

# Output Error Estimation for Chaotic Flows

Yukiko S. Shimizu\* and Krzysztof J. Fidkowski†

*Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI 48109, USA*

This paper presents two methods for estimating the effect of numerical discretization error on statistical output accuracy in chaotic unsteady flow simulations: (1) an extension of recent advances in least-squares shadowing sensitivity calculations, and (2) an approximate time-windowing approach with individual adjoint solutions on each time window. Both methods rely on output adjoints, a direct application of which is not possible for chaotic systems, in which sensitivities to initial conditions and single-point discretization errors grow exponentially. This paper shows results for two prototypical chaotic systems: the Lorenz oscillator and the modified ergodic Kuramoto-Sivashinsky partial differential equation (MEKS). In addition it presents results for a low-Reynolds number Navier-Stokes flow to demonstrate the effectivity of the error estimates. Preliminary adaptive results are also included, in which spatially-localized forms of the error estimates drive static mesh adaptation to reduce errors in statistical outputs.

## I. Introduction

Chaotic systems are becoming increasingly relevant to computational fluid dynamics (CFD) analysis and design. Specifically, large-eddy simulations (LES) of turbulent flows have the potential to provide much-needed fidelity in applications ranging from high-lift wing configurations to jet engine mixing and combustion; however, at present, they are quite expensive. This is not only due to computational costs associated with fine grids, but also due to additional costs associated with user-intensive grid generation, expert interpretation of solutions, re-runs to address problematic areas, and convergence studies to verify solutions.<sup>1,2</sup> Automated error estimation and grid adaptation<sup>3</sup> have dramatically reduced such costs for steady and deterministic unsteady problems, but their applicability to LES remains a challenge. This is due to the physics' non-deterministic nature, which stifles traditional error estimation methods, and the wide range of length scales requiring resolution, which demands meshes that tax computational resources even on today's high-performance machines. This paper presents initial research into techniques that address these challenges through the application of output-based adaptive methods to unsteady chaotic flows.

Though various factors, such as uncertainty in physical models or boundary conditions, pollute turbulent flow simulations, our attention to numerical error is driven by the fact that quantification of these errors is indispensable in ascribing confidence to simulations, avoiding tedious grid convergence studies, and automating mesh generation for robust analysis and design. Furthermore, in cases where other errors dominate, numerical error estimates would indicate this and could drive adaptation to yield more efficient meshes.

While several numerical error estimation methods exist, such as those based on approximation error, truncation error, features, etc, output-based methods hold the most promise for robustness,

---

\*Graduate Research Assistant, AIAA Member

†Associate Professor, AIAA Senior Member

as demonstrated in deterministic steady and unsteady flows. For chaotic flows with a large range of scales and potentially-important regions to resolve, accuracy of the error estimates and robustness of the adaptation with respect to distracting features will be paramount. Output-based approaches offer a mathematical framework for such accuracy and robustness by focusing discretization attention to only those regions that are important for the output.

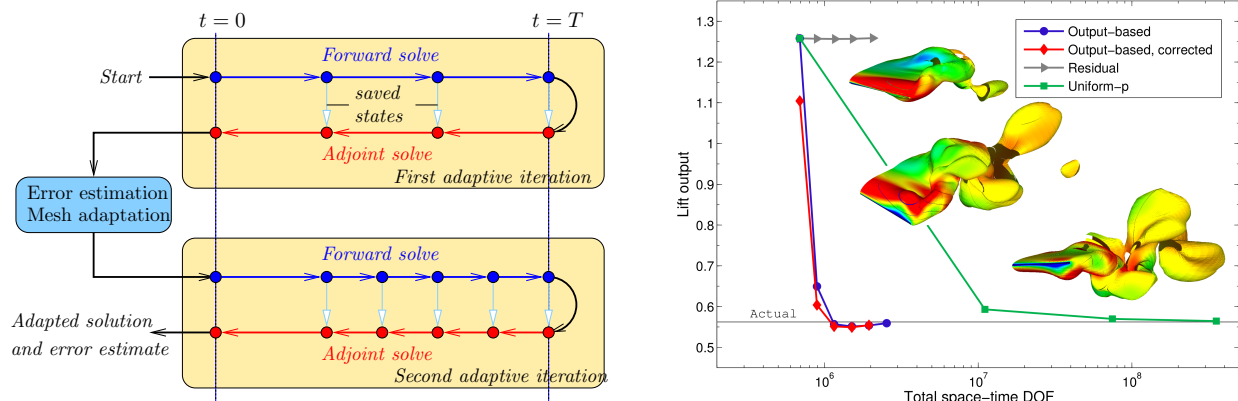
In this paper, we show that adjoint-based error estimates are possible and useful for chaotic systems, and we design an adaptive strategy driven by these estimates. The outline for the remainder of this paper is as follows. In Section II we discuss output-based error estimation and the difficulties with chaotic systems. The next two sections, III and IV, present our two approaches for estimating output error in chaotic systems. We show preliminary results in Section V and conclude with plans for the final paper in Section VI.

## II. Background

### II.A. Output-Based Methods for Unsteady Flows

Whereas output-based adaptive methods are now fairly mature for steady problems,<sup>3–8</sup> their application to unsteady simulations has been more limited<sup>9–14</sup> due largely to implementation challenges and computational expense associated with the solution of a fine-space adjoint equation, especially for nonlinear problems. Over the last few years, several groups have made progress on various fronts in tackling unsteady problems, including: temporal-only error estimation and adaptation;<sup>9, 10</sup> spatial-only error estimation and adaptation;<sup>11, 15, 16</sup> combined temporal and spatial mesh refinement with a static geometry and mesh;<sup>13, 17</sup> combined temporal and dynamic spatial refinement on static geometries;<sup>14, 18, 19</sup> combined temporal and dynamic-order spatial refinement on deformable domains.<sup>20–22</sup> In our previous work we have employed space-time discontinuous Galerkin (DG) and hybridized discontinuous Galerkin (HDG)<sup>23–28</sup> finite element discretizations using time slabs and an approximate space-time solver.<sup>13, 29</sup>

Output-based methods rely on an adjoint solution, which for unsteady problems is typically obtained by reverse time-integration and linearization about a stored primal state. Figure 1(a) shows a schematic of the adaptive process, in which the unsteady simulation is run multiple times, starting with a coarse space-time mesh that is successively improved. Important in unsteady calculations is a means to distinguish spatial and temporal errors, i.e. space-time anisotropy, and we have developed a technique to do so via projection of the fine-space adjoint to semi-refined spaces.<sup>29</sup>



(a) Unsteady output-based adaptive solution process

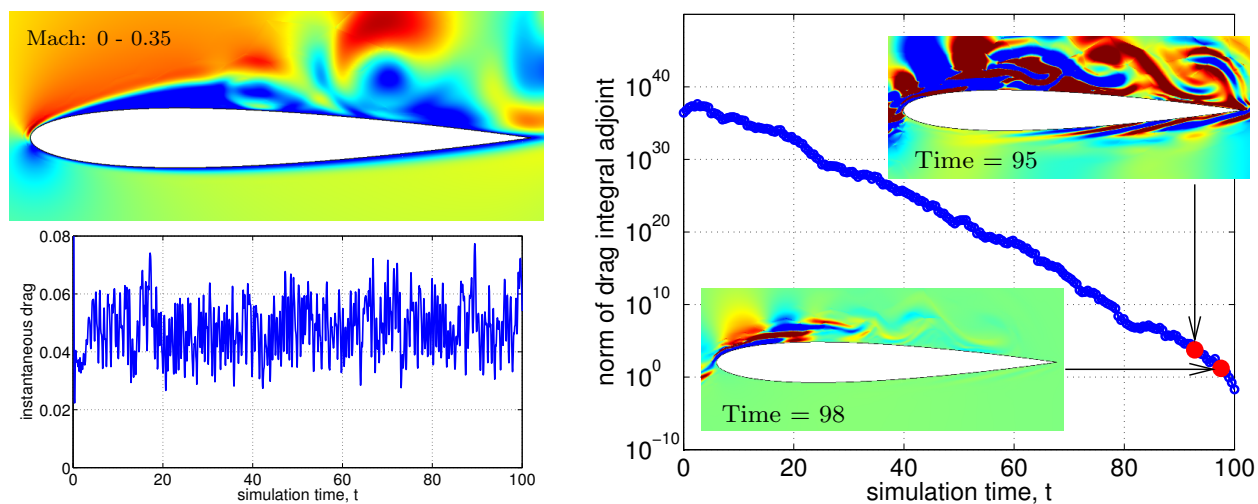
(b) Output convergence for a flapping wing simulation

**Figure 1. Schematic of an adaptive primal and adjoint solution procedure for output-based unsteady simulations and output convergence result for a representative flapping-wing Navier-Stokes simulation on a deforming domain.**

We have investigated several unsteady adaptive schemes, including time slab bisection, time node redistribution, static  $h$  spatial mesh refinement, and static/dynamic spatial order refinement. We have also considered problems with mesh motion in an ALE formulation,<sup>21</sup> and Figure 1(b) shows a dynamic-order adaptive result for a sample flapping-wing problem. In this case, output-based refinement clearly beats uniform refinement and other indicators, an advantage that extends to computational time in our implementation.

## II.B. Chaotic Systems

The example in Figure 1(b) illustrates a successful application of adjoint-based error estimation and mesh adaptation to a complex unsteady Navier-Stokes simulation at low Reynolds number and relatively short simulation time. However, for higher Reynolds numbers and longer simulation times, output quantities become chaotic and the adjoint-based error estimates fail. Specifically, the instantaneous state at late times becomes highly sensitive to initial condition perturbations, so that the reverse-integrated adjoint eventually diverges. Statistical outputs, such as time-averaged forces, are of engineering interest and can still be computed. Just as outputs of deterministic simulations, these outputs are polluted by discretization errors<sup>30,31</sup> due to insufficient spatial or temporal mesh resolution. However, output-based methods cannot be directly applied to these situations, though defining outputs as time averages does not cure the initial-condition sensitivity and associated adjoint instability. Figure 2 illustrates this point for a time-average drag calculation on an airfoil at moderate angle of attack in viscous flow: the adjoint field quickly deteriorates and its norm grows exponentially, as expected.<sup>32,33</sup>



(a) Primal snapshot and output history

(b) Drag integral adjoint history and snapshots

**Figure 2.** NACA 0012,  $M = 0.2$ ,  $Re = 10k$ ,  $\alpha = 8^\circ$ : ill-conditioning of average-drag prediction manifests itself through an unstable adjoint; i.e. the output is highly-sensitive to initial conditions.

The high sensitivity to initial conditions indicates the onset of chaotic behavior of the solution and calls for new strategies in estimating the impact of numerical error on relevant statistical outputs. A similar problem shows up in parameter sensitivity calculations, e.g. for optimization, and novel work is currently underway to address this. One technique is least-squares shadowing,<sup>32</sup> in which sensitivities are computed by solving an optimization problem for a trajectory that is close to a baseline trajectory, but in which the physics are governed by the perturbed parameter. In this work we extend this approach to estimating and reducing output errors due to insufficient discretization. Furthermore, we develop a cheaper time-windowing alternative that approximately estimates numerical error in an averaged output through multiple adjoint solutions on short time

windows of the simulation.

### II.C. Effects of Discretization Error on Chaotic Systems

To determine the impact of discretization errors on statistical outputs of chaotic problems, we begin with a prototypical chaotic system, the Lorenz oscillator,<sup>34</sup>

$$\frac{d\mathbf{u}}{dt} = \mathbf{f}(\mathbf{u}), \quad \mathbf{u} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \quad \mathbf{f} = \begin{bmatrix} \sigma(y - x) \\ x(\rho - z) - y \\ xy - \beta z \end{bmatrix}. \quad (1)$$

In this system,  $\sigma, \rho, \beta$  are parameters with baseline values of 10, 28,  $8/3$ , respectively. We integrate Eqn. 1 using a discontinuous Galerkin (DG) temporal discretization,<sup>35</sup> in which we can vary the temporal approximation order,  $r$ . We compare temporal orders  $r = 1$  (coarse) and  $r = 2$  (fine) for integrating the system from  $t = 0$  to  $t = T$ , the final time. The output of interest is the mean  $z$ -coordinate,  $\bar{J} \equiv \frac{1}{T} \int_0^T z dt$ . For  $T \rightarrow \infty$  and exact temporal integration,  $\bar{J}$  is a well-defined mean. However, for finite  $T$  and inexact temporal integration, statistical and discretization errors pollute  $\bar{J}$ .

Long integration times and high-fidelity temporal integration reduce errors but add computational expense, and hence quantifying the relative importance of each error source is of interest in ensuring optimal-efficiency calculations. We gain insight into the magnitudes of these errors numerically for the Lorenz system by varying  $T$  and comparing statistics for temporal orders  $r = 1$  with those for  $r = 2$ . Figure 3 illustrates sample trajectories obtained using  $\Delta t = .05$  and  $T = 20$ , starting with the same initial conditions. The two solutions are initially close but then drift apart. This is expected in a chaotic system, and of interest is how the discretization error affects the desired *statistical* output.

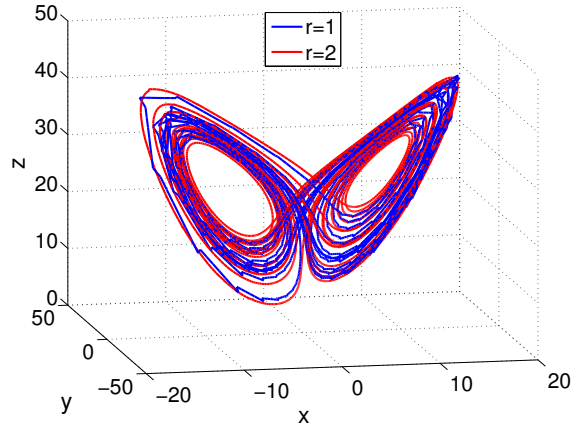


Figure 3. Sample trajectories for the Lorenz system using linear and quadratic temporal basis functions.

Figure 4 shows a quantitative comparison of output differences for  $\Delta t = .05$  and an ensemble of 500 different random initial conditions. A burn-time of  $0.2T$  is used to allow the system to settle around the attractor for each choice of initial conditions. As expected, the statistics improve (ensemble standard deviation drops) as  $T$  increases, but there is a persistent discrepancy between the two temporal accuracy orders and the ensemble means converge to different values. The difference is similar for all  $T$  and larger than the statistical errors measured by the ensemble standard deviations, which indicates that this case warrants higher fidelity time integration in lieu of longer integration times. For more complex simulations, we cannot afford such a detailed convergence study, and we need to make this decision based on more efficient error estimates. Furthermore, for a discretization of partial differential equations, numerical error arises from both spatial and temporal discretizations, and an accurate distinction between the two is vital to efficiency and convergence of adaptation.

### III. Least-Squares Shadowing

The least-squares shadowing (LSS) approach has been used successfully to compute efficient sensitivities for chaotic systems.<sup>32,36–38</sup> In this section we briefly review the LSS formulation and

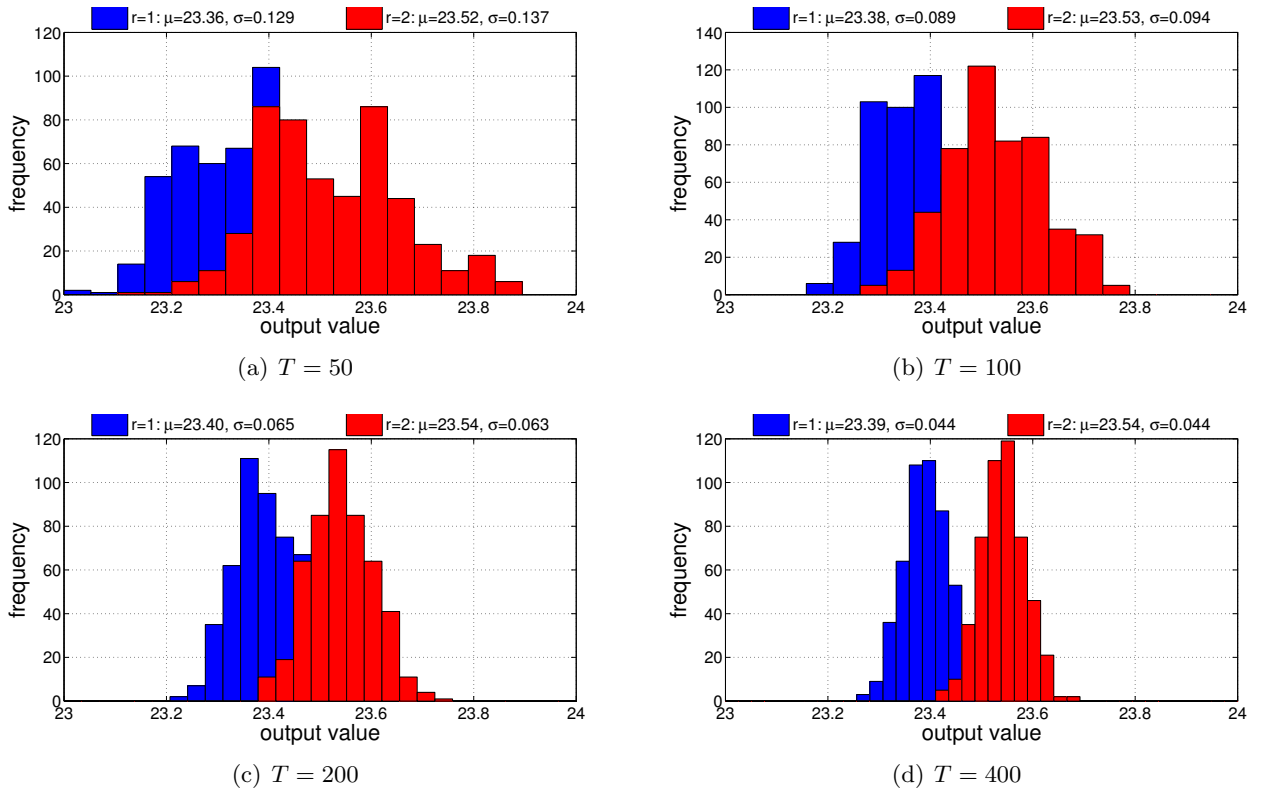


Figure 4. Lorenz oscillator: ensemble statistics of outputs computed using different integration times,  $T$ , and DG-in-time integration order,  $r$ . Each ensemble contains 500 experiments, and  $\mu$  and  $\sigma$  refer to the ensemble mean (more accurate with increasing  $r$ ) and standard deviation (smaller with increasing  $T$ ).

present its extension to output-based error estimation.

### III.A. The LSS primal problem

Consider a discrete dynamical system parametrized by  $s$ ,

$$\frac{d\mathbf{u}}{dt} = \mathbf{f}(\mathbf{u}, s), \quad (2)$$

LSS reduces to finding a tangent solution  $\mathbf{v}(t)$  that solves the following minimization statement:<sup>32</sup>

$$\min_{\mathbf{v}} \frac{1}{2} \int_0^T \|\mathbf{v}\|^2 dt, \quad \text{s.t.} \quad \frac{d\mathbf{v}}{dt} = \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \mathbf{v} + \frac{\partial \mathbf{f}}{\partial s} + \eta \mathbf{f}, \quad (3)$$

where  $\eta = d\tau/dt - 1$  is a time-dilation term. Intuitively, for a parameter perturbation  $\delta s$ ,  $\mathbf{u}(t) + \mathbf{v}(t)\delta s$  is the perturbed “shadow” trajectory that incorporates the physics of the parameter change but does not stray far from  $\mathbf{u}(t)$ . Eqn. 3 is a constrained optimization problem with Lagrangian,

$$\mathcal{L}^{\text{LSS}} = \int_0^T \left[ \frac{1}{2} \|\mathbf{v}\|^2 + \mathbf{w}^T \left( \frac{d\mathbf{v}}{dt} - \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \mathbf{v} - \frac{\partial \mathbf{f}}{\partial s} - \eta \mathbf{f} \right) \right] dt, \quad (4)$$

where  $\mathbf{w}(t)$  is the Lagrange multiplier. Requiring the Lagrangian to be stationary with respect to variations in  $\mathbf{w}$  gives the original tangent equation in Eqn. 3; requiring stationarity with respect to

variations in  $\mathbf{v}$  gives, after an integration by parts,

$$\mathcal{L}_{\mathbf{v}}^{\text{LSS}} = \mathbf{0} : \quad \mathbf{v} - \frac{d\mathbf{w}}{dt} - \frac{\partial \mathbf{f}^T}{\partial \mathbf{u}} \mathbf{w} = \mathbf{0}. \quad (5)$$

These equations constitute a linearized primal system for the tangent solution,  $\mathbf{v}$ , and the Lagrange multiplier used in the physics constraints,  $\mathbf{w}$ . The effect of  $\eta$  in Eqn. 3 can be incorporated by using a projection operator,<sup>38</sup> in which case we solve the tangent equation without the  $\eta$  term,

$$\frac{d\mathbf{v}'}{dt} = \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \mathbf{v}' + \frac{\partial \mathbf{f}}{\partial s}, \quad (6)$$

and then apply the projection  $\mathbf{v}(t) = P_t \mathbf{v}'(t) \equiv \mathbf{v}'(t) - \mathbf{v}'(t)^T \mathbf{f}(t) \mathbf{f}(t) / (\mathbf{f}(t)^T \mathbf{f}(t))$ .

### III.B. The LSS adjoint problem

We consider a time-average output,  $\bar{J} = \frac{1}{T} \int_0^T J(\mathbf{u}, s) dt$ . Linearizing about the reference primal trajectory, we define  $\bar{J}' = \frac{1}{T} \int_0^T \frac{\partial J}{\partial \mathbf{u}} \mathbf{v} dt$ . To derive adjoint equations for the linearized output, we construct the following Lagrangian,

$$\mathcal{L} = \bar{J}' - \int_0^T (\hat{\mathbf{w}}^T \mathbf{r}_{\mathbf{v}'} + \hat{\mathbf{v}}'^T \mathbf{r}_{\mathbf{w}}) dt, \quad (7)$$

where  $\hat{\mathbf{w}}$  and  $\hat{\mathbf{v}}'$  are the adjoint variables associated with the two residuals  $\mathbf{r}_{\mathbf{v}'}$  and  $\mathbf{r}_{\mathbf{w}}$  of the primal equations (6) and (5), respectively. Specifically,

$$\mathbf{r}_{\mathbf{v}'} \equiv \frac{d\mathbf{v}'}{dt} - \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \mathbf{v}' - \frac{\partial \mathbf{f}}{\partial s}, \quad \mathbf{r}_{\mathbf{w}} \equiv \frac{d\mathbf{w}}{dt} + \frac{\partial \mathbf{f}^T}{\partial \mathbf{u}} \mathbf{w} - P_t \mathbf{v}'. \quad (8)$$

The adjoint equations that govern  $\hat{\mathbf{w}}$  and  $\hat{\mathbf{v}}'$  derive from variations of  $\mathcal{L}$  with respect to  $\mathbf{v}'$  and  $\mathbf{w}$ :

$$\frac{d\hat{\mathbf{v}}'}{dt} = \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \hat{\mathbf{v}}', \quad (9)$$

$$\frac{d\hat{\mathbf{w}}}{dt} = -\frac{\partial \mathbf{f}^T}{\partial \mathbf{u}} \hat{\mathbf{w}} - P_t \hat{\mathbf{v}}' - \frac{1}{T} \frac{\partial J^T}{\partial \mathbf{u}}. \quad (10)$$

In summary, the adjoint system is given by Eqn. 9 and Eqn. 10. Due to the boundary conditions, which derive from the boundary terms in the integration by parts, the equation for  $\hat{\mathbf{v}}'$  is solved forward in time, whereas the equation for  $\hat{\mathbf{w}}$  is solved backward in time.

### III.C. Checkpoint design

The LSS adjoint system derived in the previous section can be solved by an iterative checkpointing algorithm that only requires solves of the tangent and adjoint on short *time segments*.<sup>38</sup> The algorithm proceeds by first computing a reference primal solution and then converging the time-segment endpoint tangent and adjoint solutions,  $\hat{\mathbf{v}}_k$  and  $\hat{\mathbf{w}}_k$ , by an iterative matrix-free solver such as GMRES. The residuals for time segment  $k$  are given by

$$\begin{aligned} \mathbf{R}_{k+1}^{\hat{\mathbf{w}}} &= \hat{\mathbf{v}}_{k+1} - P_{t_{k+1}} \hat{\mathbf{v}}'(t_{k+1}^-), \quad \text{when } k < K \\ \mathbf{R}_k^{\hat{\mathbf{v}}} &= \hat{\mathbf{w}}_k - P_{t_k} \hat{\mathbf{w}}(t_k^+), \end{aligned}$$

and the calculation of these requires tangent and adjoint solves on the time segment. In this algorithm, we have a total of  $2K - 1$  unknowns:  $\hat{\mathbf{v}}_0, \dots, \hat{\mathbf{v}}_{K-1}$  and  $\hat{\mathbf{w}}_1, \dots, \hat{\mathbf{w}}_{K-1}$ , and  $2K - 1$  residuals:  $\mathbf{R}_1^{\hat{\mathbf{w}}}, \dots, \mathbf{R}_{K-1}^{\hat{\mathbf{w}}}$  and  $\mathbf{R}_0^{\hat{\mathbf{v}}}, \dots, \mathbf{R}_{K-1}^{\hat{\mathbf{v}}}$ . For a large number of time segments, the stiffness of this system grows and demands efficient preconditioning strategies.

### III.D. Output error estimation via the adjoint-weighted residual

After the adjoint system solve, output sensitivities can be calculated from the  $\hat{\mathbf{w}}$  variable, since this is the adjoint that weights the residual term with  $\frac{\partial \mathbf{f}}{\partial \mathbf{s}}$ , as given in Eqn. 8. Specifically,

$$\frac{d\bar{J}}{ds} = \int_0^T \hat{\mathbf{w}}^T \frac{\partial \mathbf{f}}{\partial \mathbf{s}} dt. \quad (11)$$

To estimate the output error using the LSS adjoint, we apply Eqn. 11, using a residual perturbation computed from two different discretization spaces: a coarse one with temporal order  $r_H$ , and a fine one with temporal order  $r_h = r_H + 1$ . We solve the primal problem with order  $r_H$  and then inject the solution into the finer space. Doing so gives us a perturbation in the residual, which we weight by the fine-space adjoint to obtain the error estimate:

$$\delta \bar{J} = - \int_0^T \hat{\mathbf{w}}_h^T \left[ \frac{d\mathbf{u}_H}{dt} - f(\mathbf{u}_H) \right]_h dt = - \hat{\mathbf{W}}_h^T \mathbf{R}_h(\mathbf{u}_H), \quad (12)$$

where  $\hat{\mathbf{W}}_h$  is the vector of all of the  $\hat{\mathbf{w}}_h$  unknowns and  $\mathbf{R}_h(\mathbf{u}_H)$  is the order  $r_h$  residual vector evaluated with the order  $r_H$  injected solution.

## IV. Time Windowing

The LSS-based approach to output error estimation builds on an established mathematical foundation and, as will be seen in the results, shows promising accuracy and robustness. However, it is also computationally expensive due to the high cost of calculating the entire adjoint solution via an iterative solver. For complex chaotic problems, such as large-eddy simulations, where obtaining the reference primal solution already taxes computational resources, these additional calculations may not be practical.

When calculating sensitivities needed for optimization, the accuracy delivered by the LSS method is essential. However, when estimating output error or computing adaptive indicators, accuracy need not be critical. An adaptive simulation can proceed with just relative information about where the errors are high and low; some approximate absolute error measure can then inform a stopping criterion. This suggests that cheaper techniques than LSS may suffice for error estimation and adaptation.

### IV.A. Overview

Our second approach for error estimation is such a method, based on a simple time windowing approach illustrated in Figure 5. This method is essentially an application of existing output-based unsteady error estimation to interspersed short temporal windows of the chaotic simulation. The motivation for this approach is that statistical outputs are computed as averages over the entire temporal domain, so that error contributions aggregated over sub-intervals should be representative of the total error.

Specifically, suppose we divide our time horizon into  $N_w$  windows,  $t \in [T_i, T_{i+1}]$ ,  $i = 1..N$ , each long enough such that the window outputs,

$$\bar{J}_i \equiv \int_{T_i}^{T_{i+1}} J(\mathbf{u}(t)) dt, \quad 1 \leq i \leq N_w, \quad (13)$$

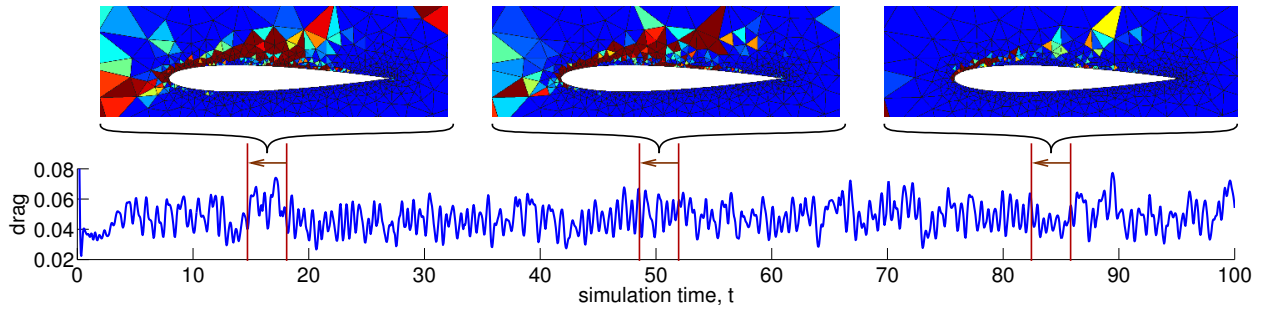


Figure 5. Schematic of an inexpensive time windowing approach to computing error estimates and adaptive indicators for chaotic flow. The adjoint-weighted residual is applied on short temporal windows interspersed throughout the temporal domain; the resulting indicators can be aggregated over all windows.

are minimally correlated. Note that we do not explicitly divide this output by  $T$ , but this factor could be incorporated into the definition of  $J$ . An estimator of the desired integral over the entire time horizon is then

$$\bar{J} = \sum_{i=1}^{N_w} \bar{J}_i. \quad (14)$$

If each time window is short enough such that the adjoint calculations on the time windows do not diverge, then we can compute an error estimate on each window using an unsteady adjoint-weighted residual technique.<sup>13</sup> The result is a  $\delta\bar{J}_i$  on each time window, and the error in the final statistical output is then estimated as

$$\delta\bar{J} = \sum_{i=1}^{N_w} \delta\bar{J}_i. \quad (15)$$

This summation over time windows extends similarly to spatially-localized error indicators. The result of such a sum is a distribution of spatial error indicators that can drive mesh adaptation. To make this method successful, the time windows must be chosen of the appropriate length: not too long, otherwise the window adjoints will become meaningless; and not too short, otherwise the window outputs will be correlated and the sum in Eqn. 14 will not be correct. For the results in the present paper, we monitor the adjoint solution to manually choose a suitable time window prior to the adjoint diverging, although this process could be automated by incorporating estimates of Lyapunov exponents. Finally, we note that for certain multiscale systems, a diverging adjoint may limit the time window to a size that is too small to decorrelate the  $\delta\bar{J}_i$ , and the resulting error estimate may not capture the interaction effects between windows. We are addressing this situation in current work.

#### IV.B. Implementation

Consider a dynamical system arising from the spatial discretization of a system of partial differential equations,

$$\bar{\mathbf{R}}(\mathbf{U}) \equiv \mathbf{M} \frac{d\mathbf{U}}{dt} + \mathbf{R}(\mathbf{U}) = \mathbf{0}. \quad (16)$$

In this equation,  $\mathbf{M} \in \mathbb{R}^{N \times N}$  is the spatial mass matrix,  $\mathbf{U} \in \mathbb{R}^N$  is the discrete state vector of basis function coefficients,  $\mathbf{R} \in \mathbb{R}^N$  is the discrete spatial residual vector, and  $\bar{\mathbf{R}}$  is the unsteady residual.



The calculation of an output error estimate over the  $i^{\text{th}}$  window requires the solution of an adjoint problem for  $\bar{J}_i$  in Eqn. 13. This could be a discrete adjoint, obtained strictly from the linearized discrete primal system, or a continuous adjoint obtained by discretizing the adjoint differential equation. In the present work we use an adjoint that is discrete in space but continuous in time, for flexibility in the choice of the time marching scheme and application to non-variational temporal discretizations. Starting with Eqn. 16, which is already discretized in space, we obtain the continuous-in-time adjoint equation by constructing an augmented Lagrangian and requiring its first variation with respect to allowable state perturbations to vanish. The result is

$$-\mathbf{M} \frac{d\boldsymbol{\Psi}}{dt} + \left( \frac{\partial \mathbf{R}}{\partial \mathbf{U}} \right)^T \boldsymbol{\Psi} + \left( \frac{\partial J}{\partial \mathbf{U}} \right)^T = \mathbf{0}. \quad (17)$$

Note that for nonlinear problems, the linearizations about  $\mathbf{U}$  in the adjoint equation depend on the primal state, which is stored to disk at every time node for a given window. We solve the adjoint over each time window using a fine approximation space, denoted by the subscript  $h$ . This consists of incrementing the spatial order by 1 and using a higher-order time marching scheme compared to the primal. The resulting error estimate for window  $i$  takes the form

$$\delta \bar{J}_i \approx - \int_{T_i}^{T_{i+1}} \boldsymbol{\Psi}_h^T \bar{\mathbf{R}}_h(\mathbf{U}_h^H) dt, \quad (18)$$

where  $\mathbf{U}_h^H$  is the injection of the primal solution from the original approximation space  $H$  to the fine space  $h$ . In the spatial domain, this is a pure injection to higher order:  $p \rightarrow p + 1$ . In the temporal domain, this involves a sufficiently-accurate reconstruction over the time interval, and for the schemes considered in this work, we use a cubic time reconstruction based on the state and its time derivative at the adjacent time nodes. The same temporal reconstruction is used when calculating  $\bar{J}_i$  and when marching the adjoint with a multi-stage method, both situations that require knowledge of the primal in between the time nodes.

#### IV.C. Adaptation

The output error estimate in Eqn. 18 can be separated into spatial and temporal components by using an adjoint that is fine only in space or only in time.<sup>29</sup> We obtain the temporal error over time window  $i$ ,  $\delta \bar{J}_i^{\text{time}}$ , by spatially projecting the fine adjoint to order  $p$  and recalculating the error via Eqn. 18 with the projected adjoint. The spatial error then results from a simple difference,  $\delta \bar{J}_i^{\text{space}} \equiv \delta \bar{J}_i - \delta \bar{J}_i^{\text{time}}$ .

The relative magnitudes of  $\delta \bar{J}_i^{\text{time}}$  and  $\delta \bar{J}_i^{\text{space}}$  can drive a space-time adaptation approach that appropriately targets the spatial and/or temporal discretizations. For the present work, we focus on the spatial discretization and simply hold the time discretization fixed at high accuracy. We also use static spatial refinement, one mesh over the entire simulation, driven by a temporally-marginalized adaptive indicator over the elements  $e$ . Based on Eqn. 18, we define the adaptive indicator for element  $e$  over time window  $i$  as

$$\epsilon_{ei} \equiv \left| \int_{T_i}^{T_{i+1}} \boldsymbol{\Psi}_{he}^T \bar{\mathbf{R}}_{he}(\mathbf{U}_h^H) dt \right|, \quad (19)$$

where the subscript  $e$  denotes restriction to element  $e$ . This is trivial in a discontinuous Galerkin spatial discretization, but similar restrictions could be formulated for other discretizations. We

sum the indicators over the windows to obtain a single indicator for each element,

$$\epsilon_e = \sum_{i=1}^{N_w} \epsilon_{ei}. \quad (20)$$

Given these indicators, we adapt the spatial mesh, presently via a simple fixed-fraction order refinement in which  $f^{\text{adapt}}$  of the elements with the largest  $\epsilon_e$  receive a unit order increment. Following adaptation, the set of  $N_w$  windows is re-run with the refined mesh for the next adaptive iteration. The initial condition for this run is the terminal state from the previous adaptive iteration, projected to the new mesh and simulated forward for a time  $T_{\text{burn}}$  to remove transients of the projection.

## V. Results

In this section we present results for both the least-squares shadowing and the time-windowing approaches to error estimation. For LSS, we restrict our attention to the Lorenz oscillator, whereas for time-windowing we show results for the modified ergodic kuramoto-sivashinky equations and adaptive results for the Navier-Stokes equations.

### V.A. Lorenz Oscillator

To test the error estimate in Eqn. 12 we run an ensemble of calculations with various simulation horizons  $T$ , time steps  $\Delta t = .05$ , and approximation orders  $r_H = 1$ , and  $r_h = 2$ . For each  $T$ , we run 50 simulations with different initial conditions. The primal problem initial conditions are chosen randomly, but to ensure that we are on an attractor during the output calculation, a burn time of  $0.5T$  is used.

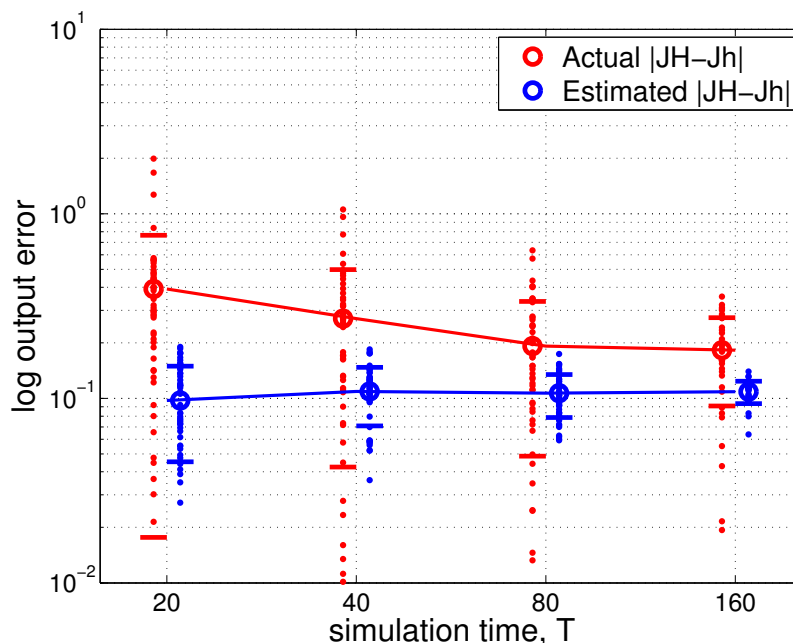


Figure 6. Lorenz system: temporal output error estimation using the LSS-adjoint weighted residual method. The points denote the output errors computed from 50 individual runs in each ensemble. The circles are the means of the ensemble errors, and horizontal line segments are drawn at  $\pm\sigma$ .

Figure 6 shows comparisons of the output-based error estimate and the actual error for the Lorenz system in which the output is the average of  $z$ . The actual error is obtained by running

the fine space primal problem with the same  $t = 0$  (post-burn) initial condition. Also, in these runs the time segment window was chosen to be  $T_k = 4$ , nearly the maximum possible while keeping the linearized calculations stable. The estimated numerical error under-predicts the actual error, particularly at low  $T$ , which is expected because for short simulation times, the statistical integration error dominates. At longer  $T$ , the numerical error estimate improves as the statistical error diminishes. We note that output error estimates are more tightly clustered than the actual errors, which indicates that reasonably high confidence output error results can be obtained without very long simulation times.

### V.B. Modified Ergodic Kuramoto-Sivashinsky

LSS exhibits accurate results for the Lorenz attractor, yet it is expensive. Therefore, in this section we turn to the cheaper, though less accurate, time windowing approach. Before considering the Navier-Stokes equations, we apply the windowing method to a one dimensional fourth order partial differential equation: the modified ergodic Kuramoto-Sivashinsky equation (MEKS). This equation was derived by Kuramoto in the context of angular phase turbulence for a system of reaction-diffusion equations modeling the Belousov-Zhabotinskii reaction in three space dimensions. In addition it was used by Sivashinsky to model thermal diffusive instabilities in laminar flame fronts.<sup>39</sup>

The governing partial differential equation reads

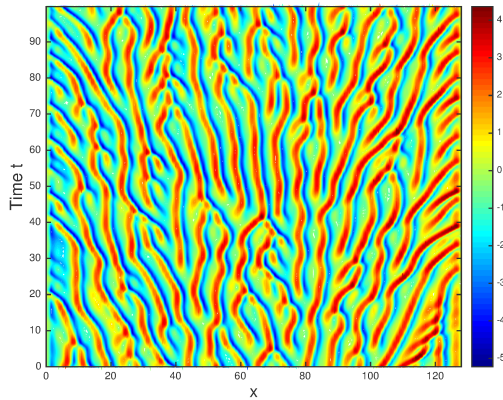
$$\frac{\partial u}{\partial t} = -(c + u)\frac{\partial u}{\partial x} - \alpha\frac{\partial^2 u}{\partial x^2} - \nu\frac{\partial^4 u}{\partial x^4}, \quad (21)$$

where  $c = 0.5$ ,  $\alpha = 1$ ,  $\nu = 0.25$ , and  $x \in [0, 128]$ . For these parameters, the system exhibits chaotic behavior, as seen Figure 7 when comparing original to perturbed trajectories. The advection term is added in order to ensure that the solution exhibits ergodic behavior.<sup>40</sup> The initial condition for the system is set to a triangular distribution on a discrete grid, where the solution is 0 at all locations except on the two middle elements, with  $u = 1$  at  $x = 64$ , dropping linearly to zero at the adjacent nodes. Periodic boundary conditions were imposed.

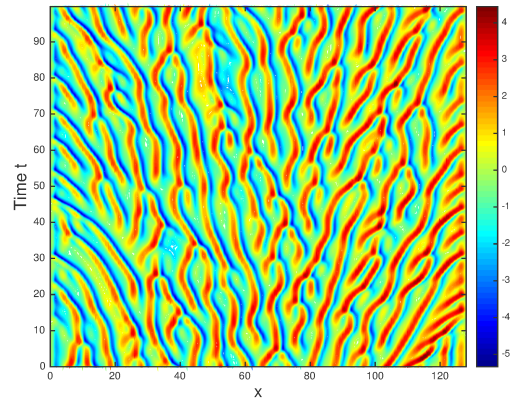
The discontinuous Galerkin method (DG) was used to discretize this equation in space while backward differentiation formulas (BDF) were used to march in time. The first order term was discretized using the upwinding method, the second order term was discretized using an interior penalty (IP) method, and the fourth order term was discretized using a modified IP method.<sup>41</sup> For the purpose of performing mesh adaptation, we are interested in error estimates between a coarse solution, subscript  $H$ , and a fine solution, subscript  $h$ . For this particular case,  $H$  refers to  $p = 1$  in space and BDF1 for time integration;  $h$  refers to  $p = 2$  in space and BDF2 for time integration. Before solving the forward solution for MEKS, a burn time is performed just as one would do for the Lorenz oscillator in order to ensure that the initial state of the simulation is indeed on the attractor. The burn time was set to  $T_{\text{burn}} = 1000$  to ensure that the state reached the attractor. Once the new initial conditions were found for the coarse solution  $H$ , the solution was injected in the fine space  $h$  and set as the initial conditions for the fine space solution. The output for each time segment was set to

$$J = \frac{u(x, t)}{X_f T_f}, \quad (22)$$

where  $T_f$  refers to the total time of the entire simulation, and  $X_f$  is the spatial domain length. A discrete adjoint solution was computed by marching backwards in time on each time window, error estimates were computed using the adjoint-weighted residual in Eqn. 19. These were then summed to estimate the total error in the output. When interpolating the state or adjoint in time, we used a linear reconstruction between time steps. Figure 8 shows the results for these runs.



(a) MEKS original trajectory



(b) MEKS perturbed trajectory

Figure 7. MEKS trajectories showing the sensitivity to initial conditions. Both trajectories were solved with  $p = 2$  and BDF2. The perturbed trajectory's initial condition (initial conditions that were found after the burn time) was changed by 0.01 at all nodes. By  $T \approx 30$ , the solutions cease to look similar.

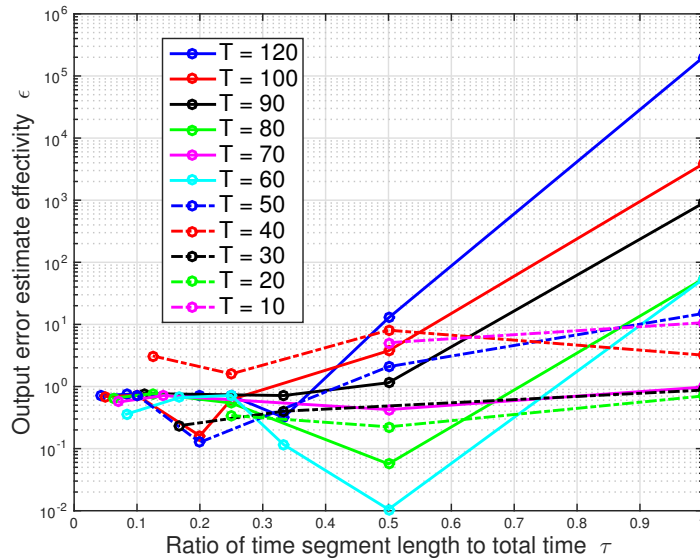
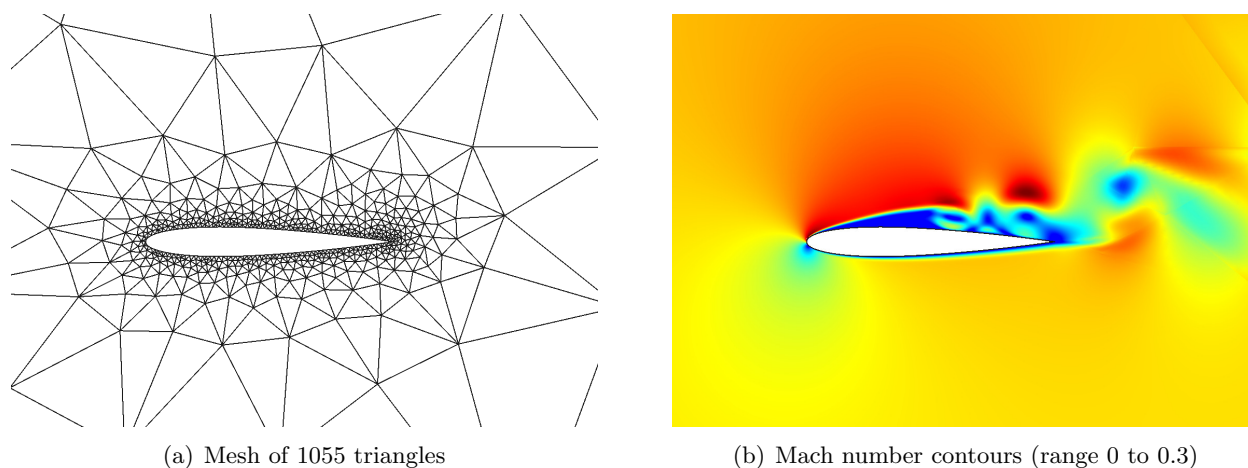


Figure 8. Modified ergodic Kuramoto-Sivashinsky: spatial and temporal output error estimation using the time windowing method. The circles refer to the different cases that were executed. The independent axis refers to the ratio of the length of the time segments and the total time  $\tau$ . The dependent axis refers to the output error estimate effectivity  $\epsilon$ , defined as the difference between the actual and estimated error nondimensionalized by the actual error. The circles at  $\epsilon = 1$  refer to the original (non time windowing) unsteady error estimation results.

As expected, at longer time lengths  $T$ , the error estimates grow exponentially, due to the divergence of the adjoint. According to Figure 8, for  $T = 120$ ,  $\tau \approx 0.3$  gives the best output error estimate effectivity, where  $\tau$  is the ratio of the window size to the total time,  $T$ . For  $T = 100$ , the best output error estimate effectivity occurs at  $\tau = 0.2$ . This changes with different time lengths; one reason is due to whether or not the system is exhibiting chaotic behavior. For smaller time lengths ( $T = 10 - 40$ ), the system does not show chaotic behavior, which leads to inaccurate error estimates when using the time-windowing method. This can be attributed to the fact that too many time windows in the simulation lead to loss of information tied to the initial conditions and coupling between the windows. For small time lengths such as these, the adjoint does not diverge and the original unsteady error calculations hold. For large  $T$ , too few windows lead to inaccurate error estimates due to divergence of the adjoint. It is thus advantageous to find the optimal  $\tau$ , typically as the maximum window size that does not yield a divergent adjoint solution.

### V.C. Compressible Navier-Stokes

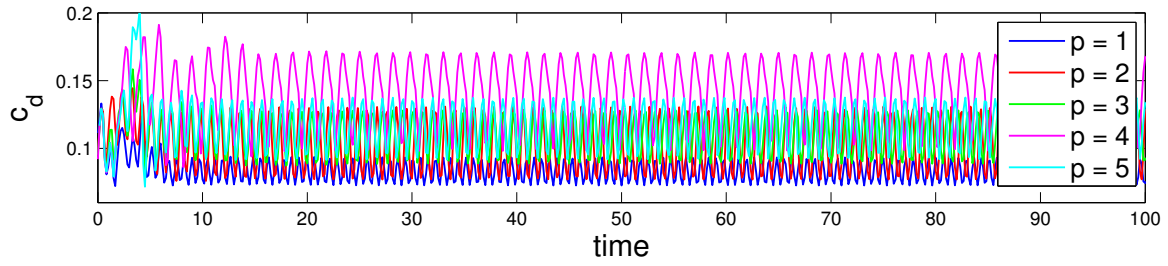
To demonstrate the time windowing method for error estimation and adaptation, we consider a compressible Navier-Stokes simulation of a NACA 0012 airfoil in viscous flow at  $M = 0.2$ ,  $Re = 10^4$ , and angle of attack  $\alpha = 8^\circ$ . The output of interest is the time average of the drag coefficient. Figure 9 shows the computational mesh used for this study, and a snapshot of the unsteady flow-field.



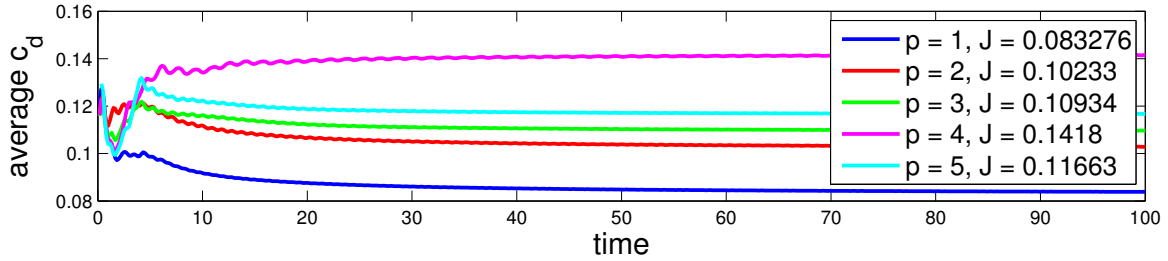
**Figure 9. NACA 0012,  $M = 0.2$ ,  $Re = 10^4$ ,  $\alpha = 8^\circ$ : computational mesh and flow-field snapshot.**

The farfield is approximately 100 chord lengths away from the airfoil, and the initial solution approximation order is uniform in space at  $p = 2$ . Prior to adapting, the state is initialized to free-stream and advanced forward for a time length of  $T_{\text{init}} = 100$  using third-order diagonally-implicit Runge-Kutta (DIRK3) time marching with  $\Delta t = .125$ . The compressible Navier-Stokes equations are solved in dimensional form with numerically convenient units, chosen such that the free-stream state is  $\mathbf{u}^T = [\rho, \rho u, \rho v, \rho E] = \left[1, \cos(\alpha), \sin(\alpha), \frac{1}{\gamma(\gamma-1)M^2} + \frac{1}{2}\right]$ , where  $\gamma = 1.4$  is the ratio of specific heats. The boundary conditions are full-state on the farfield and adiabatic no-slip wall on the airfoil.

Figure 10 shows time histories of the drag coefficient, and convergence of its time-averaged value, for different spatial orders  $p$ . The time horizon of interest is  $T = 100$ . As the mesh is coarse, discretization errors are quite large, even at moderate orders. Furthermore, the effect of the discretization error on the output is not predictable:  $p = 4$  behaves as an outlier in between  $p = 3$  and  $p = 5$ .



(a) Drag coefficient time histories



(b) Convergence of the average drag coefficient

**Figure 10. NACA 0012,  $M = 0.2$ ,  $Re = 10^4$ ,  $\alpha = 8^\circ$ : time histories and average drag coefficient convergence for uniform order refinement.**

Next, output-based order-adaptive simulations were performed using the time-windowing method described in Section IV. In these simulations, an optimal distribution of orders was generated by considering several short windows, whose summed total time was significantly less than the time horizon of interest,  $T = 100$ . Specifically, the window time lengths were uniformly set to 3, and  $N_w = 1$  and  $N_w = 4$  time windows were considered. After three adaptive iterations, using fixed fractions of  $f^{\text{adapt}} = 0.1$  or  $0.2$ , the resulting order distribution was used in the long-time simulation up to  $T = 100$ . Figure 11 shows the convergence of the time-averaged drag output, computed from the long-time simulations, versus the spatial degrees of freedom. The uniform order refinement results are also shown for comparison. We see monotonic and rapid convergence of the average drag coefficient when using the output-adapted order distributions. For  $f^{\text{adapt}} = 0.1$ , the output is comparable to uniform  $p = 5$ , but with a factor of 2.5 fewer spatial degrees of freedom. Furthermore, for this problem, there is little difference between the adaptive results obtained with 1 and 4 windows.

Figure 12 shows the time histories of the drag coefficient and the convergence of its average value for the adapted runs using  $f^{\text{adapt}} = 0.2$ ,  $N_w = 1$ . The time histories are similar, and the differences in the average drag coefficient are small. Finally, Figure 13 shows the spatial order distributions in the adapted meshes, over the three adaptive iterations, for both  $f^{\text{adapt}} = 0.2$  and  $f^{\text{adapt}} = 0.1$ . As expected, more elements are refined for the larger adaptive fraction. However, the regions targeted are similar in both cases. A crucial region appears to be on a curve that starts from the leading edge and propagates above the airfoil, along the line of leading-edge separation. Elements along the airfoil surface and large elements further along the wake are also targeted. The computational savings over uniform order refinement, especially for  $f^{\text{adapt}} = 0.1$ , comes from the many elements left at the baseline order of  $p = 2$  (blue in the figures).

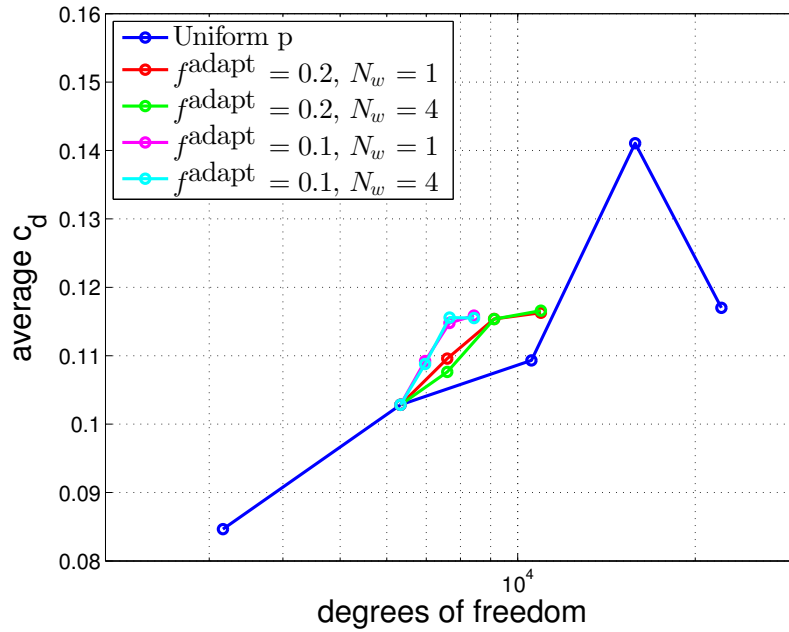
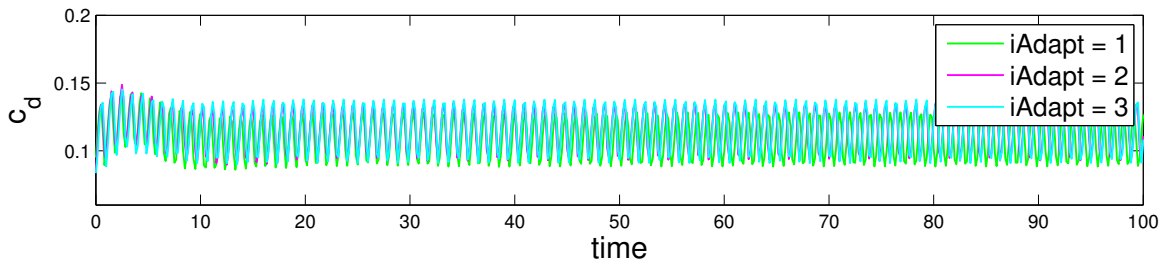
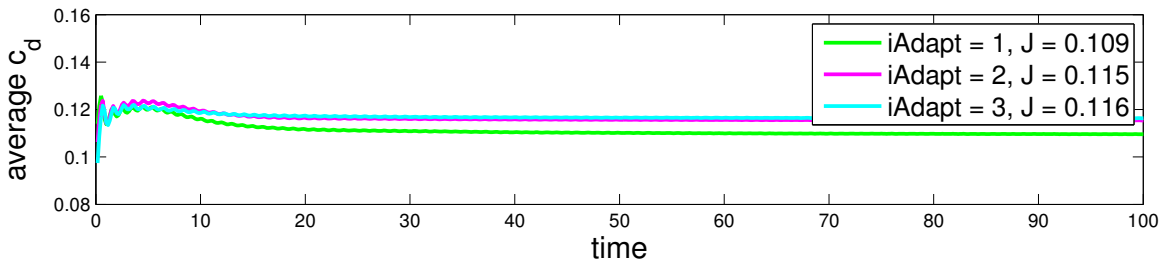


Figure 11. NACA 0012,  $M = 0.2$ ,  $Re = 10^4$ ,  $\alpha = 8^\circ$ : convergence of outputs versus degrees of freedom (spatial only) for uniform order refinement and order adaptations using two adaptive fractions.

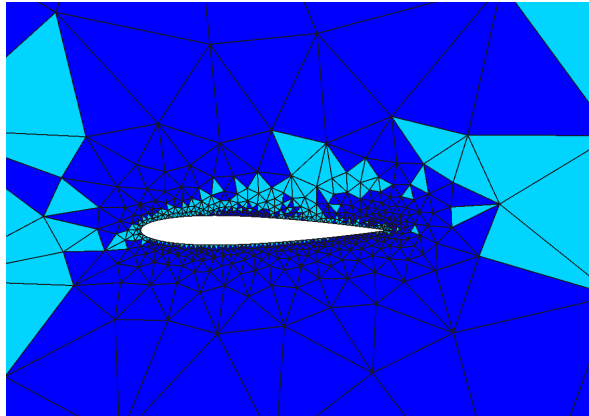


(a) Drag coefficient time histories

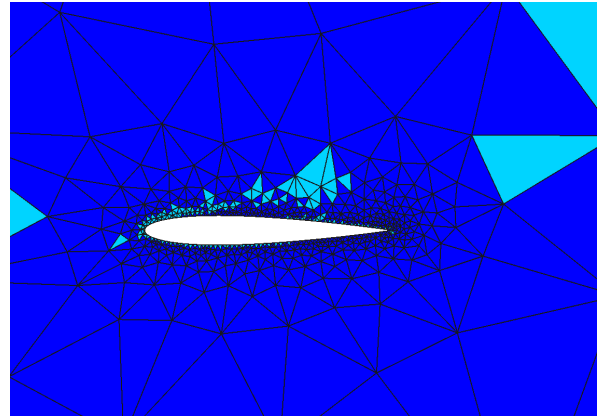


(b) Convergence of the average drag coefficient

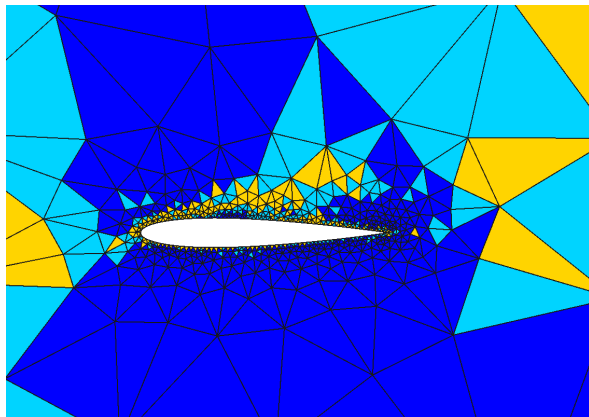
Figure 12. NACA 0012,  $M = 0.2$ ,  $Re = 10^4$ ,  $\alpha = 8^\circ$ : time histories and average drag coefficient convergence for adaptive order refinement with  $f^{\text{adapt}} = 0.2$ ,  $N_w = 1$ .



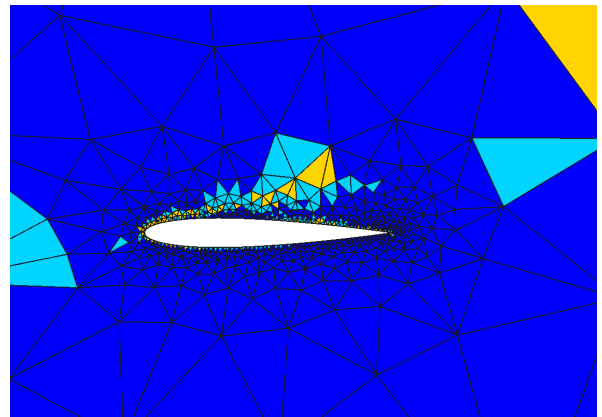
(a)  $f^{\text{adapt}} = 0.2$ , adapt iteration 1



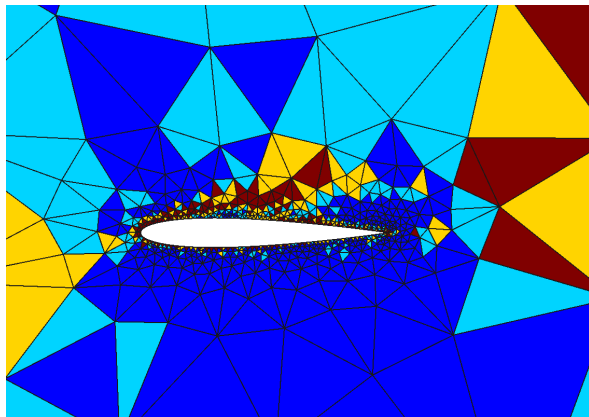
(b)  $f^{\text{adapt}} = 0.1$ , adapt iteration 1



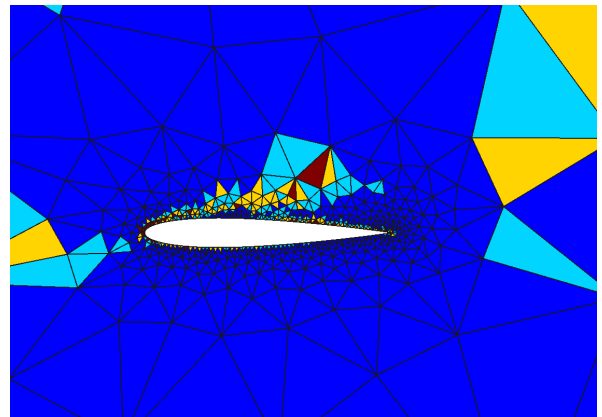
(c)  $f^{\text{adapt}} = 0.2$ , adapt iteration 2



(d)  $f^{\text{adapt}} = 0.1$ , adapt iteration 2



(e)  $f^{\text{adapt}} = 0.2$ , adapt iteration 3



(f)  $f^{\text{adapt}} = 0.1$ , adapt iteration 3

Figure 13. NACA 0012,  $M = 0.2$ ,  $Re = 10^4$ ,  $\alpha = 8^\circ$ : approximation orders for output-adapted meshes at three adaptive iterations, using one window,  $N_w = 1$ , of time length 3.



## VI. Conclusions

This paper presents two methods for estimating output error due to finite spatial and temporal discretizations for unsteady chaotic systems. The outputs of interest are statistical quantities, which are well-defined for sufficiently-long time integration. However, traditional output error estimation methods based on adjoints fail for such systems due to the high sensitivity of outputs to initial conditions and pointwise residual sources, a problem that manifests itself numerically via a divergent adjoint. The two presented approaches avoid this problem by, respectively, (1) leveraging existing work on a least-squares shadowing approach for computing the adjoint solution, and (2) employing adjoint solutions on time intervals that are sufficiently short to prevent adjoint divergence but sufficiently long to obtain usable statistics. The least-squares shadowing approach yields accurate error estimates even for short simulation times; that is, even before the outputs statistically converge. This partially mitigates the main downside of the approach, which is its high computational expense. On the other hand, the time-windowing approach provides only approximate output error estimates. However, it is relatively cheap computationally, and the error estimates appear to provide adequate error localization for adaptation. In future work will consider methods for reducing the cost of LSS error estimation, and for extending time windowing to chaotic problems with a range of temporal scales.

## Acknowledgments

This work was supported in part by funding from The Boeing Company, with technical monitor Dr. Mori Mani.

## References

- <sup>1</sup>Larsson, J. and Wang, Q., “The prospect of using large eddy and detached eddy simulations in engineering design, and the research required to get there,” *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, Vol. 372, No. 2022, 2014, pp. 20130329.
- <sup>2</sup>Pope, S. B., “Ten questions concerning the large-eddy simulation of turbulent flows,” *New Journal of Physics*, Vol. 6, No. 1, 2004, pp. 35.
- <sup>3</sup>Fidkowski, K. J. and Darmofal, D. L., “Review of Output-Based Error Estimation and Mesh Adaptation in Computational Fluid Dynamics,” *American Institute of Aeronautics and Astronautics Journal*, Vol. 49, No. 4, 2011, pp. 673–694.
- <sup>4</sup>Pierce, N. A. and Giles, M. B., “Adjoint recovery of superconvergent functionals from PDE approximations,” *SIAM Review*, Vol. 42, No. 2, 2000, pp. 247–264.
- <sup>5</sup>Becker, R. and Rannacher, R., “An optimal control approach to a posteriori error estimation in finite element methods,” *Acta Numerica*, edited by A. Iserles, Cambridge University Press, 2001, pp. 1–102.
- <sup>6</sup>Hartmann, R. and Houston, P., “Adaptive discontinuous Galerkin finite element methods for the compressible Euler equations,” *Journal of Computational Physics*, Vol. 183, No. 2, 2002, pp. 508–532.
- <sup>7</sup>Venditti, D. A. and Darmofal, D. L., “Anisotropic grid adaptation for functional outputs: application to two-dimensional viscous flows,” *Journal of Computational Physics*, Vol. 187, No. 1, 2003, pp. 22–46.
- <sup>8</sup>Nemec, M. and Aftosmis, M. J., “Error Estimation and Adaptive Refinement for Embedded-Boundary Cartesian Meshes,” AIAA Paper 2007-4187, 2007.
- <sup>9</sup>Mani, K. and Mavriplis, D. J., “Discrete adjoint based time-step adaptation and error reduction in unsteady flow problems,” AIAA Paper 2007-3944, 2007.
- <sup>10</sup>Mani, K. and Mavriplis, D. J., “Error Estimation and Adaptation for Functional Outputs in Time-Dependent Flow Problems,” *Journal of Computational Physics*, Vol. 229, 2010, pp. 415–440.
- <sup>11</sup>Barth, T. J., “Space-Time Error Representation and Estimation in Navier-Stokes Calculations,” *Complex Effects in Large Eddy Simulations*, edited by S. C. Kassinos, C. A. Langer, G. Iaccarino, and P. Moin, Springer Berlin Heidelberg, Lecture Notes in Computational Science and Engineering Vol 26, 2007, pp. 29–48.
- <sup>12</sup>Meidner, D. and Vexler, B., “Adaptive Space-Time Finite Element Methods for Parabolic Optimization Problems,” *SIAM Journal on Control Optimization*, Vol. 46, No. 1, 2007, pp. 116–142.

- <sup>13</sup>Fidkowski, K. J. and Luo, Y., “Output-based Space-Time Mesh Adaptation for the Compressible Navier-Stokes Equations,” *Journal of Computational Physics*, Vol. 230, 2011, pp. 5753–5773.
- <sup>14</sup>Fidkowski, K. J., “An Output-Based Dynamic Order Refinement Strategy for Unsteady Aerodynamics,” AIAA Paper 2012-77, 2012.
- <sup>15</sup>Krakos, J. A., *Unsteady adjoint analysis for output sensitivity and mesh adaptation*, Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, 2012.
- <sup>16</sup>Belme, A., Dervieux, A., and Alauzet, F., “Error Estimation and Adaptation for Functional Outputs in Time-Dependent Flow Problems,” *Journal of Computational Physics*, Vol. 231, 2012, pp. 6323–6348.
- <sup>17</sup>Flynt, B. T. and Mavriplis, D. J., “Discrete Adjoint Based Adaptive Error Control in Unsteady Flow Problems,” AIAA Paper 2012-0078, 2012.
- <sup>18</sup>Schmich, M. and Vexler, B., “Adaptivity with Dynamic Meshes for Space-Time Finite Element Discretizations of Parabolic Equations,” *SIAM Journal on Scientific Computing*, Vol. 30, No. 1, 2008, pp. 369–393.
- <sup>19</sup>Kast, S. M., Fidkowski, K. J., and Roe, P. L., “An Unsteady Entropy Adjoint Approach for Adaptive Solution of the Shallow-Water Equations,” AIAA Paper 2011-3694, 2011.
- <sup>20</sup>Kast, S. M., Ceze, M. A., and Fidkowski, K. J., “Output-Adaptive Solution Strategies for Unsteady Aerodynamics on Deformable Domains,” Seventh International Conference on Computational Fluid Dynamics ICCFD7-3802, 2012.
- <sup>21</sup>Kast, S. M. and Fidkowski, K. J., “Output-based Mesh Adaptation for High Order Navier-Stokes Simulations on Deformable Domains,” *Journal of Computational Physics*, Vol. 252, No. 1, 2013, pp. 468–494.
- <sup>22</sup>Fidkowski, K. J., “An Output-Based Adaptive Hybridized Discontinuous Galerkin Method on Deforming Domains,” AIAA Paper 2015-2602, 2015.
- <sup>23</sup>Nguyen, N., Peraire, J., and Cockburn, B., “An Implicit High-Order Hybridizable Discontinuous Galerkin Method for Linear Convection-Diffusion Equations,” *Journal of Computational Physics*, Vol. 228, 2009, pp. 3232–3254.
- <sup>24</sup>Cockburn, B., Gopalakrishnan, J., and Lazarov, R., “Unified hybridization of discontinuous Galerkin, mixed, and continuous Galerkin methods for second order elliptic problems,” *SIAM Journal on Numerical Analysis*, Vol. 47, No. 2, 2009, pp. 1319–1365.
- <sup>25</sup>Nguyen, N. C., Peraire, J., and Cockburn, B., “Hybridizable Discontinuous Galerkin Methods,” *Spectral and High Order Methods for Partial Differential Equations*, edited by J. S. Hesthaven, E. M. Rnquist, T. J. Barth, M. Griebel, D. E. Keyes, R. M. Nieminen, D. Roose, and T. Schlick, Vol. 76 of *Lecture Notes in Computational Science and Engineering*, Springer Berlin Heidelberg, 2011, pp. 63–84, 10.1007/978-3-642-15337-2\_4.
- <sup>26</sup>Woopen, M., Balan, A., May, G., and Schütz, J., “A comparison of hybridized and standard DGmethods for target-based hp-adaptive simulation of compressible flow,” *Computers & Fluids*, Vol. 98, 2014, pp. 3–16.
- <sup>27</sup>Dahm, J. P. and Fidkowski, K. J., “Error Estimation and Adaptation in Hybridized Discontinuous Galerkin Methods,” AIAA Paper 2014-0078, 2014.
- <sup>28</sup>Kast, S. M., Dahm, J. P., and Fidkowski, K. J., “Optimal test functions for boundary accuracy in discontinuous finite element methods,” *Journal of Computational Physics*, Vol. 298, No. 1, 2015, pp. 360 – 386.
- <sup>29</sup>Fidkowski, K. J., “Output error estimation strategies for discontinuous Galerkin discretizations of unsteady convection-dominated flows,” *International Journal for Numerical Methods in Engineering*, Vol. 88, No. 12, 2011, pp. 1297–1322.
- <sup>30</sup>Ghosal, S., “An analysis of numerical errors in large-eddy simulations of turbulence,” *Journal of Computational Physics*, Vol. 125, No. 1, 1996, pp. 187–206.
- <sup>31</sup>Kravchenko, A. and Moin, P., “On the effect of numerical errors in large eddy simulations of turbulent flows,” *Journal of Computational Physics*, Vol. 131, No. 2, 1997, pp. 310–322.
- <sup>32</sup>Wang, Q., “Forward and adjoint sensitivity computation of chaotic dynamical systems,” *Journal of Computational Physics*, Vol. 235, 2013, pp. 1–13.
- <sup>33</sup>Johnson, C., “On Computability and Error Control in CFD,” *International Journal for Numerical Methods in Fluids*, Vol. 20, 1995, pp. 777–788.
- <sup>34</sup>Lorenz, E. N., “Deterministic nonperiodic flow,” *Journal of the atmospheric sciences*, Vol. 20, No. 2, 1963, pp. 130–141.
- <sup>35</sup>Richter, T., “Discontinuous Galerkin as Time-Stepping Scheme for the Navier-Stokes Equations,” *Fourth International Conference on High Performance Scientific Computing Modeling, Simulation and Optimization of Complex Processes*, Hanoi, Vietnam, 2009.
- <sup>36</sup>Wang, Q., Hu, R., and Blonigan, P., “Least Squares Shadowing sensitivity analysis of chaotic limit cycle oscillations,” *Journal of Computational Physics*, Vol. 267, 2014, pp. 210–224.
- <sup>37</sup>Wang, Q., “Convergence of the least squares shadowing method for computing derivative of ergodic averages,” *SIAM Journal on Numerical Analysis*, Vol. 52, No. 1, 2014, pp. 156–170.

<sup>38</sup>Blonigan, P. J., Gomez, S. A., and Wang, Q., “Least Squares Shadowing for Sensitivity Analysis of Turbulent Fluid Flows,” AIAA Paper 2014-1426, 2014.

<sup>39</sup>Hyman, J. and Nicolaenko, B., “The Kuramoto-Sivashinsky equation: A bridge between PDE’s and dynamical systems,” *Physica D: Nonlinear Phenomena*, Vol. 18, 1986, pp. 113–126.

<sup>40</sup>Wang, Q., Gomez, S. A., Blonigan, P. J., Gregory, A. L., and Qian, E. Y., “Towards scalable parallel-in-time turbulent flow simulations,” *Physics of Fluids (1994-present)*, Vol. 25, No. 11, 2013, pp. 110818.

<sup>41</sup>Georgoulis, E. H., Houston, P., and Virtanen, J., “An a posteriori error indicator for discontinuous Galerkin approximations of fourth-order elliptic problems,” *IMA journal of numerical analysis*, 2009, pp. drp023.