

Output-Based Space-Time Mesh Adaptation for Unsteady Aerodynamics

Yuxing Luo* and Krzysztof J. Fidkowski†

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

An adjoint-based output error estimation algorithm is presented for unsteady problems discretized on static meshes with a space-time discontinuous Galerkin finite element method. An approximate factorization technique is used to solve both the forward and the discrete adjoint problems. A space-time anisotropy measure based on projection of the adjoint solution is used to attribute the error to spatial or temporal resolution. This measure drives a fixed-growth adaptive strategy that employs hanging-node refinement in the spatial domain and slab bisection in the temporal domain. Adaptive results for convection-dominated flows in two dimensions, including those governed by the compressible Navier-Stokes equations, demonstrate the effectivity of the output error estimate and the degree-of-freedom benefits of output-based adaptation compared to uniform space-time refinement and to cheaper heuristic indicators.

I. Introduction

Output-based adaptive methods in aerospace applications are sought for their ability to produce accurate answers with efficient use of resources. These methods not only guide adaptation, but also return output error estimates that improve robustness of the calculations. While the topic of output error estimation and mesh adaptation has been studied in depth for steady problems, unsteady problems have received considerably less attention. Granted, output error estimation is not yet mature even for steady simulations, and unsteady problems only add complexity and computational expense. However, many aerospace applications rely on unsteady simulations, and these often contain a wide range of spatial *and* temporal scales whose distribution is not known a priori. In such simulations, the risk of unquantified output errors is significant, as is the potential cost benefit of output adaptive methods that make efficient use of resources.

Numerous authors have studied output-based adaptive methods for steady-state problems.¹⁻⁷ Output error indicators specifically target for refinement areas of the computational domain that are important for predicting the output of interest. They are popular for aerospace applications because they properly account for error propagation effects through the use of adjoint solutions. While in the steady case the cost of the adjoint solution is generally no more than that of the original forward problem, the cost generally rises in the unsteady case, especially in terms of storage for nonlinear problems. However, the potential benefits of adjoint-based adaptation also rise with the inclusion of time as another dimension for adaptation.

*Research Assistant

†Assistant Professor, AIAA Member

Since the theoretical framework for output error estimation is most rigorous in a variational formulation, this work employs a space-time finite element method. Specifically, the discontinuous Galerkin (DG) method is used in both space and time. Such a discretization has been studied previously,^{8–13} and although computationally expensive, it shows potential for high accuracy and flexibility in the solution space. The latter point is important for mesh adaptation, which may require hanging nodes or variation in order throughout the space-time domain.

Previous works have investigated output error estimation and mesh adaptation for unsteady simulations. For a finite volume method with a backwards-difference time discretization and dynamic meshes, Mani and Mavriplis estimate the output error due to both the temporal resolution and to partial convergence of the unsteady residuals.^{14,15} They then adaptively refine the time steps and demonstrate an improvement in functional convergence compared to uniform refinement. For a space-time DG discretization, Barth outlines an adjoint-based error estimation procedure on static unstructured meshes.¹² He shows that the error indicator is effective at estimating the true error, and he demonstrates an adaptive procedure for the spatial-mesh. Vexler *et al*^{16,17} study output error estimation and mesh adaptation for parabolic problems discretized using continuous Galerkin in space and discontinuous Galerkin in time. By employing high-order reconstructions of the adjoint in space and time, they obtain separate spatial and temporal error estimates, and they use these error estimates to drive adaptive simulations of scalar and two-equation parabolic systems.

The present work builds on previous studies by combining spatial and temporal adaptation for problems of aerodynamic interest. A critical component to the success of the method is the decision between spatial and temporal refinement, which at present is made based on projection of the fine adjoint solution onto semi-coarsened spatial and temporal spaces. This paper presents the space-time discretization of the forward and adjoint problems, the output error estimation procedure, and the mesh adaptation algorithm for spatially-static meshes. Results show the efficacy of the adaptive approach for an unsteady convection-diffusion-reaction problem and an impulsively-started airfoil.

II. Discretization and Solution

A. Forward Discretization

In this work we consider unsteady, convection-dominated flows governed by a system of partial differential equations,

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{r}(\mathbf{u}) = 0, \quad (1)$$

where $\mathbf{u}(\mathbf{x}, t) \in \mathbb{R}^s$ is the state vector defined at every point in space, $\mathbf{x} \in \mathbb{R}^d$, and time, $t \in \mathbb{R}$. $\mathbf{r} : \mathbb{R}^s \rightarrow \mathbb{R}^s$ is a spatial operator, s is the number of governing equations, and $d = 1, 2$, or 3 is the spatial dimension.

We discretize Eqn. 1 in space and time using a discontinuous finite element method on a tensor product space-time mesh, as illustrated in Figure 1 for one spatial dimension. An approximate solution, $\mathbf{u}_H(\mathbf{x}, t)$, is sought in the finite-dimensional space \mathcal{V}_H where $\mathcal{V}_H = \mathcal{V}_H^{\text{space}} \otimes \mathcal{V}_H^{\text{time}}$. The spatial approximation space is $\mathcal{V}_H^{\text{space}} = [\mathcal{V}_H^{\text{space}}]^s$, where $\mathcal{V}_H^{\text{space}}$ is N_H -dimensional and consists of polynomials of order p in the reference spatial coordinates on each element. The temporal approximation space $\mathcal{V}_H^{\text{time}}$ is N_H^{time} -dimensional and consists of polynomials of order r in time on each slab. These spaces admit discontinuities across element/time-slab interfaces.

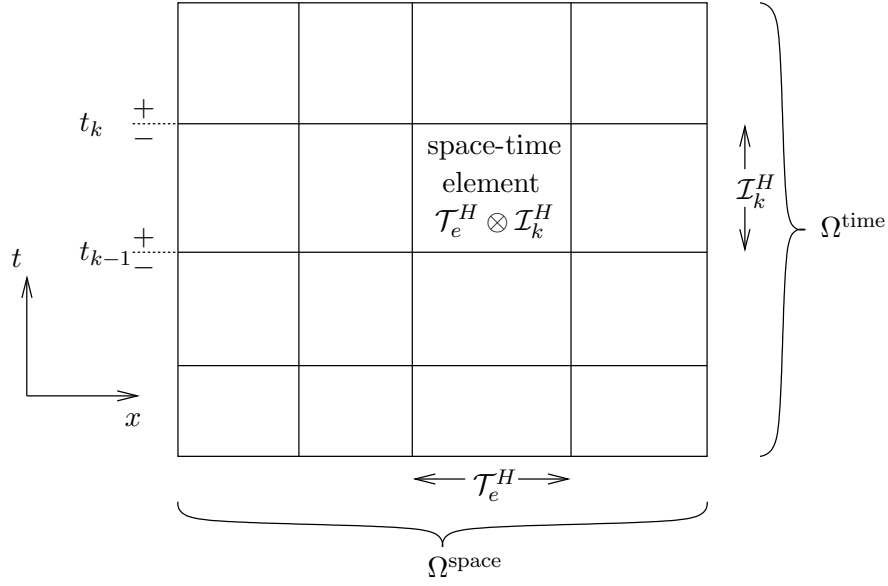


Figure 1. Sample mesh for a space-time discretization in one spatial dimension. Each space time element, $\mathcal{T}_e^H \otimes \mathcal{I}_k^H$, is given by a tensor product of a spatial element, \mathcal{T}_e^H , $e \in [1 \dots N_{H,\text{elem}}]$, and a time slab, \mathcal{I}_k^H , $k \in [1 \dots N_{H,\text{slab}}]$.

The approximate solution is expanded as

$$\mathbf{u}_H(\mathbf{x}, t) = \mathbf{u}_{H,j}^n \phi_{H,j}(\mathbf{x}) \varphi_H^n(t), \quad (2)$$

where $\phi_{H,j}(\mathbf{x})$, $j \in [1 \dots N_H]$, are basis functions for $\mathcal{V}_H^{\text{space}}$, $\varphi_H^n(t)$, $n \in [1 \dots N_H^{\text{time}}]$, are basis functions for $\mathcal{V}_H^{\text{time}}$, and $\mathbf{u}_{H,j}^n \in \mathbb{R}^s$ are expansion coefficients for the state. Summation is implied on the repeated indices. Useful will be the semi-discrete expressions

$$\mathbf{u}_H(\mathbf{x}, t) = \mathbf{u}_{H,j}(t) \phi_{H,j}(\mathbf{x}) = \mathbf{u}_H^n(\mathbf{x}) \varphi_H^n(t),$$

where $\mathbf{u}_{H,j}(t) = \mathbf{u}_{H,j}^n \varphi_H^n(t)$ and $\mathbf{u}_H^n(\mathbf{x}) = \mathbf{u}_{H,j}^n \phi_{H,j}(\mathbf{x})$. The approximate finite element solution must satisfy the weak form

$$\bar{\mathcal{R}}_H(\mathbf{u}_H, \mathbf{v}_H) = 0, \quad \forall \mathbf{v}_H \in \mathcal{V}_H,$$

where $\bar{\mathcal{R}}_H(\cdot, \cdot) : \mathcal{V}_H \otimes \mathcal{V}_H \rightarrow \mathbb{R}$ is a semilinear form obtained from Eqn. 1 by multiplying by test functions \mathbf{v}_H and integrating over the space-time domain. Substituting the expansion from Eqn. 2, and a similar one for \mathbf{v}_H , into Eqn. 1 yields the following expression for the semilinear form,

$$\bar{\mathcal{R}}_H(\mathbf{u}_H, \mathbf{v}_H) = \sum_{k=1}^{N_{H,\text{slab}}} \int_{\mathcal{I}_k^H} \left[\mathbf{v}_{H,i}^T(t) \frac{d\mathbf{u}_{H,j}(t)}{dt} \int_{\Omega^{\text{space}}} \phi_{H,j}(\mathbf{x}) \phi_{H,i}(\mathbf{x}) d\Omega + \mathcal{R}_H(\mathbf{u}_H, \mathbf{v}_{H,i}(t) \phi_{H,i}(\mathbf{x})) \right] dt,$$

where $N_{H,\text{slab}}$ is the number of time slabs, the superscript T denotes the transpose, and $\mathcal{R}_H(\cdot, \cdot) : \mathcal{V}_H^{\text{space}} \otimes \mathcal{V}_H^{\text{space}} \rightarrow \mathbb{R}$ is the semilinear form associated with the spatial discretization. The spatial discretization is a standard discontinuous Galerkin formulation, employing the second form of Bassi and Rebay¹⁸ for viscous fluxes and, in the case of compressible gas dynamics, the Roe linearization¹⁹

for convective fluxes. For compactness of notation, we now unroll all the degrees of freedom associated with $\mathbf{u}_{H,j}(t)$ into one vector, $\mathbf{U}_H(t) \in \mathbb{R}^{N_{Hs}}$. Doing the same for $\mathbf{v}_{H,i}(t)$, we obtain

$$\bar{\mathbf{R}}_H(\mathbf{u}_H, \mathbf{v}_H) = \sum_{k=1}^{N_{H,\text{slab}}} \int_{\mathcal{I}_k^H} \mathbf{V}_H^T(t) \left[\mathbf{M}_H \frac{d\mathbf{U}_H(t)}{dt} + \mathbf{R}_H(\mathbf{U}_H(t)) \right] dt = 0, \quad (3)$$

where $\mathbf{M}_H \in \mathbb{R}^{N_{Hs} \times N_{Hs}}$ is the spatial mass matrix and $\mathbf{R}_H \in \mathbb{R}^{N_{Hs}}$ is the spatial residual vector.

Since Eqn. 3 has to be satisfied for all test functions $\mathbf{V}_H(t)$, it represents a system of N_{Hs} ordinary differential equations in time. The discontinuous Galerkin temporal discretization is obtained by using $\varphi^m(t)$, $m \in [1 \dots N_H^{\text{time}}]$ in the temporal test functions. After an integration by parts, Eqn. 3 on each time slab becomes

$$- \int_{\mathcal{I}_k^H} \mathbf{M}_H \mathbf{U}_H \frac{d\varphi_H^m}{dt} dt + \mathbf{M}_H \mathbf{U}_H(t_k^-) \varphi_H^m(t_k) - \mathbf{M}_H \mathbf{U}_H(t_{k-1}^-) \varphi_H^m(t_{k-1}) + \int_{\mathcal{I}_k^H} \varphi_H^m \mathbf{R}_H(\mathbf{U}_H) dt = 0,$$

where t_k^- indicates the end of time slab \mathcal{I}_k^H , and where we have used a natural ‘‘upwind’’ flux to define the state at the time slab interfaces. The sum over time slabs is not necessary because the $\varphi^m(t)$ functions each have support over only one time slab. Substituting the temporal expansion $\mathbf{U}_H(t) = \mathbf{U}_H^n \varphi_H^n(t)$ into the above equation yields the algebraic system

$$\bar{\mathbf{R}}_H^m(\mathbf{U}_H^n) = 0, \quad (4)$$

where $\bar{\mathbf{R}}_H^m \in \mathbb{R}^{N_{Hs}}$ is the unsteady residual vector associated with basis function $\varphi^m(t)$.

For example, in the case of Lagrange polynomials in time of order $r = 1$, the solution on the first time slab is expanded as

$$\mathbf{U}_H(t) = \mathbf{U}_H^1 \varphi_H^1(t) + \mathbf{U}_H^2 \varphi_H^2(t), \quad \varphi_H^1(t) = \frac{t_1 - t}{t_1 - t_0}, \quad \varphi_H^2(t) = \frac{t - t_0}{t_1 - t_0},$$

and the two unsteady residual vectors on the first time slab are

$$\begin{aligned} \bar{\mathbf{R}}_H^1 &= \frac{\mathbf{M}_H}{2} (\mathbf{U}_H^1 + \mathbf{U}_H^2) - \mathbf{M} \mathbf{U}_H^0 + \int_{t_0}^{t_1} \varphi_H^1(t) \mathbf{R}_H(\mathbf{U}_H(t)) dt, \\ \bar{\mathbf{R}}_H^2 &= \frac{\mathbf{M}_H}{2} (-\mathbf{U}_H^1 + \mathbf{U}_H^2) + \int_{t_0}^{t_1} \varphi_H^2(t) \mathbf{R}_H(\mathbf{U}_H(t)) dt, \end{aligned} \quad (5)$$

where \mathbf{U}_H^0 is the prescribed initial condition. For all other time slabs, this vector will be the solution from the end of the previous time slab. In the case of Lagrange polynomials of order $r = 2$, the three unsteady residual vectors on the first time slab are

$$\begin{aligned} \bar{\mathbf{R}}_H^1 &= \mathbf{M}_H \left(\frac{1}{2} \mathbf{U}_H^1 + \frac{2}{3} \mathbf{U}_H^2 - \frac{1}{6} \mathbf{U}_H^3 \right) - \mathbf{M} \mathbf{U}_H^0 + \int_{t_0}^{t_1} \varphi_H^1(t) \mathbf{R}_H(\mathbf{U}_H(t)) dt, \\ \bar{\mathbf{R}}_H^2 &= \mathbf{M}_H \left(-\frac{2}{3} \mathbf{U}_H^1 + \frac{2}{3} \mathbf{U}_H^3 \right) + \int_{t_0}^{t_1} \varphi_H^2(t) \mathbf{R}_H(\mathbf{U}_H(t)) dt, \\ \bar{\mathbf{R}}_H^3 &= \mathbf{M}_H \left(\frac{1}{6} \mathbf{U}_H^1 - \frac{2}{3} \mathbf{U}_H^2 + \frac{1}{2} \mathbf{U}_H^3 \right) + \int_{t_0}^{t_1} \varphi_H^3(t) \mathbf{R}_H(\mathbf{U}_H(t)) dt. \end{aligned} \quad (6)$$

B. Discrete Unsteady Adjoint

Consider a scalar output that is a function of the unsteady state vector,

$$\text{output} = \mathcal{J}_H(\mathbf{u}_H(x, t)) = J_H(\mathbf{U}_H^n),$$

where $\mathcal{J}_H(\cdot) : \mathcal{V}_H \rightarrow \mathbb{R}$ and $J_H(\cdot) : \mathbb{R}^{N_{Hs}} \rightarrow \mathbb{R}$ are respectively the variational and discrete representations of the output. Once a basis for \mathcal{V}_H is chosen, these two representations are interchangeable. The discrete adjoint equation associated with the output and with the algebraic system in Eqn. 4 is

$$\underbrace{\left(\frac{\partial \bar{\mathbf{R}}_H^m}{\partial \mathbf{U}_H^n} \right)^T \boldsymbol{\Psi}_H^m + \left(\frac{\partial J_H}{\partial \mathbf{U}_H^n} \right)^T}_{\bar{\mathbf{R}}_H^{\psi, n}(\boldsymbol{\Psi}_H^m)} = 0, \quad (7)$$

where the linearization of the unsteady residual is computed about the forward solution for nonlinear problems.⁷ Assuming that the forward discretization and output definition are adjoint consistent, the discrete adjoint solution, $\boldsymbol{\Psi}_H^m \in \mathbb{R}^{N_{Hs}}$, gives the coefficients in a discrete representation of a function $\psi_H \in \mathcal{V}_H$ that approximates the solution to the continuous adjoint problem.

$\bar{\mathbf{R}}_H^{\psi, n}(\boldsymbol{\Psi}_H^m) \in \mathbb{R}^{N_{Hs}}$, $n \in [1 \dots N_H^{\text{time}}]$, are the unsteady adjoint residual vectors. Using an $r = 1$ Lagrange basis, on the first time slab these are

$$\begin{aligned} \bar{\mathbf{R}}_H^{\psi, 1} &= \frac{\mathbf{M}_H}{2}(\boldsymbol{\Psi}_H^1 - \boldsymbol{\Psi}_H^2) + \int_{t_0}^{t_1} \varphi_H^1 \left(\frac{\partial \mathbf{R}}{\partial \mathbf{U}}(\mathbf{U}_H(t)) \right)^T \boldsymbol{\Psi}_H(t) dt + \left(\frac{\partial J_H}{\partial \mathbf{U}_H^1} \right)^T, \\ \bar{\mathbf{R}}_H^{\psi, 2} &= \frac{\mathbf{M}_H}{2}(\boldsymbol{\Psi}_H^1 + \boldsymbol{\Psi}_H^2) - \mathbf{M}_H \boldsymbol{\Psi}_H^{r+2} + \int_{t_0}^{t_1} \varphi_H^2 \left(\frac{\partial \mathbf{R}}{\partial \mathbf{U}}(\mathbf{U}_H(t)) \right)^T \boldsymbol{\Psi}_H(t) dt + \left(\frac{\partial J_H}{\partial \mathbf{U}_H^2} \right)^T, \end{aligned} \quad (8)$$

where $\boldsymbol{\Psi}_H(t) = \boldsymbol{\Psi}_H^1 \varphi_H^1(t) + \boldsymbol{\Psi}_H^2 \varphi_H^2(t)$, and $\boldsymbol{\Psi}_H^{r+2}$ is the adjoint vector associated with the first basis function on the second time slab. Using an $r = 2$ Lagrange basis, the three adjoint residuals are

$$\begin{aligned} \bar{\mathbf{R}}_H^{\psi, 1} &= \mathbf{M}_H \left(\frac{1}{2} \boldsymbol{\Psi}_H^1 - \frac{2}{3} \boldsymbol{\Psi}_H^2 + \frac{1}{6} \boldsymbol{\Psi}_H^3 \right) + \int_{t_0}^{t_1} \varphi_H^1 \left(\frac{\partial \mathbf{R}}{\partial \mathbf{U}}(\mathbf{U}_H(t)) \right)^T \boldsymbol{\Psi}_H(t) dt + \left(\frac{\partial J_H}{\partial \mathbf{U}_H^1} \right)^T, \\ \bar{\mathbf{R}}_H^{\psi, 2} &= \mathbf{M}_H \left(\frac{2}{3} \boldsymbol{\Psi}_H^1 - \frac{2}{3} \boldsymbol{\Psi}_H^2 \right) + \int_{t_0}^{t_1} \varphi_H^2 \left(\frac{\partial \mathbf{R}}{\partial \mathbf{U}}(\mathbf{U}_H(t)) \right)^T \boldsymbol{\Psi}_H(t) dt + \left(\frac{\partial J_H}{\partial \mathbf{U}_H^2} \right)^T, \\ \bar{\mathbf{R}}_H^{\psi, 3} &= \mathbf{M}_H \left(-\frac{1}{6} \boldsymbol{\Psi}_H^1 + \frac{2}{3} \boldsymbol{\Psi}_H^2 + \frac{1}{2} \boldsymbol{\Psi}_H^3 \right) - \mathbf{M}_H \boldsymbol{\Psi}_H^{r+2} \\ &\quad + \int_{t_0}^{t_1} \varphi_H^3 \left(\frac{\partial \mathbf{R}}{\partial \mathbf{U}}(\mathbf{U}_H(t)) \right)^T \boldsymbol{\Psi}_H(t) dt + \left(\frac{\partial J_H}{\partial \mathbf{U}_H^3} \right)^T. \end{aligned} \quad (9)$$

To obtain the exact discrete adjoint, the integrals in the above equations are evaluated using the same numerical quadrature as in Eqns. 5 and 6.

C. Approximate Factorization Solution

Eqn. 4 is an algebraic and generally nonlinear system in which all of the degrees of freedom associated with a time slab are coupled together. We use a quasi-Newton method to solve this system,

where on each time slab the solution is first initialized to the state from the end of the previous time slab. The linear update equation on slab k is given by

$$\frac{\partial \bar{\mathbf{R}}_H^m}{\partial \mathbf{U}_H^n} \Delta \mathbf{U}_H^n = -\bar{\mathbf{R}}_H^m, \quad (10)$$

where m and n are the temporal indices associated with slab k . The sparsity pattern of the unsteady residual Jacobian matrix is shown in Figure 2 for $r = 1$. Time slabs are coupled together through the upwind temporal flux, and for a Lagrange basis only the final node on a time slab affects the next slab.

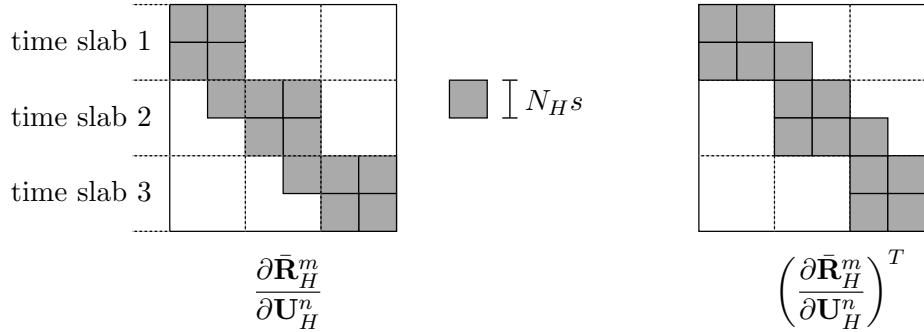


Figure 2. Sparsity patterns of the unsteady residual Jacobian matrix and its transpose for a Lagrange temporal basis of order $r = 1$.

Solving Eqn. 10 requires inverting a system of size $(r + 1)N_{Hs}$, i.e. with $(r + 1)$ times as many unknowns as in the steady problem. This can be prohibitive especially for large problems when the steady Jacobian matrix already taxes memory requirements. Hence, in this work, we employ an approximate factorization strategy introduced by Richter.¹³ For $r = 1$, this strategy relies on:

- Using a temporally constant spatial Jacobian matrix evaluated at the time slab midpoint in the linearization of the time integrals appearing in the unsteady residual:

$$\begin{aligned} \frac{\partial}{\partial \mathbf{U}_H^n} \int_{\mathcal{I}_k^H} \varphi_H^m(t) \mathbf{R}_H(\mathbf{U}_H(t)) dt &= \int_{\mathcal{I}_k^H} \varphi_H^m(t) \frac{\partial \mathbf{R}_H}{\partial \mathbf{U}_H} \Big|_{\mathbf{U}_H(t)} \varphi_H^n(t) dt \\ &\approx \underbrace{\frac{\partial \mathbf{R}_H}{\partial \mathbf{U}_H} \Big|_{\mathbf{U}_H^{k+1/2}}}_{\mathbf{A}_H} \int_{\mathcal{I}_k^H} \varphi_H^m(t) \varphi_H^n(t) dt, \end{aligned}$$

where $\mathbf{U}_H^{k+1/2}$ is the state at the midpoint of time slab k . The resulting linear update system for the first time slab is, with $\Delta t_1 = t_1 - t_0$,

$$\begin{aligned} \left[\frac{\mathbf{M}_H}{2} + \frac{\Delta t_1}{3} \mathbf{A}_H \right] \Delta \mathbf{U}_H^1 + \left[\frac{\mathbf{M}_H}{2} + \frac{\Delta t_1}{6} \mathbf{A}_H \right] \Delta \mathbf{U}_H^2 &= -\bar{\mathbf{R}}_H^1, \\ \left[-\frac{\mathbf{M}_H}{2} + \frac{\Delta t_1}{6} \mathbf{A}_H \right] \Delta \mathbf{U}_H^1 + \left[\frac{\mathbf{M}_H}{2} + \frac{\Delta t_1}{3} \mathbf{A}_H \right] \Delta \mathbf{U}_H^2 &= -\bar{\mathbf{R}}_H^2. \end{aligned} \quad (11)$$

- Employing an approximate factorization for the solution of the system in Eqn. 11, based on

$$\left[\mathbf{I}_H + \frac{2\Delta t_1}{3} \mathbf{M}_H^{-1} \mathbf{A}_H + \frac{\Delta t_1^2}{6} (\mathbf{M}_H^{-1} \mathbf{A}_H)^2 \right] \approx \left[\mathbf{I}_H + \frac{\Delta t_1}{\sqrt{6}} \mathbf{M}_H^{-1} \mathbf{A}_H \right]^2.$$

\mathbf{I}_H is the identity matrix, and the resulting three-step solution scheme is

$$\begin{aligned} \left[\mathbf{M}_H + \frac{\Delta t_1}{\sqrt{6}} \mathbf{A}_H \right] \mathbf{Y}_H &= -\bar{\mathbf{R}}_H^1 - \bar{\mathbf{R}}_H^2 - \frac{1}{3} \Delta t_1 \mathbf{A}_H \mathbf{M}_H^{-1} (2\mathbf{R}_H^2 - \mathbf{R}_H^1), \\ \left[\mathbf{M}_H + \frac{\Delta t_1}{\sqrt{6}} \mathbf{A}_H \right] \Delta \mathbf{U}_H^2 &= \mathbf{M}_H \mathbf{Y}_H, \\ \left[\mathbf{M}_H + \frac{2\Delta t_1}{3} \mathbf{A}_H \right] \Delta \mathbf{U}_H^1 &= -2\bar{\mathbf{R}}_H^1 - \left[\mathbf{M}_H + \frac{\Delta t_1}{3} \mathbf{A}_H \right] \Delta \mathbf{U}_H^1, \end{aligned}$$

where \mathbf{Y}_H is an intermediate vector. The approximate factorization bypasses the need for forming and inverting \mathbf{A}_H^2 . The above steps only require inversion of two systems, both with the same compact stencil as the steady discretization. Note that the presence of the approximation requires multiple Newton iterations even for linear systems. However, the approximation is quite accurate, with a deviation of less than 10% for the general eigenvalue problem.¹³

The same approximate factorization is applied to the solution of the adjoint system on each time slab, Eqn. 8 for $r = 1$. The only differences are in the use of \mathbf{A}_H^T in place of \mathbf{A}_H and in reversing the role of the first and second update vectors in the three-step scheme. In the present work we do not require solution of the system for $r = 2$, as only the residual for $r = 2$ will be used for output error estimation.

III. Output Error Estimation

The discretization discussed in the previous section yields a solution approximation, $\mathbf{u}_H(x, t)$, in the finite dimensional space \mathcal{V}_H . We are interested in the effect of the solution approximation on the error in a scalar output, $\mathcal{J}_H(\mathbf{u}_H(x, t))$. To make the error estimation tractable, we resolve to compare $\mathcal{J}_H(\mathbf{u}_H(x, t))$ to the output calculated from a ‘‘fine’’ solution, $\mathbf{u}_h(x, t)$, on a richer space \mathcal{V}_h . In a discontinuous Galerkin space-time discretization, the solution space can be enriched by increasing the interpolation order or refining the elements/time slabs. We only require that the coarse/fine spaces be nested, $\mathcal{V}_H \subset \mathcal{V}_h$.

A. The Adjoint-Weighted Residual

We define the output error as the difference between the output computed with the coarse versus fine solutions. If we have the adjoint solution on the fine space, $\boldsymbol{\psi}_h \in \mathcal{V}_h$, then the output error can be estimated using the adjoint-weighted residual technique,⁷

$$\delta J = \text{output error} = \mathcal{J}_H(\mathbf{u}_H) - \mathcal{J}_h(\mathbf{u}_h) \approx -\bar{\mathcal{R}}_h(\mathbf{u}_H, \boldsymbol{\psi}_h). \quad (12)$$

Note that the error estimation requires an evaluation of the fine-space unsteady residual associated with the coarse solution. The approximation sign indicates that the above expression is not exact for nonlinear problems as the fine-space adjoint is used instead of the mean-value adjoint.

The fully-discrete version of Eqn. 12 reads

$$\delta J \approx -(\boldsymbol{\Psi}_h^m)^T \bar{\mathbf{R}}_h^m(\mathbf{U}_h^{H,m}), \quad (13)$$

where $\mathbf{U}_h^{H,m}$ corresponds to the injection of the coarse solution into the fine space. Summation is implied on m , which ranges over the fine-space temporal degrees of freedom, $m \in [1 \dots N_h^{\text{time}}]$.

B. Implementation

The output error estimate in Eqn. 13 requires an adjoint solution on the fine space. The cost of obtaining this adjoint depends on the choice of the fine space \mathcal{V}_h and on the approximations employed in the adjoint solution. In this work, the fine space \mathcal{V}_h is obtained from the coarse space \mathcal{V}_H by enriching the order of the spatial approximation, $p \rightarrow p + 1$, and by enriching the order of the temporal approximation, $r \rightarrow r + 1$.

The output error estimate is computed during the adjoint solve, as described in the following solution outline:

1. Calculate the unsteady forward solution \mathbf{U}_H^n , $n \in [1 \dots N_H^{\text{time}}]$. Save the state on each time slab to disk.
2. Begin a loop backwards in time over the time slabs.
 - (a) Load the coarse solution from time slab \mathcal{I}_k^H .
 - (b) Inject the coarse solution into the fine space to obtain $\mathbf{U}_h^{H,m}$.
 - (c) Solve the fine adjoint problem using $p + 1$ and r , and reconstruct in time to order $r + 1$.
 - (d) Calculate the fine-space residual on the sub-slabs and form the inner product in Eqn. 13 for $m \in \text{dof}_h(\mathcal{I}_k^H)$, where $\text{dof}_h(\mathcal{I}_k^H)$ returns the fine-space degrees of freedom associated with time slab \mathcal{I}_k^H . Add to δJ .

The temporal reconstruction makes use of superconvergence properties of DG solutions at slab endpoints and Radau points.²⁰ Specifically, for an $r = 1$ coarse solution, the points used in the $r = 2$ reconstruction consist of the left node of the current time slab, the left node of the adjacent future time slab, and the root of the left Radau polynomial on the current time slab.

We note that solution checkpointing can be used to trade-off forward solution storage costs against computational time.²¹ We also note that the fine adjoint problem is linearized about the injected coarse forward solution, so that the fine forward solution \mathbf{U}_h^m is never required. This is an implementation choice that reduces the cost of the output error estimation. The impact of various choices of coarse versus fine spaces on the efficacy of the output error estimates will be addressed in future work.

IV. Space-Time Mesh Adaptation

The output error estimate drives an adaptive process in which the unsteady problem is solved on successively refined space-time meshes. The iterative process begins with a forward and adjoint solution on a coarse mesh. The output error is estimated using the adjoint-weighted residual method, and if the error is below a specified tolerance, the iterative process terminates. Otherwise, the output error is localized and the space-time mesh is refined, as described in this section.

A. Error Localization

The output error calculation in Eqn. 13 can be recast as a sum over space-time elements,

$$\delta J = \sum_{k=1}^{N_{\text{slab},H}} \sum_{e=1}^{N_{\text{elem},H}} \varepsilon_{k,e}, \quad (14)$$

where the error contribution of a coarse space-time element $\mathcal{T}_e^H \otimes \mathcal{I}_k^H$ is

$$\varepsilon_{k,e} = - \sum_{m \in \text{dof}_h(\mathcal{I}_k^H)} \sum_{i \in \text{dof}_h(\mathcal{T}_e^H)} (\psi_{h,i}^m)^T \bar{\mathbf{R}}_{h,i}^m (\mathbf{U}_h^{H,n}).$$

Note that both $\psi_{h,i}^m$ and $\bar{\mathbf{R}}_{h,i}^m$ are vectors in \mathbb{R}^s . $\bar{\mathbf{R}}_{h,i}^m$ is the unsteady residual associated with spatial degree of freedom i and temporal degree of freedom m . The error indicator for a space-time element is the absolute value of the elemental contribution,

$$\text{error indicator} = \epsilon_{k,e} = |\varepsilon_{k,e}|.$$

This indicator identifies the space-time elements most responsible for the error in the desired output. The adaptive strategy is to reduce the output error by refining these elements.

B. Space-Time Anisotropy and Adaptation

With a discontinuous Galerkin space-time discretization, the space-time elements could potentially be refined independently, following for example a fixed-fraction strategy in which elements with the highest error indicators are refined first. However, in the present work, we restrict the adaptation as follows. First, to allow for solution of the unsteady discrete problem via an approximate factorization, temporal refinement is limited to entire time slabs. Second, to minimize storage and complexity of the data structures, the refinement of the spatial mesh remains fixed throughout the unsteady simulation. This latter requirement is not a fundamental limitation and will be relaxed in future work.

The adaptive strategy must decide which elements and time steps to refine based on the error indicator, $\epsilon_{k,e}$. A simple strategy would be to marginalize the error indicator onto the spatial and temporal domains: the error indicator on time slab k would be $\sum_{e=1}^{N_{\text{elem},H}} \epsilon_{k,e}$, while the error indicator on element e would be $\sum_{k=1}^{N_{\text{slab},H}} \epsilon_{k,e}$. A fixed fraction of the elements and time slabs would then be refined. However such a marginalization does not account for the spatial versus temporal *source* of the error and can lead to under-performance of the adaptation. For example, if the discretization is under-resolved in the time-domain, spatial elements would needlessly be refined on account of the temporal error marginalized onto the spatial domain.

Key to an effective space-time adaptation strategy is attributing the error indicator to spatial versus temporal resolution using a measure of space-time anisotropy. In the present work, we employ two anisotropy measures. The first measure of space-time anisotropy is a heuristic based on inter-element jumps in the solution. For each space-time element, the average jump in the forward state, is computed across the spatial interfaces and across the time slab interfaces. For systems of equations, the average jumps of each state component are summed together. The relative magnitudes of the average spatial and temporal jumps are then used to calculate the anisotropy. This approach is motivated by the successful use of solution jumps in guiding anisotropic spatial refinement on quadrilateral and hexahedral elements in steady calculations.²² Specifically, for element (e, k) , we define the fraction of output error attributable to the spatial, respectively temporal, discretization as $\beta_{e,k}^{\text{space}}$, respectively $\beta_{e,k}^{\text{time}}$. For the inter-element jump measure, $\beta_{e,k}^{\text{time}}$ is defined as the ratio of the temporal jump to the sum of the temporal and spatial jumps, and $\beta_{e,k}^{\text{space}} = 1 - \beta_{e,k}^{\text{time}}$.

The second measure of space-time anisotropy is based on separate projections of the fine-space adjoint onto coarse spatial and temporal spaces. We define spatial and temporal output error

estimates as

$$\delta J^{\text{space}} \approx -\bar{\mathcal{R}}_h(\mathbf{u}_H, \boldsymbol{\psi}_{hH}), \quad \delta J^{\text{time}} \approx -\bar{\mathcal{R}}_h(\mathbf{u}_H, \boldsymbol{\psi}_{Hh}), \quad (15)$$

where $\boldsymbol{\psi}_{hH}$ and $\boldsymbol{\psi}_{Hh}$ are projections of the fine adjoint onto the coarse temporal and spatial spaces,

$$\boldsymbol{\psi}_{hH} = \Pi_H^{\text{space}} \boldsymbol{\psi}_h, \quad \boldsymbol{\psi}_{Hh} = \Pi_H^{\text{time}} \boldsymbol{\psi}_h.$$

Π_H^{space} and Π_H^{time} are least-squares projection operators from spatial order $p + 1$ to p and from temporal order $r + 1$ to r , respectively. These definitions are motivated by considering the spatial output error as the change in the output observed when the spatial discretization is refined while keeping the temporal discretization unchanged, and similarly for the temporal output error. The contributions to these quantities from space-time element (e, k) are

$$\varepsilon_{e,k}^{\text{space}} = - \sum_{m \in \text{dof}_h(\mathcal{I}_k^H)} (\mathbf{Z}_e \boldsymbol{\Psi}_{hH}^m)^T \mathbf{Z}_e \bar{\mathbf{R}}_h^m(\mathbf{U}_h^{H,n}), \quad \varepsilon_{e,k}^{\text{time}} = - \sum_{m \in \text{dof}_h(\mathcal{I}_k^H)} (\mathbf{Z}_e \boldsymbol{\Psi}_{Hh}^m)^T \mathbf{Z}_e \bar{\mathbf{R}}_h^m(\mathbf{U}_h^{H,n}),$$

where $\boldsymbol{\Psi}_{hH} \in \mathbb{R}^{N_h}$ and $\boldsymbol{\Psi}_{Hh} \in \mathbb{R}^{N_h}$ are discrete fine-space representations of the projected adjoints. The mask matrix \mathbf{Z}_e contains N_h columns and returns the degrees of freedom associated with coarse element e . In this measure, the spatial/temporal error fractions are given by

$$\beta_{e,k}^{\text{space}} = \frac{|\varepsilon_{e,k}^{\text{space}}|}{|\varepsilon_{e,k}^{\text{space}}| + |\varepsilon_{e,k}^{\text{time}}|}, \quad \beta_{e,k}^{\text{time}} = 1 - \beta_{e,k}^{\text{space}}.$$

Finally, aggregate adaptive indicators for each spatial element and for each time slab are obtained by summing over time slabs and elements as follows:

$$\begin{aligned} \text{spatial indicator on element } e &= \epsilon_e = \sum_{k=1}^{N_{\text{slab},H}} \epsilon_{k,e} \beta_{e,k}^{\text{space}}, \\ \text{temporal indicator on time slab } k &= \epsilon_k = \sum_{e=1}^{N_{\text{elem},H}} \epsilon_{k,e} \beta_{e,k}^{\text{time}}. \end{aligned} \quad (16)$$

This marginalization is illustrated in Figure 3. In the results we will also present conservative error estimates calculated as the sum of the element indicators:

$$\epsilon = \sum_{k=1}^{N_{\text{slab},H}} \sum_{e=1}^{N_{\text{elem},H}} \epsilon_{k,e}. \quad (17)$$

The adaptive strategy must decide which elements and time slabs to refine based on the localized error indicators ϵ_e and ϵ_k given in Eqn. 16. The mechanics of the adaptation are hanging-node refinement of spatial elements and bisection of time slabs. Other means of introducing degrees of freedom, such as order increments, will be investigated in future work. A fixed-growth strategy is used in which the increase in the number of space-time elements at every adaptive iteration is governed by a growth factor, f^{growth} . The budget of new space-time elements is approximate because of the tensor-product requirement of the space-time mesh, and because the hanging-node adaptation marks certain neighbors of flagged spatial elements to preserve a maximum factor of two refinement ratio between neighbors.

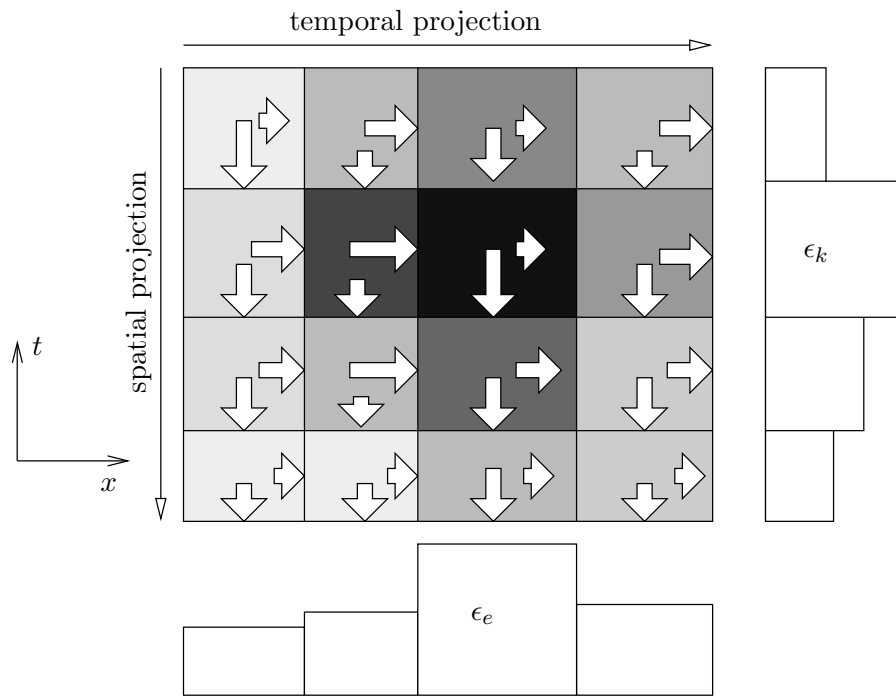


Figure 3. Marginalization of the space-time output error indicator to the spatial and temporal domains. The gray scale indicates magnitude of $\epsilon_{k,e}$. Arrow directions and sizes indicate fraction of error attributed to spatial resolution (down) versus to the temporal resolution (right). Bar graphs depict the distribution of the marginalized spatial and temporal error.

The decision of which elements or time slabs to refine is made in a greedy fashion by allocating resources to the refinement choice that addresses the largest error with the fewest additional space-time elements. To first approximation, the number of elements introduced in a time slab division is estimated as $N_{\text{elem},H}$, while the number of elements introduced in a spatial refinement is estimated as $N_{\text{slab},H}$ times the number of new subelements obtained from a hanging-node refinement. The respective adaptive indicators ϵ_k and ϵ_e are divided by these quantities and then sorted highest to lowest. The element or time slab with the highest error indicator per proposed additional number of space-time elements is chosen for refinement first, and the process continues until the growth budget is reached or surpassed. We note that in some cases the refinement could target only spatial elements or only time slabs, depending on the relative resolution in time and space.

C. Alternative Adaptive Indicators

The results compare output-based unsteady adaptation to uniform refinement in space and time and to two cheaper indicators: one based on the approximation error and one based on the unweighted residual. Such indicators are relatively simple to evaluate and have been analyzed in the past for steady problems.²³

The approximation error can be estimated in various ways. One method for a DG discretization is to consider the inter-element jumps of the solution, an idea that has been considered previously for shock-capturing purposes.^{24–26} Specifically, we define the jump indicator on space-time element (e, k) as

$$\epsilon_{e,k}^{\text{jump}} = \text{average jump in the state across space-time element boundaries.}$$

The element boundaries include interfaces between adjacent spatial elements on the same slab and between adjacent past and future time slabs. The average jump is computed on a reference space-time element so that no bias is introduced from the element size. For smooth solutions, this indicator targets areas of the space-time domain where the solution is not approximated well in the finite-dimensional space \mathcal{V}_H .

The unweighted residual indicator is given by a form similar to the output-error indicator, but without the adjoint,

$$\epsilon_{e,k}^{\text{res}} = \sum_{m \in \text{dof}_h(\mathcal{I}_k^H)} \mathbf{z}_e |\overline{\mathbf{R}}_h^m(\mathbf{U}_h^{H,n})|.$$

This indicator targets areas of the space-time domain where the partial differential equation is not satisfied. Both the approximation error indicator and the residual indicator are cheaper to evaluate than the output-based indicator, as they do not require an adjoint solution. The jump-based space-time anisotropy measure is used for both adaptive indicators.

V. Results

This section presents results of the adjoint-based unsteady adaptation for two examples of aerodynamic interest. Both of the examples are two dimensional in space and employ static body-fitted spatial meshes. The solutions are approximated with quadratic, $p = 2$, elements in space. In the time domain, the flow state is linearly approximated, $r = 1$, using a Lagrange basis with nodes at the time slab endpoints. The fine space for the adjoint is obtained by increasing the spatial and

temporal orders by one. The growth factor used in the adaptation is $f^{\text{growth}} = 2.0$, and a fixed number of adaptive iterations are run in each case. The reference “actual” solutions are computed on space-time meshes that are uniformly-refined versions of the final output-adapted meshes.

A. Two-Dimensional Scalar Convection-Diffusion-Reaction

The first example is a two-dimensional scalar convection-diffusion-reaction problem. The governing equation is

$$\frac{\partial u}{\partial t} + \nabla \cdot (\vec{V}u) - \nabla \cdot (\nu \nabla u) + S(u) = 0,$$

where u is the scalar concentration, \vec{V} is the convection velocity, ν is the diffusivity, and $S(u)$ is an Arrhenius-law reaction term given by

$$S(u) = Au(c_1 - u) \exp\left(-\frac{E}{c_2 - u}\right).$$

The constants in the reaction term are $A = 0.1$, $c_1 = 2$, $E = .05$, and $c_2 = 2.4$. The computational domain is a square and the initial condition is a two-dimensional Gaussian concentration profile with unit amplitude, as shown in Figure 4(a). The velocity field is an irrotational counterclockwise vortex flow about a point outside the lower left corner of the domain. The Peclet number based on the domain length L and the mean velocity $|\vec{V}|$ in the domain is $Pe \equiv L|\vec{V}|/\nu = 1000$. The output is a time-integral near the end of the simulation of the scalar measured at one point, as shown in Figure 4(c).

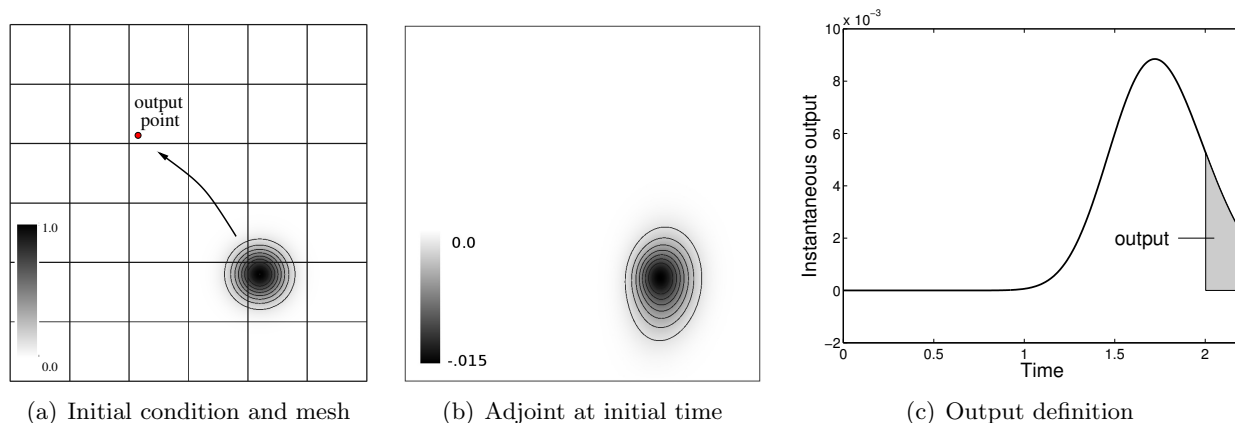


Figure 4. Scalar convection-diffusion-reaction: initial condition and point output overlaid on the coarse mesh, the adjoint solution at the initial time, and the definition of the output time-integral quantity.

The initial mesh for the adaptive runs, Figure 5(a), is a 6×6 uniform spatial grid with 4 time slabs. Figure 5(b) shows the output convergence with degrees of freedom for uniform refinement and adaptive indicators based on the output error, approximation error, and residual. The degrees of freedom are measured for the entire space-time mesh as the product of the spatial and temporal degrees of freedom. The results show that output-based adaptation converges most rapidly to the actual output value. The advantage over the other indicators depends on the desired output

error tolerance. For example, the result of output-based adaptation drops below 5% error in about 30 times fewer degrees of freedom than uniform refinement. The convergence results of the approximation-error indicator and the residual indicator are more irregular and on par or slightly worse than uniform refinement in this case.

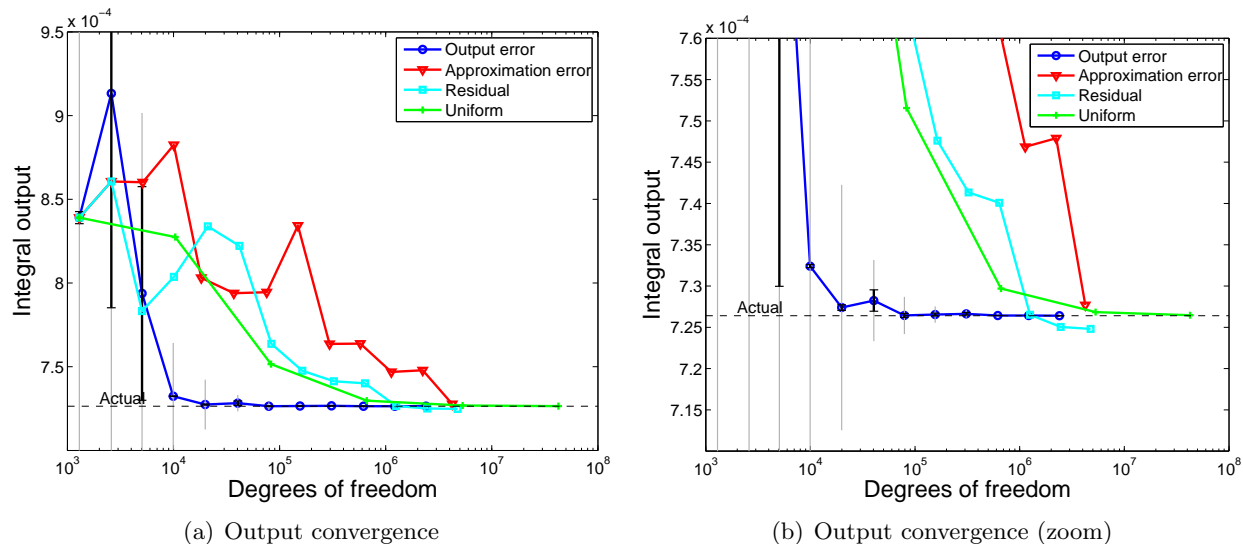


Figure 5. Scalar convection-diffusion-reaction: convergence of output under various adaptive indicators. Error bars at $\pm\delta J$ and whiskers at $\pm\epsilon$ are shown for the output-based results.

The output-adapted results in Figure 5(b) are shown with error bars at $\pm\delta J$, using the adjoint-weighted residual estimate in Eqn. 14, and at the more conservative $\pm\epsilon$, using the sum of element indicators in Eqn. 17. In this example, the estimate δJ under-predicts the output error on some of the adapted meshes, while ϵ generally over-predicts the error. Both quantities converge to zero as the adaptation proceeds.

Time histories of the scalar output reading on several adapted space-time meshes are shown in Figure 6(a). The temporal solution is discontinuous in time, and the plot shows only the instantaneous output at the end of each time slab, where the solution is superconvergent at order $2r + 1$.²⁰ The space-time meshes are chosen to have comparable degrees of freedom. In this set, which consists of the seventh adaptive iteration of each of the indicator-based methods and the second uniform refinement, the output-based adaptation result is closest to the actual time history. A more quantitative comparison is given in Figure 6(b), which plots the L_2 error in the time histories versus degrees of freedom. The output-based adaptation again shows the most rapid convergence, with savings of one to two orders of magnitude in degrees of freedom, depending on the desired error tolerance.

The adapted space-time meshes corresponding to the time histories in Figure 6(a) are shown in Figure 7. In the spatial domain, the indicators based on approximation error and residual target primarily the initial location of the scalar, where the concentration is highest. On the other hand, the output-based adaptation additionally targets the area near the output sensor. In the temporal domain, the approximation error and residual indicators target the initial time as this is where the reaction rate is highest. In contrast, the output error indicator produces a fairly uniform temporal refinement for tracking the scalar profile until the output reading at the end of the simulation.

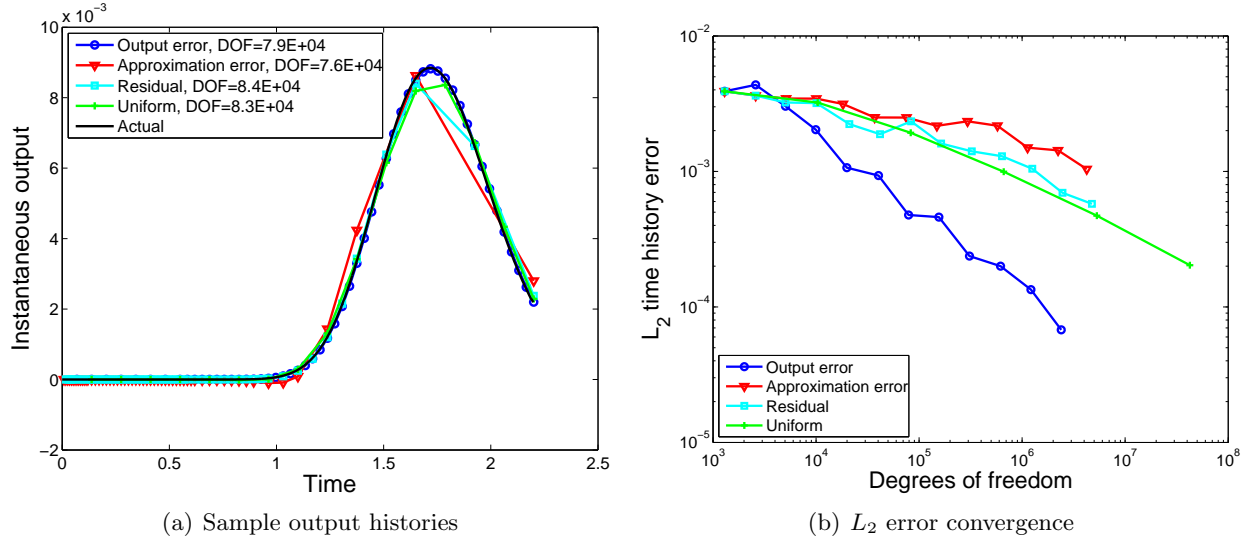


Figure 6. Scalar convection-diffusion-reaction: sample point-value time histories and convergence of the L_2 time history error for various adaptive indicators.

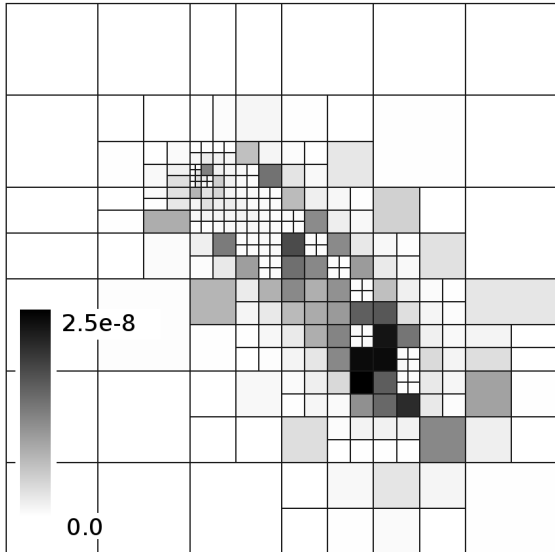
B. Impulsively-Started Airfoil

The second example is an impulsively-started NACA 0012 airfoil in viscous flow, where for $t \geq 0$, the freestream conditions are $M_\infty = 0.25$, $\alpha = 8^\circ$, $Re = 5000$. To prevent a non-physical step change in the velocity of the fluid at the airfoil surface, the initial condition at $t = 0$ consists of the freestream with the velocity blended smoothly to zero in a circular disk around the airfoil. Specifically, the velocity in the blended region is $\mathbf{v} = \mathbf{V}_\infty(1 - \cos(\pi(r - r_1)/(r_2 - r_1)))/2$ where r_1 and r_2 are radial distances from the airfoil mid-chord, set to one and three chord lengths respectively, and \mathbf{V}_∞ is the freestream velocity. The corresponding initial Mach number contours are shown in Figure 8(b). No steady solve is performed prior to the unsteady simulation.

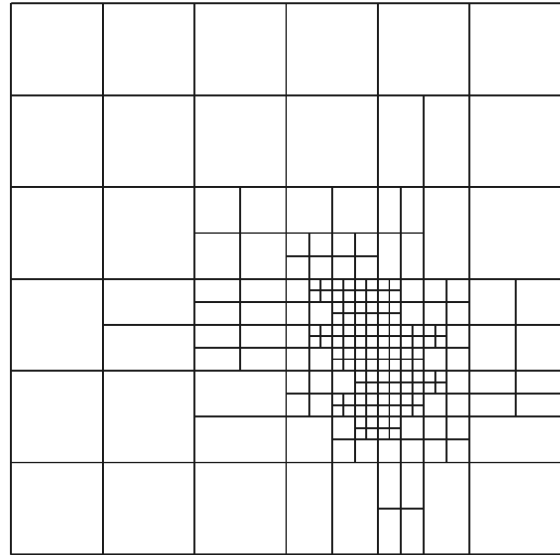
Figure 8(a) shows the entropy contours at $t = 10$ units, the final time in the simulation. By this time an alternating pattern of shed vortices has developed and is clearly visible. The output of interest is the lift coefficient integral from $t = 9$ to $t = 10$, as illustrated in Figure 9(a). A snapshot of the corresponding adjoint solution for the y -momentum equation at $t = 6$ is illustrated in Figure 8(b). A “reverse wake” is evident in the adjoint solution, signifying an oscillatory sensitivity of the output to y -momentum residual perturbations upstream.

For the adaptive runs, the initial spatial mesh contains 510 elements, see Figure 8(a), and the initial temporal mesh contains 16 time slabs. The output convergence for the various indicators is shown in Figure 9(b). The residual indicator does not perform well at all: the output varies significantly from iteration to iteration. The other indicators converge, with the fastest being output-based adaptation, followed by approximation error and then uniform refinement. The advantage of the output-based refinement with degrees of freedom is not as pronounced in this case, with a factor of 3-4 savings over the approximation error indicator. The error estimates underpredict the error in the middle stages of output-based refinement, while the conservative whiskers at $\pm\epsilon$ are more robust.

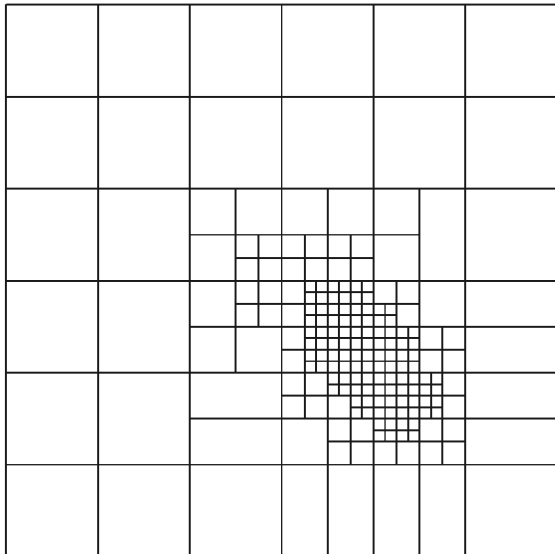
Figure 10(a) shows the time histories of the lift coefficient for adapted space-time meshes of similar size. The error in the residual-adapted case is clear: the simulation does not predict oscilla-



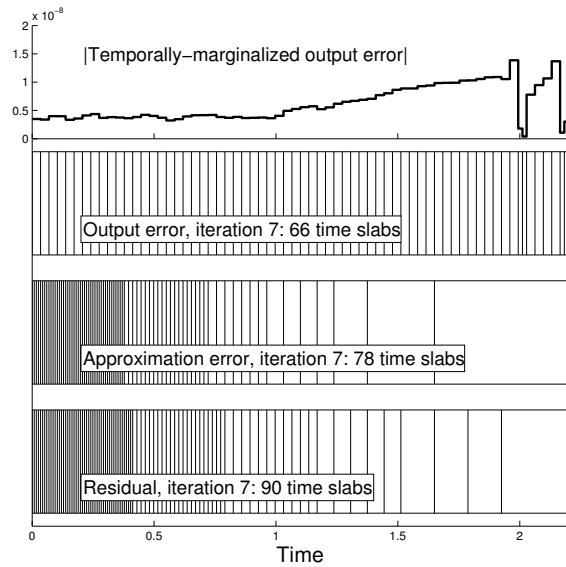
(a) Adapted on output error (262 elements)



(b) Adapted on approximation error (212 elements)



(c) Adapted on residual (202 elements)



(d) Temporal meshes

Figure 7. Scalar convection-diffusion-reaction: adapted spatial and temporal meshes for the seventh adaptive iteration. Localized output error estimates ϵ_e and ϵ_k are shown for the output-error adapted spatial and temporal meshes.

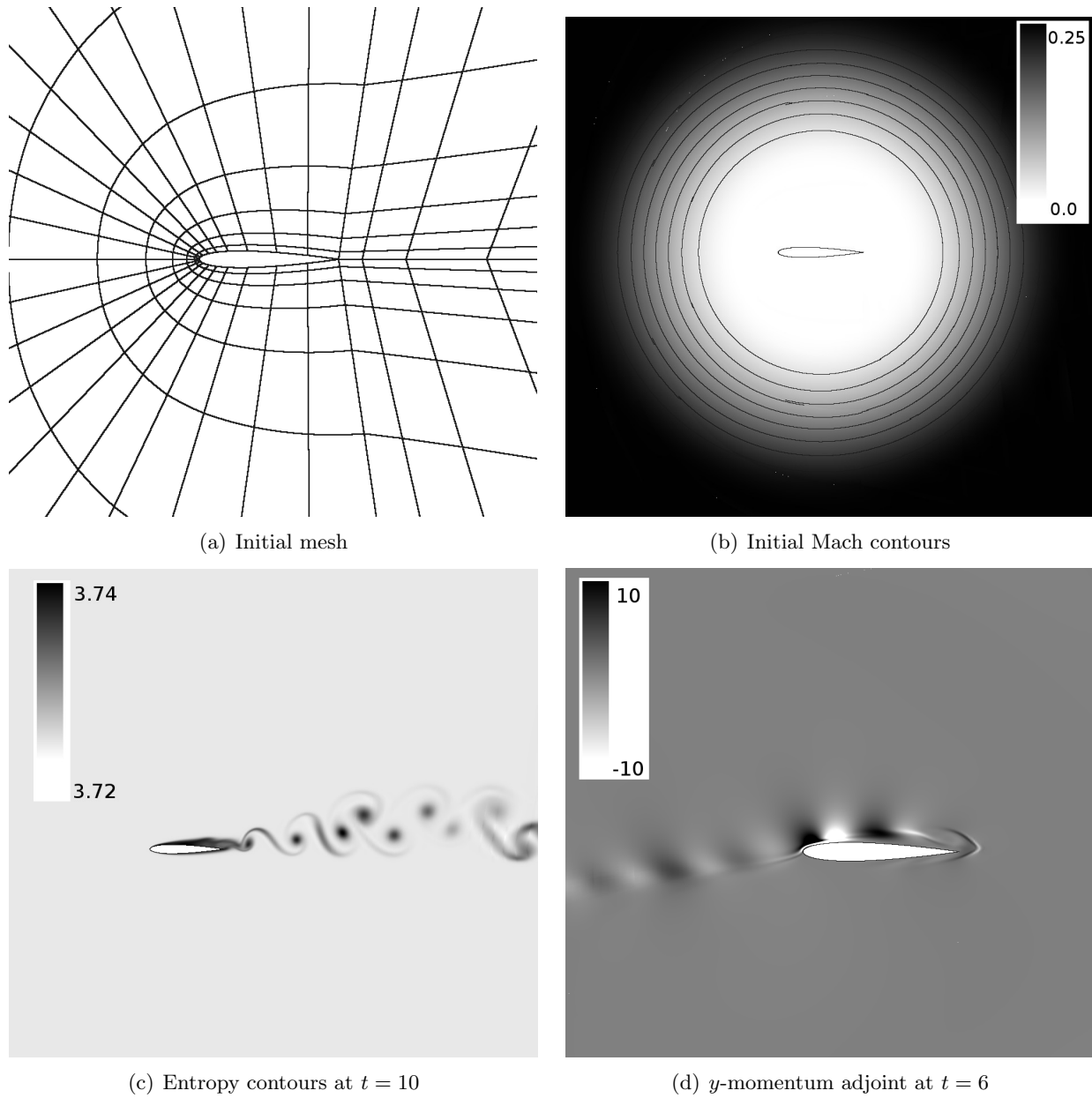


Figure 8. Impulsively-started airfoil: initial mesh, initial Mach number contours, forward state at the final time, and the adjoint state at $t = 6$.

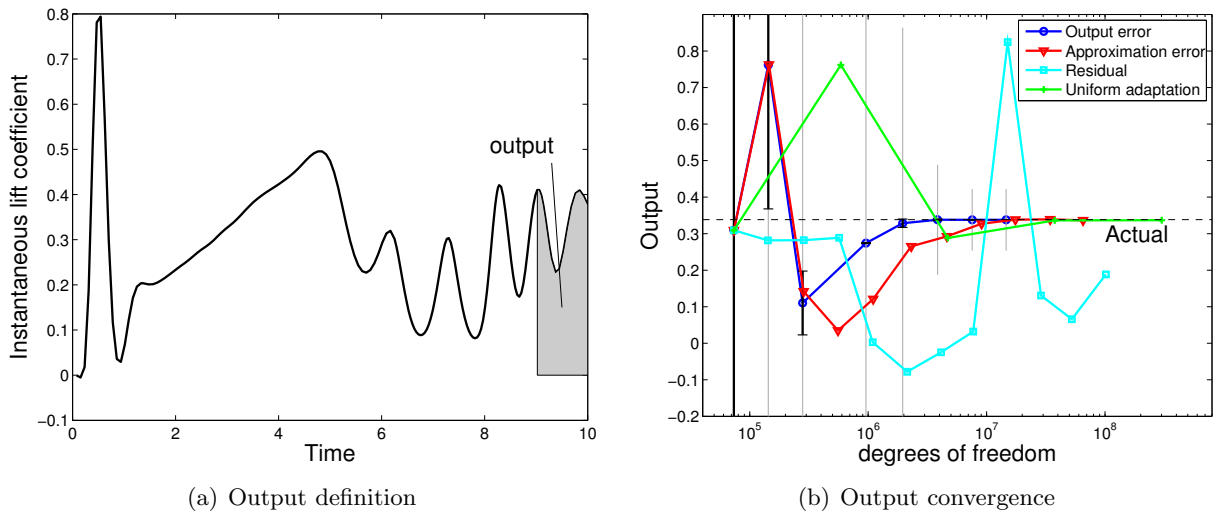


Figure 9. Impulsively-started airfoil: time integral output definition and its convergence under the adaptive indicators. Error bars at $\pm\delta J$ and whiskers at $\pm\epsilon$ are shown for the output-based results.

tory vortex shedding, but rather an increasing lift coefficient. The other three adaptive indicators track the actual time history well. Figure 10(b) shows the L_2 time history error convergence, and here output-based adaptation is fastest. Even though the output is only measured on the final 10% of the simulation time, accurate resolution prior to this metric time is important as it affects the state at the start of the output measurement.

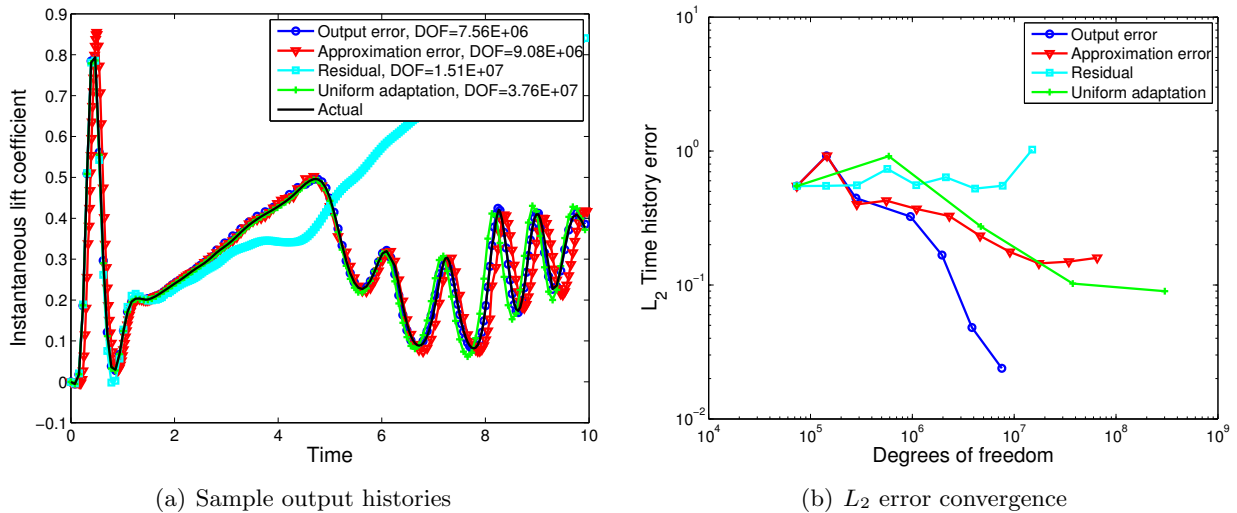


Figure 10. Impulsively-started airfoil: sample lift coefficient time histories and convergence of the L_2 time history error for various adaptive indicators.

The corresponding adapted meshes are shown in Figure 11. Output-based adaptation targets the airfoil leading and trailing edges, the boundary-layer region above the front of the airfoil, and slightly the stagnation line in front and the wake behind the airfoil. The approximation-error indicator also targets the leading and trailing edges and puts more emphasis on the wake, where

the shed vortices propagate. The residual indicator is distracted by effects of the initial condition: the velocity blending near the airfoil sends out acoustic waves that the residual indicator attempts to track as they propagate away from the airfoil.

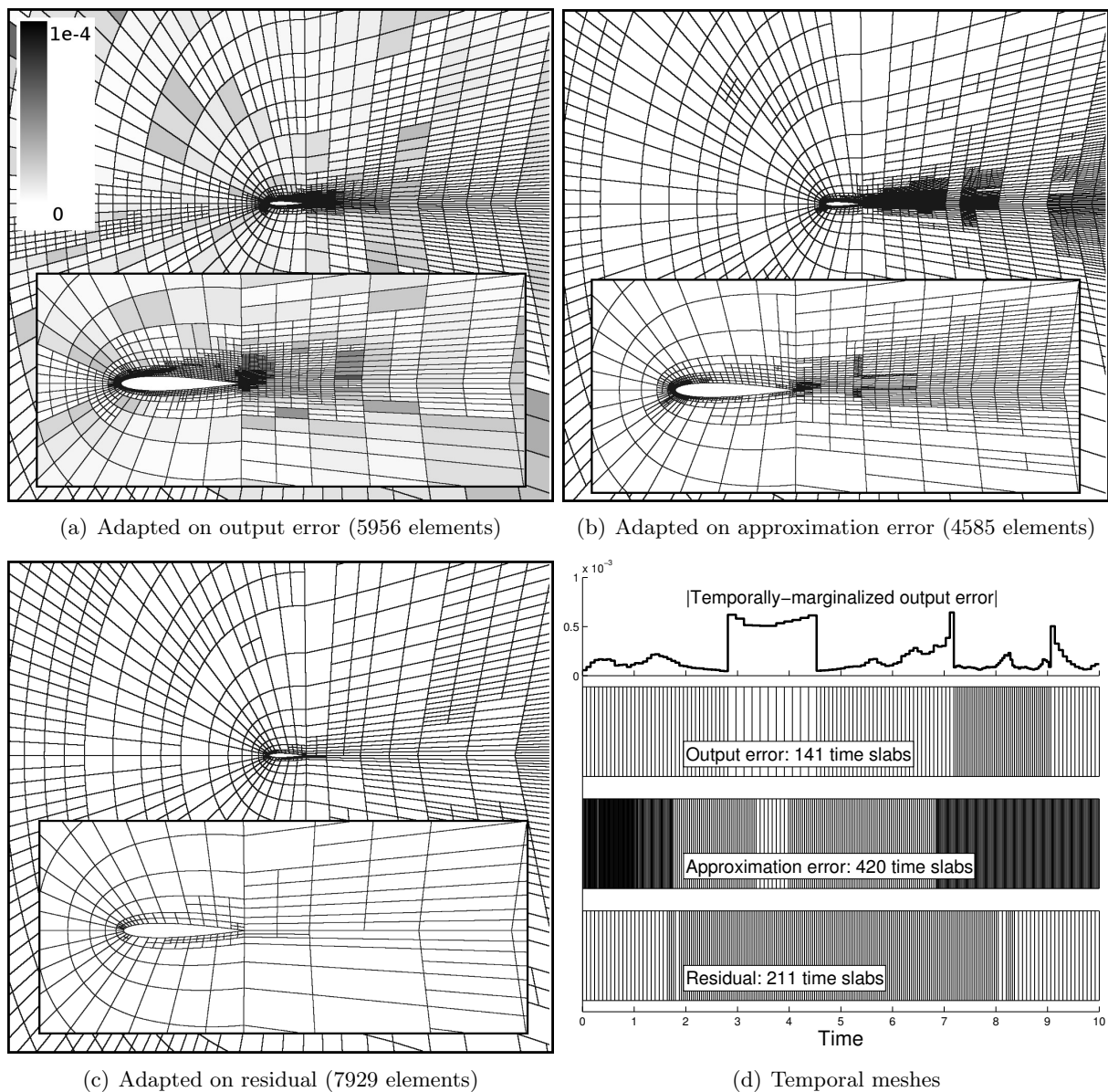


Figure 11. Impulsively-started airfoil: adapted spatial and temporal meshes for the seventh adaptive iteration. Localized output error estimates ϵ_e and ϵ_k are shown for the output-error adapted meshes.

The temporal meshes are shown in Figure 11(d). The output-based indicator creates a fairly uniform temporal refinement, with slightly higher resolution prior to the metric time. The approximation error focuses on the initial time, as it tracks the evolution of the blended velocity field, and the latter 1/3 time when the shed vortices develop. The residual indicator creates a mostly-uniform temporal mesh as it tracks the acoustic waves.

VI. Conclusions

This paper presents an adjoint-based output error estimation algorithm for unsteady simulations. A space-time discontinuous Galerkin finite element discretization is used for its flexibility in adaptation and its admission of a robust variational formulation for error estimation. An approximate factorization solution technique for the unsteady forward problem is extended to the discrete adjoint problem. Details are presented in the application of the adjoint-weighted residual method to the unsteady discretization, including the choice of the adjoint fine space for combined spatial and temporal error estimation. In addition, a space-time anisotropy measure based on projection of the fine adjoint solution is introduced to attribute the error to spatial or temporal resolution. This measure drives refinement in space and time on static spatial meshes.

Adaptive results for a two-dimensional convection-diffusion-reaction problem and for a two-dimensional compressible Navier-Stokes airfoil problem demonstrate the effectivity of the output error estimate and the benefits in terms of degrees of freedom of output-based adaptation compared to uniform space-time refinement and to cheaper heuristic adaptation schemes. Future work will investigate the computational cost of the error estimation and more sophisticated adaptive strategies including dynamically-refined spatial meshes and unsteady problems involving grid motion.

References

- ¹Pierce, N. A. and Giles, M. B., “Adjoint recovery of superconvergent functionals from PDE approximations,” *SIAM Review*, Vol. 42, No. 2, 2000, pp. 247–264.
- ²Becker, 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.
- ³Hartmann, 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.
- ⁴Venditti, 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.
- ⁵Sen, S., Veroy, K., Huynh, D., Deparis, S., Nguyen, N., and Patera, A., ““Natural norm” a posteriori error estimators for reduced basis approximations,” *Journal of Computational Physics*, Vol. 217, 2006, pp. 37–62.
- ⁶Nemec, M. and Aftosmis, M. J., “Error estimation and adaptive refinement for embedded-boundary Cartesian meshes,” AIAA Paper 2007-4187, 2007.
- ⁷Fidkowski, K. J. and Darmofal, D. L., “Output-based error estimation and mesh adaptation: Overview and recent results,” *American Institute of Aeronautics and Astronautics Journal*, 2010, Accepted.
- ⁸Bar-Yoseph, P. and Elata, D., “An efficient L2 Galerkin finite element method for multi-dimensional non-linear hyperbolic systems,” *International Journal for Numerical Methods in Engineering*, Vol. 29, 1990, pp. 1229–1245.
- ⁹Lowrie, R. B., Roe, P. L., and van Leer, B., “Properties of space-time discontinuous Galerkin,” Los Alamos Technical Report LA-UR-98-5561, 1998.
- ¹⁰van der Ven, H. and van der Vegt, J., “Space-time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows II. Efficient flux quadrature,” *Computer Methods in Applied Mechanical Engineering*, Vol. 191, 2002, pp. 4747–4780.
- ¹¹Klaij, C., van der Vegt, J., and van der Ven, H., “Space-time discontinuous Galerkin method for the compressible Navier-Stokes equations,” *Journal of Computational Physics*, Vol. 217, 2006, pp. 589–611.
- ¹²Barth, 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.
- ¹³Richter, 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*, 2009.
- ¹⁴Mani, K. and Mavriplis, D. J., “Discrete adjoint based time-step adaptation and error reduction in unsteady flow problems,” AIAA Paper 2007-3944, 2007.
- ¹⁵Mani, 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.
- ¹⁶Meidner, 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.

¹⁷Schmich, 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.

¹⁸Bassi, F. and Rebay, S., “GMRES discontinuous Galerkin solution of the compressible Navier-Stokes equations,” *Discontinuous Galerkin Methods: Theory, Computation and Applications*, edited by K. Cockburn and Shu, Springer, Berlin, 2000, pp. 197–208.

¹⁹Roe, P. L., “Approximate Riemann solvers, parameter vectors, and difference schemes,” *Journal of Computational Physics*, Vol. 43, 1981, pp. 357–372.

²⁰Adjerid, S., Devine, K. D., Flaherty, J., and Krivodonova, L., “A posteriori error estimation for discontinuous Galerkin solutions of hyperbolic problems,” *Computer Methods in Applied Mechanical Engineering*, Vol. 191, 2002, pp. 1097–1112.

²¹Griewank, A. and Walther, A., “Revolve: An implementation of checkpointing for the reverse or adjoint mode of computational differentiation,” *ACM Transactions on Mathematical Software*, Vol. 26, No. 1, 2000, pp. 19–45.

²²Leicht, T. and Hartmann, R., “Multitarget error estimation and adaptivity in aerodynamic flow simulations,” *International Journal for Numerical Methods in Fluids*, Vol. 56, 2008, pp. 2111–2138.

²³Zhang, X. D., Vallet, M.-G., Dompierre, J., Labbe, P., Pelletier, D., Trepanier, J.-Y., Camarero, R., Lassaline, J. V., Manzano, L. M., and Zingg, D. W., “Mesh adaptation using different error indicators for the Euler equations,” AIAA Paper 2001-2549, 2001.

²⁴Dolejsi, V., Feistauer, M., and Schwab, C., “On some aspects of the discontinuous Galerkin finite element method for conservation laws,” *Mathematics and Computers in Simulation*, Vol. 61, No. 3, 2003, pp. 333–346.

²⁵Krivodonova, L., Xin, J., Remacle, J., Chevaugeon, N., and Flaherty, J., “Shock detection and limiting with discontinuous Galerkin methods for hyperbolic conservation laws,” *Applied Numerical Mathematics*, Vol. 48, No. 3, 2004, pp. 323–338.

²⁶Barter, G. E., *Shock Capturing with PDE-Based Artificial Viscosity for an Adaptive Higher-Order Discontinuous Galerkin Finite Element Method*, Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, 2008.