Output-Based Space-Time Mesh Adaptation for the Compressible Navier-Stokes Equations

Krzysztof J. Fidkowski*, Yuxing Luo

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

Abstract

This paper presents an output-based adaptive algorithm for unsteady simulations of convection-dominated flows. A space-time discontinuous Galerkin discretization is used in which the spatial meshes remain static in both position and resolution, and in which all elements advance by the same time step. Error estimates are computed using an adjoint-weighted residual, where the discrete adjoint is computed on a finer space obtained by order enrichment of the primal space. An iterative method based on an approximate factorization is used to solve both the forward and adjoint problems. The output error estimate drives a fixed-growth adaptive strategy that employs hanging-node refinement in the spatial domain and slab bisection in the temporal domain. Detection of space-time anisotropy in the localization of the output error is found to be important for efficiency of the adaptive algorithm, and two anisotropy measures are presented: one based on inter-element solution jumps, and one based on projection of the adjoint. Adaptive results are shown for several two-dimensional convection-dominated flows, including the compressible Navier-Stokes equations. For sufficiently-low accuracy levels, output-based adaptation is shown to be advantageous in terms of degrees of freedom when compared to uniform refinement and to adaptive indicators based on approximation error and the unweighted residual. Time integral quantities are used for the outputs of interest, but entire time histories of the integrands are also compared and found to converge rapidly under the proposed scheme. In addition, the final output-adapted space-time meshes are shown to be relatively insensitive to the starting mesh.

Keywords: Output error estimation, Space-time mesh adaptation, Discontinuous Galerkin, Compressible Navier-Stokes, Unsteady adjoint

1. Introduction

Numerical simulations of large-scale unsteady aerodynamics problems generate massive amounts of data in the form of a state description on the entire space-time domain. Processing this data can be a challenging task, yet for engineering purposes often only a few scalar quantities or time histories are of interest. Robustness of the simulations in the context of engineering analysis and design depends on the verification measures that are employed for the outputs of interest. In this

*Corresponding author

Email address: kfid@umich.edu (Krzysztof J. Fidkowski)

work we present one such verification measure in the form of a posteriori output error estimates, and we use the estimates to drive a combined spatial and temporal adaptive solution scheme.

A posteriori output error estimates have received considerable attention for steady problems [1, 2, 3, 4, 5, 6, 7], where they have been used not only to provide error bars on outputs but also to drive mesh adaptation. A recent review of adaptive methods based on these estimates can be found in [7]. The robustness and efficiency of these methods arise from the fact that they specifically target for refinement those and only those areas of the computational domain that are important for predicting the output of interest. They are popular for convection-dominated aerospace applications because through the use of adjoint solutions, they properly account for error propagation effects, which can be troublesome for many heuristic adaptive indicators.

Research into output-based error estimation for unsteady problems has been more limited, in part due to implementation challenges and to computational expense. While in the steady case the cost of an adjoint solution is generally comparable to that of the primal problem on the same space, the cost rises in the unsteady case, especially in terms of storage for nonlinear problems. However, from a theoretical point of view, there is no fundamental limitation preventing the implementation of unsteady adjoint capability into aerospace simulation codes. We believe that there is a need for unsteady adjoints as many aerospace applications rely on unsteady simulations that 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 benefit of solution-adaptive methods that use resources efficiently. Moreover, with the continuing growth of processing power, output error estimates and adaptive methods are now within computational reach for many unsteady aerospace applications.

The theoretical framework for output error estimation is most rigorous in a variational formulation, and hence this work employs a space-time finite element discretization. Specifically, the discontinuous Galerkin (DG) method is used in both space and time. DG has been studied previously for spatial [8, 9, 10, 11, 12, 13], temporal [14, 15, 16, 17], and combined space-time [18, 19, 20, 21, 22] discretizations. Although computationally expensive, DG shows potential for high accuracy and is flexible in the admissible solution space. The latter point is important for our present work, which employs non-conforming elements in the spatial domain, and for our future work, which will consider dynamic mesh refinement. We note that less some of the mathematical rigor and with additional approximations, we expect the methods presented in our work to extend to other discretizations, including finite volume/difference in space and multistep/multistage in time.

A few 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 mesh motion, Mani and Mavriplis use the discrete adjoint to estimate the output error due to both the temporal resolution and to partial convergence of the unsteady residuals [23, 24]. 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 [22]. 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* [25, 26] 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

into a tractable algorithm for the compressible Navier-Stokes equations. Primary contributions of this work consist of an estimate of space-time anisotropy of the solution and a mesh optimization algorithm for making decisions between spatial and temporal refinement. These contributions are highlighted in an extensive set of results. The results also demonstrate several robustness aspects of the output-based adaptation, including independence of the initial space-time mesh and favorable convergence of output histories in addition to integral quantities.

The remainder of this paper is organized as follows: Section 2 presents the space-time discontinuous Galerkin discretization and the iterative solution technique for both the primal and the discrete adjoint problems. The output error estimation procedure via the adjoint-weighted residual is described with implementation details in Section 3. Section 4 discusses error localization, including the measures of space-time anisotropy, and mesh adaptation. It also presents the approximation error and residual indicators that will be used for comparison. Various adaptive results for problems in two spatial dimensions are presented in Section 5. These problems include gust encounters in subsonic and transonic regimes and an impulsively-started airfoil. We end with concluding remarks in Section 6.

2. Discretization and Solution

2.1. Primal Discretization

The canonical problem for this work consists of a system of s partial differential equations in d spatial dimensions,

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

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, and $\mathbf{r} : \mathbb{R}^s \to \mathbb{R}^s$ is a differential operator in $\mathbf{x}$.

We discretize (1) using a discontinuous Galerkin finite element method on tensor-product space-time meshes, an example of which is illustrated in Figure 1 for $d = 2$. Although the present work only considers problems in which the nodes of the spatial mesh remain fixed in time, mesh motion could be incorporated through suitable mapping of a reference spatial domain.

We seek an approximate solution, $\mathbf{u}_H(\mathbf{x}, t)$, in the finite-dimensional space $\mathcal{V}_H = [\mathcal{V}_H^{\text{space}}]^s \otimes \mathcal{V}_H^{\text{time}}$. The spatial approximation space $\mathcal{V}_H^{\text{space}}$ consists of polynomials of order p in the reference spatial coordinates of each element. The temporal approximation space, $\mathcal{V}_H^{\text{time}}$, consists of polynomials of order r in time on each of $N_{H,\text{slab}}$ time slabs. Both of these spaces admit discontinuities across elements/time slabs.

Let $\{\varphi_H^n(t)\}$, $n = 1, \ldots, (r+1)N_{H,\text{slab}}$, be a basis for $\mathcal{V}_H^{\text{time}}$. Using implied summation on n, we expand the solution as

$$\mathbf{u}_H(\mathbf{x}, t) = \mathbf{u}_H^n(\mathbf{x})\varphi_H^n(t) = \mathbf{I}_H(\mathbf{x})\mathbf{U}_H^n\varphi_H^n(t). \tag{2}$$

We do not focus on the spatial discretization and simply assume that at time node n $\mathbf{u}_H(\mathbf{x}, t)$ is described by a vector of N_H numbers, $\mathbf{U}_H^n \in \mathbb{R}^{N_H}$. $\mathbf{I}_H(\mathbf{x}) \in \mathbb{R}^{s \times N_H}$ is a position-dependent interpolation matrix that contains the details of the spatial basis.

The variational finite element statement reads: find $\mathbf{u}_H \in \mathcal{V}_H$ such that

$$\overline{\mathcal{R}}_H(\mathbf{u}_H, \mathbf{v}_H) = 0, \qquad \forall \mathbf{v}_H \in \mathcal{V}_H, \tag{3}$$

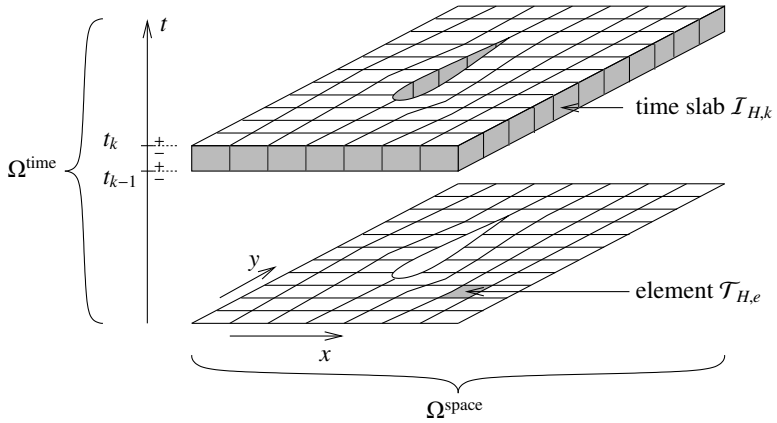


Figure 1: Portion of a sample mesh for a space-time discretization in two spatial dimensions. Each space-time element, $\mathcal{T}_{H,e} \otimes \mathcal{I}_{H,k}$, is given by a tensor product of a spatial element, $\mathcal{T}_{H,e}$, $e = 1, \ldots, N_{H,\text{elem}}$, and a time slab, $\mathcal{I}_{H,k}$, $k = 1, \ldots, N_{H,\text{slab}}$. The spatial mesh, which can be unstructured, is assumed to be invariant in time.

where the semilinear form $\overline{\mathcal{R}}_H(\cdot,\cdot) : \mathcal{V}_H \otimes \mathcal{V}_H \to \mathbb{R}$ is obtained by multiplying (1) by a test function $\mathbf{v}_H$ and integrating by parts over the space-time domain. The discrete version of (3) is a set of $N_H(r+1)N_{H,\text{slab}}$ equations that we group by time nodes into $(r+1)N_{H,\text{slab}}$ unsteady residual vectors, $\overline{\mathbf{R}}_H^m \in \mathbb{R}^{N_H}$. For clarity and without loss of generality, we consider only the first time slab, on which the $r+1$ unsteady residual equations are

$$\overline{\mathbf{R}}_H^m \equiv a^{m,n}\mathbf{M}_H\mathbf{U}_H^n - \varphi_H^m(t_0)\mathbf{M}_H\mathbf{U}_H^0 + \int_{t_0}^{t_1} \varphi_H^m(t)\mathbf{R}_H(\mathbf{U}_H(t))\,dt = \mathbf{0}, \tag{4}$$

where $\mathbf{M}_H \in \mathbb{R}^{N_H \times N_H}$ is the spatial mass matrix and $\mathbf{R}_H \in \mathbb{R}^{N_H}$ is the spatial residual vector. $\mathbf{U}_H^0$ is the prescribed initial condition. For all other time slabs, the solution from the end of the previous time slab is used in place of $\mathbf{U}_H^0$. For the spatial discretization, we use discontinuous Galerkin with the Roe inviscid flux [27] and the second form of Bassi and Rebay [28] for the viscous flux. These choices are arbitrary, however. We note that the temporal discretization and solution scheme impose no restrictions on the spatial discretization.

The matrix $a^{m,n}$ on the first time slab is given by

$$a^{m,n} = -\int_{t_0}^{t_1} \varphi_H^n \frac{d\varphi_H^m}{dt}\,dt + \varphi_H^n(t_1)\varphi_H^m(t_1). \tag{5}$$

Using a Lagrange temporal basis on equally-spaced nodes, we have on the first time slab,

$$\text{for } r=1,\ a^{m,n} = \begin{bmatrix} 1/2 & 1/2 \\ -1/2 & 1/2 \end{bmatrix}; \quad \text{for } r=2,\ a^{m,n} = \begin{bmatrix} 1/2 & 2/3 & -1/6 \\ -2/3 & 0 & 2/3 \\ 1/6 & -2/3 & 1/2 \end{bmatrix}. \tag{6}$$

The integral in (4) is evaluated with $(r+1)$ points of Gauss quadrature for order $2r+1$ accuracy.

2.2. Adjoint Discretization

We consider a scalar output calculated from the unsteady state vector,

$$\text{output} = J_H(\mathbf{u}_H(x,t)), \tag{7}$$

where the subscript H accounts for the particular quadrature and/or geometry fidelity associated with the spatial mesh. By (2), we can think of the output as a function of the discrete vectors $\mathbf{U}_H^n$.

The discrete adjoint solution, $\mathbf{\Psi}_H^m \in \mathbb{R}^{N_H}$, $m \in 1, \ldots, (r+1)N_{H,\text{slab}}$, consists of vectors whose components are the sensitivities of J_H to residual source perturbations added to (4). The discrete adjoint equation is

$$\underbrace{\left(\frac{\partial \overline{\mathbf{R}}_H^m}{\partial \mathbf{U}_H^n}\right)^T \mathbf{\Psi}_H^m + \left(\frac{\partial J_H}{\partial \mathbf{U}_H^n}\right)^T}_{\overline{\mathbf{R}}_H^{\psi,n}(\mathbf{\Psi}_H^m)} = 0, \tag{8}$$

where the linearization of the unsteady residual is computed about the forward solution for nonlinear problems. For an adjoint-consistent output and primal discretization [29, 30], the space-time function

$$\boldsymbol{\psi}_H(\mathbf{x}, t) = \mathbf{I}_H(\mathbf{x}) \mathbf{\Psi}_H^m \varphi_H^m(t)$$

approximates the solution to the continuous adjoint problem.

For each $n = 1, \ldots, (r+1)N_{H,\text{slab}}$, $\overline{\mathbf{R}}_H^{\psi,n}(\mathbf{\Psi}_H^m) \in \mathbb{R}^{N_H}$ is an unsteady adjoint residual vector. On the first time slab, the $r+1$ adjoint residual vectors are

$$\begin{aligned} \overline{\mathbf{R}}_H^{\psi,n} &= a^{m,n} \mathbf{M}_H \mathbf{\Psi}_H^m - \varphi_H^n(t_1) \mathbf{M}_H \mathbf{\Psi}_H^{\text{next}} \\ &\quad + \int_{t_0}^{t_1} \varphi_H^n \left(\frac{\partial \mathbf{R}}{\partial \mathbf{U}}(\mathbf{U}_H(t))\right)^T \mathbf{\Psi}_H(t)\, dt + \left(\frac{\partial J_H}{\partial \mathbf{U}_H^n}\right)^T, \end{aligned} \tag{9}$$

where $\mathbf{\Psi}_H(t) = \mathbf{\Psi}_H^m \varphi_H^m(t)$, and $\mathbf{\Psi}_H^{\text{next}}$ is the adjoint vector associated with the start of the second time slab. These formulas extend naturally to all other time slabs. To obtain the exact discrete adjoint, the integrals in the above equation are evaluated using the same numerical quadrature as in (4).

2.3. Iterative Unsteady Solution

The residual equations in (4) constitute a possibly nonlinear system in which all of the degrees of freedom within a time slab are coupled together. Each time slab is also coupled to its predecessor through the solution at the end of the previous slab. We solve this system using an iterative method introduced by Richter [31]. The starting point for this method is a Newton-Raphson update equation on time slab k,

$$\frac{\partial \overline{\mathbf{R}}_H^m}{\partial \mathbf{U}_H^n} \Delta \mathbf{U}_H^n = -\overline{\mathbf{R}}_H^m, \tag{10}$$

where m and n are the temporal indices associated with slab k.

Two approximations are employed to make the solution of (10) tractable. First, a temporally constant spatial Jacobian matrix, $\mathbf{A}_H \in \mathbb{R}^{N_H \times N_H}$, is evaluated at the time slab midpoint and used in the linearization of the time integrals appearing in the unsteady residuals. The result is

$$(a^{m,n} \mathbf{M}_H + b^{m,n} \Delta t_k \mathbf{A}_H)\, \Delta \mathbf{U}_H^n = -\overline{\mathbf{R}}_H^m, \tag{11}$$

$$\text{where} \quad b^{m,n} = \frac{1}{\Delta t_k} \int_{\mathcal{I}_{H,k}} \varphi_H^m(t) \varphi_H^n(t) dt, \tag{12}$$

and $\Delta t_k = t_k - t_{k-1}$.

Second, an approximate factorization is used in the solution of the $r+1$ residual equations resulting in a multistep scheme that does not require formation or inversion of squares or higher powers of $\mathbf{A}_H$. The scheme consists of three steps and two matrix inversions for $r = 1$ and six steps and three matrix inversions for $r = 2$. These schemes are presented in Appendix A. The matrices involved are just combinations of the mass matrix and $\mathbf{A}_H$, so that they have the same compact stencil as the steady-state problem. Inversion is performed iteratively using the generalized minimal residual method preconditioned by an element-line smoother.

3. Output Error Estimation

An approximation of the solution in the finite dimensional space $\mathcal{V}_H$ will generally not be "exact" in the sense that it will change as the solution space is enriched. A scalar output computed with this solution will therefore be affected by solution discretization errors. Our goal is to estimate the resulting output error. Central to the process will be an approximation to the fine space adjoint solution, $\boldsymbol{\psi}_h(x,t) \in \mathcal{V}_h$, which will be used in an adjoint-weighted residual calculation to produce an estimate that accounts for discretization errors in both space and time. The following subsections describe the details of this process.

3.1. The Adjoint-Weighted Residual

In a variational setting, the theory of adjoint-based output error estimation extends naturally from steady to unsteady discretizations. By virtue of Galerkin orthogonality, the output error estimate evaluates to zero for all approximate adjoint solutions that are in the same space as the primal solution. Therefore, the adjoint must be approximated on a space $\mathcal{V}_h$ that is finer than $\mathcal{V}_H$. In a discontinuous Galerkin space-time discretization, the solution space can be enriched by increasing the interpolation order and/or refining the elements/time slabs. We use a fine space that is obtained from the coarse space by incrementing the spatial and temporal orders by 1.

For our purpose, the output error is defined as the difference between the output computed with the coarse solution and that computed with a fine solution. 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 [2, 7],

$$\delta J = \text{output error} \approx -\overline{\mathcal{R}}_h(\mathbf{u}_H, \boldsymbol{\psi}_h) = -\left(\boldsymbol{\Psi}_h^m\right)^T \overline{\mathbf{R}}_h^m(\mathbf{U}_h^{H,n}), \tag{13}$$

where in the first expression $\overline{\mathcal{R}}_h(\mathbf{u}_H, \boldsymbol{\psi}_h)$ is the weak form from (3). In the second expression, $\mathbf{U}_h^{H,n} \in \mathbb{R}^{N_h}$ corresponds to the injection of the coarse solution into the fine space, and summation is implied on m, which ranges over the fine-space temporal degrees of freedom. N_h is the number of spatial degrees of freedom on the fine space. The approximation sign indicates that the above expression is not exact when the adjoint equation is solved approximately or when the problem is nonlinear. The choice of order enrichment for the fine space simplifies the residual calculation on the fine space as the spatial and temporal meshes remain fixed.

The cost of the fine adjoint solution depends on the choice of the fine space $\mathcal{V}_h$ and on the approximations employed. As the space $\mathcal{V}_h$ is made richer, the accuracy of the error estimate improves but the computational cost rises, resulting in a trade-off between cost and accuracy. In this work we solve the adjoint problem exactly with spatial order $p+1$ and temporal order r, and we then reconstruct the adjoint to order $r+1$. The temporal reconstruction makes use

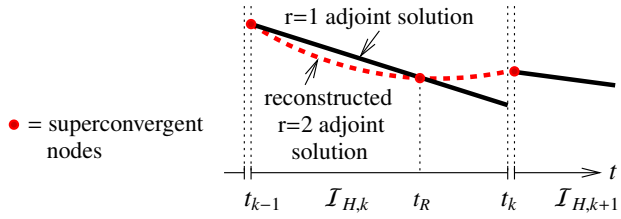


Figure 2: Illustration of temporal reconstruction of the fine-space adjoint solution. Shown is the reconstruction of an $r = 2$ fine adjoint solution (dashed line) from an $r = 1$ coarse solution (solid lines) using the left-node from the adjacent future time slab and superconvergent nodes on the current time slab. t_R indicates the root of the left Radau polynomial for $r = 1$.

of superconvergence properties of DG solutions at slab endpoints and Radau points [32]. It is illustrated in Figure 2 for $r = 1$. We note that whereas reconstruction of the forward solution may yield non-physical state values, no such problem exists in the reconstruction of the adjoint variables, which are not subject to physical constraints. In addition, the temporal reconstruction is essentially local: the additional information required for a high-order interpolant comes in only from the left node of the adjacent future time slab.

3.2. Implementation

The error estimate is computed concurrently with the adjoint solve, which is performed after the primal solve. Whether or not the problem is nonlinear, the primal state is always stored to hard disk during the forward solve in the form of $r + 1$ vectors, each of size N_H, per time slab. During the adjoint solve, the primal state on the current slab is loaded into memory, injected into the fine space, and used to calculate the fine-space residual. The adjoint problem is then approximated on the current time slab by solving at spatial order $p + 1$ and temporal order r, and then reconstructing in time to order $r + 1$. All linearizations required in the adjoint problem are performed using the injected coarse forward solution, so that the fine forward solution is never required. A coarse-space projection is subtracted from the fine-space adjoint to minimize errors in the adjoint-weighted-residual calculation. The adjoint-weighted residual result from the time slab increments the output error and running totals of the spatial and temporal adaptive indicators, which are described in the next section.

4. Spatial and Temporal 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 is illustrated in Figure 3. The following subsections describe the error localization and adaptive optimization strategies.

4.1. Error Localization

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

$$\delta J \approx -\left(\boldsymbol{\Psi}_h^m\right)^T \overline{\mathbf{R}}_h^m(\mathbf{U}_h^{H,m}) = \sum_{k=1}^{N_{\text{slab},H}} \sum_{e=1}^{N_{\text{elem},H}} \varepsilon_{e,k},$$

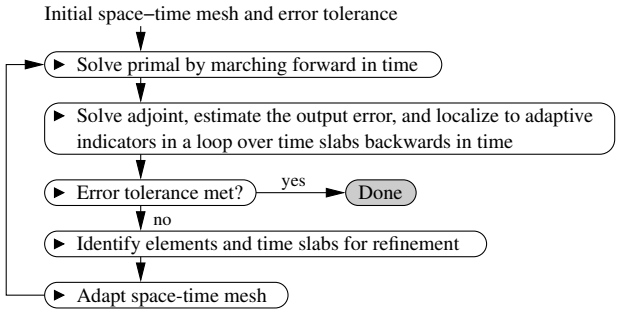


Figure 3: Steps of the proposed unsteady adaptive strategy.

where the error contribution of a coarse space-time element $\mathcal{T}_{H,e} \otimes \mathcal{I}_{H,k}$ is

$$\varepsilon_{e,k} = - \sum_{m \in \text{dof}_h(\mathcal{I}_k^H)} \left(\mathbf{Z}_e \boldsymbol{\Psi}_h^m \right)^T \mathbf{Z}_e \overline{\mathbf{R}}_h^m (\mathbf{U}_h^{H,n}).$$

The mask matrix $\mathbf{Z}_e$ contains N_h columns and returns the degrees of freedom associated with coarse element e. The error indicator for a space-time element is obtained by taking the absolute value of the elemental contribution,

$$\text{space-time element error indicator} = \epsilon_{e,k} = \left| \varepsilon_{e,k}^H \right|. \tag{14}$$

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 in space and time.

The indicator in (14) is sufficient for isotropic space-time refinement in which elements are targeted for refinement in both space and time. However, isotropic refinement can produce inefficient meshes. For example, if the discretization is under-resolved in the time domain, spatial elements would needlessly be refined on account of the temporal error. Therefore, important for an efficient adaptation strategy is a measure of the space-time anisotropy of the error. We present two such 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 state, $\mathbf{u}_H(\mathbf{x}, t)$, 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 [33]. 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

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

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

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

Π_H^{space} and Π_H^{time} are least-squares projection operators to spatial order p and temporal order 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

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

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. In this measure, the spatial/temporal error fractions are given by

$$\beta_{e,k}^{\text{space}} = \frac{|\mathcal{E}_{e,k}^{\text{space}}|}{|\mathcal{E}_{e,k}^{\text{space}}| + |\mathcal{E}_{e,k}^{\text{time}}|}, \qquad \beta_{e,k}^{\text{time}} = 1 - \beta_{e,k}^{\text{space}}.$$

Finally, as discussed in the next section, in the present work we use tensor-product space-time meshes in which the temporal discretization remains in slabs and the spatial resolution is static. Independent refinement of individual space-time elements is not possible in this setting, and instead aggregate adaptive indicators are sought for each spatial element and for each time slab. These aggregate indicators are obtained by summing over time slabs and elements as follows:

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

In the results we will also present conservative error estimates calculated as the sum of the element indicators:

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

4.2. Adaptation

In theory, with a discontinuous Galerkin space-time discretization, the space-time elements could 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 the iterative method described in Section 2.3, temporal refinement is limited to entire time slabs. That is, at each time, all elements take time steps of the same size. Second, to minimize storage and complexity of the data structures, the refinement of the spatial mesh remains fixed throughout the unsteady simulation. Clearly this latter requirement limits the efficiency of the adaptation, especially for problems that exhibit spatially localized sources of error that move in time. Dynamic spatial refinement is a topic of future work.

The mesh is adapted by hanging-node refinement of spatial elements and bisection of time slabs. 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 (16). In the present 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, $(1 - f^{\text{growth}})$ times the current number of 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.

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.

4.3. 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 [34].

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, which have been considered previously for shock-capturing purposes [35, 36, 37]. Specifically, we define the jump indicator on space-time element (e, k) as

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

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 on 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,

$$\epsilon^{\text{res}}_{e,k} = \sum_{m \in \text{dof}_h(\mathcal{I}^H_k)} \mathbf{Z}_e \left| \overline{\mathbf{R}}^m_h(\mathbf{U}^{H,n}_h) \right|.$$

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.

4.4. Temporal Robustness Adaptation

On each time slab, the primal and adjoint solvers decrease the unsteady residuals to a low relative tolerance using the iterative solver described in Section 2.3. We do not take advantage of estimating the output error due to inexact solves [24], and instead for simplicity we use a fixed relative residual tolerance of eight orders of magnitude. Since the iterative solver is based on a Newton-Raphson method, it may occasionally fail to converge for time slabs that are large relative to the spatial resolution. This is usually observed for initial coarse meshes that are under-resolved in time. In such cases, a temporal robustness adaptation is performed: the current time slab is bisected and the iterative solver begins anew on the smaller time slab. The adaptation only affects the current time slab, so that each robustness adaptation adds only one time slab to the temporal discretization. Both the primal and adjoint solvers employ this measure.

5. Results

This section presents results of adjoint-based unsteady adaptive runs for four problems. All results are in two spatial dimensions on static body-fitted quadrilateral meshes. The solutions are approximated using $p = 2$ tensor product polynomials in space and $r = 1$ polynomials in time. Solutions obtained using the various adaptive indicators are compared to "actual" solutions, which are computed on space-time meshes that are obtained by uniformly-refining the final output-adapted meshes. The growth factor used in the adaptation iterations is $f^{\text{growth}} = 2.0$. A near-zero error tolerance is prescribed for the output adaptation so that the adaptive process does not terminate; instead a fixed number of adaptive iterations is run to enable comparisons at similar degrees of freedom.

We do not present timing results as these are subject to variability resulting from specifics of code implementation and optimization, and to approximations made in the adjoint solution, which have to be further investigated. As a rough estimate, with the authors' current implementation, the cost of adjoint-based error estimation for the problems considered is on average five times that of a forward solve on the same mesh.

5.1. Two-Dimensional Scalar Convection-Diffusion-Reaction

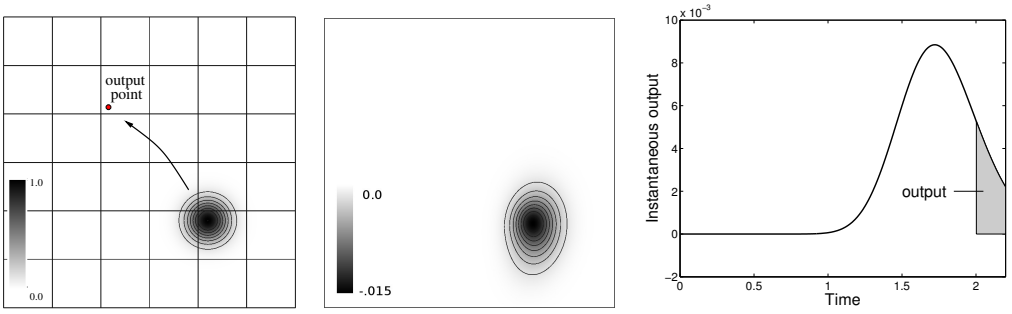
The first example is a demonstration of the adaptive method for a two-dimensional scalar convection-diffusion-reaction problem. The governing equation for the scalar concentration u is

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

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

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

where $A = 0.1$, $c_1 = 2$, $E = .05$, and $c_2 = 2.4$. The computational domain is a square, $(x, y) \in [0, 2] \otimes [0, 2]$, and the initial condition is a Gaussian concentration profile centered at $(x, y) = (2.1, 0.9)$ with unit amplitude and variance of 1/50, as shown in Figure 4(a). The velocity field is induced by an irrotational counterclockwise vortex flow centered at $(x, y) = (-1, -1)$ and with a vortex strength of $\Gamma = 3$; i.e. at a point a distance l away from the vortex center, the velocity magnitude is $|\vec{V}| = \Gamma/(2\pi l)$ and the direction is counterclockwise tangential relative to the vortex



(a) Initial condition and mesh (b) Adjoint at initial time (c) Output definition

Figure 4: Scalar convection-diffusion-reaction: initial condition and point output overlayed on the coarse mesh, the adjoint solution at the initial time, and the definition of the output time-integral quantity. The velocity field is induced by an irrotational vortex outside the lower left corner of the domain.

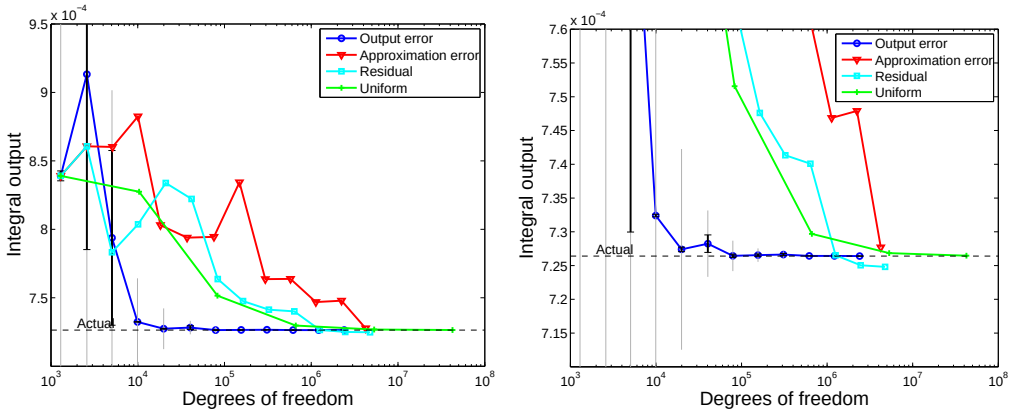
center. The kinematic viscosity is 0.01, so that the Peclet number based on the vortex strength is $Pe \equiv \Gamma/\nu = 300$. The output is a time-integral near the end of the simulation, from $t = 2$ to $t = 2.2$, of the scalar measured at one point, $(x, y) = (1.05, 2.05)$, as shown in Figure 4(a,c).

The initial mesh for the adaptive runs is a 6×6 uniform spatial grid with 4 time slabs. Figure 5b 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.

The output-adapted results in Figure 5b are shown with error bars at $\pm\delta J$, using the adjoint-weighted residual estimate in (13), and at the more conservative $\pm\epsilon$, using the sum of element indicators in (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.

The time histories of the scalar output reading on several adapted space-time meshes are shown in Figure 6a. The temporal solution is discontinuous in time, and the plot shows the instantaneous output at the end of each time slab, where the solution is superconvergent at order $2r + 1$ [32]. To obtain a representative continuous curve, these superconvergent points are interpolated using a temporal cubic spline, which is expected to retain the optimal order $2r + 1$ accuracy on the time slab interiors. The space-time meshes in Figure 6a 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.

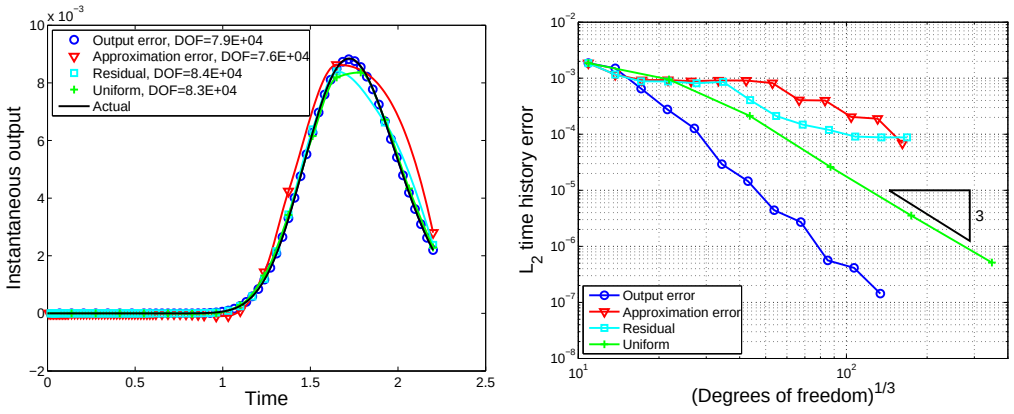
A more quantitative comparison is given in Figure 6b, which plots the L_2 error in the time histories versus the cube root of the degrees of freedom. The cube root is used to aid in the illustration of the convergence rate under uniform refinement of three dimensions: two spatial and one temporal. We remark that the observed third-order convergence for uniform refinement



(a) Output convergence (b) Output convergence (zoom)

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. "Actual" output value is computed on a uniformly-refined final output-adapted space-time mesh.

is consistent with the expected rate for a spatially and temporally smooth problem with $p = 2$ and $r = 1$, since the spatial error is expected to converge at a rate of $p + 1 = 3$, and the temporal error at a rate of $2r + 1 = 3$ for a time integral quantity. From this figure, we see that 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.

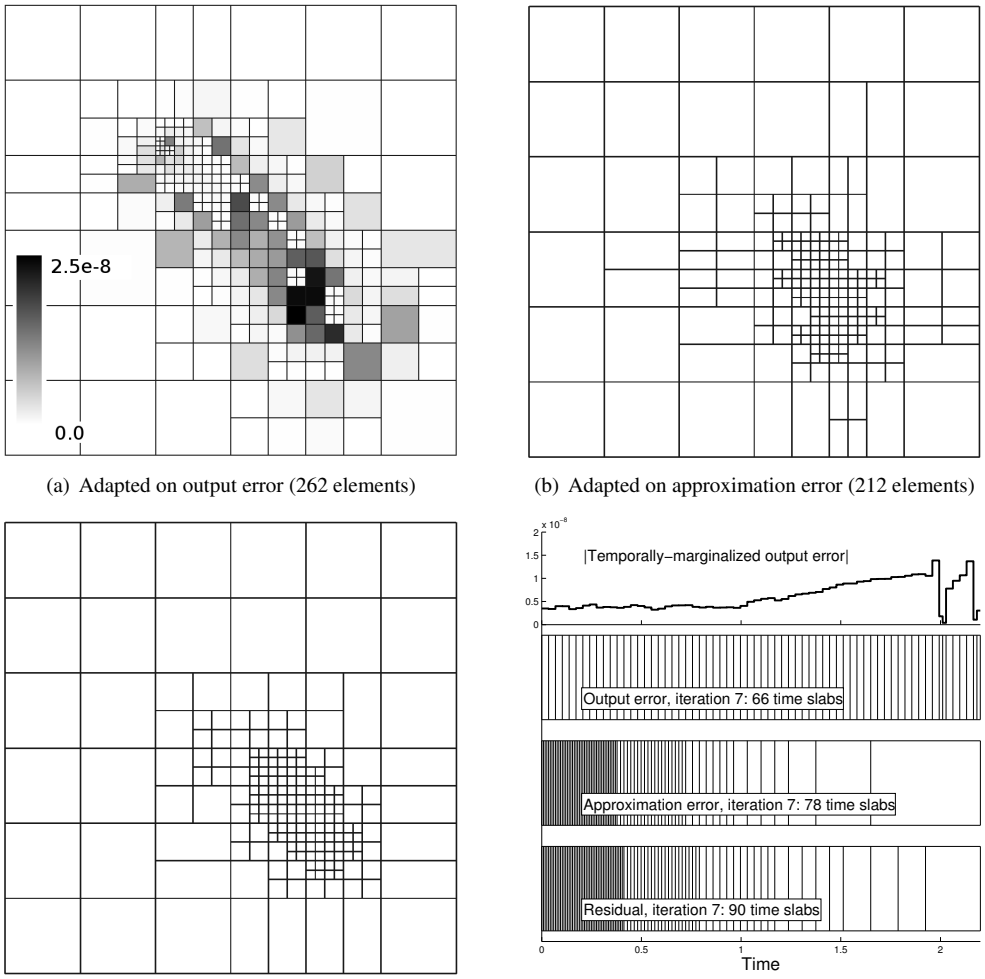


(a) Sample output histories (b) L_2 error convergence

Figure 6: Scalar convection-diffusion-reaction: sample point-value time histories and convergence of the L_2 time history error for various adaptive indicators. "Actual" time history is computed on a uniformly-refined final output-adapted space-time mesh.

The adapted space-time meshes corresponding to the time histories in Figure 6a 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 due to the subsequent reaction. On the other hand, the output-based adaptation also 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.



(a) Adapted on output error (262 elements)

(b) Adapted on approximation error (212 elements)

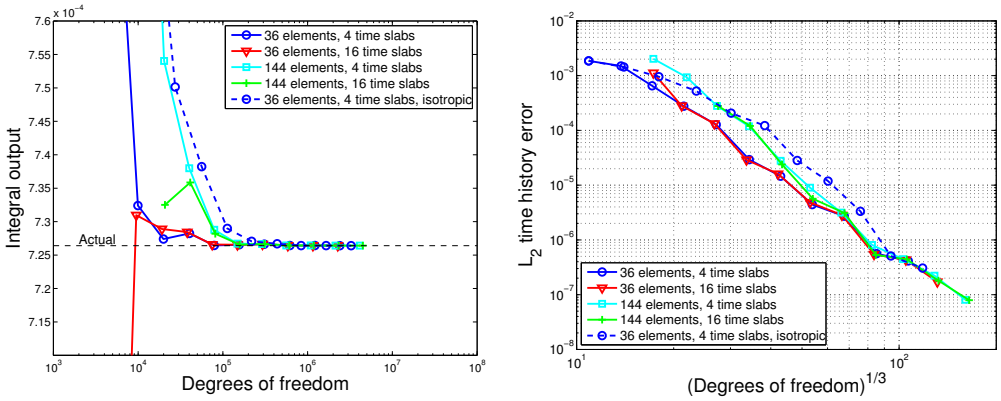
(c) Adapted on residual (202 elements)

(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 compares convergence histories of output-based adaptation for four different initial space-time meshes. These meshes consist of: the baseline mesh of 36 elements and 4 time-slabs; the baseline refined temporally to 16 time slabs; the baseline refined spatially to 144 elements; and the baseline refined both spatially and temporally. By 2×10^5 degrees of freedom, the outputs from the various refinements are indistinguishable on the plot in Figure 8a. In addition, Figure 8b shows the convergence of the L_2 error in the output time histories. The differences in the initial

degrees of freedom are quickly washed out as the adaptive scheme targets the same space-time regions for refinement.



(a) Output convergence

(b) L_2 error convergence

Figure 8: Scalar convection-diffusion-reaction: relative insensitivity of the output convergence to initial mesh when using output-based adaptation. The legend entries refer to the size of the initial mesh. Isotropic refers to ascribing the space-time element error $\epsilon_{e,k}$ equally to the spatial and temporal discretization instead of using the anisotropy measure described in Section 4.1.

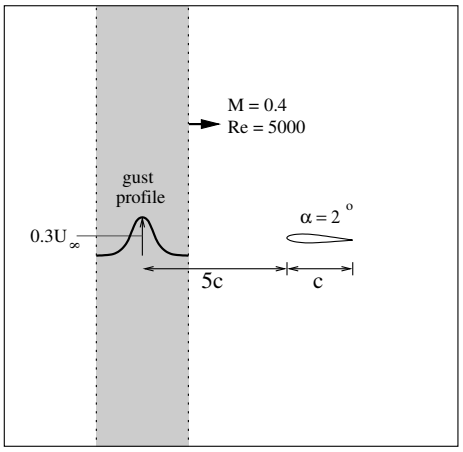
Also shown in Figure 8a as a dashed curve is the convergence of the output using the baseline initial mesh but without space-time anisotropy detection. Instead, the space-time elemental error indicator $\epsilon_{e,k}$ is split equally between the spatial and temporal discretizations: $\beta_{e,k}^{\text{space}} = \beta_{e,k}^{\text{time}} = 0.5$. As indicated in the figure, the resulting output convergence is slower by almost an order of magnitude in terms of degrees of freedom.

5.2. Subsonic Laminar Gust Encounter

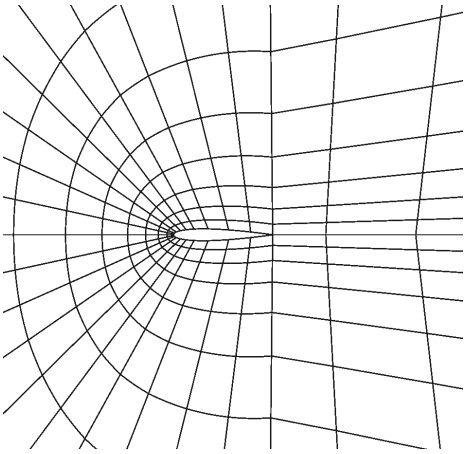
The second example is of a vertical gust encounter for a NACA 0012 airfoil in laminar viscous flow at freestream Mach number $M_\infty = 0.4$, angle of attack $\alpha = 2°$, and Reynolds number $Re = 5000$. The initial condition is illustrated schematically in Figure 9a. Prior to unsteady simulation, a steady solution is converged for a uniform freestream, imposed approximately 50 chord lengths away from the airfoil. The gust field is then superimposed on the solution by perturbing the vertical velocity with a Gaussian profile centered five chord lengths ahead of the airfoil. The standard deviation of the Gaussian is a quarter chord length, and the profile is shifted so that the perturbation is zero one chord length ahead of the airfoil. The result is an initial condition that satisfies the physical flow tangency and no-slip boundary conditions on the airfoil.

The coarse initial spatial mesh of 510 elements is illustrated in Figure 9b. The elements are curved using cubic shape polynomials to represent the geometry. The initial temporal mesh consists of eight uniform time steps. The primal state in the form of entropy contours is shown at the half-way point of the simulation in Figure 9b. A faint signature of the gust field is visible 1.5 chord lengths behind the airfoil. The wake at this time is clearly unsteady, and an alternating pattern of shed vortices is visible.

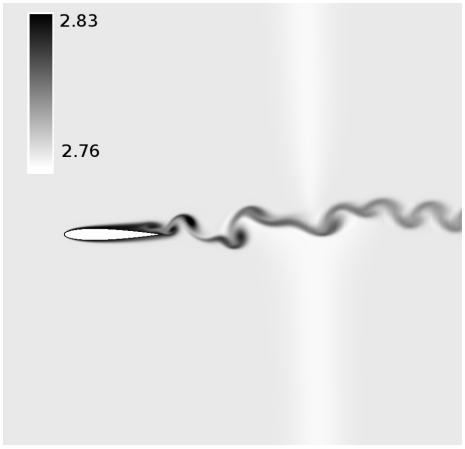
The temporal domain spans 15 time units, where one time unit is the chord length divided by the freestream velocity magnitude. The output of interest is chosen as the time integral of the lift coefficient from $t = 5$ to $t = 15$. This quantity is illustrated in Figure 10a for a time history



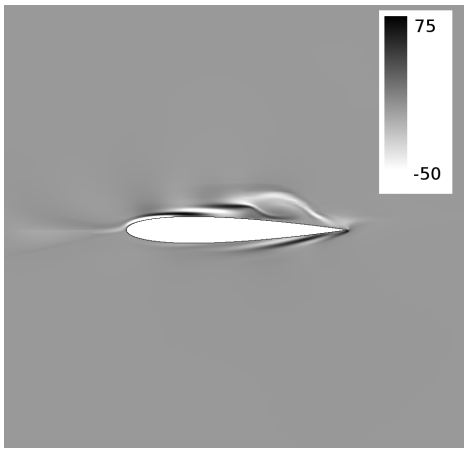
(a) Problem setup



(b) Initial mesh



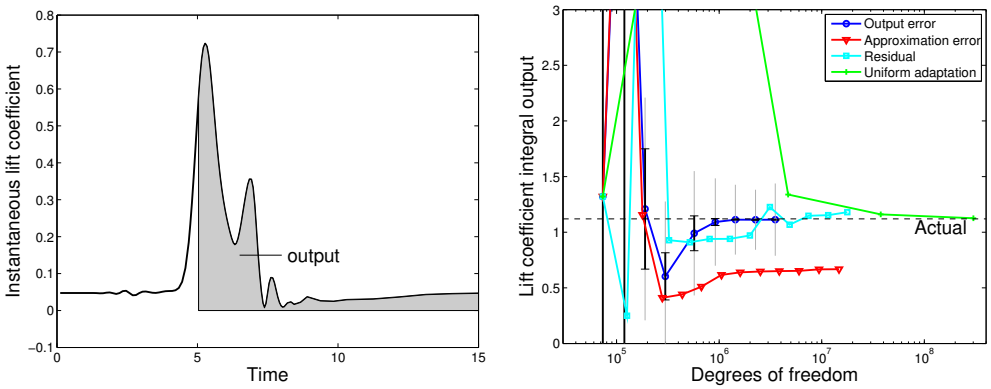
(c) Entropy at $t = 7.5$



(d) x-momentum adjoint at $t = 7.5$

Figure 9: Subsonic laminar gust encounter: problem setup showing a schematic of the initial condition, the starting coarse mesh, and snapshots of the primal and adjoint solutions.

computed with a fine solution. Note that the output time horizon captures most of the gust encounter and the subsequent time. A snapshot of the unsteady x-momentum adjoint solution, taken at $t = 7.5$, is shown in Figure 9d. The contours show that at this time the output is sensitive to x-momentum residuals in the boundary layer above the leading edge, off the top-middle surface, and below the trailing edge. This information is automatically incorporated into the error estimates that drive the adaptation.



(a) Output definition

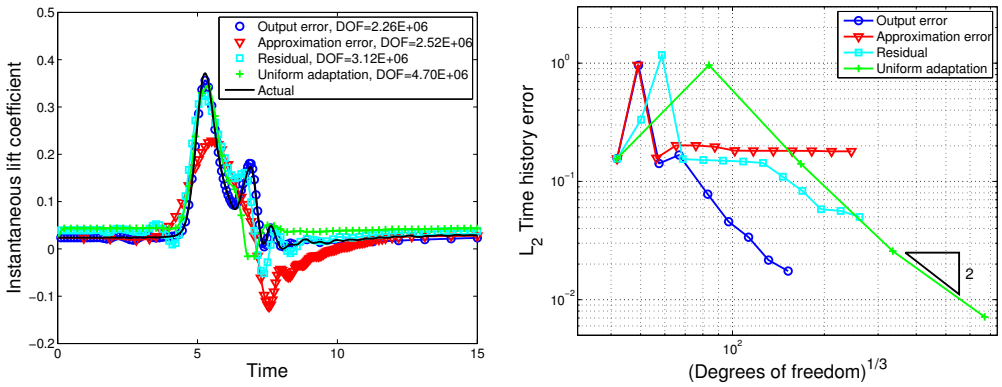
(b) Output convergence

Figure 10: Subsonic laminar gust encounter: time integral output definition and its convergence under various adaptive indicators. Error bars at $\pm\delta J$ and whiskers at $\pm\epsilon$ are shown for the output-based results. "Actual" output is computed on a uniformly-refined final output-adapted space-time mesh.

Starting with the initial space-time mesh, four adaptive runs were performed using the three indicators (output error, approximation error, and residual) and uniform refinement in space and time. The output at each adaptation iteration is shown in Figure 10b versus total degrees of freedom. Error bars at $\pm\delta J$ and $\pm\epsilon$ are included on the output-based plot. On the initial coarse meshes the output error is very high for all of the methods, and the error estimate reflects this. As convergence proceeds, the error and error estimate are quickly reduced for the output-based run as it converges to the actual value. Adaptation on approximation error produces a convergence plot that looks reasonable in isolation; however, the value to which it appears to converge is drastically incorrect. The residual indicator yields a somewhat more accurate output but exhibits oscillations in the latter stages of convergence. Finally, uniform refinement yields results that approach the true output, but with significantly more degrees of freedom. For example, for 10% output error, the factor difference in degrees of freedom between output-based and uniform refinement is approximately 40.

The time histories of the lift coefficient for several adapted space-time meshes at comparable degree of freedom counts are shown in Figure 11a. In this set, the output-based adapted mesh most closely tracks the true time history of the lift coefficient. The quantitative comparison in Figure 11b confirms that this is the case in the L_2 error norm. In this metric, residual-based adaptation comes next with an order of magnitude more degrees of freedom, followed by uniform refinement. Uniform refinement exhibits a convergence rate of approximately 2.0 in this plot against the cube root of degrees of freedom, which is less than the optimal expected rate of 3.0. This suboptimal behavior could be caused by high-order spatial derivative discontinuities in the initial gust profile. Finally, we note that adaptation on the approximation error indicator levels

off at a high error value and clearly fails to converge.



(a) Sample output histories

(b) L_2 error convergence

Figure 11: Subsonic laminar gust encounter: sample lift coefficient time histories and convergence of the L_2 time history error for various adaptive indicators. "Actual" time history is computed on a uniformly-refined final output-adapted space-time mesh.

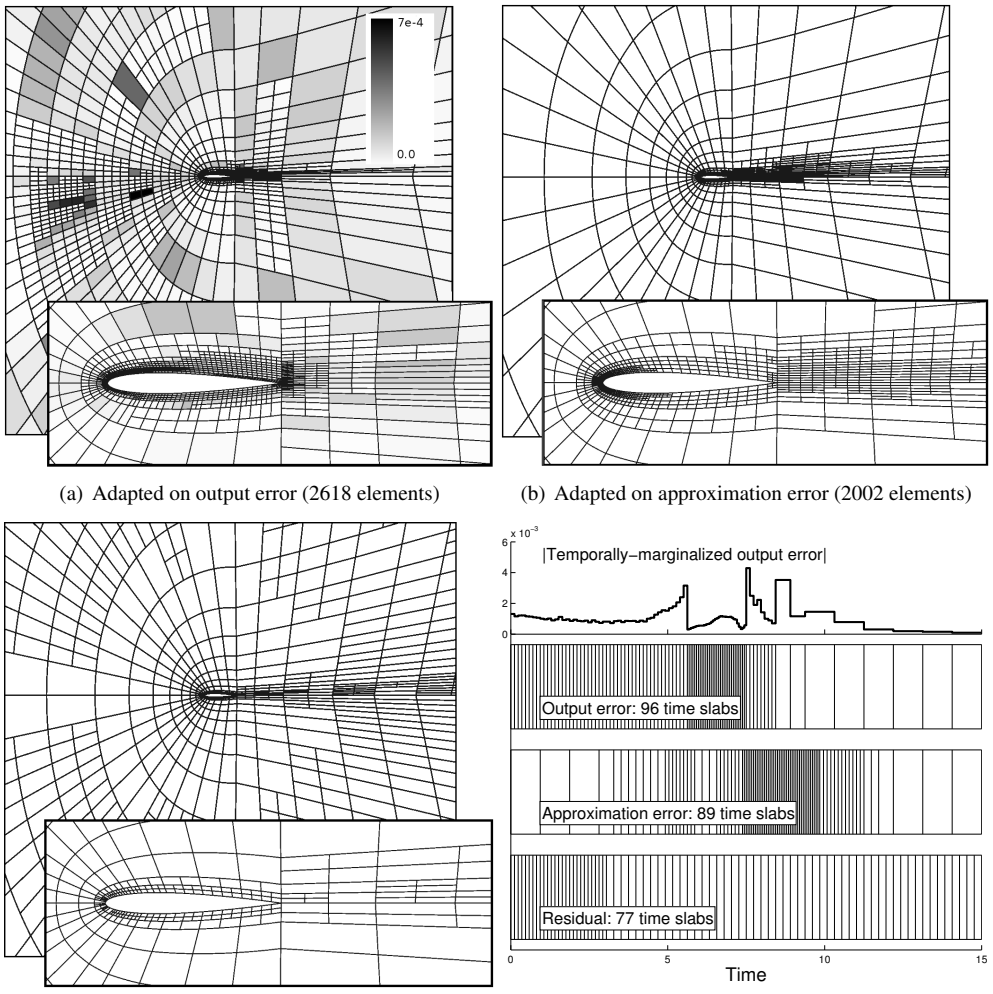
The adapted spatial and temporal meshes at the seventh adaptive iteration for each indicator are shown in Figure 12. The output error indicator localized to the spatial elements and time slabs is overlayed on the output-adapted meshes. Focusing first on the spatial meshes, we see marked differences in the areas targeted for refinement by the three indicators. The output-based indicator refines the propagation path of the gust ahead of the airfoil, the leading and trailing edges where the shed vortices are formed, and portions of the boundary layer and wake. On the other hand, the approximation error indicator focuses primarily on the airfoil leading edge and the wake: areas where the solution exhibits the largest spatial variation. It leaves the area in front of the airfoil quite coarse, as the approximation error there is lower; however, this results in errors in propagating the gust field, and these errors are significant for the lift prediction. Finally, the residual indicator is distracted by propagation of the entire gust field and targets areas far above and below the airfoil.

Looking next at the temporal meshes in Figure 12d, we again see important differences. The output-based adaptation focuses on the time before the encounter, so as to accurately propagate the gust, and on the time at the peak of the encounter. It leaves the time steps in the latter portion of the simulation, when the lift is not changing much, relatively coarse. The approximation-error indicator focuses instead on the time immediately after the gust encounter, when the vortex shedding is strongest. Finally, the residual indicator produces a nearly uniform temporal refinement, suitable for propagating the gust field throughout the simulation.

5.3. Transonic Inviscid Gust Encounter

The third example is another gust encounter simulation with a NACA 0012 airfoil, but this time in inviscid transonic flow at $M_\infty = 0.95$ and $\alpha = 0°$. The gust setup is as illustrated in Figure 9a, except that the peak gust velocity is now $0.5U_\infty$, where U_∞ is the freestream velocity magnitude. A steady solve is performed prior to each unsteady simulation.

Figure 13 shows the primal and adjoint solution snapshots at one time during the gust encounter. The vertical gust field, situated towards the aft of the airfoil, is visible on the Mach



(a) Adapted on output error (2618 elements)

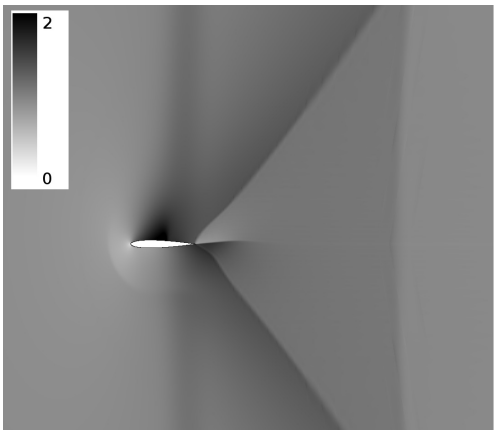
(b) Adapted on approximation error (2002 elements)

(c) Adapted on residual (2875 elements)

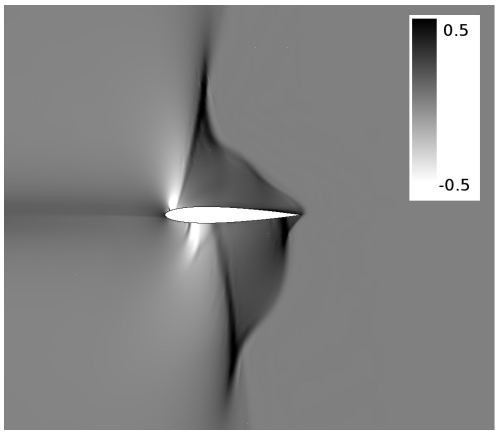
(d) Temporal meshes

Figure 12: Subsonic laminar gust encounter: 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.

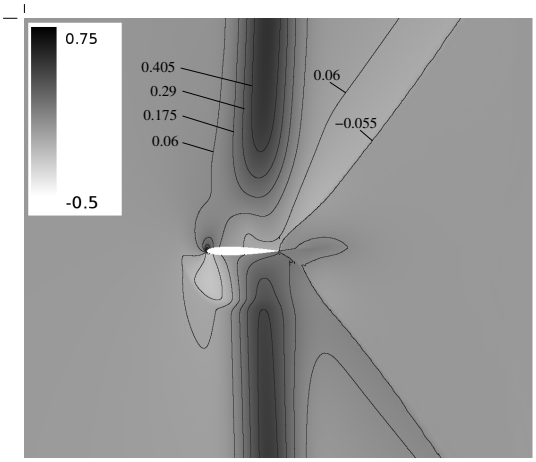
number contours. A "fishtail" shock structure is also evident behind the airfoil, and it is slightly perturbed by the gust.



(a) Mach number contours at $t = 6$



(b) Mass-conservation adjoint at $t = 6$



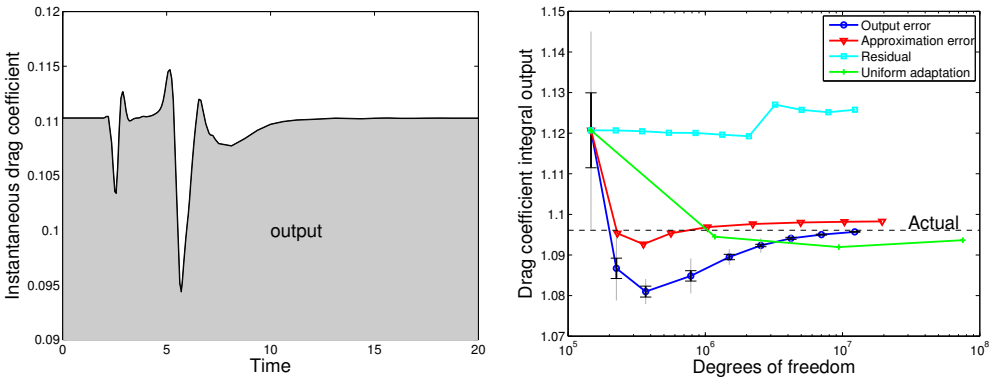
(c) y-Momentum contours at $t = 6$

Figure 13: Transonic inviscid gust encounter: primal and adjoint states in the middle of the simulation. Contours of y-momentum, in units of freestream velocity magnitude, indicate that at this time the airfoil is just past the middle of the gust.

The temporal domain spans 20 time units, where again each time unit is the chord length divided by the freestream speed. The output of interest is the time integral of the drag coefficient over the entire simulation, as illustrated in Figure 14a. A snapshot of the mass-conservation adjoint solution for this output is shown in Figure 13b. We note an asymmetric "lambda" structure of the adjoint solution that is formed by characteristic curves in the supersonic regions.

The initial spatial mesh of 510 elements is the same as in the previous example, and the initial temporal mesh contains 32 time slabs. The output convergence from four adaptive runs with the different indicators is given in Figure 14b. The unweighted residual-based adaptation does not converge at all, while the approximation-error adaptation "converges" but to the wrong value.

The output-based adaptation converges to the actual value, with a similar advantage in terms of degrees of freedom as in the previous examples. The error bars on the outputs under-predict the actual error, as do the more conservative whiskers given by the sum of the adaptive indicators. This observation is attributed to the more complex structure of this problem and to the fact that error estimation in the presence of shocks is still an open area of research for high order methods. Nevertheless, adaptive refinement based on these error indicators performs very well.



(a) Output definition

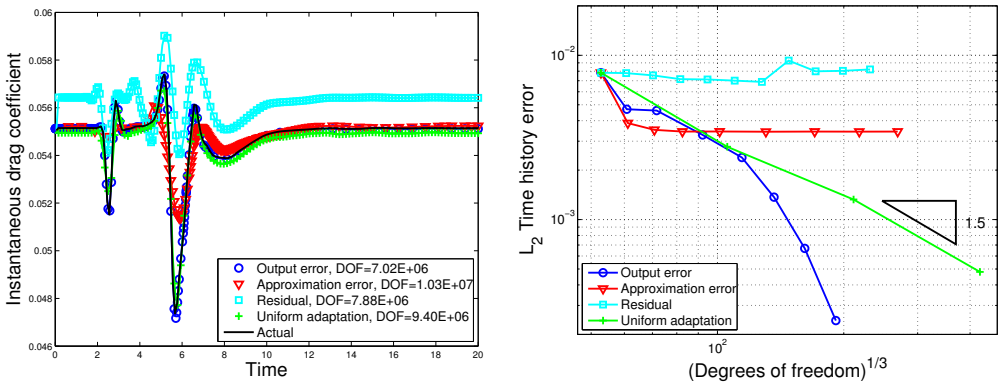
(b) Output convergence

Figure 14: Transonic inviscid gust encounter: 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. "Actual" output is computed on a uniformly-refined final output-adapted space-time mesh.

Time histories of the drag coefficient are shown in Figure 15a, for adapted space-time meshes with similar degrees of freedom. The residual-adapted result is incorrect right away even in the initial steady drag coefficient, leading to a consistent over-prediction of the output. The approximation-error adaptation does not predict the peak drag changes well, although the extent of this error is masked somewhat in the integral output. On the other hand, uniform refinement and output-based adaptation track the actual time history well.

Figure 15b shows the L_2 time history error convergence. Non-convergence of the adaptations based on residual and approximation-error is evident. Uniform refinement shows a convergence rate of about 1.5 on the final refinement, a suboptimal rate that is likely caused by spatial discontinuities at the shocks. In this metric, the difference between output-based adaptation and uniform refinement is lower: about an order of magnitude for sufficiently low error tolerance. However, the success of uniform refinement rests on the quality of the initial space-time mesh, which in this case was relatively well-suited for the problem. A mismatch in the spatial and temporal resolutions on the initial mesh would remain uncorrected in uniform refinement compared to output-based adaptation.

The adapted meshes corresponding to the time histories in Figure 15a are shown in Figure 16. Output-based adaptation again targets propagation of the gust field ahead of airfoil, and the leading and trailing edges of the airfoil. More notably, the adaptation does not target much of the wake, leaving the fishtail shock structure quite under-resolved. Conversely, adaptation on the approximation error is distracted by the shock structure, leaving the area ahead of and above/below the airfoil relatively coarse. The residual-based adaptation targets the entire gust field (the figures do not show the entire spatial domain) and not the airfoil vicinity, resulting in



(a) Sample output histories

(b) L_2 error convergence

Figure 15: Transonic inviscid gust encounter: sample drag coefficient time histories and convergence of the L_2 time history error for various adaptive indicators. "Actual" time history is computed on a uniformly-refined final output-adapted space-time mesh.

errors in the steady drag value.

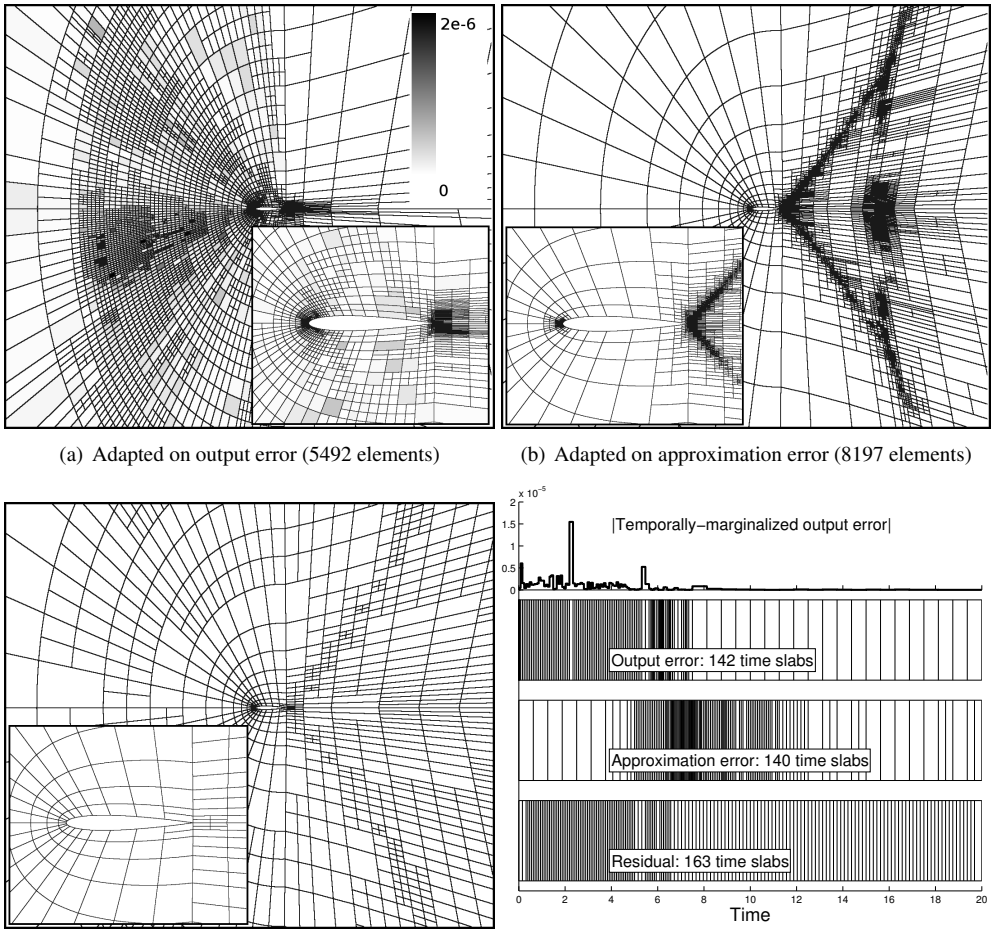
The adapted temporal meshes in Figure 16d show that the output-based indicator targets the time before and during the encounter. The approximation error indicator focuses on the gust encounter and the time shortly thereafter, leaving the propagation of the gust prior to the encounter prone to error. Finally, the residual-based indicator shows mostly uniform refinement as it attempts to propagate the gust accurately throughout the entire simulation.

5.4. Impulsively-Started Airfoil

The final example is an impulsively-started NACA 0012 airfoil in viscous flow, where for $t \geq 0$, the freestream conditions are $M_\infty = 0.25$, $\alpha = 8°$, $Re = 5000$. 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, $r_1 \leq r \leq r_2$ 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 $\mathbf{V}_\infty$ is the freestream velocity. No steady solve is performed prior to the unsteady simulation.

Figure 17a 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 18a. A snapshot adjoint solution for the y-momentum equation at $t = 6$ is illustrated in Figure 17b. 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 same initial spatial mesh of 510 elements is used as in the previous examples, and the initial temporal mesh contains 16 time slabs. The output convergence for the various indicators is shown in Figure 18b. The residual indicator does not perform well at all again: 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



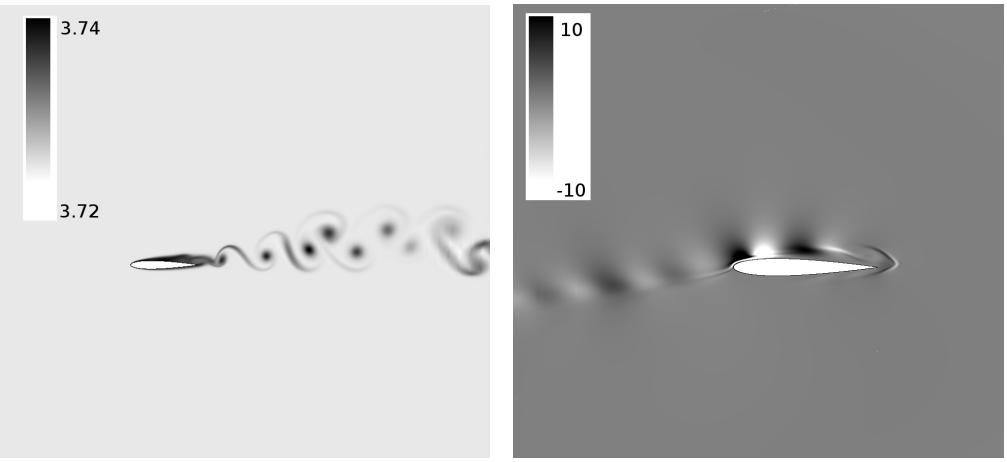
(a) Adapted on output error (5492 elements)

(b) Adapted on approximation error (8197 elements)

(c) Adapted on residual (5373 elements)

(d) Temporal meshes

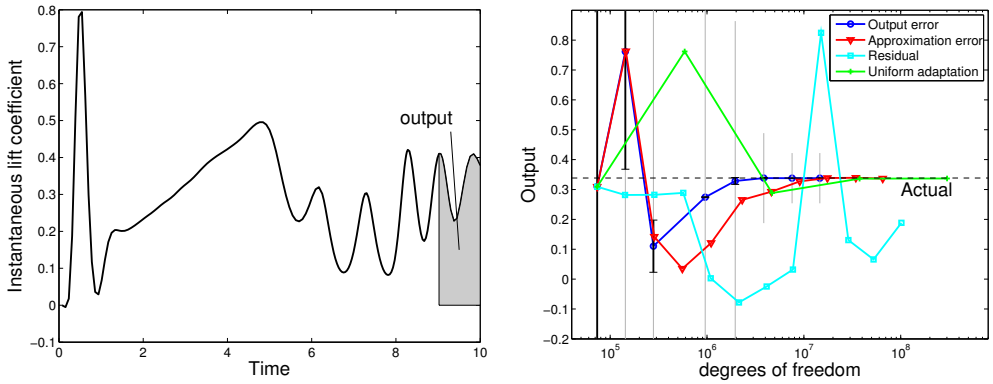
Figure 16: Transonic inviscid gust encounter: adapted spatial and temporal meshes for the seventh adaptive iteration. Localized output error estimates ϵ_e and ϵ_k are shown for the output-error adapted meshes.



(a) Entropy contours at $t = 10$ (b) y-momentum adjoint at $t = 6$

Figure 17: Impulsively-started airfoil: primal state at the final time and the adjoint state at $t = 6$.

pronounced in this case, with a factor of 3-4 savings over the approximation error indicator. The error estimates under-predict the error in the middle stages of output-based refinement, while the conservative whiskers at $\pm\epsilon$ are more robust.

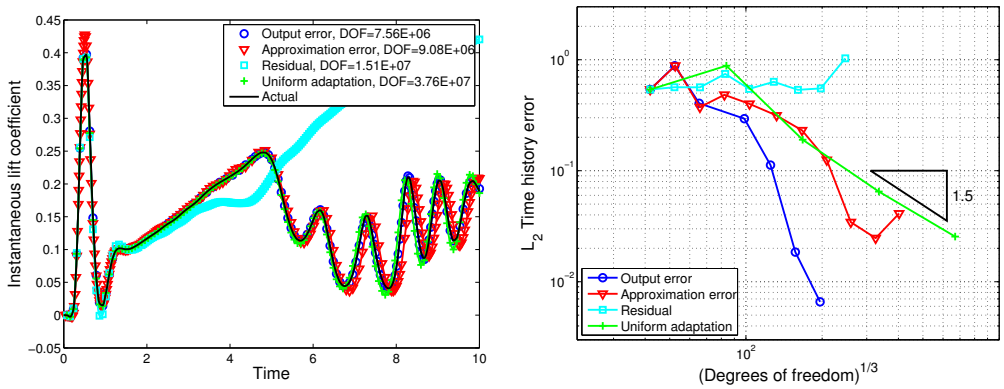


(a) Output definition (b) Output convergence

Figure 18: 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. "Actual" output is computed on a uniformly-refined final output-adapted space-time mesh.

Figure 19a shows the time histories of the lift coefficient for adapted space-time meshes of similar size. The source of the error in the residual-adapted case is clear: it does not predict oscillatory vortex shedding, but rather an increasing lift coefficient. The other three adaptive indicators track the actual time history well. Figure 19b shows the L_2 time history error convergence for all of the methods, versus the cube root of the degrees of freedom. Under uniform refinement, the observed convergence rate is a suboptimal 1.5, which could be caused by high-order

derivative discontinuities in the blended initial condition or the resolution not yet being in the asymptotic regime. Of primary interest, however, is the performance of output-based adaptation, which remains the fastest out of the methods tested. 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.



(a) Sample output histories

(b) L_2 error convergence

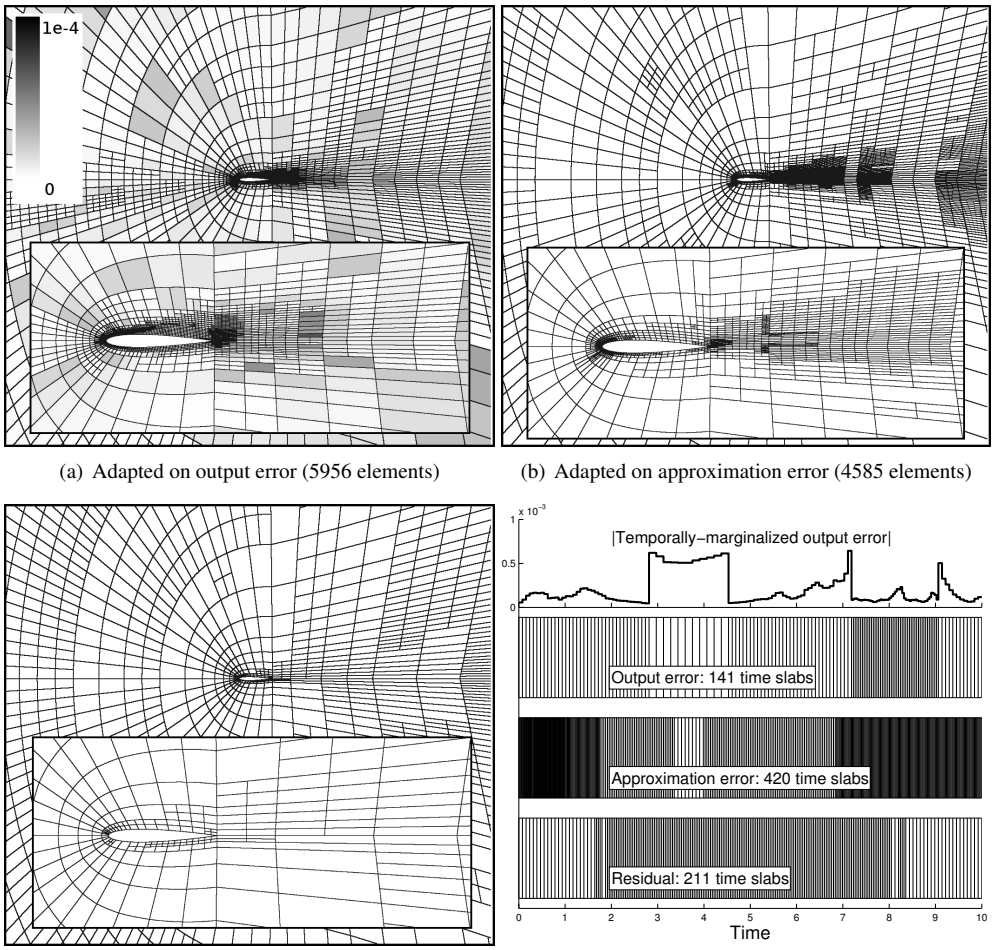
Figure 19: Impulsively-started airfoil: sample lift coefficient time histories and convergence of the L_2 time history error for various adaptive indicators. "Actual" time history is computed on a uniformly-refined final output-adapted space-time mesh.

The corresponding adapted meshes are shown in Figure 20. 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.

The temporal meshes are shown in Figure 20d. 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 again creates a mostly-uniform temporal mesh as it tracks the acoustic waves.

6. Conclusions

This paper presents an output-based adaptive solution strategy for unsteady simulations of convection-dominated flows, including those governed by the compressible Navier-Stokes equations. The discretization is performed on tensor product space-time meshes in which the spatial resolution is static and the temporal refinement is in slabs. Output error estimates are obtained using an unsteady adjoint-weighted residual in which the discrete adjoint is obtained on a refined space using the same approximate solver used for the primal. The adaptive strategy requires multiple unsteady runs, with refinement of both the spatial and temporal meshes after every run.



(a) Adapted on output error (5956 elements)

(b) Adapted on approximation error (4585 elements)

(c) Adapted on residual (7929 elements)

(d) Temporal meshes

Figure 20: 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 proposed strategy adds non-trivial costs to the solver: the forward state must be stored at each time step for linearization in the adjoint solve, and the adjoint must be solved on an enriched space. One source of cost reduction is approximation in the adjoint solution, and this work presents a temporal reconstruction that avoids solving the adjoint on a temporally-enriched space. However, the adjoint solution remains more expensive than the primal on the cases tested. This expense must be weighed against the benefits, which consist of error estimates for outputs of interest and long-run cost reductions through a targeted adaptive solution strategy.

In this work, adaptive results are presented for a scalar convection-diffusion-reaction problem and for the compressible Euler and Navier-Stokes equations in two spatial dimensions. An important observation is that, at least for the cases tested, output-based adaptation is demonstrated to be advantageous in terms of degrees of freedom. The factor of advantage over the next closest strategy ranges from four to a couple orders of magnitude, depending on the case and on the error tolerance. Computational time considerations reduce this advantage, but the precise amount is not presented as it is sensitive to implementation and approximation choices. Further research into these choices, and into the efficiency of the primal problem, is required to make quantitative computational time comparisons. Moreover, we note that for highly nonlinear problems, such as the unsteady Reynolds-averaged Navier-Stokes (RANS) equations, the cost of the linear adjoint solution relative to the primal is expected to be very much reduced due to the expense of the primal solve.

The comparison strategies include uniform refinement and indicators based on the unweighted residual and a measure of the approximation error. Neither the unweighted-residual indicator nor the approximation-error indicator is found to be robust. The residual indicator is distracted by the entire gust field for the gust-encounter simulation, and by acoustic transients from the impulsively-started airfoil. The approximation-error indicator targets space-time areas of poor resolution that do not necessarily affect the output of interest; moreover, the adaptive results often "converge" to an incorrect value. In contrast, the output-based indicator targets only those areas of the space-time domain important for the prediction of the output of interest. Space-time anisotropy is shown to be an important factor affecting the performance of output-based adaptation; when properly incorporated into the adaptive algorithm, the convergence is found to be relatively insensitive to the initial space-time mesh. In addition, due to the conservative localization of the error, the entire time history is well-predicted in addition to the single scalar integral quantity.

From the point of view of efficiency, the most important area of future work is the incorporation of dynamic spatial refinement. This will require changes in the implementation, to accommodate time-varying refinement, and in the theory, in terms of error localization and adaptive optimization. However, significant savings are expected, especially for problems with moving features such as shocks or vortices. Mesh motion is another path forward, to allow analysis of more interesting unsteady problems including aeroelasticity and fluid-structure interaction. Additional physics, such as unsteady RANS, and more complex configurations, including three spatial dimensions, are also candidates for future work. We expect the adjoint-based adaptation to be even more attractive for these challenging problems in which the primal problems are more expensive and proper allocation of degrees of freedom is paramount for efficiency and robustness.

Appendix A. Approximate iterative solution schemes for $r = 1$ and $r = 2$

This appendix presents the multistep solution schemes of the system in (11) based on the approximate factorization discussed in Section 2.3. For $r = 1$, the scheme consists of three steps

and two matrix inversions:

$$\begin{aligned}
\mathbf{C}^1_{H,1}\mathbf{Y}_H + \left(1 - \frac{1}{3}\mathbf{B}_H\right)\overline{\mathbf{R}}^1_H + \left(1 + \frac{2}{3}\mathbf{B}_H\right)\overline{\mathbf{R}}^2_H &= 0 \\
\mathbf{C}^1_{H,1}\Delta\mathbf{U}^2_H - \mathbf{M}_H\mathbf{Y}_H &= 0 \\
\mathbf{C}^2_{H,1}\Delta\mathbf{U}^1_H + 2\overline{\mathbf{R}}^1_H + \left[\mathbf{M}_H + \frac{\Delta t_1}{3}\mathbf{A}_H\right]\Delta\mathbf{U}^1_H &= 0
\end{aligned}$$

where $\mathbf{B}_H = \Delta t_1 \mathbf{A}_H \mathbf{M}_H^{-1}$ and

$$\mathbf{C}^1_{H,1} = \left[\mathbf{M}_H + \frac{\Delta t_1}{\sqrt{6}}\mathbf{A}_H\right], \quad \mathbf{C}^2_{H,1} = \left[\mathbf{M}_H + \frac{2\Delta t_1}{3}\mathbf{A}_H\right].$$

For $r = 2$, six steps and three matrix inversions are required:

$$\begin{aligned}
\mathbf{C}^1_{H,2}\mathbf{Y}^1_H + \left(\mathbf{I}_H - \frac{2}{5}\mathbf{B}_H + \frac{1}{20}\mathbf{B}^2_H\right)\overline{\mathbf{R}}^1_H + \\
\left(\mathbf{I}_H + \frac{1}{10}\mathbf{B}_H - \frac{1}{40}\mathbf{B}^2_H\right)\overline{\mathbf{R}}^2_H + \left(\mathbf{I}_H + \frac{3}{5}\mathbf{B}_H - \frac{3}{20}\mathbf{B}^2_H\right)\overline{\mathbf{R}}^3_H &= 0 \\
\mathbf{C}^1_{H,2}\mathbf{Y}^2_H - \mathbf{M}_H\mathbf{Y}^1_H &= 0 \\
\mathbf{C}^1_{H,2}\Delta\mathbf{U}^3_H - \mathbf{M}_H\mathbf{Y}^2_H &= 0 \\
\mathbf{C}^2_{H,2}\mathbf{Y}^3_H + \left(\frac{3}{2}\mathbf{I}_H + \frac{3}{20}\mathbf{B}_H +\right)\overline{\mathbf{R}}^1_H + \left(\frac{9}{8}\mathbf{I}_H + \frac{3}{10}\mathbf{B}_H\right)\overline{\mathbf{R}}^2_H + \left(\frac{1}{2}\mathbf{I}_H + \frac{1}{4}\mathbf{B}_H + \frac{1}{40}\mathbf{B}^2_H\right)\Delta\mathbf{U}^3_H &= 0 \\
\mathbf{C}^2_{H,2}\Delta\mathbf{U}^2_H - \mathbf{M}_H\mathbf{Y}^3_H &= 0 \\
\mathbf{C}^3_{H,2}\Delta\mathbf{U}^1_H + 2\overline{\mathbf{R}}^1_H + \left(\frac{4}{3}\mathbf{I}_H + \frac{2}{15}\mathbf{B}_H\right)\mathbf{M}_H\Delta\mathbf{U}^2_H + \left(-\frac{1}{3}\mathbf{I}_H - \frac{1}{15}\mathbf{B}_H\right)\mathbf{M}_H\Delta\mathbf{U}^3_H &= 0
\end{aligned}$$

where

$$\mathbf{C}^1_{H,2} = \left[\mathbf{M}_H + 60^{-1/3}\Delta t_1 \mathbf{A}_H\right], \quad \mathbf{C}^2_{H,2} = \left[\mathbf{M}_H + \sqrt{\frac{3}{20}}\Delta t_1 \mathbf{A}_H\right], \quad \mathbf{C}^3_{H,2} = \left[\mathbf{M}_H + \frac{4}{15}\Delta t_1 \mathbf{A}_H\right].$$

References

[1] N. A. Pierce, M. B. Giles, Adjoint recovery of superconvergent functionals from PDE approximations, SIAM Review 42 (2) (2000) 247–264.

[2] R. Becker, R. Rannacher, An optimal control approach to a posteriori error estimation in finite element methods, in: A. Iserles (Ed.), Acta Numerica, Cambridge University Press, 2001, pp. 1–102.

[3] R. Hartmann, P. Houston, Adaptive discontinuous Galerkin finite element methods for the compressible Euler equations, Journal of Computational Physics 183 (2) (2002) 508–532.

[4] D. A. Venditti, D. L. Darmofal, Anisotropic grid adaptation for functional outputs: application to two-dimensional viscous flows, Journal of Computational Physics 187 (1) (2003) 22–46.

[5] S. Sen, K. Veroy, D. Huynh, S. Deparis, N. Nguyen, A. Patera, "Natural norm" a posteriori error estimators for reduced basis approximations, Journal of Computational Physics 217 (2006) 37–62.

[6] M. Nemec, M. J. Aftosmis, Error estimation and adpative refinement for embedded-boundary Cartesian meshes, AIAA Paper 2007-4187 (2007).

[7] K. J. Fidkowski, D. L. Darmofal, Review of output-based error estimation and mesh adaptation in computational fluid dynamics, American Institute of Aeronautics and Astronautics Journal 49 (4) (2011) 673–694.

[8] W. Reed, T. Hill, Triangular mesh methods for the neutron transport equation, Los Alamos Scientific Laboratory Technical Report LA-UR-73-479 (1973).
[9] F. Bassi, S. Rebay, High–order accurate discontinuous finite element solution of the 2-D Euler equations, Journal of Computational Physics 138 (1997) 251–285.
[10] B. Cockburn, C.-W. Shu, Runge-Kutta discontinuous Galerkin methods for convection-dominated problems, Journal of Scientific Computing 16 (3) (2001) 173–261.
[11] P. Houston, R. Hartmann, E. Süli, Adaptive discontinuous Galerkin finite element methods for compressible fluid flows, in: M. Baines (Ed.), Numerical Methods for Fluid Dynamics VII, ICFD, Vol. 8, 2001, pp. 347–353.
[12] F. Brezzi, L. Marini, E. Süli, Discontinuous Galerkin methods for first-order hyperbolic problems, Mathematical Models and Methods in Applied Sciences 14 (2004) 1893–1903.
[13] K. J. Fidkowski, T. A. Oliver, J. Lu, D. L. Darmofal, *p*-Multigrid solution of high–order discontinuous Galerkin discretizations of the compressible Navier-Stokes equations, Journal of Computational Physics 207 (2005) 92–113.
[14] C. Johnson, A. Szepessy, P. Hansbo, Discontinuous Galerkin methods for ordinary differential equations, Mathematics of Computation 36 (154) (1981) 455–473.
[15] K. Eriksson, C. Johnson, V. Thomee, Adaptive finite element methods for parabolic problems I: A linear model problem, Mathematical Modelling and Numerical Analysis 19 (4) (1985) 611–643.
[16] K. Eriksson, C. Johnson, Adaptive finite element methods for parabolic problems I: A linear model problem, SIAM Journal on Numerical Analysis 28 (1) (1991) 43–77.
[17] W. Liu, H. Ma, T. Tang, N. Yan, A posteriori error estimates for discontinuous Galerkin time-stepping method for optimal control problems governed by parabolic equations, SIAM Journal on Numerical Analysis 42 (3) (2004) 1032–1061.
[18] P. Bar-Yoseph, D. Elata, An efficient L2 Galerkin finite element method for multi-dimensional non-linear hyperbolic systems, International Journal for Numerical Methods in Engineering 29 (1990) 1229–1245.
[19] R. B. Lowrie, P. L. Roe, B. van Leer, Properties of space-time discontinuous Galerkin, Los Alamos Technical Report LA-UR-98-5561 (1998).
[20] H. van der Ven, J. van der Vegt, Space-time discontinuous Galerkin fintie element method with dynamic grid motion for inviscid compressible flows II. Efficient flux quadrature, Computer Methods in Applied Mechanical Engineering 191 (2002) 4747–4780.
[21] C. Klaij, J. van der Vegt, H. van der Ven, Space-time discontinuous Galerkin method for the compressible Navier-Stokes equations, Journal of Computational Physics 217 (2006) 589–611.
[22] T. J. Barth, Space-time error representation and estimation in Navier-Stokes calculations, in: S. C. Kassinos, C. A. Langer, G. Iaccarino, P. Moin (Eds.), Complex Effects in Large Eddy Simulations, Springer Berlin Heidelberg, Lecture Notes in Computational Science and Engineering Vol 26, 2007, pp. 29–48.
[23] K. Mani, D. J. Mavriplis, Discrete adjoint based time-step adaptation and error reduction in unsteady flow problems, AIAA Paper 2007-3944 (2007).
[24] K. Mani, D. J. Mavriplis, Error estimation and adaptation for functional outputs in time-dependent flow problems, Journal of Computational Physics 229 (2010) 415–440.
[25] D. Meidner, B. Vexler, Adaptive space-time finite element methods for parabolic optimization problems, SIAM Journal on Control Optimization 46 (1) (2007) 116–142.
[26] M. Schmich, B. Vexler, Adaptivity with dynamic meshes for space-time finite element discretizations of parabolic equations, SIAM Journal on Scientific Computing 30 (1) (2008) 369–393.
[27] P. L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, Journal of Computational Physics 43 (1981) 357–372.
[28] F. Bassi, S. Rebay, GMRES discontinuous Galerkin solution of the compressible Navier-Stokes equations, in: K. Cockburn, Shu (Eds.), Discontinuous Galerkin Methods: Theory, Computation and Applications, Springer, Berlin, 2000, pp. 197–208.
[29] D. N. Arnold, F. Brezzi, B. Cockburn, L. D. Marini, Unified analysis of discontinuous Galerkin methods for elliptic problems, SIAM Journal on Numerical Analysis 39 (5) (2002) 1749–1779.
[30] J. Lu, An a posteriori error control framework for adaptive precision optimization using discontinuous Galerkin finite element method, Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts (2005).
[31] T. Richter, Discontinuous Galerkin as time-stepping scheme for the Navier-Stokes equations, in: Fourth International Conference on High Performance Scientific Computing Modeling, Simulation and Optimization of Complex Processes, 2009.
[32] S. Adjerid, K. D. Devine, J. Flaherty, L. Krivodonova, A posteriori error estimation for discontinuous Galerkin solutions of hyperbolic problems, Computer Methods in Applied Mechanical Engineering 191 (2002) 1097–1112.
[33] T. Leicht, R. Hartmann, Multitarget error estimation and adaptivity in aerodynamic flow simulations, International Journal for Numerical Methods in Fluids 56 (2008) 2111–2138.
[34] X. D. Zhang, M.-G. Vallet, J. Dompierre, P. Labbe, D. Pelletier, J.-Y. Trepanier, R. Camarero, J. V. Lassaline, L. M. Manzano, D. W. Zingg, Mesh adaptation using different error indicators for the Euler equations, AIAA Paper

2001-2549 (2001).
[35] V. Dolejsi, M. Feistauer, C. Schwab, On some aspects of the discontinuous Galerkin finite element method for conservation laws, Mathematics and Computers in Simulation 61 (3) (2003) 333–346.
[36] L. Krivodonova, J. Xin, J. Remacle, N. Chevaugeon, J. Flaherty, Shock detection and limiting with discontinuous Galerkin methods for hyperbolic conservation laws, Applied Numerical Mathematics 48 (3) (2004) 323–338.
[37] G. E. Barter, 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).