INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2011; **88**:1297–1322
Published online 18 May 2011 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/nme.3224

Output error estimation strategies for discontinuous Galerkin discretizations of unsteady convection-dominated flows

Krzysztof J. Fidkowski*,†

Department of Aerospace Engineering, University of Michigan, 1320 Beal Avenue 3029 FXB, Ann Arbor, MI 48109, U.S.A.

SUMMARY

We study practical strategies for estimating numerical errors in scalar outputs calculated from unsteady simulations of convection-dominated flows, including those governed by the compressible Navier–Stokes equations. The discretization is a discontinuous Galerkin finite element method in space and time on static spatial meshes. Time-integral quantities are considered for scalar outputs and these are shown to superconverge with temporal refinement. Output error estimates are calculated using the adjoint-weighted residual method, where the unsteady adjoint solution is obtained using a discrete approach with an iterative solver. We investigate the accuracy versus computational cost trade-off for various approximations of the fine-space adjoint and find that exact adjoint solutions are accurate but expensive. To reduce the cost, we propose a local temporal reconstruction that takes advantage of superconvergence properties at Radau points, and a spatial reconstruction based on nearest-neighbor elements. This inexact adjoint yields output error estimates at a computational cost of less than 2.5 times that of the forward problem for the cases tested. The calculated error estimates account for numerical error arising from both the spatial and temporal discretizations, and we present a method for identifying the percentage contributions of each discretization to the output error. Copyright © 2011 John Wiley & Sons, Ltd.

Received 8 July 2010; Accepted 13 April 2011

KEY WORDS: verification; discontinuous Galerkin; Navier–Stokes; unsteady adjoint; output error

1. INTRODUCTION

While numerical simulations of complex phenomena generate massive amounts of data, often only a few scalar output quantities are of practical interest. The robustness of these simulations improves greatly when verification measures in the form of error estimates are available along with the scalar outputs. *A posteriori* output error estimation has recently received considerable attention for steady problems, where it has been used not only to provide error bars on outputs but also to drive mesh adaptation. Research into unsteady problems has been more limited, in part due to implementation challenges and computational expense. Nevertheless, many real-world phenomena are unsteady, and with the continuing growth of processing power, output error estimates for complex unsteady simulations are now becoming tractable.

For steady-state problems, output-based *a posteriori* error estimates have been studied in depth by numerous authors [1–6]. The central feature in all these methods is an adjoint-weighted residual calculation that targets scalar outputs of interest. While output error estimates have

*Correspondence to: Krzysztof J. Fidkowski, Department of Aerospace Engineering, University of Michigan, 1320 Beal Avenue 3029 FXB, Ann Arbor, MI 48109, U.S.A.
†E-mail: kfid@umich.edu

been employed for various discretizations, most rigorous are ones that admit a functional form of the solution, such as a variational finite element method. When a variational formulation is also used for the temporal discretization, as is the case in this work, the adjoint-weighted residual method extends naturally to unsteady problems. However, while the theoretical extension of the adjoint-weighted residual method is straightforward, unsteady problems do pose implementation and computational challenges for large simulations, most notably in the solution of a fine-space unsteady adjoint problem. These challenges motivate the present investigation into approximations that enable accurate and cost-effective error estimates for problems of engineering interest.

Unsteady output error estimates have been studied to some extent in previous works. For a finite volume method with a backwards-difference time discretization and dynamic meshes, Mani and Mavriplis [7, 8] estimate the output error due to both the temporal resolution and partial convergence of the unsteady residuals. They solve the discrete adjoint equations by marching backwards in time, and they reconstruct a fine adjoint solution by high-order spline interpolation. For a space–time discontinuous Galerkin (DG) discretization, Barth [9] outlines an adjoint-based error estimation procedure on static unstructured meshes. He shows that an adjoint-weighted residual estimate is capable of estimating the true error as long as the problem is sufficiently well-resolved in space and time. Vexler and coworkers [10, 11] study output error estimates for parabolic problems discretized using continuous Galerkin in space and DG in time. By employing high-order reconstructions of the adjoint in space and time, they obtain separate spatial and temporal error estimates, and they demonstrate error effectivities that approach unity with refinement.

In this work, the DG method is used for both the spatial and temporal discretizations of unsteady convection-dominated systems. DG is chosen for its variational formulation and for its flexibility in admissible solution spaces. DG has been studied in depth for spatial [12–17], temporal [18–21], and combined space–time [9, 22–25] discretizations. We specifically consider a tensor–product solution space formed by independent spatial and temporal discretizations. This space lends itself to an iterative temporal solution scheme, introduced by Richter [26], that simplifies the temporal solver to a straightforward extension of an implicit steady-state solver. We note that while the solver itself is expensive relative to other time-marching schemes, it does yield certain attractive superconvergence properties that we discuss in Section 3.3.

The present work is most closely related to the works of Barth [9] and Vexler and coworkers [10, 11], which also study unsteady output error estimation using the adjoint-weighted residual method. An important contribution of this work is a systematic investigation into the effectivity versus cost trade-off of various approximations in the output error estimation. For example, Barth solves the adjoint problem exactly on a higher order discretization, while Vexler and coworkers reconstruct a high-order adjoint interpolant. Other approximations are also possible, including a local temporal reconstruction that we introduce in Section 3.4. We make recommendations on the most accurate and cost-effective implementation choices for the convection-dominated flows of interest. In addition, this work presents a division of the output error into separate contributions arising from spatial and temporal discretizations. This information could be used to identify and efficiently reduce the dominant component of the error by refining in space or in time. A similar decomposition has been studied previously for reconstructed adjoint solutions [10] and has been used, upon appropriate localization, to drive an adaptive algorithm [11]. We do not consider adaptation in this work and instead focus on the accuracy and cost of the space–time error division under various enabling approximations made in the error estimation.

The remainder of this paper is organized as follows. Section 2 presents the spatial and temporal discretizations of the forward problem and the iterative solver. Section 3 discusses various aspects of output error estimation, including the discrete adjoint problem, the adjoint-weighted residual method, superconvergence of integral outputs, fine-space adjoint approximations, and the separation of the output error into spatial and temporal contributions. Results for two convection-dominated flows are given in Section 4, followed by concluding remarks in Section 5.

2. THE FORWARD PROBLEM

In this section, we describe the spatial and temporal discretizations of a general system of partial differential equations using discontinuous finite elements in space and time on a tensor product space–time mesh. We also outline an iterative solution strategy for elements of linear and quadratic order in time. No assumptions are made here about the underlying form of the differential equations, but in the results we restrict our attention to convection-dominated flows.

2.1. Discretization

We consider a system of partial differential equations of the form

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

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 $\mathbf{x}$. s is the number of governing equations, and d = 1, 2, or 3 is the spatial dimension. We discretize (1) using a discontinuous finite element method on a tensor product space–time mesh, as illustrated in Figure 1 for $d = 2$.

An approximate solution, $\mathbf{u}_H(\mathbf{x}, t)$, is sought in the finite-dimensional space $\boldsymbol{\mathcal{V}}_H$, where $\boldsymbol{\mathcal{V}}_H = \boldsymbol{\mathcal{V}}_H^{\text{space}} \otimes \mathcal{V}_H^{\text{time}}$. The spatial approximation space is $\boldsymbol{\mathcal{V}}_H^{\text{space}} = [\mathcal{V}_H^{\text{space}}]^s$, where $\mathcal{V}_H^{\text{space}}$ is of dimension $N_{H,\text{dof}}^{\text{space}}$ 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 of dimension $N_{H,\text{dof}}^{\text{time}}$ and consists of polynomials of order r in time on each slab. These spaces admit discontinuities across elements and time slabs.

Let $\{\phi_{H,j}(\mathbf{x})\}$, $j = 1, \ldots, N_{H,\text{dof}}^{\text{space}}$, be a basis for $\mathcal{V}_H^{\text{space}}$ and $\{\varphi_H^n(t)\}$, $n = 1, \ldots, N_{H,\text{dof}}^{\text{time}}$, be a basis for $\mathcal{V}_H^{\text{time}}$. We approximate the solution as

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

where $\mathbf{u}_{H,j}^n \in \mathbb{R}^s$ are expansion coefficients for the state and summation is implied on the repeated indices j and n. The following semi-discrete expressions will also be used:

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

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

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

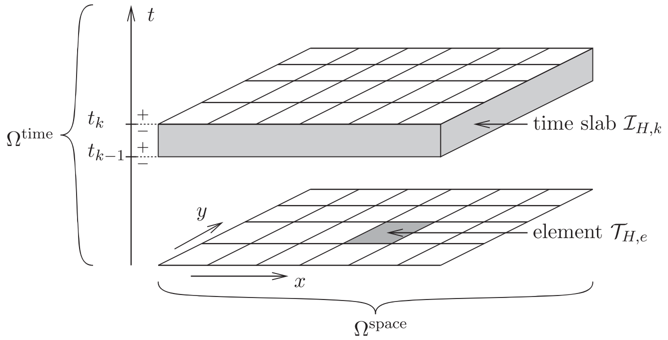


Figure 1. Portion of a sample mesh for a discretization in two spatial dimensions and time. Each space–time element, $\mathcal{T}_{H,e} \otimes \mathcal{I}_{H,k}$, is given by a tensor product of a spatial element, $\mathcal{T}_{H,e}$, $e = 1, \ldots, N_{H,\text{elem}}$, and a time slab, $\mathcal{I}_{H,k}$,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

DOI: 10.1002/nme

where $\bar{\mathscr{R}}_H(\cdot,\cdot):\mathscr{V}_H\otimes\mathscr{V}_H\to\mathbb{R}$ is a semilinear form obtained from (1) by multiplying by test functions $\mathbf{v}_H$ and integrating over the space–time domain, taking into account the discontinuous nature of the test functions and the approximated solution. Substituting the expansion from (2), and a similar one for $\mathbf{v}_H$, into (1) yields the following expression for the semilinear form:

$$\bar{\mathscr{R}}_H(\mathbf{u}_H,\mathbf{v}_H)=\sum_{k=1}^{N_{H,\mathrm{slab}}}\left\{\int_{\mathscr{I}_{H,k}}\left[\mathbf{v}_{H,i}^{\mathrm{T}}(t)\frac{\mathrm{d}\mathbf{u}_{H,j}(t)}{\mathrm{d}t}M_{H,ij}+\mathscr{R}_H(\mathbf{u}_H,\mathbf{v}_H)\right]\mathrm{d}t\right.$$
$$\left.+M_{H,ij}\mathbf{v}_{H,i}^{\mathrm{T}}(t_{k-1}^{+})(\mathbf{u}_{H,j}(t_{k-1}^{+})-\mathbf{u}_{H,j}(t_{k-1}^{-}))\right\}, \tag{4}$$

where $N_{H,\mathrm{slab}}$ is the number of time slabs, the superscript T denotes the transpose, and

$$M_{H,ij}=\int_{\Omega^{\mathrm{space}}}\phi_{H,j}(\mathbf{x})\phi_{H,i}(\mathbf{x})\,\mathrm{d}\Omega.$$

t_{k-1}^{-} indicates the end of time slab $\mathscr{I}_{H,k-1}$, and t_{k-1}^{+} indicates the beginning of time slab $\mathscr{I}_{H,k}$. The term involving the jump in the state across time slab interfaces in (4) couples time slabs, and we note that $\mathbf{u}_H(t_0^{-})$ refers to the initial condition. $\mathscr{R}_H(\cdot,\cdot):\mathscr{V}_H^{\mathrm{space}}\otimes\mathscr{V}_H^{\mathrm{space}}\to\mathbb{R}$ is the semilinear form associated with the spatial discretization. The spatial discretization in this work is a discontinuous Galerkin formulation that employs the second form of Bassi and Rebay [27] for viscous fluxes and the Roe linearization [28] for convective fluxes. Details on this formulation are given in numerous steady-state studies, including [17]. However, we note that the error estimation results are not restricted to a specific spatial discretization. For compactness of notation, we now unroll all the degrees of freedom associated with $\mathbf{u}_{H,j}(t)$ into one vector, $\mathbf{U}_H(t)\in\mathbb{R}^{N_H}$, where $N_H\equiv sN_{H,\mathrm{dof}}^{\mathrm{space}}$. Doing the same for $\mathbf{v}_{H,i}(t)$, we have

$$\mathbf{U}_H(t)=\begin{bmatrix}\mathbf{u}_{H,1}(t)\\\mathbf{u}_{H,2}(t)\\\vdots\\\mathbf{u}_{H,N_{H,\mathrm{dof}}^{\mathrm{space}}}(t)\end{bmatrix},\quad \mathbf{V}_H(t)=\begin{bmatrix}\mathbf{v}_{H,1}(t)\\\mathbf{v}_{H,2}(t)\\\vdots\\\mathbf{v}_{H,N_{H,\mathrm{dof}}^{\mathrm{space}}}(t)\end{bmatrix}.$$

Equation (4) then becomes

$$\bar{\mathscr{R}}_H(\mathbf{u}_H,\mathbf{v}_H)=\sum_{k=1}^{N_{H,\mathrm{slab}}}\left\{\int_{\mathscr{I}_{H,k}}\mathbf{V}_H^{\mathrm{T}}(t)\left[\mathbf{M}_H\frac{\mathrm{d}\mathbf{U}_H(t)}{\mathrm{d}t}+\mathbf{R}_H(\mathbf{U}_H(t))\right]\mathrm{d}t\right.$$
$$\left.+\mathbf{V}_H^{\mathrm{T}}(t_{k-1}^{+})\mathbf{M}_H(\mathbf{U}_H(t_{k-1}^{+})-\mathbf{U}_H(t_{k-1}^{-}))\right\}, \tag{5}$$

where $\mathbf{M}_H=M_{H,ij}\otimes\mathbf{I}_s\in\mathbb{R}^{N_H\times N_H}$, with $\mathbf{I}_s$ the $s\times s$ identity matrix, is the spatial mass matrix, and $\mathbf{R}_H\in\mathbb{R}^{N_H}$ is the spatial residual vector.

Since (5) has to be satisfied for all test functions $\mathbf{V}_H(t)$, whose components are in $\mathscr{V}_H^{\mathrm{time}}$, it represents a system of N_H coupled ordinary differential equations in time. The DG temporal discretization is obtained by using $\varphi_H^m(t)$, $m=1,\ldots,N_{H,\mathrm{dof}}^{\mathrm{time}}$, for the temporal test functions. The local support of these test functions requires that the contribution of each time slab to the sum in (5) vanishes. After an integration by parts, (5) on each time slab becomes

$$\mathbf{0}=-\int_{\mathscr{I}_{H,k}}\mathbf{M}_H\mathbf{U}_H\frac{\mathrm{d}\varphi_H^m}{\mathrm{d}t}\mathrm{d}t+\mathbf{M}_H\mathbf{U}_H(t_k^{-})\varphi_H^m(t_k)-\mathbf{M}_H\mathbf{U}_H(t_{k-1}^{-})\varphi_H^m(t_{k-1})$$
$$+\int_{\mathscr{I}_{H,k}}\varphi_H^m\mathbf{R}_H(\mathbf{U}_H(t))\,\mathrm{d}t. \tag{6}$$

DOI: 10.1002/nme

Coupling to the previous time slab is achieved by the presence of $\mathbf{U}_H(t_{k-1}^-)$, which arises from the jump terms in (5), and which here can be viewed as a natural causal flux at the time slab interface. Substituting the temporal expansion $\mathbf{U}_H(t)=\mathbf{U}_H^n\varphi_H^n(t)$ into the above equation yields the algebraic system

$$\bar{\mathbf{R}}_H^m(\mathbf{U}_H^n)=0, \tag{7}$$

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

For clarity and without loss of generality, we now consider only the first time slab. Also, for the functions $\varphi_H^m(t)$ we use Lagrange polynomials of order r with equally spaced nodes that include the slab endpoints. The $r+1$ unsteady residual vectors on the first time slab then take the form

$$\bar{\mathbf{R}}_H^m=a_r^{m,n}\mathbf{M}_H\mathbf{U}_H^n-\varphi_H^m(t_0^+)\mathbf{M}_H\mathbf{U}_H^0+\int_{t_0}^{t_1}\varphi_H^m(t)\mathbf{R}_H(\mathbf{U}_H(t))\,\mathrm{d}t, \tag{8}$$

where $\mathbf{U}_H^0$ 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 $\mathbf{U}_H^0$. The coefficient matrix $a_r^{m,n}$ is

$$a_r^{m,n}=-\int_{t_0}^{t_1}\varphi_H^n\frac{\mathrm{d}\varphi_H^m}{\mathrm{d}t}\,\mathrm{d}t+\varphi_H^n(t_1^-)\varphi_H^m(t_1^-). \tag{9}$$

Specifically, for $r=1$ and $r=2$ we have

$$a_1^{m,n}=\begin{bmatrix}\frac{1}{2} & \frac{1}{2}\\ -\frac{1}{2} & \frac{1}{2}\end{bmatrix},\quad a_2^{m,n}=\begin{bmatrix}\frac{1}{2} & \frac{2}{3} & -\frac{1}{6}\\ -\frac{2}{3} & 0 & \frac{2}{3}\\ \frac{1}{6} & -\frac{2}{3} & \frac{1}{2}\end{bmatrix}.$$

In practice, the integral in (8) is evaluated numerically. In the present work, we employ Gauss quadrature with $r+1$ points for order $2r+1$ accuracy.

2.2. *Solution*

Equation (7) is a possibly non-linear system in which all 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 use an approximate Newton method to solve this system. On each time slab, the solution is first initialized to the state from the end of the previous time slab. In subsequent iterations, the linear update equation on slab k is given by

$$\left.\frac{\partial\bar{\mathbf{R}}_H^m}{\partial\mathbf{U}_H^n}\right|_{\mathbf{U}_H^n}\Delta\mathbf{U}_H^n=-\bar{\mathbf{R}}_H^m(\mathbf{U}_H^n), \tag{10}$$

where m and n are the temporal indices associated with slab k and summation is implied on n on the left-hand side.

Solving (10) requires inverting a system with $(r+1)N_H$ unknowns, i.e. $(r+1)$ as many as in the steady problem. A solver that requires storage of the Jacobian matrix for the entire slab can be prohibitive especially for large problems in which the steady Jacobian matrix already taxes memory requirements. Hence, in this work, we employ an approximate factorization strategy introduced by Richter [26]. This strategy is based on two approximations. First, a temporally constant spatial Jacobian matrix, $\mathbf{A}_H$, is evaluated at the time slab midpoint and used in the linearization of the

DOI: 10.1002/nme

time integrals appearing in the unsteady residual in (8),

$$\frac{\partial}{\partial \mathbf{U}_H^n}\int_{\mathscr{I}_{H,k}} \varphi_H^m(t)\mathbf{R}_H(\mathbf{U}_H(t))\,\mathrm{d}t = \int_{\mathscr{I}_{H,k}} \varphi_H^m(t)\left.\frac{\partial \mathbf{R}_H}{\partial \mathbf{U}_H}\right|_{\mathbf{U}_H(t)} \varphi_H^n(t)\,\mathrm{d}t$$

$$\approx \underbrace{\left.\frac{\partial \mathbf{R}_H}{\partial \mathbf{U}_H}\right|_{\mathbf{U}_H^{k-1/2}}}_{\mathbf{A}_H}\int_{\mathscr{I}_{H,k}} \varphi_H^m(t)\varphi_H^n(t)\,\mathrm{d}t,$$

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

$$(a_r^{m,n}\mathbf{M}_H + b_r^{m,n}\Delta t_1 \mathbf{A}_H)\Delta \mathbf{U}_H^n = -\bar{\mathbf{R}}_H^m$$

$$\text{where } b_r^{m,n} = \frac{1}{\Delta t_1}\int_{t_0}^{t_1} \varphi_H^m(t)\varphi_H^n(t)\,\mathrm{d}t. \tag{11}$$

Specifically for $r=1$ and $r=2$,

$$b_1^{m,n} = \begin{bmatrix} \frac{1}{3} & \frac{1}{6} \\ \frac{1}{6} & \frac{1}{3} \end{bmatrix}, \quad b_2^{m,n} = \begin{bmatrix} \frac{2}{15} & \frac{1}{15} & -\frac{1}{30} \\ \frac{1}{15} & \frac{8}{15} & \frac{1}{15} \\ -\frac{1}{30} & \frac{1}{15} & \frac{2}{15} \end{bmatrix}.$$

Second, an approximate factorization is used in the solution of the system in (11). For $r=1$, with $\mathbf{F}_H = \Delta t_1 \mathbf{M}_H^{-1}\mathbf{A}_H$, the approximation is

$$[\mathbf{I}_H + \tfrac{2}{3}\mathbf{F}_H + \tfrac{1}{6}\mathbf{F}_H^2] \approx [\mathbf{I}_H + \tfrac{1}{\sqrt{6}}\mathbf{F}_H]^2,$$

where superscripts on $\mathbf{F}_H$ are powers and $\mathbf{I}_H$ is the identity matrix. The resulting three-step solution scheme is given in Appendix A. The approximate factorization bypasses the need for forming and inverting the square of $\mathbf{A}_H$. The iterative steps only require inversion of two systems, both of which have 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 [26].

For $r=2$, again with $\mathbf{F}_H = \Delta t_1 \mathbf{M}_H^{-1}\mathbf{A}_H$, the approximations made are [26]

$$\mathbf{I}_H + \tfrac{3}{5}\mathbf{F}_H + \tfrac{3}{20}\mathbf{F}_H^2 + \tfrac{1}{60}\mathbf{F}_H^3 \approx (\mathbf{I}_H + 60^{-1/3}\mathbf{F}_H)^3,$$

$$\mathbf{I}_H + \tfrac{3}{5}\mathbf{F}_H + \tfrac{3}{20}\mathbf{F}_H^2 \approx (\mathbf{I}_H + \sqrt{\tfrac{3}{20}}\mathbf{F}_H)^2,$$

and the resulting six-step scheme is given in Appendix A. As in the case of $r=1$, the scheme only requires inversion of matrices that have the same compact stencil as the steady discretization. For $r=2$, three different matrices need to be inverted.

Finally, we note that convergence and stability of the iterative method are important considerations. Appendix B presents an iterative error amplification analysis for both $r=1$ and $r=2$. The result is that the approximate solver is unconditionally stable for all stable (damped) modes of the spatial discretization. In addition, the convergence rate, as measured by the magnitude of the error amplification factor, is faster for $r=1$ than for $r=2$ and improves as Δt decreases.

3. OUTPUT ERROR ESTIMATION

In general, the solution $\mathbf{u}_H(x,t)$ in the finite dimensional space $\mathscr{V}_H$ will not be 'exact' in the sense that it will change as the solution space is enriched. A scalar output computed with this solution,

$\mathscr{J}_H(\mathbf{u}_H(x,t))$, will therefore be affected by discretization errors in $\mathbf{u}_H(x,t)$. Our goal is to estimate the resulting output error. Specifically, we consider the difference between $\mathscr{J}_H(\mathbf{u}_H(x,t))$ and the output calculated from a 'fine' solution, $\mathbf{u}_h(x,t)$, on a richer space $\mathscr{V}_h$. We assume that the coarse and fine spaces are nested, $\mathscr{V}_H \subset \mathscr{V}_h$. In fact, the output error estimation algorithm considered here will not require $\mathbf{u}_h(x,t)$. Instead, central to the error estimation will be an approximation to the fine-space adjoint solution, $\boldsymbol{\psi}_h(x,t) \in \mathscr{V}_h$. The choices of the fine space and of the approximate solvers for $\boldsymbol{\psi}_h(x,t)$ are not unique, and we will show that certain choices are advantageous in terms of accuracy and efficiency of the output error estimates.

3.1. The discrete adjoint

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

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

where $\mathscr{J}_H(\cdot): \mathscr{V}_H \to \mathbb{R}$ and $J_H(\cdot): \mathbb{R}^{N_H} \to \mathbb{R}$ are, respectively, the variational and discrete representations of the output. Once a basis for $\mathscr{V}_H$ is chosen, these two representations are interchangeable. The discrete adjoint equation associated with the output and with the algebraic system in (7) is

$$\underbrace{\left(\frac{\partial \bar{\mathbf{R}}_H^m}{\partial \mathbf{U}_H^n}\right)^{\mathrm{T}} \boldsymbol{\Psi}_H^m + \left(\frac{\partial J_H}{\partial \mathbf{U}_H^n}\right)^{\mathrm{T}}}_{\bar{\mathbf{R}}_H^{\psi,n}(\boldsymbol{\Psi}_H^m)} = 0, \tag{12}$$

where the linearization of the unsteady residual is computed about the forward solution for nonlinear problems, and summation is implied on m. The discrete adjoint vector, $\boldsymbol{\Psi}_H^m \in \mathbb{R}^{N_H}$, consists of unrolled expansion coefficients, $\boldsymbol{\psi}_{H,i}^m$, for $\boldsymbol{\psi}_H(\mathbf{x},t) \in \mathscr{V}_H$. When the forward discretization and output definition are adjoint consistent [29, 30], $\boldsymbol{\psi}_H(\mathbf{x},t)$ approximates the solution to the continuous adjoint problem.

For each $n = 1, \ldots, N_{H,\text{dof}}^{\text{time}}$, $\bar{\mathbf{R}}_H^{\psi,n}(\boldsymbol{\Psi}_H^m) \in \mathbb{R}^{N_H}$, is an unsteady adjoint residual vector. For our order r Lagrange temporal basis, the unsteady adjoint residual vectors on the first time slab are

$$\begin{aligned}\bar{\mathbf{R}}_H^{\psi,n} &= a_r^{m,n} \mathbf{M}_H \boldsymbol{\Psi}_H^m - \varphi_H^n(t_1) \mathbf{M}_H \boldsymbol{\Psi}_H^{\text{next}} \\ &\quad + \int_{t_0}^{t_1} \varphi_H^n \left(\frac{\partial \mathbf{R}}{\partial \mathbf{U}}(\mathbf{U}_H(t))\right)^{\mathrm{T}} \boldsymbol{\Psi}_H(t)\,\mathrm{d}t + \left(\frac{\partial J_H}{\partial \mathbf{U}_H^n}\right)^{\mathrm{T}},\end{aligned} \tag{13}$$

where $a_r^{m,n}$ is defined in (9), $\boldsymbol{\Psi}_H(t) = \boldsymbol{\Psi}_H^m \varphi_H^m(t)$, and $\boldsymbol{\Psi}_H^{\text{next}}$ is the adjoint vector associated with the first basis function on 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 (8).

The iterative solver used in the forward problem is also applied to the solution of the adjoint system on each time slab, i.e. $\bar{\mathbf{R}}_H^{\psi,n}(\boldsymbol{\Psi}_H^m) = 0$. At the cost of requiring multiple Newton iterations for a linear problem, the iterative scheme based on the approximate factorization makes the adjoint solution tractable for large-scale problems and permits reuse of the forward solver. The iterative solver described in Section 2.2 requires only two minor changes when applied to the adjoint system: $\mathbf{A}_H^{\mathrm{T}}$ is used in place of $\mathbf{A}_H$ and the order of the update vectors is reversed in the multistep scheme.

3.2. The adjoint-weighted residual

We define the output error as the difference between the output computed with the coarse solution (subscript H) versus that computed with a fine solution (subscript h). If we have the adjoint solution on the fine space, $\boldsymbol{\psi}_h \in \mathscr{V}_h$, then the output error can be calculated using the adjoint-weighted

DOI: 10.1002/nme

residual technique [2, 6],

$$\begin{aligned}\delta J = \text{output error} &= \mathscr{J}_H(\mathbf{u}_H) - \mathscr{J}_h(\mathbf{u}_h)\\ &= \underbrace{\bar{\mathscr{R}}_h(\mathbf{u}_H, \delta\boldsymbol{\psi}_h)}_{\text{estimate}} + \underbrace{R^{(2)}(\|\delta\mathbf{u}_h\|, \|\delta\boldsymbol{\psi}_h\|)}_{\text{remainder}},\end{aligned} \tag{14}$$

where $\bar{\mathscr{R}}_h(\mathbf{u}_H, \delta\boldsymbol{\psi}_h)$ is the weak form from (3), and $R^{(2)}(\|\delta\mathbf{u}_h\|, \|\delta\boldsymbol{\psi}_h\|)$ is a remainder term that is second order in $\delta\mathbf{u}_h \equiv \mathbf{u}_H - \mathbf{u}_h$, and $\delta\boldsymbol{\psi}_h \equiv \boldsymbol{\psi}_H - \boldsymbol{\psi}_h$ [2]. Neglecting the remainder term yields the output error estimate that we use in this work. We note that the estimate requires an evaluation of the fine-space unsteady residual associated with the coarse solution. For spatially and temporally smooth solutions and outputs, we expect that in suitable norms, $\|\delta\mathbf{u}_h\|$ and $\|\delta\boldsymbol{\psi}_h\|$ will converge as $(\Delta x)^{p+1}$ and $(\Delta t)^{r+1}$, so that the remainder term will converge as $(\Delta x)^{2p+2}$ and $(\Delta t)^{2r+2}$. The remainder term is non-zero when the adjoint equation is solved only approximately or when the problem is non-linear. In the latter case, the mean-value adjoint can be approximated to address linearization errors [1, 31], or the accuracy of the error estimate in (14) can be improved by including a term consisting of an adjoint residual weighted by the forward solution error [2]. These options, which add computational cost to the error estimation, are not considered in the present work, which focuses on the cost versus accuracy trade-off for fine-space approximation strategies of the adjoint solution.

The fully discrete version of the error estimate in (14) reads

$$\delta J \approx (\delta\boldsymbol{\Psi}_h^m)^{\mathrm{T}} \bar{\mathbf{R}}_h^m(\mathbf{U}_h^{H,n}), \tag{15}$$

where $\delta\boldsymbol{\Psi}_h^m = \boldsymbol{\Psi}_h^{H,m} - \boldsymbol{\Psi}_h^m$, and $\mathbf{U}_h^{H,n}$ and $\boldsymbol{\Psi}_h^{H,m}$ correspond to the injections of the coarse forward, respectively adjoint, solutions into the fine space. Summation is implied on m, which ranges over the fine-space temporal degrees of freedom, $m = 1, \ldots, N_h^{\text{time}}$.

3.3. Integral outputs

A scalar output could be as simple as a component of the solution measured at some point in space and time, or as complex as a weighted time and space integral of a non-linear function of the state. So far we have not made any requirements on the output. Indeed, the discrete adjoint can be calculated for any output that is differentiable with respect to the state. However, the form of the output does have an effect on its convergence and on the computability of the associated error estimate. This effect is related to the choice of output and to the consistency with which the output and governing equations are discretized [29, 30, 32–35]. The cited works study the requirements on the form of the output and spatial discretization for achieving optimal output convergence, and we adhere to these requirements in our work.

For unsteady problems, studies have shown that integral quantities are more suitable than point-value quantities for computing error estimates, especially for the Navier–Stokes equations at high Reynolds numbers [9, 36]. Starting from the semi-discrete form and assuming adjoint consistency in the spatial discretization, we consider outputs that are expressed as integral quantities over the temporal domain,

$$J_H = \int_{\Omega_{\text{time}}} f(\mathbf{U}_H(t), t)\,\mathrm{d}t. \tag{16}$$

Assuming that $f(\mathbf{U}_H(t), t)$ is a sufficiently smooth function of the discrete spatial state vector and of time, and that the forward problem is driven by temporally smooth data, integral outputs superconverge with temporal refinement. This result can be shown by noting that when the unsteady adjoint problem is driven by temporally smooth data, the rate of convergence of point-wise errors in the adjoint approximation is the same as that of the forward approximation. If these are both $(\Delta t)^{r+1}$, the adjoint–weighted residual formulation yields an output error that converges as $(\Delta t)^{2r+1}$—the product of the forward and adjoint approximation errors with a $(\Delta t)^{-1}$ factor that accounts for the temporal differentiation operator in the residual. This argument is analyzed in previous works, including [37].

An alternate proof not based on the adjoint solution follows from Adjerid *et al.* [38], whose results indicate that for DG discretizations of one-dimensional hyperbolic conservation laws, equivalently the temporal discretization in a semi-discrete formulation, weighted integrals of the solution error over an element (time slab in our case) converge at a local rate of $(\Delta t)^{2r+2}$, for a global integral error of $(\Delta t)^{2r+1}$. An outline of the superconvergence proof, which relies on the smoothness of $\partial f/\partial U_H$, is given in Appendix C.

3.4. Enabling approximations

The output error estimate in (15) requires an adjoint solution on the fine space. The cost of obtaining this adjoint depends on the choice of the fine space $\mathscr{V}_h$ and on the approximations employed in the adjoint solution. As the space $\mathscr{V}_h$ is made richer, the accuracy of the error estimate improves but the computational cost rises, resulting in a trade-off between cost and accuracy.

We measure the accuracy of the output error estimate through an *effectivity*, η_{eff}, defined as

$$\eta_{\text{eff}} \equiv \frac{\delta J}{\mathscr{J}_H(\mathbf{u}_H) - \mathscr{J}(\mathbf{u})}, \tag{17}$$

where $\mathscr{J}(\mathbf{u})$ is the output of the exact solution. A value of $\eta_H^e = 1$ means that the output error estimate is exact. In practice, the effectivity will be lower or higher than 1 due to the use of a finite-dimensional space in our output error definition and due to approximations made in the output error estimation.

The cost of the error estimation is measured by the processing (CPU) time relative to that of the forward solution. Specifically, we define the *CPU factor*, η_{CPU}, as

$$\eta_{\text{CPU}} \equiv \frac{\text{CPU time of adjoint solve and error estimation}}{\text{CPU time of forward solve}}. \tag{18}$$

The ratio of CPU times makes η_{CPU} less sensitive to algorithmic implementation differences in routines that are shared between the forward and the adjoint codes. We note that both the effectivity and the CPU factor depend on the choice of fine space and on any approximations made in the adjoint solution.

In this work, we compare six combinations of fine-space choice and adjoint approximation. These are as follows:

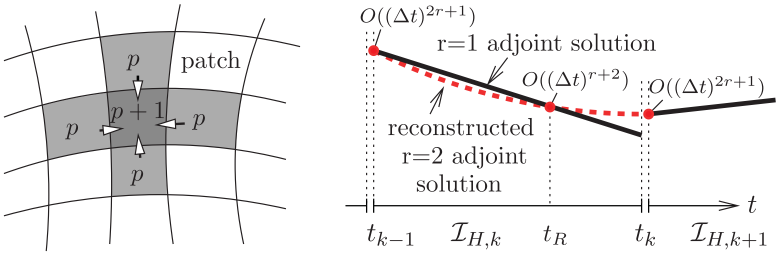
A: '$p+1, r+1$'. ψ_h^{A} is obtained by solving exactly the discrete adjoint equation, (12), on a fine space $\mathscr{V}_h$ that is obtained from the coarse space $\mathscr{V}_H$ by increasing the spatial order, p, by 1 and the temporal order, r, by 1.

B: '$p+1, \Delta t/2$'. ψ_h^{B} is obtained by solving exactly the discrete adjoint equation on a fine space $\mathscr{V}_h$ that is obtained from the coarse space $\mathscr{V}_H$ by increasing the spatial order, p, by 1, and by bisecting each time slab: $\Delta t \to \Delta t/2$.

C: 'Reconstruct($p+1$), $r+1$'. ψ_h^{C} is obtained by solving the adjoint problem on a space in which the temporal order, r, is incremented by 1 but in which the spatial order remains p. This solution is then *reconstructed* spatially to order $p+1$ using a least-squares, nearest-neighbor patch reconstruction, illustrated in Figure 2(a) and described in further detail in [39]. Note that the accuracy of the reconstruction depends on the spatial smoothness of the solution relative to the mesh size.

D: '$p+1$, Reconstruct($r+1$)'. ψ_h^{D} is obtained by solving the adjoint problem on a space in which the spatial order, p, is incremented by 1 but in which the temporal order remains r. This solution is then reconstructed temporally to order $r+1$ using a local interpolation of the solution at superconvergent nodes. Specifically, the $r+2$ points required for the order $r+1$ interpolant consist of, in the adjoint case, the left time nodes of the current and future time slabs and the r 'left Radau' points on the current time slab. The left Radau polynomials are obtained by summing two consecutive Legendre polynomials [38]. The Radau points are the

DOI: 10.1002/nme



(a) Spatial reconstruction (b) Temporal reconstruction (adjoint)

Figure 2. Illustration of spatial and temporal reconstructions. (a) Shows a patch of nearest-neighbor elements used for spatial high-order reconstruction via least-squares interpolation and (b) shows reconstruction of an $r=1$ adjoint solution to $r=2$ 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 $r=1$.

roots of the Radau polynomials, and for $r=1$, the single root of the left Radau polynomial on time slab k is at $t_\mathrm{R}=(t_{k-1}+2t_k)/3$; see Figure 2(b). The adjoint solution converges at a rate of $(\Delta t)^{r+2}$ at the left Radau points and at a rate of $(\Delta t)^{2r+1}$ at the time slab left endpoints. For smooth temporal solutions, we therefore expect an order $r+1$ polynomial interpolant of these values to converge at the high-order rate of $(\Delta t)^{r+2}$.

Remark 1
The order $(\Delta t)^{r+2}$ point-wise convergence of the proposed interpolant follows directly from one-dimensional interpolation theory. The claim is that for smooth temporal adjoint solutions, a polynomial interpolant over a time slab of the solution at $r+2$ time nodes, where the solution is at least $O((\Delta t)^{r+2})$ accurate, is itself $O((\Delta t)^{r+2})$ accurate everywhere within the time slab. To sketch the proof, we consider the difference between the order $r+1$ polynomial interpolant and a Taylor-series expansion of the adjoint about an arbitrary point within the time slab. This difference can be written as $\Delta P^{r+1}(t)+O((\Delta t)^{r+2})$, where $\Delta P^{r+1}(t)$ is a polynomial of order $r+1$. At the $r+2$ interpolation points, we have that $\Delta P^{r+1}(t)=O((\Delta t)^{r+2})$. Constructing a Vandermonde system to solve for the coefficients in $\Delta P^{r+1}(t)$ yields a right-hand side vector with $O((\Delta t)^{r+2})$ terms, so that $\Delta P^{r+1}(t)$ is point-wise $O((\Delta t)^{r+2})$ over the entire time slab. Therefore, the reconstructed adjoint is $O((\Delta t)^{r+2})$ accurate everywhere within the time slab. We also note that the result extends naturally to the forward problem when interpolating the superconvergent right Radau points of the current time slab and the right time nodes of the current and past time slabs.

E: 'Reconstruct($p+1, r+1$)'. ψ_h^E is obtained by solving the adjoint problem on a space in which the spatial order remains p, and in which the temporal order remains r. This solution is then reconstructed spatially to order $p+1$, and temporally to order $r+1$. The reconstructions are the same as described in choices C and D.

F: 'Smooth($p+1$,$r+1$)'. ψ_h^F is calculated by an inexact solve of the discrete adjoint equation on a fine space $\mathcal{V}_h$ that is obtained from the coarse space $\mathcal{V}_H$ by increasing the spatial order, p, by 1 and the temporal order, r, by 1. The inexact solve consists of several smoothing iterations of the solver described in Section 2.2, effected by prescribing a less strict adjoint residual tolerance. In this work, the adjoint residual decrease tolerance was set to the square root of the tolerance on the forward problem residual.

Subdivision of the spatial elements is not considered for the fine space $\mathcal{V}_h$ in the interests of implementation simplicity and keeping the growth of degrees of freedom in check—in three dimensions, uniform subdivision of hexahedral elements would lead to a factor of eight increase in the number of spatial degrees of freedom, compared with a factor of $(\frac{4}{3})^3 \approx 2.37$ in a $p=2$ to $p=3$ order increment. In addition, the spatial aspect of fine-space choices has been investigated in previous studies [30, 31, 39–42], and hence we focus primarily on the temporal aspect.

DOI: 10.1002/nme

The time slab reconstruction in choices D and E takes advantage of point-wise superconvergence properties of a temporal DG discretization to avoid having to solve for the adjoint on a finer temporal space. These superconvergence properties have been studied previously [18, 19, 38], and have been used to define reconstructions and correction functions for the forward problem [43, 44]. We note that while superconvergence has been studied for multidimensional hyperbolic systems, of interest in this work is the particular one-dimensional case when DG is used for the temporal discretization. Hence, the character of the underlying partial differential equation, as dictated by the spatial operator, does not affect our temporal superconvergence result.

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, we note that the adjoint temporal reconstruction outlined above 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. In this regard, the reconstruction respects the physics of the adjoint equation, in which information propagates backward in time.

3.5. Separation of spatial and temporal errors

The tensor product space–time discretization used in this work allows for a simple estimation of the relative magnitudes of the spatial and temporal errors. We define these errors as

$$\begin{aligned}\delta J_H^{\text{space}} &= \mathscr{J}(\mathbf{u}_H)-\mathscr{J}(\mathbf{u}_{hH})\\ &= \bar{\mathscr{R}}_{hH}(\mathbf{u}_H,\delta\boldsymbol{\psi}_{hH})+R^{(2)}(\|\delta\mathbf{u}_{hH}\|,\|\delta\boldsymbol{\psi}_{hH}\|),\\ \delta J_H^{\text{time}} &= \mathscr{J}(\mathbf{u}_H)-\mathscr{J}(\mathbf{u}_{Hh})\\ &= \bar{\mathscr{R}}_{Hh}(\mathbf{u}_H,\delta\boldsymbol{\psi}_{Hh})+R^{(2)}(\|\delta\mathbf{u}_{Hh}\|,\|\delta\boldsymbol{\psi}_{Hh}\|),\end{aligned} \tag{19}$$

where the subscript hH, respectively Hh, denotes a solution or residual evaluation in the semi-refined space $\mathscr{V}_h^{\text{space}}\otimes\mathscr{V}_H^{\text{time}}$, respectively $\mathscr{V}_H^{\text{space}}\otimes\mathscr{V}_h^{\text{time}}$, and

$$\begin{aligned}\delta\mathbf{u}_{hH}&\equiv\mathbf{u}_H-\mathbf{u}_{hH}, &\quad \delta\boldsymbol{\psi}_{hH}&\equiv\boldsymbol{\psi}_H-\boldsymbol{\psi}_{hH},\\ \delta\mathbf{u}_{Hh}&\equiv\mathbf{u}_H-\mathbf{u}_{Hh}, &\quad \delta\boldsymbol{\psi}_{Hh}&\equiv\boldsymbol{\psi}_H-\boldsymbol{\psi}_{Hh}.\end{aligned} \tag{20}$$

The adjoint-weighted residual expressions in (19) are straightforward extensions of the result in (14) to modified fine spaces. Ignoring the remainder terms leaves the error estimates that are used in the present study, with expected accuracy that is second order in the forward and adjoint errors on the semi-refined spaces.

The interpretation of (19) is that the spatial error in an output is the change in the output that occurs when the spatial resolution is refined, while keeping the temporal resolution fixed. A similar interpretation holds for the temporal output error. However, the definition of these errors is not unique. For example, we can define the 'fine' error measures

$$\begin{aligned}\delta J_h^{\text{space}} &= \mathscr{J}(\mathbf{u}_{Hh})-\mathscr{J}(\mathbf{u}_h)\\ &= \bar{\mathscr{R}}_h(\mathbf{u}_{Hh},\delta\boldsymbol{\psi}_h^{\text{space}})+R^{(2)}(\|\delta\mathbf{u}_{Hh}\|,\|\delta\boldsymbol{\psi}_h^{\text{space}}\|),\\ \delta J_h^{\text{time}} &= \mathscr{J}(\mathbf{u}_{hH})-\mathscr{J}(\mathbf{u}_h)\\ &= \bar{\mathscr{R}}_h(\mathbf{u}_{hH},\delta\boldsymbol{\psi}_h^{\text{time}})+R^{(2)}(\|\delta\mathbf{u}_{hH}\|,\|\delta\boldsymbol{\psi}_h^{\text{time}}\|),\end{aligned} \tag{21}$$

where $\delta\boldsymbol{\psi}_h^{\text{space}}\equiv\boldsymbol{\psi}_{Hh}-\boldsymbol{\psi}_h$ and $\delta\boldsymbol{\psi}_h^{\text{time}}\equiv\boldsymbol{\psi}_{hH}-\boldsymbol{\psi}_h$. The definitions in (19) and (21) are related by the following expression (see also Figure 3):

$$\delta J=\mathscr{J}(\mathbf{u}_H)-\mathscr{J}(\mathbf{u}_h)=\delta J_H^{\text{space}}+\delta J_h^{\text{time}}=\delta J_H^{\text{time}}+\delta J_h^{\text{space}}.$$

One measure of the spatial error is the average of $\delta J_H^{\text{space}}$ and $\delta J_h^{\text{space}}$, and similarly for the temporal error. In this case, the sum of the averaged spatial and temporal errors is the output error.

DOI: 10.1002/nme

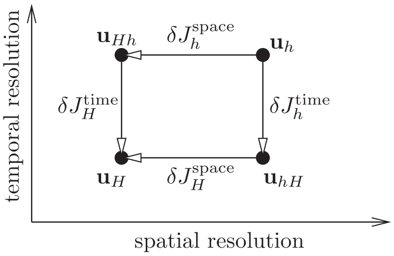

$\mathbf{u}_{Hh}$
$\delta J_h^{\mathrm{space}}$
$\mathbf{u}_h$
$\delta J_H^{\mathrm{time}}$
$\delta J_h^{\mathrm{time}}$
$\mathbf{u}_H$
$\delta J_H^{\mathrm{space}}$
$\mathbf{u}_{hH}$
temporal resolution
spatial resolution

Figure 3. Illustration of the relationship between various spatial and temporal error measures for two spatial and temporal resolution levels.

However, calculating $\delta J_h^{\mathrm{space}}$ and $\delta J_h^{\mathrm{time}}$ via the adjoint-weighed residual approach requires the states $\mathbf{u}_{Hh}$ and $\mathbf{u}_{hH}$, which are not available without additional solves or reconstructions. For this reason, we restrict our attention to the measures $\delta J_H^{\mathrm{space}}$ and $\delta J_H^{\mathrm{time}}$. These errors do not necessarily sum to the total output error, but they are computable without additional state solutions.

We assess the relative magnitudes of the spatial and temporal errors through the *temporal error fraction*,

$$f_{\mathrm{temporal}} = \frac{|\delta J_H^{\mathrm{time}}|}{|\delta J_H^{\mathrm{space}}| + |\delta J_H^{\mathrm{time}}|}. \tag{22}$$

In the results, we compare the temporal error fraction calculated using the various adjoint solutions to the actual error fraction that results from using a very fine space or time resolution. To avoid additional computations, the adjoints $\boldsymbol{\psi}_{hH}$ and $\boldsymbol{\psi}_{Hh}$ required in the calculation of $\delta J_H^{\mathrm{space}}$ and $\delta J_H^{\mathrm{time}}$ are approximated by projecting the fine-space adjoint, $\boldsymbol{\psi}_h$, onto the coarse temporal and spatial spaces, respectively:

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

where $\Pi_H^{\mathrm{space},h}$ and $\Pi_H^{\mathrm{time},h}$ are least-squares projection operators onto the coarse spatial and temporal spaces.

4. RESULTS

For the results, we consider two unsteady simulations: one governed by a scalar convection–diffusion-reaction equation, and the other by the laminar compressible Navier–Stokes equations. In both cases, the solution is discretized using $p=2$ polynomials in space and $r=1$ polynomials in time. Consequently, $p=3$ and $r=2$ are used for some of the fine-space adjoint solutions. The simulations are performed on static, quadrilateral spatial meshes, and uniform temporal spacing. Both the forward and the adjoint schemes employ an element-line preconditioned Generalized Minimal Residual solver [45]. The solutions are converged to a prescribed reduction in the L_1 norm of the unsteady residual that is 10 orders of magnitude in the first case and 7 in the second case. These criteria were determined empirically as sufficiently low to ensure that the iterative convergence errors were negligible compared with the spatial and temporal discretization errors. In addition, the convergence criteria could be expressed in norms other than L_1, and the choice of this norm was made arbitrarily. In each simulation, a study is performed on the effectivity/cost tradeoff and the spatial/temporal division of the error estimates. In addition, a demonstration of convergence rates discussed in Section 3 is included in the first example.

DOI: 10.1002/nme

4.1. Scalar convection diffusion reaction

The governing equation for the scalar convection diffusion reaction simulation is

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

where $\vec{V} = (1, 0)$ is the convection velocity, $\nu = 10^{-3}$ 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 $A = 1.0$, $c_1 = 2$, $E = 0.05$, and $c_2 = 2.4$. The spatial domain, two units wide and one unit high, and the coarse mesh are illustrated in Figure 4(a). Using $L = 1$ as the characteristic length, the Peclet number for this problem is $Pe \equiv L|\vec{V}|/\nu = 1000$. The initial condition, $u_0(x, y)$, consists of a piecewise bi-linear distribution for u centered at $(x, y) = (0.5, 0.5)$,

$$u_0(x, y) = \begin{cases} (1 - |x - 0.5|)(1 - |y - 0.5|) & \text{for } |x - 0.5| \leqslant 0.25 \text{ and } |y - 0.5| \leqslant 0.25, \\ 0 & \text{otherwise.} \end{cases} \tag{23}$$

This initial condition was chosen because it is representable exactly on all the meshes in the spatial refinement study. Symmetry conditions are imposed on the top and bottom boundaries, the left boundary is an inflow with zero concentration of u, and the right boundary is an outflow. The temporal domain is $t \in [0, 3]$, enough time for the majority of the scalar to convect out of the spatial domain.

The output of interest is the net outflow of the scalar from the domain in the time interval $[0, 3]$. This is given by a time integral of the scalar flux integrated along the outflow boundary,

$$\mathscr{J}(u) = \int_0^3 \int_{\text{outflow boundary}} (\vec{V} u) \cdot \mathbf{n} \, \mathrm{d}l \, \mathrm{d}t,$$

where $\mathbf{n} = (1, 0)$ is the unit outward normal on the outflow boundary. An example time history of the boundary flux integral is shown in Figure 4(b), where the output is shown as the area under the curve.

Figure 5 shows the convergence of the output error under uniform spatial and temporal refinements. In the spatial convergence plot, Figure 5(a), the number of time slabs is fixed at a high resolution, and the output error is measured as the spatial mesh is uniformly refined. The actual output error, the difference compared with a truth solution on a refined mesh, converges at a rate of about 4.2 between the final two meshes run, i.e. $\Delta x^{4.2}$ where Δx is a measure of the element size, for the $p = 2$ spatial discretization. We note that this is a convection-dominated flow, and $2p+1$

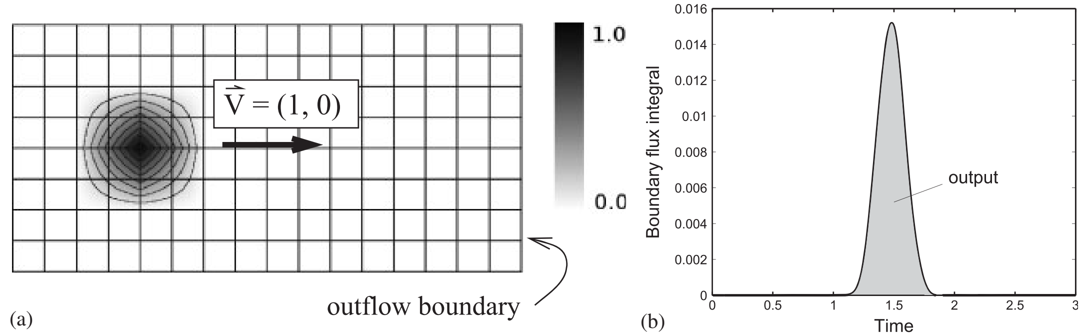


Figure 4. Scalar convection–diffusion-reaction problem setup. The coarse mesh and initial condition contours are shown at left. The time history of the outflow boundary flux integral used in the output calculation is shown on the right: (a) initial condition and (b) flux integral output.

DOI: 10.1002/nme

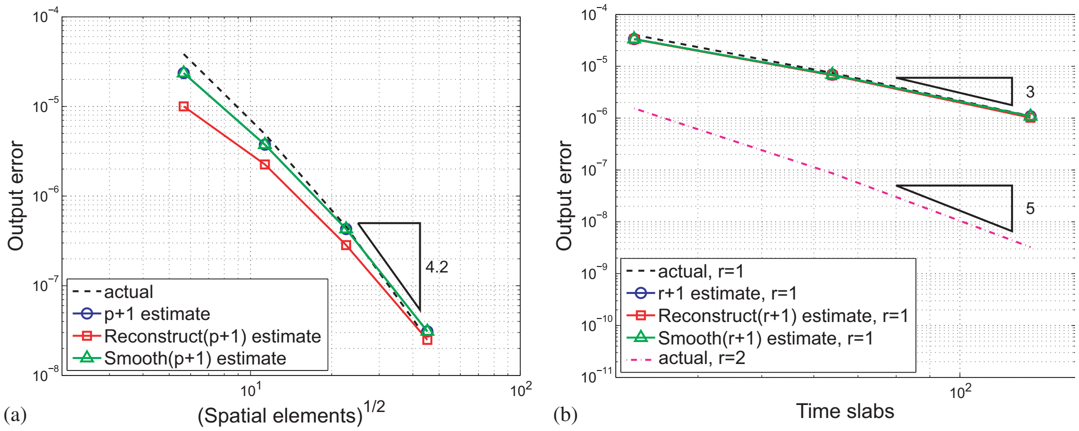


Figure 5. Scalar convection–diffusion-reaction error convergence. 'Actual' corresponds to the output error relative to a truth solution on a mesh that is very fine in space or in time: (a) spatial convergence, $p=2$ and (b) temporal convergence.

Table I. Scalar convection–diffusion-reaction: accuracy verification of the adjoint temporal reconstruction in methods D and E for $r=1$.

N_{slab}	32	64	128	256	512
L_2 error	1.75×10^{-2}	4.89×10^{-3}	8.64×10^{-4}	1.24×10^{-4}	1.64×10^{-5}
Rate	—	1.84	2.50	2.80	2.92

Shown is the convergence of the continuous space–time L_2 error norm in the reconstructed adjoint relative to the exact $r+1$ adjoint, $\|\psi_h^{\text{Recon}(r+1)}-\psi_h^{r+1}\|_{L_2}$, versus number of time slabs.

superconvergence for an output measured on a downwind boundary in a DG discretization has been previously studied for hyperbolic problems [44]. In the temporal convergence plot, Figure 5(b), the number of elements is fixed to yield high spatial resolution, and the output error is measured as the time slabs are uniformly refined. The actual output error is shown for both $r=1$ and $r=2$ temporal orders. The superconvergent rate of $2r+1$ discussed in Section 3.3 is clearly evident.

Also shown in Figure 5 is the convergence of output error *estimates* obtained using (15) and a selection of the fine-space choices presented in Section 3.4. In general, these estimates are consistent with each other and with the actual error. The only noticeable difference is observed for the estimate produced by spatial reconstruction, which somewhat under-predicts the error on coarse meshes. The difference disappears as the resolution is increased.

To further verify the accuracy of the temporal reconstruction used in methods D and E, the convergence of the L_2 space–time error in the reconstructed adjoint was measured relative to the actual $r+1$ adjoint for increasingly finer temporal resolutions, at a fixed spatial resolution of 128 elements. The results are presented in Table I. As discussed in Remark 1, for an $r=1$ adjoint reconstructed to $r=2$, we expect an order $r+2=3$ rate of convergence for the error in the reconstructed adjoint. As shown in the table, for sufficiently high temporal resolutions, the asymptotic convergence rate approaches this value, with a rate of 2.92 measured between the finest two temporal meshes.

The results of a more detailed study of the error effectivity arising from the different adjoint solutions are shown in Table II. The error effectivity is calculated for each of the six adjoints on a set of meshes with varying spatial and temporal resolutions. Several observations are in order. First, as spatial and temporal resolution are increased together, the effectivity approaches 1 for all of the adjoint choices. Second, for low spatial resolution, choices based on the spatial reconstruction (C and E) severely under-predict the error. Third, the results of the temporal reconstruction also show a drop in effectivity at low temporal resolution, although this drop is not severe for the coarsest

Table II. Scalar convection–diffusion-reaction: error effectivities, η_{eff}, calculated using the six different fine-space adjoints with various spatial and temporal resolutions.

$N_{\mathrm{elem}} \backslash N_{\mathrm{slab}}$	32	64	128
(a) $\mathscr{V}_h^A: p+1, r+1$			
32	0.71	0.64	0.62
128	0.82	0.86	0.79
512	0.82	0.91	0.95
(b) $\mathscr{V}_h^B: p+1, \Delta t/2$			
32	0.65	0.63	0.61
128	0.71	0.77	0.74
512	0.71	0.79	0.84
(c) $\mathscr{V}_h^{\mathrm{C}}$: Reconstruct$(p+1), r+1$			
32	0.36	0.29	0.26
128	0.74	0.67	0.52
512	0.81	0.88	0.83
(d) $\mathscr{V}_h^D: p+1$, Reconstruct$(r+1)$			
32	0.64	0.63	0.62
128	0.72	0.82	0.78
512	0.73	0.86	0.92
(e) $\mathscr{V}_h^{\mathrm{E}}$: Reconstruct$(p+1, r+1)$			
32	0.34	0.28	0.26
128	0.66	0.65	0.51
512	0.72	0.83	0.81
(f) $\mathscr{V}_h^{\mathrm{F}}$: Smooth$(p+1, r+1)$			
32	0.71	0.64	0.62
128	0.82	0.86	0.79
512	0.82	0.91	0.95

temporal resolution of 32 slabs. Fourth, the smoothing results, choice F, are nearly identical to the full-solve results, choice A. This indicates that a few iterations of the approximate multistep solver on the fine-space adjoint problem yield sufficient accuracy for effective error estimation in this case.

Although some of the adjoint choices are more accurate, computational expense is also a factor. We measure the expense by the processing time of the adjoint solve and output error estimation relative to the forward solve, as defined by the CPU factor, η_{CPU}, in (18). Figure 6 compares the effectivity to the CPU factor for the methods tested. For clarity, only four corners of the resolution test matrix are shown: these are the combinations of the coarsest and finest resolutions in space and time. The ideal location in this plot is in the top left corner: effectivity near 1 and low CPU factor.

From Figure 6 we see that the adjoint based on exact solve, while effective, is also the most expensive. For example, at the finest resolution, choice A $\{p+1, r+1\}$ results in the adjoint and error estimation being over 10 times as expensive as the forward solve. Choice B $\{p+1, \Delta t/2\}$ is cheaper with a CPU factor of 8.5 at the finest resolution. We note that the CPU factor varies somewhat with spatial and temporal resolutions. This effect is due to the interaction between the mesh size dependence and interpolation order dependence of the solver.

In contrast, the adjoints obtained by reconstruction or approximate solution are cheaper, with CPU factors around 1.5–5. In the effectivity versus cost tradeoff, good adjoint choices are D, $\{p+1, \text{Reconstruct}(r+1)\}$, and F, $\{\text{Smooth}(p+1, r+1)\}$. The spatial reconstruction choices, C and E, are computationally cheap but not very robust for coarse meshes in this case.

Space–time error estimates in the form of the temporal error fraction are shown in Figure 7 for all choices involving order enrichment in space and time. The temporal error fraction is computable for temporal subdivision as well, but the computation involves additional storage of adjoints on

DOI: 10.1002/nme

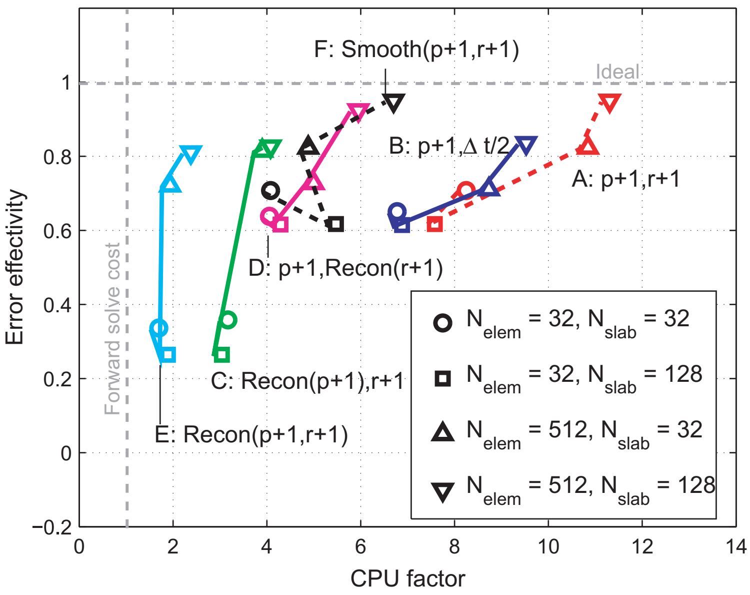


Figure 6. Scalar convection–diffusion-reaction: effectivity versus CPU factor for various spatial and temporal resolution combinations. For clarity, points belonging to the same adjoint choice are connected on the plot.

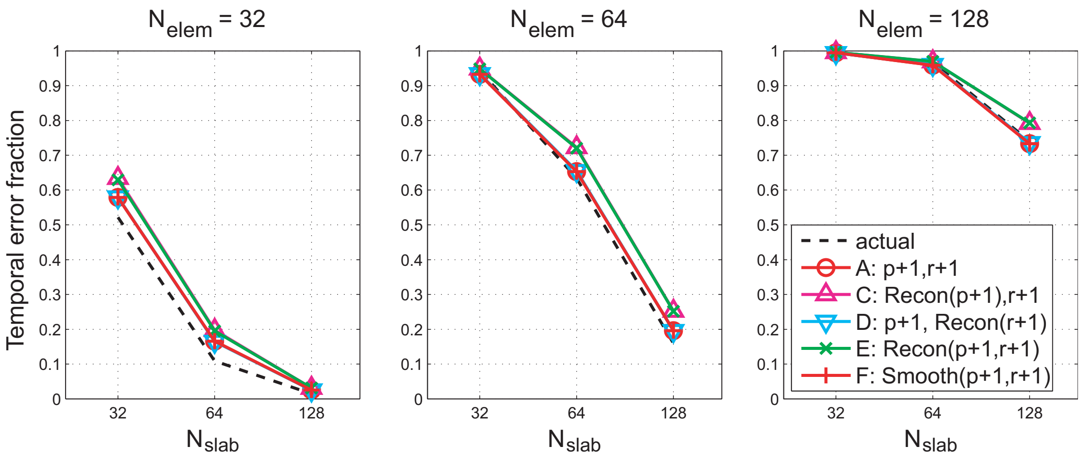


Figure 7. Scalar convection–diffusion-reaction: separation of the output error into spatial and temporal contributions using the temporal error fraction. The ‘actual’ temporal error fractions are computed from (22) using very fine spatial and temporal discretization spaces in the expressions for $\delta J_H^{\mathrm{space}}$ and $\delta J_H^{\mathrm{time}}$ in (19).

the sub-slabs, and for this reason it was not implemented. However, based on the other results, we do not expect significant differences in the temporal error fractions for the adjoints based on temporal subdivision.

Since f_{temporal}, defined in (22), estimates the fraction of the output error that is due to temporal resolution, $1-f_{\mathrm{temporal}}$ estimates the error due to the spatial resolution. In Figure 7, f_{temporal} is shown for each adjoint choice as a function of N_{slab}, and the results are grouped by N_{elem}. The actual temporal error fraction computed using highly refined spatial and temporal meshes is also shown for comparison. The estimated temporal error fractions are quite accurate, with differences generally less than 10%. In addition, the behavior with varying resolution is as expected wherein

DOI: 10.1002/nme

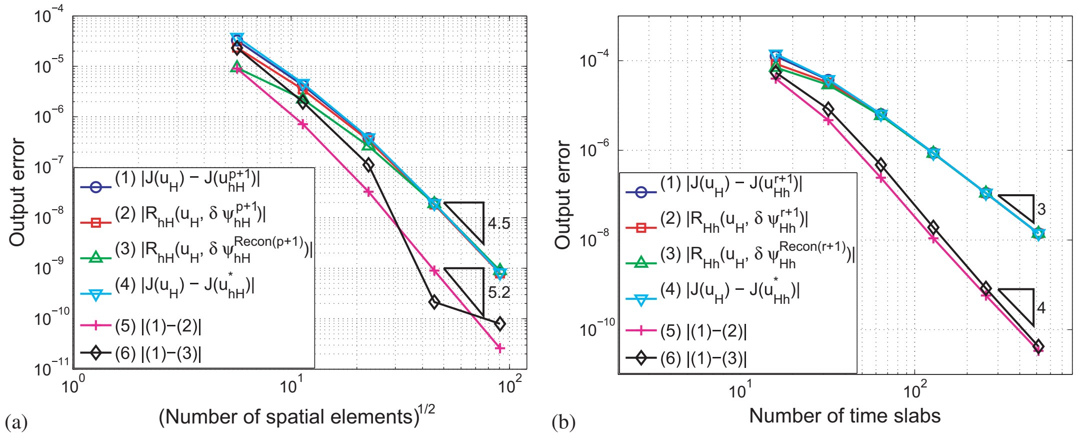


Figure 8. Scalar convection–diffusion-reaction: convergence of actual and estimated spatial and temporal errors appearing in (19). $\mathbf{u}^*_{hH}$, respectively $\mathbf{u}^*_{Hh}$, is the solution on a mesh heavily semi-refined in space, respectively time: (a) Spatial convergence of $\delta J_H^{\text{space}}$and (b) temporal convergence of δJ_H^{time}.

the temporal error fraction increases as the spatial mesh is resolved and decreases as the number of time slabs increases.

To verify the order of accuracy of the spatial and temporal error estimates $\delta J_H^{\text{space}}$ and δJ_H^{time} in (19), we present a more detailed analysis of these error estimates and associated remainder terms under spatial refinement and under temporal refinement. Figure 8 shows the results of this convergence study. First, in Figure 8(a), various quantities related to spatial error estimate, $\delta J_H^{\text{space}}$, are shown for a fixed temporal mesh of 128 slabs. The quantities include: (1) the actual difference between the output computed with $p=2$ and the output computed with $p+1=3$; (2) the output error estimate using an exact $p+1$ adjoint solve; (3) the output error estimate using a reconstructed $p+1$ adjoint; (4) the actual difference between the output computed with $p=2$ and the output computed on a heavily refined spatial mesh; (5) the remainder term in (19) when using the exact $p+1$ adjoint; and (6) the remainder term in (19) when using the reconstructed $p+1$ adjoint. The quantities are plotted versus the square-root of the number of spatial elements to make the negative slope consistent with the convergence rate in terms of Δx. We note that all of the output errors, estimates, and actual values are nearly identical for the fine meshes, and they converge at a rate of approximately 4.5 between the final two refinements. This superconvergent rate is consistent with that observed in Figure 5. On the other hand, the remainder terms converge faster, at a rate of 5.2 between the final two refinements. Finer meshes would likely improve this rate to the optimum value of $2p+2=6$, as discussed in Section 3.2, although machine precision issues impeded acquisition of additional data points.

In Figure 8(b), similar results are shown for the temporal error estimate, δJ_H^{time}, on a fixed spatial mesh of 128 elements. In this case, $r=1$, and all output error estimates agree and converge at the optimum rate of $2r+1=3$. The remainder terms using the actual or reconstructed adjoint at order $r+1$ converge at a faster rate, which is the optimal $2r+2=4$ discussed in Section 3.2. We note that estimates based on the temporally reconstructed adjoint are nearly identical to the estimates based on the actual order $r+1$ adjoint. This is in contrast to the study for $\delta J_H^{\text{space}}$ in Figure 8(a), where the remainder term from the spatially reconstructed adjoint converges at a somewhat more erratic rate.

4.2. *Impulsively started airfoil*

In this example, the flow is governed by the laminar compressible Navier–Stokes equations,

$$\frac{\partial \mathbf{u}}{\partial t}+\frac{\partial}{\partial x_i}[\mathbf{F}_i^{\mathrm{I}}(\mathbf{u})-\mathbf{F}_i^{\mathrm{V}}(\mathbf{u}, \nabla \mathbf{u})]=0,$$

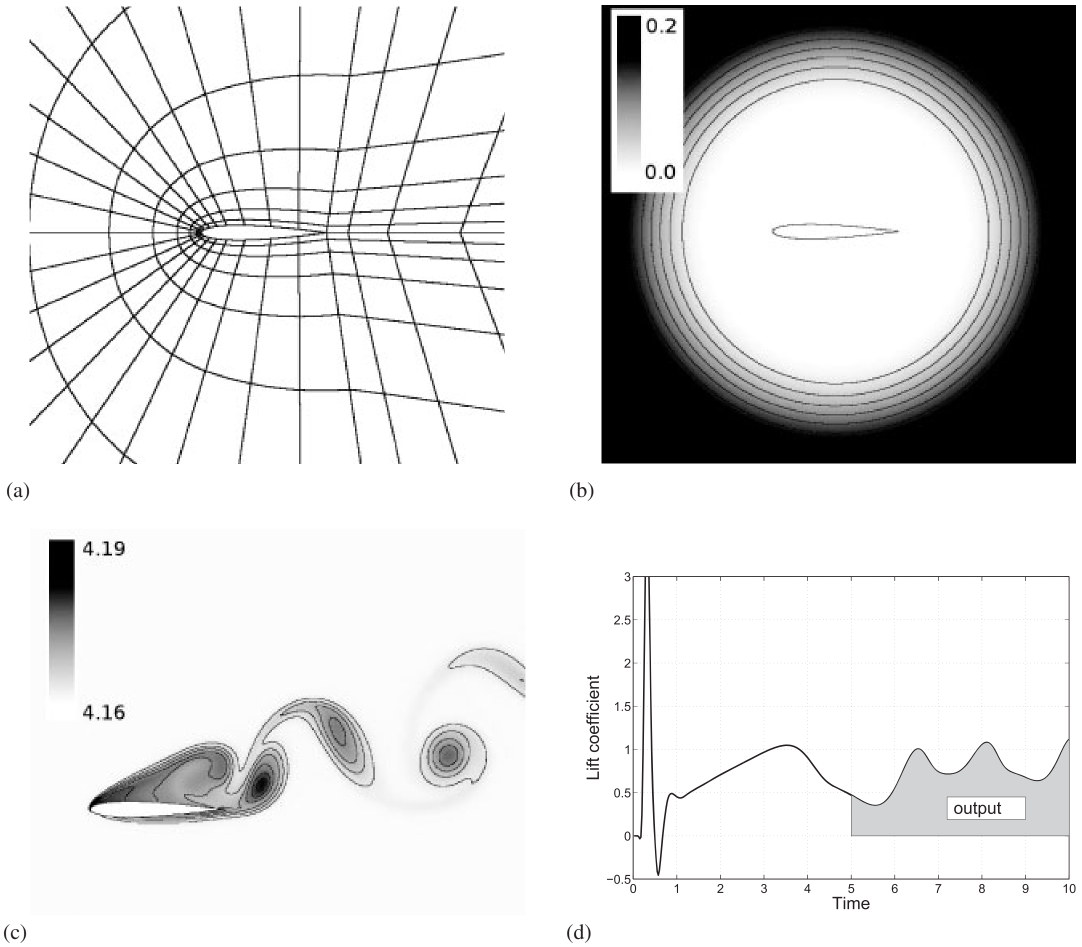


Figure 9. Impulsively started airfoil problem setup. The initial condition consists of a smooth velocity variation that matches the freestream to zero velocity at the airfoil. The output of interest is the integral of the lift coefficient over the second half of the simulation. Time is in units of airfoil chord divided by the freestream speed: (a) Coarse mesh; (b) initial Mach number contours; (c) final-time entropy contours; and (d) lift coefficient history.

where **u** is a length $s=4$ state vector of conserved variables, $\mathbf{F}_i^{\mathrm{I}}$ is the inviscid flux, and $\mathbf{F}_i^{\mathrm{V}}$ is the viscous flux. Summation is implied on the spatial dimension index, $i=1,2$. Additional details on the equations can be found in previous works, including [17].

The geometry is an NACA 0012 airfoil with a closed trailing-edge gap, illustrated with a coarse mesh of quartic quadrilateral elements in Figure 9(a). Adiabatic wall boundary conditions are prescribed on the airfoil, and freestream flow at Mach number $M=0.2$ and Reynolds number $Re=1000$ is prescribed on the farfield boundary, which is approximately 50 chord lengths away from the airfoil. The freestream flow is at an angle of $\alpha=20^\circ$ relative to the chord line of the airfoil.

In a true impulsive start, the initial condition would consist of freestream flow throughout the domain. The evolving system would simulate an airfoil accelerated from rest to freestream velocity in zero time. Although computationally tractable, the non-physical nature of this situation has an effect on the temporal accuracy of the solution, which is now limited by the discontinuous step change input at $t=0$. Therefore, in order to observe temporal superconvergence, we modify the initial condition to be consistent with the airfoil boundary conditions. Specifically, we blend the velocity to zero in a circular disk around the airfoil, as shown in Figure 9(b).

DOI: 10.1002/nme

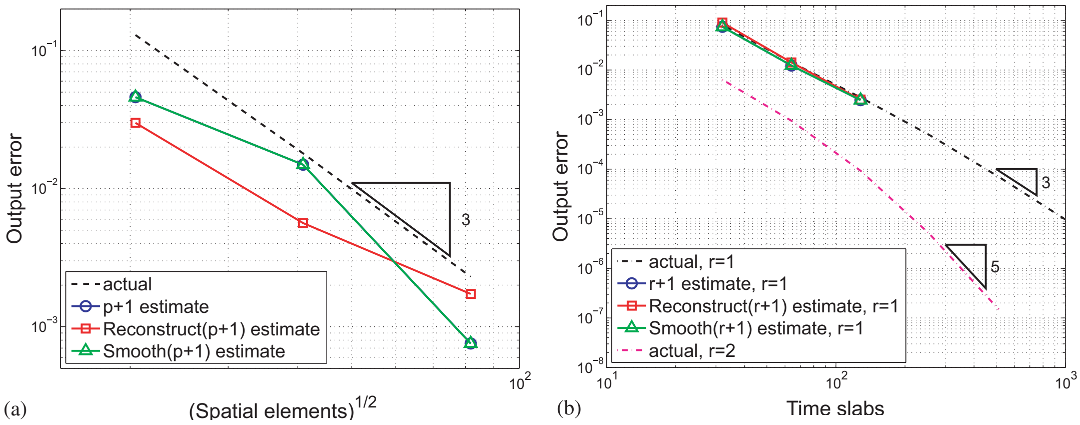


Figure 10. NACA 0012 error convergence. ‘Actual’ corresponds to the output error relative to a truth solution on a mesh refined in space or time: (a) Spatial convergence, $p=2$ and (b) temporal convergence.

The simulation time is set to the time required for the freestream to traverse 10 chord lengths of the airfoil. In this time, the airfoil interacts with the freestream flow and begins to shed vorticity from the leading and trailing edges. A entropy contour visualization of the final-time flowfield is shown in Figure 9(c).

The output of interest in this case is an integral of the lift coefficient, which is the lift nondimensionalized by the product of the freestream dynamic pressure and the chord length, over the second half of the simulation. The output definition is illustrated graphically in Figure 9(d). The oscillatory nature of the lift at this time is due to the onset of vortex shedding.

Figure 10 shows how the output error converges under spatial and temporal refinements. As in the previous example, the spatial error plot is generated by fixing the number of time slabs, while the temporal error plot is generated by fixing the number of elements. The actual output error converges at order $p+1$ under spatial refinement, which is optimal in this case in the presence of a singularity at the trailing edge. For a sufficiently fine temporal resolution, the $r=1$ and $r=2$ results indicate that the error superconverges at order $2r+1$ under temporal refinement, as expected given the smooth temporal data driving the problem.

Figure 10 also shows the convergence of the output error estimates obtained from (15) by an exact solve, reconstruction, and smoothing on an order-incremented space. In the spatial convergence plot, we see that the error estimates under-predict the error, and this is due to the relatively coarse spatial meshes used. Note that the inexact adjoint solve, ‘Smooth(p+1)’, yields nearly the same error estimates as an exact adjoint solve for this problem. The temporal convergence plot shows that all three error estimates predict the actual output error very well.

The error effectivities for each of the six adjoint choices considered are shown in Table III for nine combinations of spatial and temporal resolutions. On the coarsest spatial mesh, the effectivities are low for all the adjoint choices and temporal resolutions. The effectivities improve with the increasing spatial resolution, although on the finest space–time meshes considered they are not as close to one as in the previous example. Spatial reconstruction, in choices C and E, performs better in this problem than in the previous example, especially on the finest spatial mesh. In addition, we note that the inexact adjoint solve, choice F, yields nearly identical effectivities to the exact adjoint solve, choice A.

Figure 11 compares the effectivities of the various adjoint choices to the computational expense, as measured by the CPU factor. As in the previous example, only the four combinations of coarsest and finest spatial and temporal resolutions are shown. The difference in effectivities is less pronounced across the various adjoint choices in this case. In terms of cost, the most expensive error estimates come from choice A, the exact adjoint solve on $p+1, r+1$, which costs 7–11 times as much as the forward solve. The cost of the exact solve in choice B, $p+1, \Delta t/2$, is lower at three to five times the forward solve. As in the previous example, the adjoints obtained from temporal

DOI: 10.1002/nme

Table III. NACA 0012: error effectivities, η_{eff}, calculated using the six different fine-space adjoints with various spatial and temporal resolutions.

$N_{\mathrm{elem}} \backslash N_{\mathrm{slab}}$	32	64	128
(a) $\mathcal{V}_h^A : p+1, r+1$			
418	0.61	0.40	0.35
1672	0.95	0.82	0.80
6688	0.54	0.78	0.65
(b) $\mathcal{V}_h^B : p+1, \Delta t/2$			
418	0.46	0.29	0.25
1672	0.87	0.87	0.93
6688	0.90	0.91	1.26
(c) $\mathcal{V}_h^{\mathrm{C}}$: Reconstruct$(p+1), r+1$			
418	0.33	0.25	0.23
1672	0.97	0.60	0.37
6688	0.82	0.80	0.76
(d) $\mathcal{V}_h^{\mathrm{D}} : p+1$, Reconstruct$(r+1)$			
418	0.40	0.36	0.35
1672	0.90	0.74	0.77
6688	0.85	0.66	0.63
(e) $\mathcal{V}_h^{\mathrm{E}}$: Reconstruct$(p+1, r+1)$			
418	0.30	0.24	0.23
1672	0.90	0.58	0.37
6688	0.98	0.67	0.71
(f) $\mathcal{V}_h^{\mathrm{F}}$: Smooth$(p+1, r+1)$			
418	0.61	0.40	0.35
1672	0.95	0.82	0.80
6688	0.96	0.79	0.65

reconstruction, choice D, and inexact solution, choice F, perform consistently well and cost about four times as much as the forward problem. Choice C, employing spatial reconstruction only, is cheaper at approximately 2.5 times the cost of a forward solve. However, the cheapest strategy is the reconstruction in both space and time, choice E, which costs less than 1.5 times a forward solve. While in this case the effectivity of choice E is comparable to the other strategies, we saw in the previous example that the spatial reconstruction is not always robust.

The temporal error fractions for this example are shown in Figure 12. Aside from spatial reconstruction on the medium mesh, the temporal error fractions for all of the estimates tested are reasonably close to the actual error. In the worst cases, the estimates over-predict the temporal error fraction by a factor of 1.8 on the coarsest space–time mesh, and by a factor of 1.3 on the finest space–time mesh. The trends are again consistent with expectations, wherein the temporal error fraction decreases with temporal refinement and increases with spatial refinement.

5. CONCLUSIONS

This paper presents approximation strategies for computing scalar output quantities in unsteady simulations of convection-dominated flows. The forward and adjoint problems are discretized with a DG method in space and time on spatially static meshes. All elements are advanced with the same time step using an implicit iterative solution scheme based on an approximate factorization of the non-linear system. This scheme has been introduced previously in the literature and is extended to the discrete adjoint system in this work. The unsteady adjoint solution is used to compute output error estimates via the adjoint-weighted residual method. Owing to Galerkin orthogonality, the adjoint problem must be solved on a space that is finer than that of the forward problem. The choice

DOI: 10.1002/nme

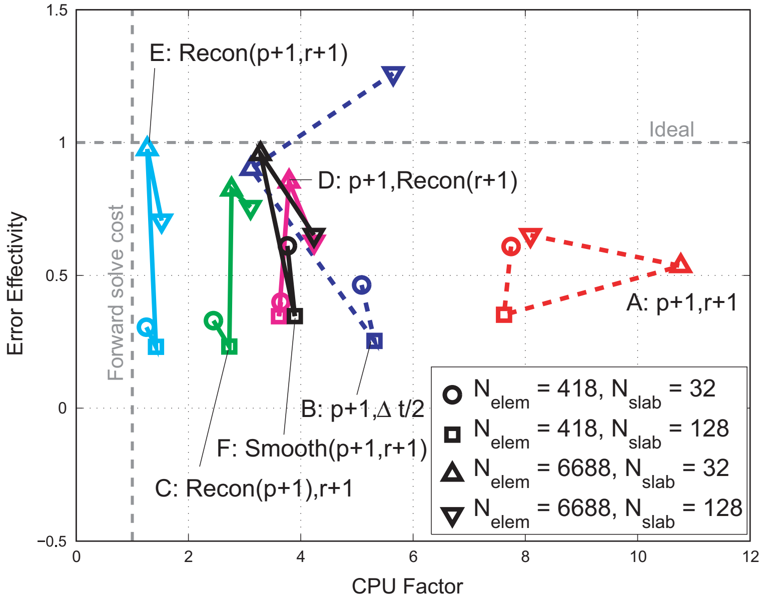


Figure 11. NACA 0012: effectivity versus CPU factor for various spatial and temporal resolution combinations. For clarity, points belonging to the same adjoint choice are connected on the plot.

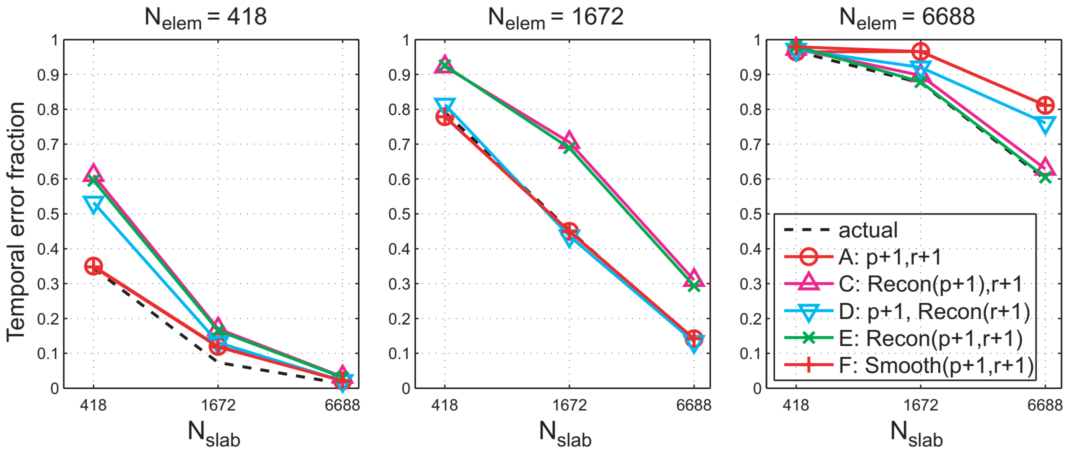


Figure 12. NACA 0012: separation of the output error into spatial and temporal contributions using the temporal error fraction. The 'actual' temporal error fractions are computed from (22) using very fine spatial and temporal discretization spaces in the expressions for $\delta J_H^{\text{space}}$ and δJ_H^{time} in (19).

of the finer space is not unique, and we investigate several common approximations, focusing on particularly the temporal discretization.

The two flows studied include one governed by a scalar convection diffusion reaction equation and the other by the compressible Navier–Stokes system. Time-integral output quantities are considered and these are shown to superconverge at a rate of $2r+1$ with temporal refinement, where r is the temporal approximation order. We find that adjoints obtained by exact solutions on enriched spaces yield accurate error estimates but at a significant computational cost. For the problems tested, the cost of these estimates is up to 10 times the cost of the forward solution. A cheaper alternative is an inexact adjoint solution obtained via a local smoother, and this is shown

DOI: 10.1002/nme

to yield consistently effective error estimates. The cheapest option tested is reconstruction in space and/or time, although the reconstruction loses effectivity on some of the coarse meshes.

In addition, we present a technique for separating the output error into contributions from the spatial and temporal discretizations. This calculation adds minimal expense as it is based on a projection of the fine-space adjoint onto coarser spatial and temporal meshes. The results show that the calculated spatial and temporal error fractions are quite accurate compared with the actual errors. This agreement indicates that the error estimation procedure is capable of differentiating errors that arise from insufficient spatial versus temporal resolution.

Even with the proposed approximations, output error estimation remains a non-trivial undertaking for unsteady problems. First, the forward states must be made available at every time step to compute the linearizations required in the adjoint solve. Second, our results indicate that the computational cost of the error estimation is generally several times that of the forward problem. However, one must be careful when generalizing these conclusions. For example, we note that for highly non-linear problems, such as the Reynolds-averaged Navier–Stokes equations, the cost of the adjoint solution relative to the forward solution could be significantly lower due to the linearity of the adjoint problem. Furthermore, storage requirements are less of an issue on modern high-performance architectures, and if necessary they can be alleviated with solution checkpointing [46].

Future work will focus on using localized forms of the output error estimates to drive adaptation of the spatial and temporal meshes. Although the adjoint solution and error estimation add to the solution cost, they have the potential to dramatically improve the robustness and efficiency of unsteady calculations. The improvements are expected for problems that exhibit propagation effects, such as convection-dominated flows, in which the optimal spatial and temporal resolutions are not known *a priori*. For such cases, an adaptive algorithm can target those spatial and temporal elements most important for the prediction of the output of interest. The associated error estimates can then guide the adaptation to produce discretizations that meet user-specified output tolerances.

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 2.2. For $r=1$, the three-step scheme is

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

$$\left[\mathbf{M}_H + \frac{\Delta t_1}{\sqrt{6}}\mathbf{A}_H\right]\Delta\mathbf{U}_H^2 = \mathbf{M}_H \mathbf{Y}_H,$$

$$\left[\mathbf{M}_H + \frac{2\Delta t_1}{3}\mathbf{A}_H\right]\Delta\mathbf{U}_H^1 = -2\bar{\mathbf{R}}_H^1 - \left[\mathbf{M}_H + \frac{\Delta t_1}{3}\mathbf{A}_H\right]\Delta\mathbf{U}_H^2,$$

where $\mathbf{Y}_H$ is an intermediate vector.

For $r=2$, the six-step scheme is

$$[\mathbf{M}_H + 60^{-1/3}\Delta t_1 \mathbf{A}_H]\mathbf{Y}_H^1 + \mathbf{M}_H \mathbf{G}_H^1 = 0,$$

$$[\mathbf{M}_H + 60^{-1/3}\Delta t_1 \mathbf{A}_H]\mathbf{Y}_H^2 - \mathbf{M}_H \mathbf{Y}_H^1 = 0,$$

$$[\mathbf{M}_H + 60^{-1/3}\Delta t_1 \mathbf{A}_H]\Delta\mathbf{U}_H^3 - \mathbf{M}_H \mathbf{Y}_H^2 = 0,$$

$$[\mathbf{M}_H + \sqrt{\tfrac{3}{20}}\Delta t_1 \mathbf{A}_H]\mathbf{Y}_H^3 + \mathbf{M}_H \mathbf{G}_H^2 = 0,$$

$$[\mathbf{M}_H + \sqrt{\tfrac{3}{20}}\Delta t_1 \mathbf{A}_H]\Delta\mathbf{U}_H^2 - \mathbf{M}_H \mathbf{Y}_H^3 = 0,$$

$$[\mathbf{M}_H + \tfrac{4}{15}\Delta t_1 \mathbf{A}_H]\Delta\mathbf{U}_H^1 + \mathbf{M}_H \mathbf{G}_H^3 = 0,$$

where $\mathbf{Y}_H^m$ are intermediate vectors and

$$\begin{aligned}
\mathbf{M}_H\mathbf{G}_H^1 &= \bar{\mathbf{R}}_H^1+\bar{\mathbf{R}}_H^2+\bar{\mathbf{R}}_H^3+\Delta t_1\mathbf{A}_H\mathbf{M}_H^{-1}[-\tfrac{2}{5}\bar{\mathbf{R}}_H^1+\tfrac{1}{10}\bar{\mathbf{R}}_H^2+\tfrac{3}{5}\bar{\mathbf{R}}_H^3\\
&\quad+\Delta t_1\mathbf{A}_H\mathbf{M}_H^{-1}(\tfrac{1}{20}\bar{\mathbf{R}}_H^1-\tfrac{1}{40}\bar{\mathbf{R}}_H^2+\tfrac{3}{20}\bar{\mathbf{R}}_H^3)],\\
\mathbf{M}_H\mathbf{G}_H^2 &= \tfrac{3}{2}\bar{\mathbf{R}}_H^1+\tfrac{9}{8}\bar{\mathbf{R}}_H^2+\tfrac{1}{2}\mathbf{M}_H\Delta\mathbf{U}_H^3\\
&\quad+\Delta t_1\mathbf{A}_H[\mathbf{M}_H^{-1}(\tfrac{3}{20}\bar{\mathbf{R}}_H^1+\tfrac{3}{10}\bar{\mathbf{R}}_H^2+\tfrac{1}{40}\Delta t_1\mathbf{A}_H\Delta\mathbf{U}_H^3)+\tfrac{1}{4}\Delta\mathbf{U}_H^3],\\
\mathbf{M}_H\mathbf{G}_H^3 &= 2\bar{\mathbf{R}}_H^1+\mathbf{M}_H(\tfrac{4}{3}\Delta\mathbf{U}_H^2-\tfrac{1}{3}\Delta\mathbf{U}_H^3)+\Delta t_1\mathbf{A}_H[\tfrac{2}{15}\Delta\mathbf{U}_H^2-\tfrac{1}{15}\Delta\mathbf{U}_H^3].
\end{aligned}$$

APPENDIX B: STABILITY AND CONVERGENCE OF THE APPROXIMATE ITERATIVE SOLVER

In this appendix, we present a linear stability and convergence analysis of the iterative solver based on the approximate factorization discussed in Section 2.2, for $r=1$ and $r=2$. The starting point is a scalar, linear, model equation,

$$\frac{\partial U}{\partial t}=\lambda U, \tag{B1}$$

where U is the scalar state and λ is a complex scalar. We consider a DG temporal discretization of this equation on a time slab of size Δt. The value $\lambda\Delta t$ determines the stability of the iterative schemes presented in Appendix A.

Denoting by U_k^n the state at the kth iteration of the approximate factorization solver, with $n=[1,\dots,r+1]$, and by U_*^n the true solution on the time slab, the error at the kth iteration is given by

$$E_{k+1}^n=S^{n,m}E_k^m, \quad S^{n,m}=I^{n,m}+\frac{\partial\Delta U^n}{\partial U^m}, \tag{B2}$$

where $E_k^m=U_k^m-U_*^m$, $I^{n,m}$ is the identity matrix, and ΔU^n is the state update from the approximate solver. Of interest for stability and accuracy is the maximum-magnitude eigenvalue of $S^{m,n}$, corresponding to the slowest-converging mode, as a function of $\lambda\Delta t$. Figure B1 shows the numerical results of the eigenvalue calculation for $r=1$ and $r=2$.

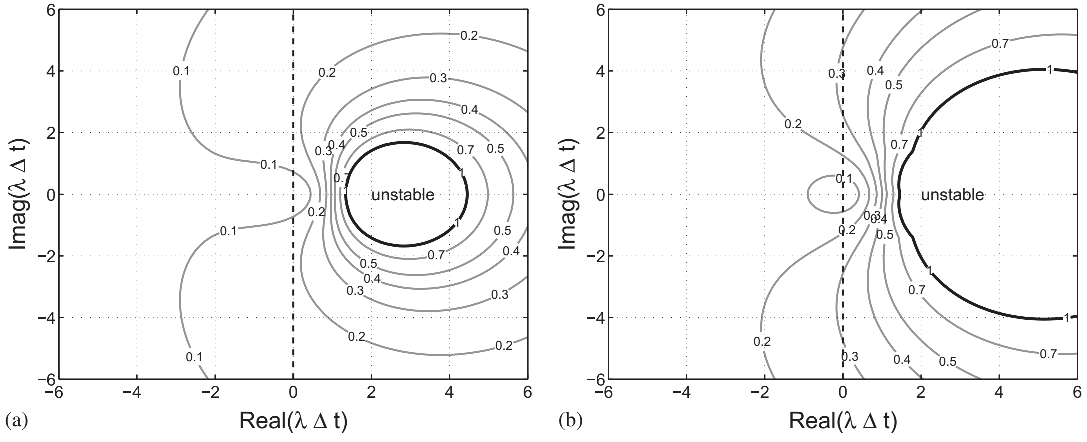


Figure B1. Contours of the maximum eigenvalue of the iterative error amplification matrix $S^{m,n}$ for $r=1$ and $r=2$. The unstable region corresponds to eigenvalue magnitudes greater than 1. Δt is the time slab size, and λ is defined in (B1): (a) $r=1$ and (b) $r=2$.

DOI: 10.1002/nme

We note that these contours are different from the standard eigenvalue stability diagrams for time-marching schemes wherein the present analysis concerns iterative stability of an approximate solver on a single time step. As shown, the iterative solver is unconditionally stable for λ with negative real part, corresponding to damped modes in the model equation. For sufficiently small Δt, the solver is also stable for exponentially growing modes.

The magnitude of the maximum eigenvalue of $S^{m,n}$ also determines the convergence rate of the approximate iterative solver. We see that for λ with negative real part, the maximum error amplification is about 0.19 for $r=1$ and about 0.32 for $r=2$—both of these correspond to very lightly damped modes. In addition, as Δt decreases, the iterative convergence rate increases, and in particular, the amplification factor approaches zero as $\Delta t \to 0$.

APPENDIX C: TEMPORAL SUPERCONVERGENCE OF INTEGRAL OUTPUTS

Extending the results of Adjerid *et al.* [38], we show that integral outputs calculated from solutions of temporal DG discretizations superconverge at a rate of $(\Delta t)^{2r+1}$. The result applies directly to other one-dimensional hyperbolic conservation laws discretized with DG. The starting point is an expansion for the temporal solution error, $\boldsymbol{\epsilon} = \mathbf{U}_H(t) - \mathbf{U}(t)$, on one-time slab,

$$\boldsymbol{\epsilon}(\xi) = \sum_{k=r+1}^{2r+1} \mathbf{Q}_k(\xi)\Delta t^k + O(\Delta t^{2r+2}), \tag{C1}$$

where $\mathbf{U}(t)$ is the solution with no temporal error, $\xi \in (-1, 1)$ is a slab reference coordinate, and $\mathbf{Q}_k(\xi)$ are vector-valued polynomials satisfying [38]

$$\int_{-1}^{1} \mathbf{Q}_k(\xi)\xi^j \, \mathrm{d}\xi = 0 \quad \text{for } k \geqslant r+1, \; j \geqslant 0, \text{ and } j+k < 2r+1. \tag{C2}$$

The temporal error in the integral output defined in (16) is

$$\begin{aligned} \text{temporal error} &= J(\mathbf{U}_H) - J(\mathbf{U}) \\ &= \int_{\Omega^{\text{time}}} [f(\mathbf{U}_H) - f(\mathbf{U})]\,\mathrm{d}t \\ &= \int_{\Omega^{\text{time}}} \left[\frac{\mathrm{d}f}{\mathrm{d}\mathbf{U}}\boldsymbol{\epsilon} + \frac{1}{2}\boldsymbol{\epsilon}^{\mathrm{T}} \frac{\mathrm{d}^2 f}{\mathrm{d}\mathbf{U}^2}\boldsymbol{\epsilon} + O(\boldsymbol{\epsilon}^3)\right]\mathrm{d}t. \end{aligned}$$

Terms involving powers of $\boldsymbol{\epsilon}$ higher than one are already $O(\Delta t)^{2r+2}$. Thus, we need only to consider the leading linear term in $\boldsymbol{\epsilon}$. Working on one time slab and expanding $\mathrm{d}f/\mathrm{d}\mathbf{U}$ in ξ about the slab center, $\xi = 0$, we obtain the following slab contribution to the output error:

$$\text{slab contribution} = \int_{-1}^{1} \sum_{j=0}^{\infty} \mathbf{a}_j^{\mathrm{T}} \xi^j \Delta t^{j+1} \boldsymbol{\epsilon}\, \mathrm{d}\xi,$$

where the $\mathbf{a}_j^{\mathrm{T}}$ are expansion coefficients for $\mathrm{d}f/\mathrm{d}\mathbf{U}$, and the factor Δt^{j+1} comes from the transformation from t to ξ in the time integral over the slab. Substituting the leading-order terms for $\boldsymbol{\epsilon}$ from (C1), the slab contribution becomes

$$\text{slab contribution} = \sum_{j=0}^{\infty} \mathbf{a}_j^{\mathrm{T}} \sum_{k=r+1}^{2r+1} \int_{-1}^{1} \mathbf{Q}_k(\xi)\xi^j \Delta t^{k+j+1}\, \mathrm{d}\xi.$$

Using (C2), we have that non-zero contributions to the output error occur when $j+k \geqslant 2r+1$, for which the local output error is $O(\Delta t^{2r+2})$ by the above equation. Summing over slabs, we obtain a global output error of $O(\Delta t^{2r+1})$.

The above result relies on smoothness in the output integrand, so that the above derivatives exist, and smoothness in the data driving the differential equation [38]. In particular, for point-value

DOI: 10.1002/nme

outputs $f(\mathbf{U}_H(t), t)$ is a delta distribution in time and, with the exception of the Radau points [38], superconvergence results do not hold. We note that in a DG discretization, non-smooth $f(\mathbf{U}_H(t), t)$ can still yield superconvergent outputs if $f(\mathbf{U}_H(t), t)$ is non-smooth only at the time slab interfaces, since in this case an expansion for $f(\mathbf{U}_H(t), t)$ still exists on each slab. In the adjoint interpretation of this situation, the discontinuous approximation space allows for optimal convergence of the adjoint, which will not be smooth at the time slab interfaces where f is not smooth.

REFERENCES

1. Pierce NA, Giles MB. Adjoint recovery of superconvergent functionals from PDE approximations. *SIAM Review* 2000; **42**(2):247–264.
2. Becker R, Rannacher R. An optimal control approach to a posteriori error estimation in finite element methods. In *Acta Numerica*, Iserles A (ed.). Cambridge University Press: Cambridge, 2001; 1–102.
3. Hartmann R, Houston P. Adaptive discontinuous Galerkin finite element methods for the compressible Euler equations. *Journal of Computational Physics* 2002; **183**(2):508–532.
4. Venditti DA, Darmofal DL. Anisotropic grid adaptation for functional outputs: application to two-dimensional viscous flows. *Journal of Computational Physics* 2003; **187**(1):22–46.
5. Sen S, Veroy K, Huynh D, Deparis S, Nguyen N, Patera A. 'Natural norm' a posteriori error estimators for reduced basis approximations. *Journal of Computational Physics* 2006; **217**:37–62.
6. Fidkowski KJ, Darmofal DL. Review of output-based error estimation and mesh adaptation in computational fluid dynamics. *American Institute of Aeronautics and Astronautics Journal* 2011; **49**(4):673–694.
7. Mani K, Mavriplis DJ. Discrete adjoint based time-step adaptation and error reduction in unsteady flow problems. *AIAA Paper 2007-3944*, 2007.
8. Mani K, Mavriplis DJ. Error estimation and adaptation for functional outputs in time-dependent flow problems. *Journal of Computational Physics* 2010; **229**:415–440.
9. Barth TJ. Space–time error representation and estimation in Navier–Stokes calculations. In *Complex Effects in Large Eddy Simulations*, Kassinos SC, Langer CA, Laccarino G, Moin P (eds). Lecture Notes in Computational Science and Engineering, vol. 26. Springer: Berlin, Heidelberg, 2007; 29–48.
10. Meidner D, Vexler B. Adaptive space–time finite element methods for parabolic optimization problems. *SIAM Journal on Control Optimization* 2007; **46**(1):116–142.
11. Schmich M, Vexler B. Adaptivity with dynamic meshes for space–time finite element discretizations of parabolic equations. *SIAM Journal on Scientific Computing* 2008; **30**(1):369–393.
12. Reed W, Hill T. Triangular mesh methods for the neutron transport equation. *Los Alamos Scientific Laboratory Technical Report LA-UR-73-479*, 1973.
13. Bassi F, Rebay S. High-order accurate discontinuous finite element solution of the 2-D Euler equations. *Journal of Computational Physics* 1997; **138**:251–285.
14. Cockburn B, Shu CW. Runge–Kutta discontinuous Galerkin methods for convection-dominated problems. *Journal of Scientific Computing* 2001; **16**(3):173–261.
15. Houston P, Hartmann R, Süli E. Adaptive discontinuous Galerkin finite element methods for compressible fluid flows. In *Numerical Methods for Fluid Dynamics VII*, Baines M (ed.). ICFD, 2001; 347–353.
16. Brezzi F, Marini L, Süli E. Discontinuous Galerkin methods for first-order hyperbolic problems. *Mathematical Models and Methods in Applied Sciences* 2004; **14**:1893–1903.
17. Fidkowski KJ, Oliver TA, Lu J, Darmofal DL. p-Multigrid solution of high-order discontinuous Galerkin discretizations of the compressible Navier–Stokes equations. *Journal of Computational Physics* 2005; **207**:92–113.
18. Johnson C, Szepessy A, Hansbo P. Discontinuous Galerkin methods for ordinary differential equations. *Mathematics of Computation* 1981; **36**(154):455–473.
19. Eriksson K, Johnson C, Thomee V. Adaptive finite element methods for parabolic problems I: a linear model problem. *Mathematical Modelling and Numerical Analysis* 1985; **19**(4):611–643.
20. Eriksson K, Johnson C. Adaptive finite element methods for parabolic problems I: a linear model problem. *SIAM Journal on Numerical Analysis* 1991; **28**(1):43–77.
21. Liu W, Ma H, Tang T, Yan N. A posteriori error estimates for discontinuous Galerkin time-stepping method for optimal control problems governed by parabolic equations. *SIAM Journal on Numerical Analysis* 2004; **42**(3):1032–1061.
22. Bar-Yoseph P, Elata D. An efficient L2 Galerkin finite element method for multi-dimensional non-linear hyperbolic systems. *International Journal for Numerical Methods in Engineering* 1990; **29**:1229–1245.
23. Lowrie RB, Roe PL, van Leer B. Properties of space–time discontinuous Galerkin. *Los Alamos Technical Report LA-UR-98-5561*, 1998.
24. van der Ven H, van der Vegt J. 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* 2002; **191**:4747–4780.
25. Klaij C, van der Vegt J, van der Ven H. Space–time discontinuous Galerkin method for the compressible Navier–Stokes equations. *Journal of Computational Physics* 2006; **217**:589–611.

DOI: 10.1002/nme

26. Richter T. Discontinuous Galerkin as time-stepping scheme for the Navier–Stokes equations. *Fourth International Conference on High Performance Scientific Computing Modeling*, *Simulation and Optimization of Complex Processes*, Hanoi, Vietnam, 2009.
27. Bassi F, Rebay S. GMRES discontinuous Galerkin solution of the compressible Navier–Stokes equations. In *Discontinuous Galerkin Methods*: *Theory*, *Computation and Applications*, Cockburn B, Karniadakis G, Shu C-W (eds). Springer: Berlin, 2000; 197–208.
28. Roe PL. Approximate Riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics* 1981; **43**:357–372.
29. Arnold DN, Brezzi F, Cockburn B, Marini LD. Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM Journal on Numerical Analysis* 2002; **39**(5):1749–1779.
30. Lu J. An a posteriori error control framework for adaptive precision optimization using discontinuous Galerkin finite element method. *Ph.D. Thesis*, Massachusetts Institute of Technology, Cambridge, MA, 2005.
31. Barth T, Larson M. A posteriori error estimates for higher order Godunov finite volume methods on unstructured meshes. In *Finite Volumes for Complex Applications III*, Herban R, Kröner D (eds). Hermes Penton: London, 2002; 41–63.
32. Giles MB, Süli E. Adjoint methods for PDEs: a posteriori error analysis and postprocessing by duality. *Acta Numerica* 2002; **11**:145–236.
33. Harriman K, Houston P, Senior B, Süli E. hp-version discontinuous Galerkin methods with interior penalty for partial differential equations with nonnegative characteristic form. *Technical Report NA 02/21*, Oxford University Computing Lab Numerical Analysis Group, 2002.
34. Hartmann R. Adjoint consistency analysis of discontinuous Galerkin discretizations. *SIAM Journal on Numerical Analysis* 2007; **45**(6):2671–2696.
35. Oliver TA, Darmofal DL. Impact of turbulence model irregularity on high-order discretizations. *AIAA Paper 2009-953*, 2009.
36. Hoffman J, Johnson C. Adaptive finite element methods for incompressible fluid flow. In *Error Estimation and Adaptive Discretization Methods for Computational Fluid Dynamics*, Barth T, Deconinck H (eds). Lecture Notes in Computational Science and Engineering, vol. 25. Springer: Heidelberg, 2003; 97–157.
37. Giles M, Pierce N. Adjoint error correction for integral outputs. In *Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics*. Lecture Notes in Computational Science and Engineering, vol. 25. Springer: Berlin, 2002.
38. Adjerid S, Devine KD, Flaherty J, Krivodonova L. A posteriori error estimation for discontinuous Galerkin solutions of hyperbolic problems. *Computer Methods in Applied Mechanical Engineering* 2002; **191**:1097–1112.
39. Fidkowski KJ. A simplex cut-cell adaptive method for high-order discretizations of the compressible Navier–Stokes equations. *Ph.D. Thesis*, Massachusetts Institute of Technology, Cambridge, MA, 2007.
40. Rannacher R. Adaptive Galerkin finite element methods for partial differential equations. *Journal of Computational and Applied Mathematics* 2001; **128**:205–233.
41. Venditti DA, Darmofal DL. Grid adaptation for functional outputs: application to two-dimensional inviscid flows. *Journal of Computational Physics* 2002; **176**(1):40–69.
42. Solín P, Demkowicz L. Goal-oriented hp-adaptivity for elliptic problems. *Computer Methods in Applied Mechanics and Engineering* 2004; **193**:449–468.
43. Huynh H. Collocation and Galerkin time-stepping methods. *AIAA Paper 2009-4323*, 2009.
44. Cockburn B, Luskin M, Shu CW, Süli E. Enhanced accuracy by post-processing for finite element methods for hyperbolic equations. *Mathematics of Computation* 2002; **72**(242):577–606.
45. Fidkowski KJ. A high-order discontinuous Galerkin multigrid solver for aerodynamic applications. *M.S. Thesis*, M.I.T., Department of Aeronautics and Astronautics, June 2004.
46. Griewank A, Walther A. Revolve: An implementation of checkpointing for the reverse or adjoint mode of computational differentiation. *ACM Transactions on Mathematical Software* 2000; **26**(1):19–45.

DOI: 10.1002/nme