in the limit of infinitely large LSS window. It is expected that the mismatch can be further reduced

Table 1. Sensitivity  $df/dM_\infty$  computed with various methods.

Method	Sensitivity $df/dM_\infty$
Steady Complex	0.004026323380482
Unsteady Complex	0.004026323380483
Steady Adjoint	0.004026323380482
Unsteady Adjoint	0.004026323380483
LSS Adjoint	0.004026323378342

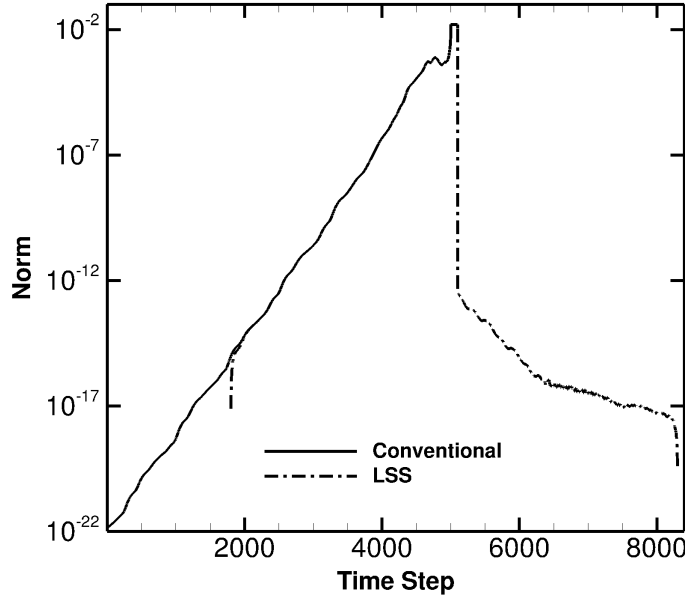


Figure 6. Time variation of  $l_2$  norm of the conventional unsteady and LSS adjoint solutions. The averaging window is from time step 5,001 to 5,100. The LSS window is from time step 1,801 to 8,400.

if the size of the LSS window is increased or the contributions from the time steps beyond the averaging window are ignored.

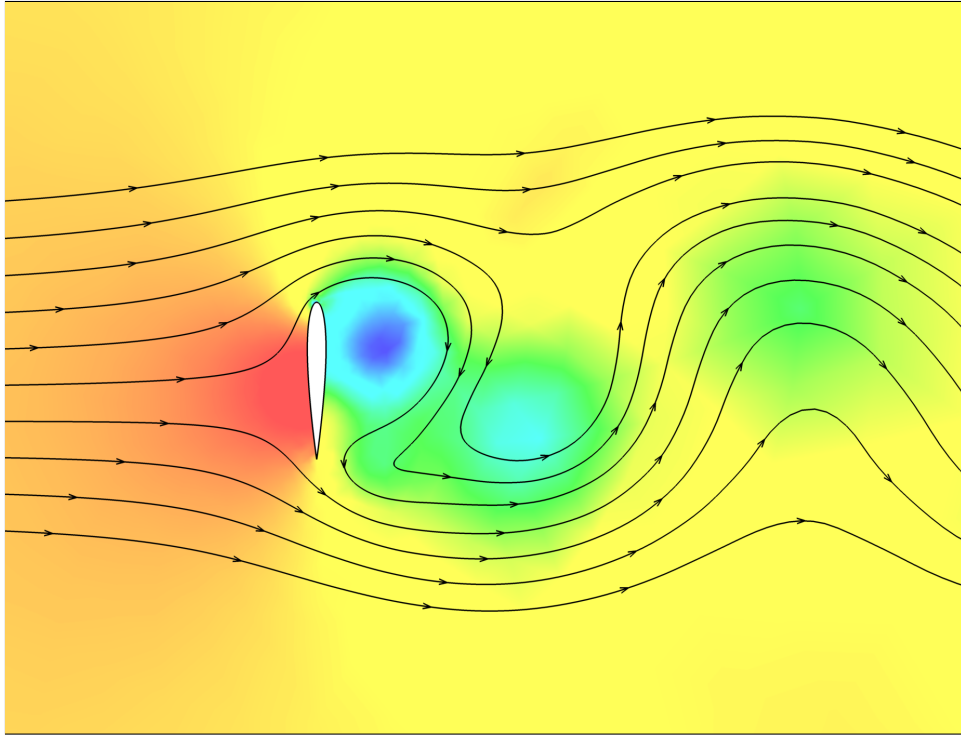
The LSS sensitivity analysis was considerably more expensive than the baseline CFD analysis. The solution of the LSS equations was obtained on 6,500 Intel Xeon E5-2680v3 cores and required 81 GB of input data and more than 130,000 GMRES iterations to converge the  $l_2$  norm of the residual to machine zero. The corresponding CFD analysis problem was solved to machine precision orders of magnitude faster on a single core.

## VI. LSS for Chaotic Vortex Shedding

### VI.A. Test Case

To evaluate the LSS methodology for a chaotic flow field, inviscid flow over a NACA 0012 airfoil is simulated using the grid shown in Fig. 5, freestream conditions corresponding to  $M_\infty = 0.25$ , an angle of attack of 90 degrees, and a nondimensional time step  $\Delta t = 0.1$ . Chaotic vortex shedding from the upper surface of the airfoil occurs for the stated conditions as shown in Fig. 7. This relatively simple test case exhibits chaotic behavior representative of more complex turbulent flow simulations, yet yields an LSS problem that is computationally tractable based on the current implementation. The current test case evaluates the ability of adjoint LSS to compute sensitivity of a time-averaged chaotic-flow quantity with respect to the freestream Mach number.

The objective function is the same as defined in Eq. (56). For this objective function, the term  $\frac{\partial f}{\partial \mathbf{D}}$



**Figure 7. Instantaneous streamlines and density field for chaotic vortex shedding.**

in Eq. (49) is ignored and the sensitivity  $\frac{df}{d\mathbf{D}}$  is equated to the adjoint contribution  $\sum_{i=1}^m [\mathbf{\Lambda}^i]^T \frac{\partial \mathbf{R}^i}{\partial \mathbf{D}}$ . The chaotic time history of  $F^i$  is observed after 100,000 time steps in Fig. 8. The breakdown of the conventional unsteady adjoint approach for the current problem is illustrated as follows. The averaging window for the objective function, Eq. (56), is defined between time steps  $m_0 = 30,001$  and  $m_1 = 31,000$ . The solution of the conventional unsteady adjoint formulation computed backwards in time from this averaging window grows exponentially as shown in Fig. 9.

## VI.B. Results

Two LSS windows have been examined with an averaging window located in the middle of each as shown in Fig. 10. In each case, the flow has been initialized using freestream conditions and integrated forward in time for 13,000 time steps. The long LSS window has been defined from time step 13,001 to time step 15,000 and the corresponding averaging window is defined between time steps 13,501 and 14,500. The short LSS window has been defined from time step 13,001 to time step 14,100 and the corresponding averaging window is defined between time steps 13,501 and 13,600. A buffer of several time steps before and after the averaging window is required because the accuracy of the shadow trajectory approximation diminishes toward the boundaries of the LSS window.<sup>33</sup> The current empirical choice of a 500-time-step buffer proved sufficient, but a rigorous method for defining a suitable size of the LSS window is needed.

As mentioned in Section V, the computational mesh contains 2,218 grid points and 11,090 spatial degrees of freedom. Even for such a small grid, the computational cost is high. For the long LSS window, the total size of the KKT Schur complement matrix is 22,180,000 rows and roughly 24 GB of input data is required. The system was solved on 2,000 cores. The short LSS window requires 13 GB of input data and was solved using 1,100 cores. Each sensitivity has converged to at least 3 decimal places.

Figure 11 shows convergence histories of the residual  $l_2$  norm and the Mach number sensitivity for both LSS window sizes. Three tests have been conducted for each window with 13,000, 50,000, and 100,000 run-up time steps before starting the LSS window to assess the sensitivity to the proximity of the LSS window to the initial conditions. Intuitively, a sufficient run-up time is needed to establish chaotic dynamics that is essentially independent of initial conditions. As expected, the computations in the short LSS window are more sensitive to the run-up time than the corresponding computations in the long LSS window. The short-

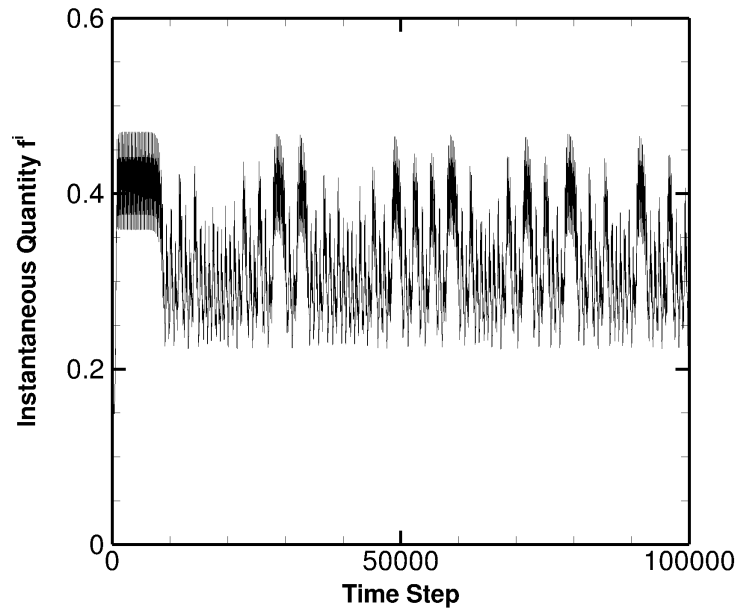


Figure 8. Variation of instantaneous quantity,  $F^i$ , Eq. (57).

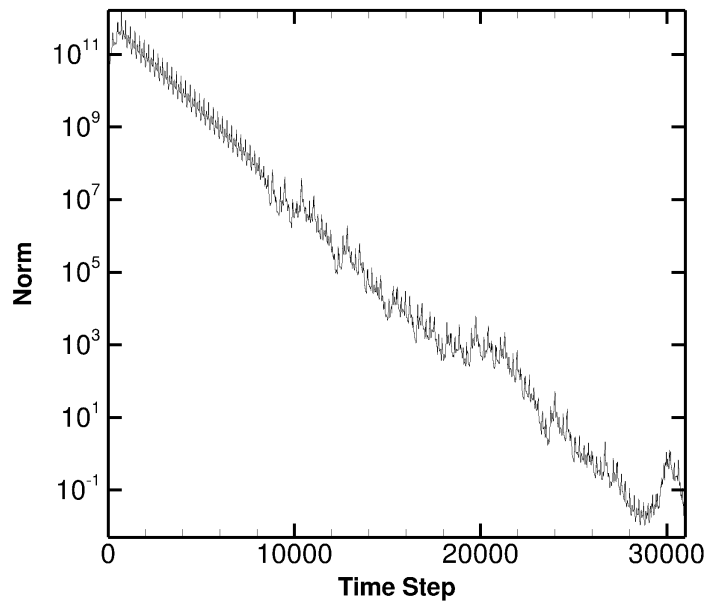


Figure 9. The  $l_2$  norm of the conventional unsteady adjoint for the flow from Fig. 8. The adjoint solution is computed over the first 31,000 steps. The averaging window is from step 30,001 to 31,000.

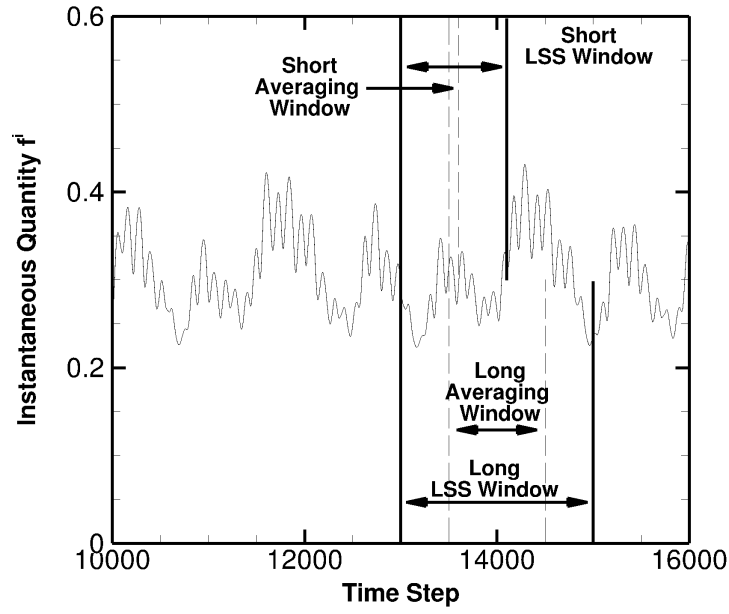


Figure 10. Variation of instantaneous quantity,  $F^i$ , Eq. (57), and two LSS windows.

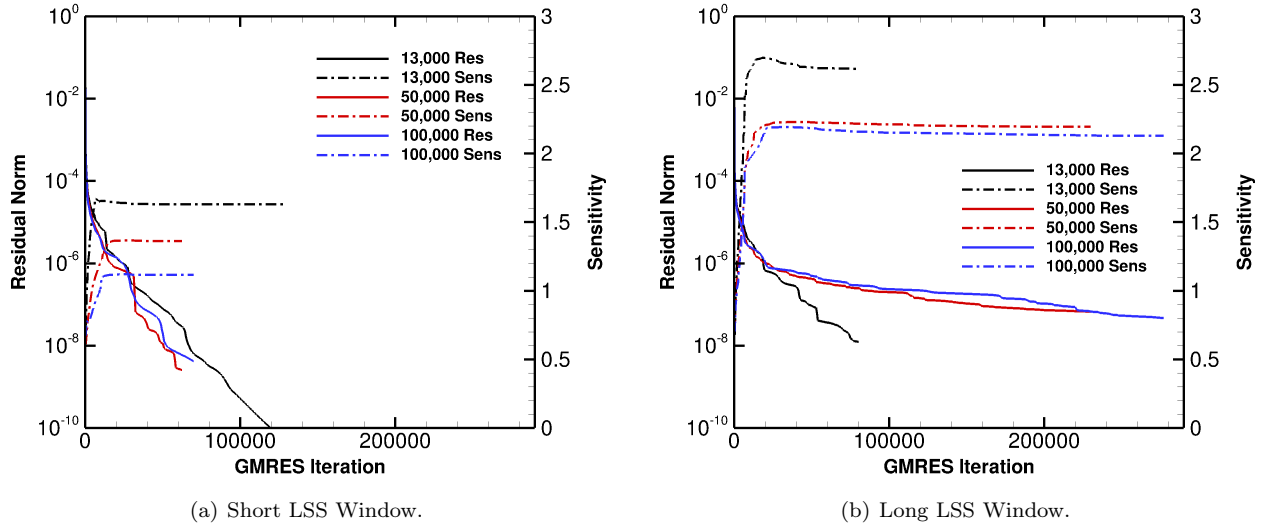


Figure 11. Iterative convergence of the residual  $l_2$  norm and sensitivity for two LSS windows.

Table 2. Sensitivities computed for various LSS Windows. Values within the 99% confidence interval of the linear regression slope are indicated in bold.

Window Start	Sensitivity $df/dM_\infty$	
	Short LSS Window	Long LSS Window
13,000	1.63	<b>2.618</b>
50,000	1.362	<b>2.195</b>
100,000	1.116	<b>2.129</b>

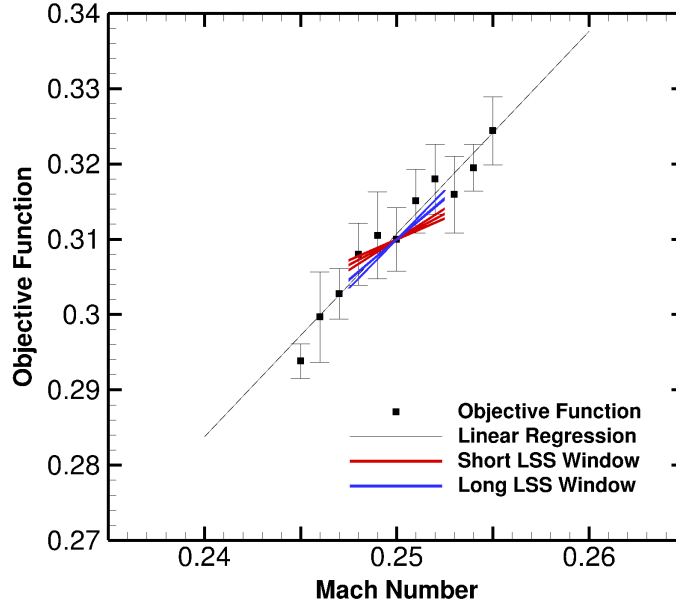


Figure 12. Objective function, Eq. (56), computed for a range of Mach numbers.

window sensitivities are significantly different from each other and from those of the long-window simulations. The long-window sensitivities computed with longer run-up intervals are similar. It appears that the short run-up interval of 13,000 time steps is not sufficient. The corresponding sensitivities are significantly higher than the other sensitivities computed with longer run-up intervals. Interestingly, residuals in the long-window computations with the short run-up time converge better than the residuals in the same window with longer run-up intervals. As expected, the residual convergence in the long-window computations is significantly slower than in the short-window computations.

All sensitivities computed with LSS are presented in Table 2. The sensitivities are compared with a linear regression of long-time averages of the objective function computed at freestream Mach numbers  $0.245 \leq M_\infty \leq 0.255$ . The objective function has been computed by averaging  $F^i$  between time steps 10,001 and 1,000,000. The first 10,000 steps have been excluded to mitigate any possible impact of initial transients on the time average. The computed linear fit is

$$f \approx 2.693M_\infty - 0.363, \quad (58)$$

where the 99% confidence interval of the slope is estimated to be  $[2.006, 3.380]$  using the standard error of the slope and the appropriate t-score of the data.

The long-window sensitivities fall within the 99% confidence interval of the regression slope. This is in contrast with the short LSS window results, which all fell outside the 99% confidence interval. It is likely that the short averaging windows are not able to capture a representative behavior of the flow. Figure 12 shows the LSS sensitivities plotted with the linear regression and the raw objective functions. The long-window LSS sensitivities match the linear regression well. Figure 12 also includes  $3\sigma$  error bars, where  $\sigma$

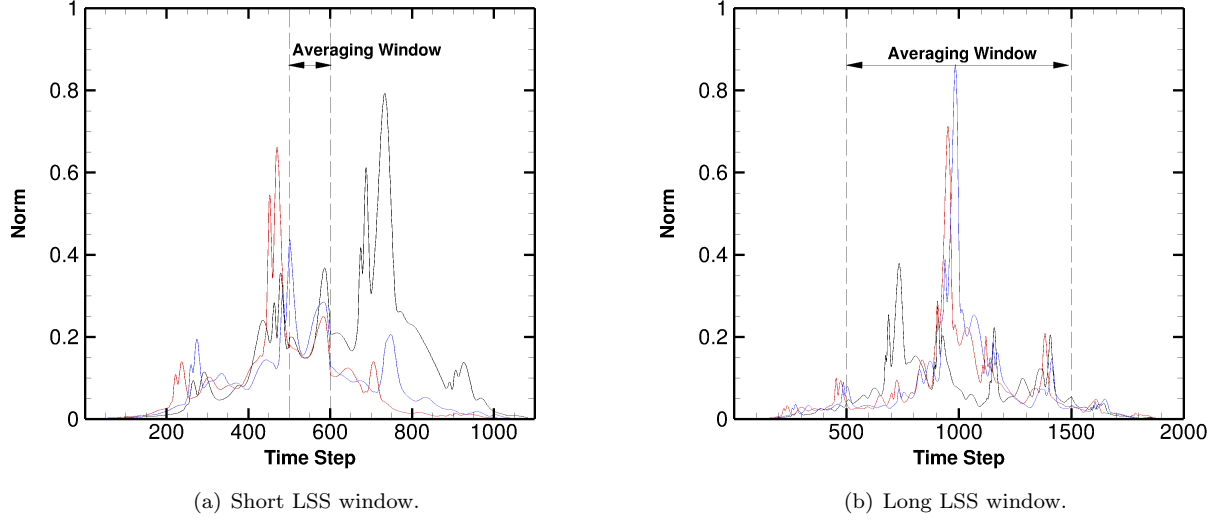


Figure 13. The  $l_2$  norm of the adjoint LSS solution for two LSS windows. The black, red, and blue lines correspond to 13,000, 50,000, and 100,000 run-up time steps, respectively.

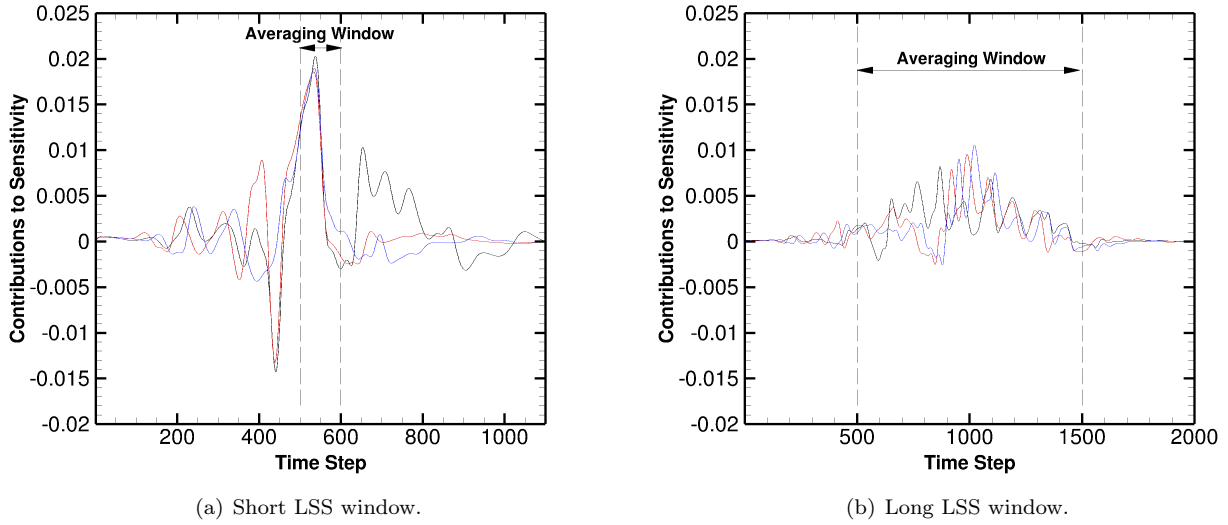


Figure 14. Instantaneous contributions  $[\Lambda^i]^T \frac{\partial \mathbf{R}^i}{\partial \mathbf{D}}$  to sensitivity  $\frac{df}{d\mathbf{D}}$  for two LSS windows. The black, red, and blue lines correspond to 13,000, 50,000, and 100,000 run-up time steps, respectively.

is an estimate of the standard deviation computed with the batch mean method.<sup>36</sup> The error bars were computed over four separate equal length time intervals between time steps 10,000 and 1,000,000. The long-time averaging intervals ensure that the objective functions computed on these intervals are essentially independent. Additional details are provided in Appendix C.

Figure 13 shows the time evolution of the  $l_2$  norm of the LSS adjoint solutions. In contrast to the conventional adjoint solution shown in Fig. 9, the LSS adjoint solution does not grow exponentially backwards in time. The magnitude of the adjoint solution at any time step for all cases studied remains bounded. The adjoint solution decays to zero towards the ends of the LSS window. Figure 14 shows contributions to the sensitivity from each time step. The non-zero values of  $[\Lambda^i]^T \frac{\partial \mathbf{R}^i}{\partial \mathbf{D}}$  observed past the end of the averaging window are much smaller for the long LSS window than for the short LSS window.

## VII. Conclusions and Future Outlook

A new Least Squares Shadowing (LSS) approach for computing sensitivities of long-time averages of chaotic flow solutions has been implemented in a large-scale computational fluid dynamics solver offering mature adjoint capabilities. Implementation of LSS in a practical framework is challenging. Moreover, established verification methods such as complex-variable procedures are not applicable. New, detailed verification procedures for an LSS implementation have been developed and presented. A chaotic flow solution and corresponding LSS sensitivities have been computed and assessed. With sufficient computational windows and time-averaging intervals, the computed sensitivities are accurate and can be used for design. However, the computational cost of the current LSS implementation is high.

Computing sensitivities of chaotic CFD simulations is a difficult problem that continues to lack an adequate solution. High-fidelity aerospace design problems are characterized by many thousands of design parameters and a high computational cost of CFD analysis. Several sensitivity analysis methods that have been developed recently in other communities for chaotic dynamical systems are not suitable for high-resolution LES and DNS computations of turbulent flow. The LSS approach described in this paper represents the most efficient method developed to date; however, its application remains prohibitively expensive for practical aerospace simulations.

The complexity of LSS sensitivity analysis is many orders of magnitude higher than that of conventional CFD analysis. A work unit can be defined as the computational operations required to perform one iteration of a nonlinear CFD analysis at all time steps within an LSS window. A typical simulation requires 10-15 such iterations to sufficiently reduce the residuals at each time step. Thus, the total cost of a typical CFD analysis may require 10-15 work units.

The computational expense of a GMRES iteration for the current LSS implementation is driven largely by the cost of performing a matrix-vector multiplication. However, it is not straightforward to evaluate this expense in terms of work units. The fill of the LSS matrix is many times larger than the fill of the approximate linearization matrix used in the CFD analysis. A low-end estimate is that the cost of one GMRES matrix-vector multiplication is about 2 work units. Based on the cases reported here, more than 250,000 GMRES iterations may be required to sufficiently reduce the LSS system residuals for a small chaotic dynamical CFD-based system. Thus, the cost of an LSS sensitivity analysis is at least 500,000 work units, or approximately 4 to 5 orders of magnitude more than that of the baseline CFD analysis. Revolutionary breakthroughs in solution technologies for LSS sensitivity analysis as well as increased availability of leadership-class computing resources are needed to enable practical high-fidelity design for aerospace applications involving chaotic flows.

## Appendices

### A. Linearizing Least Squares Shadowing

The shadowing lemma can be applied directly to the discretized nonlinear governing equations, Eq. (16). For any  $\delta$  and a reference solution  $\mathbf{Q}^i$  corresponding to a set of design parameters  $\mathbf{D}$ , there exists a solution,  $\mathbf{Q}'^i$ , corresponding to the set of design parameters with a perturbed parameter  $\mathbf{D}_j = \mathbf{D}_j + \varepsilon$ . The solution  $\mathbf{Q}'^i$  satisfies  $\|\mathbf{Q}'^i - \mathbf{Q}^i\| < \delta$  for all time  $t^i$ , where  $\|\cdot\|$  is the  $l_2$  vector norm. Note that  $\mathbf{Q}'^i$  corresponds to a smooth transformation of time,  $\tau_j^i$ ,

$$\tau_j^i = (1 + \varepsilon \eta_j^i) t^i, \quad \eta_j^0 = 0. \quad (59)$$

The flow solution  $\mathbf{Q}'^i$  is the “shadow trajectory”, and, according to the shadowing lemma, its existence is guaranteed for a range of chaotic dynamical systems.<sup>14</sup> The assumption of ergodicity allows the initial condition for  $\mathbf{Q}'^i$  to be replaced with a least-squares minimization problem. For the first order BDF1 temporal discretization, this discrete least-squares minimization problem is

$$\min_{\mathbf{Q}'^i, \eta^i} \frac{1}{2} \sum_{i=0}^m (\mathbf{Q}'^i - \mathbf{Q}^i)^T \mathcal{V}(\mathbf{Q}'^i - \mathbf{Q}^i) + \alpha^2 \left( 1 - \frac{d\tau_j^i}{dt^i} \right)^2 \quad (60)$$

$$\text{s.t.} \quad \mathcal{V} \frac{\mathbf{Q}'^i - \mathbf{Q}'^{i-1}}{\Delta \tau} + \mathbf{R}^i(\mathbf{Q}'^i, \mathbf{D}_j + \varepsilon) = 0, \quad (61)$$

where  $\Delta\tau = \frac{d\tau_j^i}{dt^i} \Delta t$  is the transformed time step.

To linearize Eq. (60), the following identities are used:

$$\lim_{\varepsilon \rightarrow 0} \mathbf{Q}'^i = \mathbf{Q}^i, \quad \lim_{\varepsilon \rightarrow 0} \frac{\mathbf{Q}'^i - \mathbf{Q}^i}{\varepsilon} = \frac{\partial \mathbf{Q}^i}{\partial \mathbf{D}_j} \equiv \mathbf{V}_j^i. \quad (62)$$

Dividing the minimization statement in Eq. (60) by  $\varepsilon^2$  and taking the limit of  $\varepsilon \rightarrow 0$  results in

$$\begin{aligned} & \lim_{\varepsilon \rightarrow 0} \min_{\mathbf{Q}'^i, \eta^i} \frac{1}{2} \sum_{i=0}^m \left( \frac{\mathbf{Q}'^i - \mathbf{Q}^i}{\varepsilon} \right)^T \mathcal{V} \left( \frac{\mathbf{Q}'^i - \mathbf{Q}^i}{\varepsilon} \right) + \alpha^2 [\eta_j^i]^2 \\ &= \min_{\mathbf{V}_j^i, \eta_j^i} \frac{1}{2} \sum_{i=0}^m [\mathbf{V}_j^i]^T \mathcal{V} \mathbf{V}_j^i + \alpha^2 [\eta_j^i]^2. \end{aligned} \quad (63)$$

The governing equation for the shadow trajectory  $\mathbf{Q}'^i$ , Eq. (61), is linearized around the reference solution  $\mathbf{Q}^i$ ,

$$\begin{aligned} & \lim_{\varepsilon \rightarrow 0} \frac{1}{\varepsilon} \left\{ \left( \mathcal{V} \frac{\mathbf{Q}'^i - \mathbf{Q}^{i-1}}{\frac{d\tau_j^i}{dt^i} \Delta t} + \mathbf{R}^i(\mathbf{Q}'^i, \mathbf{D}_j + \varepsilon) \right) - \left( \mathcal{V} \frac{\mathbf{Q}^i - \mathbf{Q}^{i-1}}{\Delta t} + \mathbf{R}^i(\mathbf{Q}^i, \mathbf{D}) \right) \right\} \\ &= \lim_{\varepsilon \rightarrow 0} \left\{ \frac{\mathcal{V}}{\Delta t} \left( \frac{\mathbf{Q}'^i - \mathbf{Q}^{i-1}}{(1 + \varepsilon \eta_j^i) \varepsilon} - \frac{\mathbf{Q}^i - \mathbf{Q}^{i-1}}{\varepsilon} \right) + \left( \frac{\mathbf{R}(\mathbf{Q}'^i, \mathbf{D}_j + \varepsilon) - \mathbf{R}(\mathbf{Q}^i, \mathbf{D})}{\varepsilon} \right) \right\} \\ &= \lim_{\varepsilon \rightarrow 0} \left\{ \frac{\mathcal{V}}{\Delta t (1 + \varepsilon \eta_j^i)} \left( \frac{\mathbf{Q}'^i - \mathbf{Q}^i}{\varepsilon} + \frac{\mathbf{Q}^{i-1} - \mathbf{Q}^{i-1}}{\varepsilon} - \eta_j^i (\mathbf{Q}^i - \mathbf{Q}^{i-1}) \right) \right\} + \frac{\partial \mathbf{R}^i}{\partial \mathbf{Q}^i} \mathbf{V}_j^i + \frac{\partial \mathbf{R}^i}{\partial \mathbf{D}_j} \\ &= \mathcal{V} \frac{\mathbf{V}_j^i - \mathbf{V}_j^{i-1}}{\Delta t} - \eta_j^i \frac{\mathbf{Q}^i - \mathbf{Q}^{i-1}}{\Delta t} + \frac{\partial \mathbf{R}^i}{\partial \mathbf{Q}^i} \mathbf{V}_j^i + \frac{\partial \mathbf{R}^i}{\partial \mathbf{D}_j} \\ &= \mathcal{V} \frac{\mathbf{V}_j^i - \mathbf{V}_j^{i-1}}{\Delta t} + \frac{\partial \mathbf{R}^i}{\partial \mathbf{Q}^i} \mathbf{V}_j^i + \frac{\partial \mathbf{R}^i}{\partial \mathbf{D}_j} + \eta_j^i \mathbf{R}^i. \end{aligned} \quad (64)$$

Combining Eqs. (63) and (64), the following least-squares minimization problem for the tangent solution is obtained

$$\min_{\mathbf{V}_j^i, \eta_j^i} \frac{1}{2} \sum_{i=0}^m [\mathbf{V}_j^i]^T \mathcal{V} \mathbf{V}_j^i + \alpha^2 [\eta_j^i]^2 \quad \text{s.t.} \quad \mathcal{V} \frac{\mathbf{V}_j^i - \mathbf{V}_j^{i-1}}{\Delta t} + \frac{\partial \mathbf{R}^i}{\partial \mathbf{Q}^i} \mathbf{V}_j^i + \frac{\partial \mathbf{R}^i}{\partial \mathbf{D}_j} + \eta_j^i \mathbf{R}^i = 0. \quad (65)$$

## B. Computing Sensitivities using a Shadow Trajectory

An objective function can be written in the following form:

$$f(\mathbf{D}) = \omega \left( \frac{1}{\Delta T} \int_{T_0}^{T_0 + \Delta T} F(\mathbf{Q}) dt \right)^p \Delta t. \quad (66)$$

The corresponding discrete form of this expression is

$$\omega \left( \frac{1}{m_1 - m_0 + 1} \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \right)^p \Delta t. \quad (67)$$

The objective function for a shadow trajectory  $\mathbf{Q}'$  is

$$f(\mathbf{D}) = \omega \left( \frac{1}{\Delta T'} \int_{T_0}^{T_0 + \Delta T'} F(\mathbf{Q}') d\tau \right)^p \Delta t, \quad (68)$$

where  $\Delta T'$  is shadow averaging window,

$$\Delta T' = \tau(T_0 + \Delta T) - T_0 = \int_{T_0}^{T_0 + \Delta T} \frac{d\tau}{dt} dt. \quad (69)$$

The discrete version of Eq. (68) is

$$f(\mathbf{D}) = \omega \left( \frac{1}{\sum_{i=m_0}^{m_1} \frac{d\tau^i}{dt^i}} \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \frac{d\tau^i}{dt^i} \right)^p \Delta t, \quad (70)$$

$$\sum_{i=m_0}^{m_1} \frac{d\tau^i}{dt^i} = (m_1 - m_0 + 1)(1 + \zeta), \quad (71)$$

where  $\zeta = \sum_{i=m_0}^{m_1} \left( \frac{d\tau^i}{dt^i} - 1 \right) = \sum_{i=m_0}^{m_1} \eta_j^i$ .

The perturbation of the objective function,  $\delta f$ , is the following difference

$$\delta f = \omega \left( \frac{1}{(m_1 - m_0 + 1)(1 + \zeta)} \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \frac{d\tau^i}{dt^i} \right)^p \Delta t - \omega \left( \frac{1}{m_1 - m_0 + 1} \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \right)^p \Delta t.$$

Assuming  $\varepsilon \ll 1$  and  $\varepsilon^p \approx 0$ , for  $p \geq 2$ ,

$$\begin{aligned} \delta f &= \omega \frac{1}{(m_1 - m_0 + 1)^p + p\zeta(m_1 - m_0 + 1)^{p-1}} \left( \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i)(1 + \varepsilon\eta_j^i) \right)^p \Delta t \\ &\quad - \omega \left( \frac{1}{m_1 - m_0 + 1} \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \right)^p \Delta t, \\ \delta f &= \omega \frac{1}{(m_1 - m_0 + 1)^p + p\zeta(m_1 - m_0 + 1)^{p-1}} \left\{ \left( \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \right)^p \right. \\ &\quad \left. + \varepsilon p \left( \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \right)^{p-1} \left( \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \eta_j^i \right) - \left( 1 + \frac{p\zeta}{m_1 - m_0 + 1} \right) \left( \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \right)^p \right\} \Delta t. \end{aligned}$$

Dividing by  $\varepsilon$  and taking the limit as  $\varepsilon \rightarrow 0$ ,

$$\begin{aligned} \frac{df}{d\mathbf{D}_j} &= \lim_{\varepsilon \rightarrow 0} \frac{\delta f}{\varepsilon} = \omega \frac{1}{(m_1 - m_0 + 1)^p + p\zeta(m_1 - m_0 + 1)^{p-1}} \left\{ \frac{1}{\varepsilon} \left[ \left( \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \right)^p - \left( \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \right)^p \right] \right. \\ &\quad \left. + p \left( \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \right)^{p-1} \left( \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \eta_j^i \right) \right. \\ &\quad \left. - p \left( \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \right)^{p-1} \frac{\sum_{i=m_0}^{m_1} \eta_j^i}{m_1 - m_0 + 1} \left( \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \right) \right\} \Delta t. \end{aligned}$$

The following limit identities hold:

$$\lim_{\varepsilon \rightarrow 0} \mathbf{Q}^i = \mathbf{Q}^i, \quad \lim_{\varepsilon \rightarrow 0} \tau^i(t^i) = t^i, \quad (72)$$

and

$$\lim_{\varepsilon \rightarrow 0} \frac{F(\mathbf{Q}^i) - F(\mathbf{Q}^i)}{\varepsilon} = \frac{\partial F^i}{\partial \mathbf{Q}^i} \frac{\partial \mathbf{Q}^i}{\partial \mathbf{D}_j} + \frac{\partial F^i}{\partial \mathbf{D}_j} = \frac{\partial F^i}{\partial \mathbf{Q}^i} \mathbf{V}^i + \frac{\partial F^i}{\partial \mathbf{D}_j}, \quad F^i \equiv F(\mathbf{Q}^i). \quad (73)$$

Consequently,

$$\begin{aligned} \frac{df}{d\mathbf{D}_j} = & \omega \frac{p}{(m_1 - m_0 + 1)^p} \left( \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \right)^{p-1} \left\{ \left( \sum_{i=m_0}^{m_1} \frac{\partial F}{\partial \mathbf{Q}^i} \mathbf{V}_j^i \right) + \left( \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \eta_j^i \right) \right. \\ & \left. - \frac{\sum_{i=m_0}^{m_1} \eta_j^i}{m_1 - m_0 + 1} \left( \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \right) \right\} \Delta t + \omega \frac{p}{(m_1 - m_0 + 1)^p} \left( \sum_{i=m_0}^{m_1} F(\mathbf{Q}^i) \right)^{p-1} \frac{\partial F}{\partial \mathbf{D}_j} \Delta t. \end{aligned} \quad (74)$$

$$\frac{df}{d\mathbf{D}_j} = \sum_{i=m_0}^{m_1} ([\mathbf{g}^i]^T \mathbf{V}_j^i + h_i \eta_j^i) + \frac{\partial f}{\partial \mathbf{D}_j}, \quad (75)$$

where

$$[\mathbf{g}^i]^T = \frac{\partial f}{\partial \mathbf{Q}^i} = \omega \Delta t \frac{p \bar{F}^{p-1}}{m_1 - m_0 + 1} \frac{\partial F^i}{\partial \mathbf{Q}^i}, \quad (76)$$

$$h^i = \omega \Delta t \frac{p \bar{F}^{p-1}}{m_1 - m_0 + 1} (F(\mathbf{Q}^i) - \bar{F}), \quad (77)$$

$$\bar{F} = \frac{1}{m_1 - m_0 + 1} \sum_{j=m_0}^{m_1} F(\mathbf{Q}^j), \quad (78)$$

$$\frac{\partial f}{\partial \mathbf{D}_j} = \omega \Delta t \frac{p \bar{F}^{p-1}}{m_1 - m_0 + 1} \sum_{i=m_0}^{m_1} \frac{\partial F^i}{\partial \mathbf{D}_j}. \quad (79)$$

### C. Computing Error Estimates for Time-Averaged Quantities

An objective function,  $f_{\Delta T}$ , computed over an averaging window of length  $\Delta T$  is

$$f_{\Delta T} = \frac{1}{\Delta T} \int_{T_0}^{T_0 + \Delta T} F(t) dt. \quad (80)$$

The error in  $f_{\Delta T}$  relative to the same objective function,

$$f_\infty = \lim_{\Delta T \rightarrow \infty} f_{\Delta T}, \quad (81)$$

computed over an infinite averaging window is estimated using a batch statistic variance estimator.<sup>36</sup> Assuming that samples of  $f_{\Delta T}$  for a given  $\Delta T$  are independent and distributed normally, there is a 99.6% probability that the next sample of  $f_{\Delta T}$  is within the range of  $f_\infty \pm 3\sigma$ , where

$$\sigma = \frac{1}{N} \left( \sum_{i=1}^N (f^i - \bar{f})^2 \right)^{0.5} \quad (82)$$

is the standard deviation of the normal distribution. Here,  $N$  is the number of samples,

$$f^i = \frac{1}{\Delta T} \int_{T_0 + (i-1)\Delta T}^{T_0 + i\Delta T} F(t) dt, \quad (83)$$

and

$$\bar{f} = \frac{1}{N} \sum_{i=1}^N f^i = \frac{1}{N\Delta T} \sum_{i=1}^N \int_{T_0 + (i-1)\Delta T}^{T_0 + i\Delta T} F(t) dt = \frac{1}{N\Delta T} \int_{T_0}^{T_0 + N\Delta T} F(t) dt = f_{N\Delta T}. \quad (84)$$

## Acknowledgments

Research for this paper was conducted with Government support under FA9550-11-C-0028 and awarded by the Department of Defense, Air Force Office of Scientific Research, National Defense Science and Engineering Graduate (NDSEG) Fellowship, 32 CFR 168a. The authors would also like to thank Mujeeb Malik, manager of the Revolutionary Computational Aerosciences project in the NASA Aeronautics Research Mission Directorate.

## References

- <sup>1</sup>Jameson, A., "Aerodynamic design via control theory," *Journal of Scientific Computing*, Vol. 3, No. 3, 1988, pp. 233–260.
- <sup>2</sup>Reuther, J., Jameson, A., Alonso, J. J., Rimlinger, M. J., and Sanders, D., "Constrained multipoint aerodynamic shape optimization using an adjoint formulation and parallel computers," *Journal of Aircraft*, Vol. 36, No. 1, 1999, pp. 51–74.
- <sup>3</sup>Martins, J. R. R. A., Alonso, J. J., and Reuther, J. J., "A Coupled-Adjoint Sensitivity Analysis Method for High-Fidelity Aero-Structural Design," *Optimization and Engineering*, Vol. 6, No. 1, 2005, pp. 33–62.
- <sup>4</sup>Nielsen, E. and Diskin, B., "Discrete Adjoint-Based Design for Unsteady Turbulent Flows on Dynamic Overset Unstructured Grids," *AIAA Journal*, Vol. 51, No. 6, 2013, pp. 1355–1373.
- <sup>5</sup>Jones, W. T., Nielsen, E. J., Lee-Rausch, E. M., and Acree, C. W., "Multi-point Adjoint-Based Design of Tilt-Rotors in a Noninertial Reference Frame," AIAA 2014-0290, Jan. 2014.
- <sup>6</sup>Venditti, D. and Darmofal, D., "Grid adaptation for functional outputs: Application to two-dimensional inviscid flow," *Journal of Computational Physics*, Vol. 176, 2002, pp. 40–69.
- <sup>7</sup>Giles, M. and Süli, E., "Adjoint methods for PDEs: a posteriori error analysis and postprocessing by duality," *Acta Numerica*, Vol. 11, 2002, pp. 145–236.
- <sup>8</sup>Gunzburger, M., *Perspectives in Flow Control and Optimization*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2002.
- <sup>9</sup>Bodony, D. and Natarajan, M., "Controller selection and placement in compressible turbulent flows," *Proceedings of the 2012 Summer Program*, Center for Turbulence Research, Stanford, Dec. 2012, pp. 35–42.
- <sup>10</sup>Wang, Q., *Uncertainty Quantification for Unsteady Fluid Flow using Adjoint-based Approaches*, PhD dissertation, Stanford University, 2009.
- <sup>11</sup>Wang, Q., Duraisamy, K., Alonso, J., and Iaccarino, G., "Risk Assessment of Scramjet Unstart Using Adjoint-Based Sampling Methods," *AIAA Journal*, Vol. 50, No. 3, 2012, pp. 581–592.
- <sup>12</sup>Lorenz, E., "Deterministic Nonperiodic Flow," *Journal of the Atmospheric Sciences*, Vol. 20, 1963, pp. 130–141.
- <sup>13</sup>Lea, D., Allen, M., and Haine, T., "Sensitivity analysis of the climate of a chaotic system," *Tellus*, Vol. 52A, 2000, pp. 523–532.
- <sup>14</sup>Wang, Q., Hui, R., and Blonigan, P., "Least Squares Shadowing sensitivity analysis of chaotic limit cycle oscillations," *Journal of Computational Physics*, Vol. 267, June 2014, pp. 210–224.
- <sup>15</sup>Ruelle, D., "Differentiation of SRB states," *Communications in Mathematical Physics*, Vol. 187, 1997, pp. 227–241.
- <sup>16</sup>Eyink, G., Haine, T., and Lea, D., "Ruelle's linear response formula, ensemble adjoint schemes and Lévy flights," *Nonlinearity*, Vol. 17, No. 5, 2004, pp. 1867–1889.
- <sup>17</sup>Lea, D., Haine, T., Allen, M., and Hansen, J., "Sensitivity analysis of the climate of a chaotic ocean circulation model," *Journal of the Royal Meteorological Society*, Vol. 128, 2002, pp. 2587–2605.
- <sup>18</sup>Thuburn, J., "Climate sensitivities via a Fokker-Planck adjoint approach," *Quarterly Journal of the Royal Meteorological Society*, Vol. 131, No. 605, January 2005, pp. 73–93.
- <sup>19</sup>Abramov, R. and Majda, A., "Blended response algorithms for linear fluctuation-dissipation for complex nonlinear dynamical systems," *Nonlinearity*, Vol. 20, No. 12, 2007, pp. 2793.
- <sup>20</sup>Pilyugin, S. Y., *Shadowing in dynamical systems*, Vol. 1706 of *Lecture Notes in Mathematics*, Springer-Verlag, Berlin Heidelberg, 1st ed., 1999.
- <sup>21</sup>Doedel, E. J. and Friedman, M. J., "Numerical computation of heteroclinic orbits," *Journal of Computational and Applied Mathematics*, Vol. 26, No. 1-2, June 1989, pp. 155–170.
- <sup>22</sup>Sanchez, J. and Net, M., "On the Multiple Shooting Continuation of Periodic Orbits by Newton-Krylov Methods," *International Journal of Bifurcation and Chaos*, Vol. 20, No. 1, 2010, pp. 43–61.
- <sup>23</sup>Farmer, J. D. and Sidorowich, J. J., "Optimal Shadowing and Noise Reduction," *Physica D*, Vol. 47, 1991, pp. 373–392.
- <sup>24</sup>Blonigan, P. and Wang, Q., "Least Squares Shadowing sensitivity analysis of a modified Kuramoto-Sivashinsky equation," *Chaos, Solitons, and Fractals*, Vol. 64, July 2014, pp. 16–25.
- <sup>25</sup>Gomez, S., *Parallel Multigrid for Large-Scale Least Squares Sensitivity*, Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, 2013.
- <sup>26</sup>Biedron, R. T., Carlson, J.-R., Derlaga, J. M., Gnoffo, P. A., Hammond, D. P., Jones, W. T., Kleb, W. L., Lee-Rausch, E. M., Nielsen, E. J., Park, M. A., Rumsey, C. L., Thomas, J. L., and Wood, W. A., "FUN3D Manual: 12.8," Tech. Rep. NASA/TM2015-218807, NASA Langley Research Center, Hampton, Virginia, May 2015.
- <sup>27</sup>Anderson, W. K. and Bonhaus, D. L., "An Implicit Upwind Algorithm for Computing Turbulent Flow on Unstructured Grids," *Comp. and Fluids*, Vol. 23, No. 1, 1994, pp. 1–21.
- <sup>28</sup>Biedron, R. T. and Thomas, J. L., "Recent Enhancements to the FUN3D Flow Solver for Moving-Mesh Applications," AIAA 2009-1360, Jan. 2009.

- <sup>29</sup>Gnoffo, P. A. and White, J. A., “Computational Aerothermodynamic Simulation Issues on Unstructured Grids,” AIAA 2004-2371, 2004.
- <sup>30</sup>Spalart, P. R. and Allmaras, S. R., “A One-Equation Turbulence Model for Aerodynamic Flows,” *La Recherche Aerospaciale*, Vol. 1, No. 1, 1994, pp. 5–21.
- <sup>31</sup>Lyness, J. N. and Moler, C. B., “Numerical Differentiation of Analytic Functions,” *SIAM Journal of Numerical Analysis*, Vol. 4, No. 2, 1967, pp. 202–210.
- <sup>32</sup>Anderson, W. K., Newman, J. C., Whitfield, D. L., and Nielsen, E. J., “Sensitivity Analysis for Navier-Stokes Equations on Unstructured Meshes Using Complex Variables,” *AIAA Journal*, Vol. 39, No. 1, 2001, pp. 56–63.
- <sup>33</sup>Wang, Q., “Convergence of the least squares shadowing method for computing derivative of ergodic averages,” *SIAM Journal of Numerical Analysis*, Vol. 52, No. 1, 2014, pp. 156–170.
- <sup>34</sup>Saad, Y. and Schultz, M. H., “GMRES: A Generalized Minimum Residual Method for Solving Nonsymmetric Linear Systems,” *SIAM Journal on Scientific and Statistical Computing*, Vol. 7, No. 3, 1986, pp. 856–869.
- <sup>35</sup>Saad, Y., *SPARSKIT: a basic tool kit for sparse matrix computations*, version 2 ed., June 1994, URL: <http://www-users.cs.umn.edu/~saad/software/SPARSKIT/> [cited 2 December 2015].
- <sup>36</sup>Fishman, G. S., “Grouping Observations in Digital Simulation,” *Management Science*, Vol. 24, No. 5, 1978, pp. 510–521.