AIAA AVIATION Forum
17-21 June 2019, Dallas, Texas
AIAA Aviation 2019 Forum

10.2514/6.2019-2951

Unsteady Combined Entropy and Output-based Adjoint Approach for Mesh Refinement and Error Estimation

Kevin T. Doetsch* and Krzysztof J. Fidkowski†
University of Michigan, Ann Arbor, MI 48109

This paper presents a strategy for estimating temporal and spatial numerical errors in scalar outputs of unsteady fluid dynamics simulations using a combination of two error indicators: one based on a user-specified engineering output, and the other based on entropy variables. This work is an extension of a combined error indicator approach for steady-state simulations introduced in our previous work. Using an entropy-based approach to calculate error estimates is computationally advantageous as it does not require the solution of an auxiliary adjoint equation. This advantage is significant for unsteady problems, where the reverse adjoint solve is eliminated, as the entropy variables are computed directly from the state. Unfortunately, unlike the output-based approach that specifically targets regions of the spatial domain that are critical to an output, the entropy-based spatial error indicator is prone to target all regions of the domain where spurious entropy is generated. Through a combined approach, in which both spatial and temporal errors are combined, the limitations of the entropy-based approach are mitigated, leading to much better output error estimates that in many instances are comparable to those obtained using only the output-based approach. The main approach for combining the indicators investigated in this work is one that uses a coarser output-based adjoint that, while less accurate by itself, is far less computationally expensive. In addition, the use of masks to help improve the spatial error estimates using the entropy variables is also considered. This work focuses predominantly on ways of combining the error indicators for a simulation that uses the compressible Navier-Stokes equations with mesh optimization via error sampling and synthesis (MOESS).

I. Introduction

Through the use of error estimates and optimized meshes, solution-based adaptive methods are becoming popular approaches for obtaining accurate solutions from Computational Fluid Dynamics (CFD) simulations [1–7]. These methods rely on an indicator to drive the mesh adaptation to produce the lowest possible output error estimate. Much of the work in this area has focused on steady problems [8–11]. In the literature, extensive work has been done studying the optimal indicator in terms of robustness, accuracy, and expense. Two previously-studied indicators are those based on engineering scalar outputs [4, 10, 12] and one based on entropy variables [13, 14]. While output-based indicators typically are chosen over the entropy variable indicator because of their overall better reliability to produce an accurate error estimate, entropy variables pose a significant advantage in that they are far less computationally expensive to compute relative to output-based adjoints. In our previous work [15], a new indicator was introduced for steady-state flow simulations that combined both the entropy-based and output-based indicators to yield an entirely new indicator. The current work presented in this paper focuses on applying that same approach of combining the two indicators for unsteady simulations.

Output-based adaptive methods are advantageous since mesh resolution is governed by the desired output of interest. This leads to mesh refinement in areas that are critical to the prediction of the particular output in consideration [3, 4, 9, 16]. Robust identification of important regions of the mesh is possible through a linear adjoint solution, which yields the sensitivity of the output of interest to residual source perturbations. An adjoint-weighted residual method excels in targeting these regions of the domain, leading to an accurate output error estimate. Despite this, the output-based approach does have drawbacks. The first is that it is not always clear which output should be chosen prior to adaptation. In addition, the solution of the adjoint can be computationally expensive, especially for unsteady simulations, where the adjoint problem has to be marched backward in time and states have to be read back in from disk or recalculated to obtain the required linearizations.

*Graduate Research Assistant, Department of Aerospace Engineering, kevintd@umich.edu, and AIAA Member
†Associate Professor, Department of Aerospace Engineering, kfid@umich.edu, and AIAA Senior Member.

The entropy variable approach alleviates both of these issues with the output-based approach. Since the adaptation is based on the entropy variables, there is no need to choose a single output of interest to govern the mesh refinement. This alleviates a burden on the user and produces an all-around "good" solution. More significant is that the entropy variables are far less computationally expensive to obtain since they are computed from a direct variable transformation of the conservative state. Unfortunately, the entropy variables target all regions of the mesh where spurious entropy is generated, regardless of whether or not they affect specific outputs of engineering interest

The logic behind the combined approach is that it should ideally take advantage of the strengths of each indicator while limiting their respective downsides. As previously documented [15], this approach works very well for many types of steady-state simulations. Not only does the combined approach outperform the entropy-based approach significantly, it even exceeds the performance of output-based approaches in some of the cases that were analyzed. Since these simulations were all steady, the cost for computing the adjoint required to obtain the output-based indicator was not nearly as severe as it is for unsteady simulations. This is the motivation behind examining this combined indicator approach for unsteady simulations.

Unsteady simulations pose significantly more computational cost for output-based adjoint approaches due to the unsteady adjoint equation. Current research has put an increased focus on extending output-based adaption to unsteady problems using a variety of adaptation mechanics [17–23]. The issue with output-based approaches is that the solution of the unsteady adjoint equation is expensive, as it must be marched backward in time. This leads to significantly higher computational costs compared to steady-state simulations. It also requires saving the primal state at each time step or at checkpoints, further adding to the computational cost.

The entropy variable indicator approach is less computationally expensive, but it is not as reliable as the output-based approach due to the tendency to refine areas of the mesh where spurious entropy is being generated that do not play a significant role in the accurate estimation of a particular engineering output. This is where the benefit of the combined approach exists as it can use a coarser output-based adjoint indicator, in tandem with the entropy-based indicator, to get a much more accurate output error estimate at a cheaper computational expense.

The outline of this paper is as follows. Section II reviews unsteady adjoints and shows that entropy variables satisfy an unsteady adjoint equation. Section III introduces the semi-discrete form of the governing equation. This section also reviews the derivation of the continuous-in-time adjoint equation used in the output-error estimates. Section IV outlines the localization of the error and the space-time optimization procedure. The variations of the combined indicator approach are discussed in Section V. Finally, Section VI presents the results of a compressible Navier-Stokes simulation using the new unsteady combined indicator approach.

II. Governing Equations and Unsteady Adjoint

Consider a system of partial differential equations that arises from an unsteady conservation law,

$$\mathbf{r}(\mathbf{u}) = \partial_t \mathbf{u} + \partial_i \boldsymbol{\mathcal{F}}_i = \mathbf{0}, \tag{1}$$

where $\mathbf{u}(\vec{x}, t) \in \mathbb{R}^s$ is the conservative state vector, i is the spatial index, and $\boldsymbol{\mathcal{F}}_i$ is a general flux that may include convective and diffusive contributions. The computational domain is assumed to be a tensor product between a spatial domain Ω and a temporal domain T, where

$$\vec{x} \in \Omega, \quad t \in T = [t_0, t_f].$$

The boundary of the spatial domain is denoted by $\partial\Omega$, while the boundary of the temporal domain is denoted by ∂T and consists of the endpoint times t_0 and t_f.

A. Unsteady Adjoint Equation

A continuous adjoint, steady or unsteady, is a Green's function that relates residual source perturbations in the original equation to an associated output of interest. It can equivalently be defined as a Lagrange multiplier on the residual when used in constrained optimization.

For a scalar output $J(\mathbf{u})$, a Lagrangian can be defined as

$$\mathcal{L} = J(\mathbf{u}) - \int_T \int_\Omega \boldsymbol{\psi}^{\mathrm{T}} \mathbf{r}(\mathbf{u}) \, d\Omega \, dt, \tag{2}$$

where the Lagrange multiplier, $\boldsymbol{\psi} \in \mathcal{V}$, is the adjoint solution [9, 24]. Enforcing stationary of the Lagrangian with respect to allowable variations in the state yields the adjoint equation,

$$J'[\mathbf{u}]\delta\mathbf{u} - \int_T \int_\Omega \boldsymbol{\psi}^{\mathrm{T}} \mathbf{r}'[\mathbf{u}]\delta\mathbf{u}\, d\Omega\, dt = 0, \quad \forall \quad \delta\mathbf{u} \in \boldsymbol{\mathcal{V}}^{\mathrm{perm}}, \tag{3}$$

where $\boldsymbol{\mathcal{V}}^{\mathrm{perm}}$ denotes the space of allowable state variations, that is, those allowed by the boundary conditions. The primes in the above equation denote Fréchét linearization with respect to the arguments in the square brackets. Using Eqn. 1, Eqn. 3 can be integrated by parts to yield an interior differential equation and boundary conditions for the adjoint,

$$\begin{aligned} J'[\mathbf{u}]\delta\mathbf{u} &= \int_T \int_\Omega \boldsymbol{\psi}^{\mathrm{T}} \mathbf{r}'[\mathbf{u}]\delta\mathbf{u}\, d\Omega\, dt \\ &= \int_T \int_\Omega \boldsymbol{\psi}^{\mathrm{T}} \left[\partial_t \delta\mathbf{u} + \partial_i \left(\boldsymbol{\mathcal{F}}_i'[\mathbf{u}]\delta\mathbf{u}\right)\right] d\Omega dt \\ &= -\int_T \int_\Omega \left[\partial_t \boldsymbol{\psi}^{\mathrm{T}} + \partial_i \boldsymbol{\psi}^{\mathrm{T}} \boldsymbol{\mathcal{F}}_i'[\mathbf{u}]\right] \delta\mathbf{u} d\Omega dt + \int_T \int_{\delta\Omega} \boldsymbol{\psi}^{\mathrm{T}} \boldsymbol{\mathcal{F}}_i'[\mathbf{u}]\delta\mathbf{u}\, n_i\, dS\, dt + \left[\int_\Omega \boldsymbol{\psi}^{\mathrm{T}} \delta\mathbf{u}\, d\Omega\right]_{t_0}^{t_f} \end{aligned} \tag{4}$$

where dS is the differential area element on $\partial\Omega$ and n_i is the i^{th} component of the outward pointing spatial normal.

The first term in the right-hand-side of Eqn. 4 is the adjoint equation on the interior space-time-domain. It can be rewritten as

$$\partial_t \boldsymbol{\psi} + \left(\boldsymbol{\mathcal{F}}_i'[\mathbf{u}]\right)^{\mathrm{T}} \partial_i \boldsymbol{\psi} = \mathbf{0}, \quad \forall\, \delta\mathbf{u} \in \boldsymbol{\mathcal{V}}^{\mathrm{perm}}, \tag{5}$$

while the second and third terms are associated with the boundary and initial conditions, respectively. It is important to note that Eqn. 5 does not constitute the complete adjoint equation when J contains contributions from the interior of the space-time domain[25].

B. Entropy Variables as Unsteady Adjoints

The previous section introduced the continuous unsteady adjoint, $\boldsymbol{\psi}$. This section demonstrates that entropy variables can also be cast as unsteady adjoints. For second-order conservation laws, Eqn. 1 can be rewritten in the following quasi-linear form:

$$\partial_t \mathbf{u} + \mathbf{A}_i \partial_i \mathbf{u} - \partial_i \left(\mathbf{K}_{ij} \partial_j \mathbf{u}\right) = \mathbf{0}, \tag{6}$$

where $\mathbf{A}_i \partial_i \mathbf{u}$ is the inviscid flux and $-\mathbf{K}_{ij}\partial_j \mathbf{u}$ is the viscous flux. We assume the existence of a scalar entropy conservation law that takes the form

$$U_t + \partial_i F_i = 0, \tag{7}$$

where $F_i(\mathbf{u})$ is the entropy flux associated with the entropy function $U(\mathbf{u})$. Both the entropy function and flux satisfy the compatibility relation $U_{\mathbf{u}}\mathbf{A}_i = (F_i)_{\mathbf{u}}$. The entropy variables are defined as $\mathbf{v} = U_{\mathbf{u}}^{\mathrm{T}}$, and they symmetrize the unsteady set of conservation laws in the sense that[26, 27]: the transformation Jacobian matrix, $\mathbf{u}_{\mathbf{v}}$, is symmetric, positive definite; and $\mathbf{A}_i \mathbf{u}_{\mathbf{v}}$ is symmetric. In addition, the entropy variables $\mathbf{v}$ must also symmetrize $\mathbf{K}_{ij}$, in the sense that $\widetilde{\mathbf{K}}_{ij} = \widetilde{\mathbf{K}}_{ji}^{\mathrm{T}}$, where $\widetilde{\mathbf{K}}_{ij} = \mathbf{K}_{ij}\mathbf{u}_{\mathbf{v}}$[27]. Substituting $\partial_i \mathbf{u} = \mathbf{u}_{\mathbf{v}} \partial_i \mathbf{v}$ into Eqn. 6 and taking the transpose yields the following equation for the entropy variables,

$$\partial_t \mathbf{v}^{\mathrm{T}} \mathbf{u}_{\mathbf{v}} + \partial_i \mathbf{v}^{\mathrm{T}} \mathbf{A}_i \mathbf{u}_{\mathbf{v}} - \partial_i \left(\partial_j \mathbf{v}^{\mathrm{T}} \widetilde{\mathbf{K}}_{ji}\right) = \mathbf{0}. \tag{8}$$

Due to the sign of the second term being negative, this is no longer a strict mathematical adjoint. However, the entropy variables still act as an adjoint, since they represent the sensitivity of a particular output to residual perturbations. The associated output can be determined by substituting the entropy variables in place of $\boldsymbol{\psi}$, and the inviscid and viscous flux terms in place of the general flux term in the output linearization, Eqn. 4. Using the aforementioned properties of the entropy variables, the entropy variables serve as an "adjoint" solution for an output that takes the form of

$$J = \int_T \int_{\partial\Omega} F_i n_i dS dt + \int_T \int_\Omega \partial_i \mathbf{v}^{\mathrm{T}} \widetilde{\mathbf{K}}_{ij} \partial_j \mathbf{v}\, d\Omega\, dt - \int_T \int_{\partial\Omega} \mathbf{v}^{\mathrm{T}} \widetilde{\mathbf{K}}_{ij} \partial_j \mathbf{v}\, n_i\, dS\, dt + \left[\int_\Omega U dS\right]_{t0}^{t_f}. \tag{9}$$

The first term in Eqn. 9 is the net outflow of entropy (U) through the spatial domain boundary, integrated over time. The second term is the generation of entropy due to viscous dissipation within shear layers, vortices, and shocks, again integrated over time. The third term is the entropy diffusion across the spatial boundary integrated over time. Finally, the last term is the net outflow of entropy from the space-time domain through the initial and final time points. The output J is zero for an exact solution to the differential equation, but it is typically not zero in a numerical simulation on a discretized space-time mesh. This is because most stable numerical schemes, though conservative, introduce spurious entropy. Adapting on J using the entropy variables as adjoints is the driver for the entropy-based approach for spatial and temporal refinement. Since the calculation of $\mathbf{v}$ is far cheaper than that of $\boldsymbol{\psi}$, this approach is computationally advantageous. However, it is not as reliable as it targets all regions of the spatial domain that exhibit spurious entropy generation.

C. Entropy Function

The entropy function that yields entropy variables that symmetrize both the inviscid and viscous term in the compressible Navier-Stokes equations, with heat-conduction included, is unique up to additive and multiplicative constants[27],

$$U = -\rho S/\left(\gamma - 1\right), \qquad S = \ln P - \gamma \ln \rho, \tag{10}$$

where P is the pressure, ρ is the density, γ is the ratio of specific heats, and S is the physical entropy. Differentiating U with respect to the conservative state $\mathbf{u}$ yields the entropy variables,

$$\mathbf{v} = U_{\mathbf{u}}^{\mathrm{T}} = \left[\frac{\gamma - S}{\gamma - 1} - \frac{1}{2}\frac{\rho V^2}{P},\ \frac{\rho u_i}{P},\ -\frac{\rho}{P}\right]^{\mathrm{T}}, \tag{11}$$

where $V^2 = u_i u_i$ is the square of the velocity magnitude, and $P = (\gamma - 1)(\rho E - \rho V^2/2)$, where E is the total energy per unit mass. The entropy variables are therefore obtained using a nonlinear transformation of the conservative variables. The corresponding entropy flux function is $F_i = u_i U$.

III. Discretization

Using a semi-discrete approach, the spatial terms in the governing equations are discretized using a discontinuous Galerkin (DG) method. In this method the state is approximated by a sum of polynomial basis functions of order p_e on each element Ω_e, with no continuity constraints imposed across the elements. The weak form of Eqn. 1 is found by multiplying it with the basis functions on the same approximation space, integrating by parts, and coupling elements via unique fluxes. The DG discretization of the compressible Navier-Stokes equations uses the Roe approximate Riemann solver[28] for the convective flux and the second form of Bassi and Rebay for the viscous flux[29]. The DG approach yields a system of ordinary differential equations,

$$\bar{\mathbf{R}}(\mathbf{U}) \equiv \mathbf{M}\frac{d\mathbf{U}}{dt} + \mathbf{R}(\mathbf{U}) = \mathbf{0}, \tag{12}$$

where $\mathbf{U} \in \mathbb{R}^N$ is the discrete state vector of basis function coefficients, $\mathbf{R}$ is the discrete spatial residual vector, $\mathbf{M}$ is the mass matrix, and $\bar{\mathbf{R}}$ is the strong-form unsteady residual.

Unlike the spatial discretization, which used a specific form of the discretization, general time marching schemes were used to advance Eqn. 1 in time from $t = 0$ to $t = T$. For this paper, implicit Runge-Kutta methods (DIRK3, DIRK4) were used.

A. Continuous-in-Time Adjoint

Consider an unsteady output of the form

$$\bar{J} \equiv \int_0^T J(\mathbf{U}(t), t)dt + J_T(\mathbf{U}(T)), \tag{13}$$

where J and J_T are functionals of the spatial distribution of the state via the discrete coefficients $\mathbf{U}$. It is important to note that J_T is the function of only the final-time state, $\mathbf{U}(T)$. To derive the unsteady adjoint equation, a Lagrangian is

defined as

$$\mathcal{L} \equiv \bar{J} + \int_0^T \mathbf{\Psi}^{\mathrm{T}} \bar{\mathbf{R}} dt = \bar{J} + \int_0^T \mathbf{\Psi}^{\mathrm{T}} \left(\mathbf{M} \frac{d\mathbf{U}}{dt} + \mathbf{R}(\mathbf{U}) \right) dt. \tag{14}$$

Integrating the first term in the integral by parts, substituting the unsteady form of the output, and further simplifying yields the unsteady adjoint equation [30],

$$-\mathbf{M} \frac{d\mathbf{\Psi}}{dt} + \frac{\partial \mathbf{R}}{\partial \mathbf{U}}^{\mathrm{T}} \mathbf{\Psi} + \frac{\partial J^{\mathrm{T}}}{\partial \mathbf{U}} = \mathbf{0}, \tag{15}$$

subject to the terminal condition

$$\mathbf{\Psi}(T) = -\mathbf{M}^{-1} \frac{dJ_T^{\mathrm{T}}}{d\mathbf{U}}. \tag{16}$$

The terminal condition makes it necessary to solver for the adjoint backwards in time using any time-integration scheme. We can define a reverse time variable, $\tau = T - t$, in order to rewrite Eqn. 15 as

$$\mathbf{M} \frac{d\mathbf{\Psi}}{d\tau} + \frac{\partial \mathbf{R}}{\partial \mathbf{U}} \mathbf{\Psi} + \frac{\partial J^{\mathrm{T}}}{\partial \mathbf{U}} = \mathbf{M} \frac{d\mathbf{\Psi}}{d\tau} + \mathbf{R}^{\mathbf{\Psi}}(\mathbf{U}, \mathbf{\Psi}) = \mathbf{0}, \tag{17}$$

where $\mathbf{R}^{\mathbf{\Psi}}$ is the adjoint residual. The above equation is almost identical to the primal form, the only difference being the leading sign on the unsteady term. This makes it possible to solve backwards in time for the adjoint using the same techniques required to solve for the primal.

B. Temporal Reconstruction

Temporal reconstruction of the primal solution at various time nodes is necessary for nonlinear problems because the adjoint residual in Eqn. 17 depends on the primal state $\mathbf{U}$. In multi-state time integration, as well as error estimation, the primal state is not known for many of the intermediate time nodes. This makes it difficult, without significantly more storage or computation, to obtain an accurate representation of the adjoint residual. To alleviate this issue, we solve for both the primal and adjoint at the intermediate time nodes using a temporal reconstruction with a prescribed order of accuracy. Details on this reconstruction can be found in [30].

C. Output Error Estimation

Given a particular engineering output, $\bar{\mathbf{J}}$, an adjoint solution obtained from an adjoint-weighted residual method can be used to estimate numerical errors and drive adaption[9, 10]. For unsteady simulations, both the temporal and spatial errors must be quantified. Denoting by $\mathbf{U}_H(t)$ the approximate primal solution obtained from a given time integration method, and by $\mathbf{U}_h^H(t)$ its injection into a higher-order space, $p \to p+1$, the unsteady output error can be estimated through an adjoint-weighted residual [30],

$$\delta \bar{J} \approx -\int_0^T \mathbf{\Psi}_h^{\mathrm{T}} \bar{\mathbf{R}}_h(\mathbf{U}_h^H) dt, \tag{18}$$

where the fine-space (order $p+1$) adjoint $\mathbf{\Psi}_h$ and residual $\bar{\mathbf{R}}_h$ are calculated using a high-order time integration method.

IV. Adaptive Mechanics

The following section details the adaptation of the spatial and temporal discretization to obtain an accurate output error estimate. The nomenclature in this section uses $\mathbf{\Psi}$ for the unsteady output-based adjoint. Replacing $\mathbf{\Psi}$ with $\mathbf{v}$ in the following equations yields an error estimate based on the entropy variables instead.

A. Error Localization

The error estimate in Eqn. 18 is a time integral of an inner product between vectors containing spatially-local data specific to each element. The output error can be written as

$$\delta \bar{J} \approx -\int_0^T \mathbf{\Psi}_h^{\mathrm{T}} \bar{\mathbf{R}}_h(\mathbf{U}_h^H) dt = \sum_{n=1}^{N_t} \sum_{e=1}^{N_e} \underbrace{\int_{t^{n-1}}^{t^n} -\mathbf{\Psi}_{h,e}^{\mathrm{T}} \bar{\mathbf{R}}_{h,e} \left(\mathbf{U}_h^H \right) dt}_{\mathcal{E}_e^n}, \tag{19}$$

where N_t is the number of time steps, N_e is the number of elements, and the subscript e on the adjoint and residual denotes restriction to element e[30]. The contribution to the overall error estimate at time step n, $\mathcal{E}_e^n$, is available for each element. The temporal error contribution can be calculated using a spatially down-projected adjoint [31],

$$\mathcal{E}_e^{n,time} = -\int_{t^{n-1}}^{t^n} \left(\mathbf{I}_H^h \boldsymbol{\Psi}_{h,e}\right)^{\mathrm{T}} \bar{\mathbf{R}}_{H,e}(\mathbf{U}_H)dt, \tag{20}$$

where $\mathbf{I}_H^h$ is a least-squares spatial projection operator from order $p_e + 1$ to order p_e on element e. Note that the residual in Eqn. 20 is still computed using the fine time-integration method. The temporal error is then used to calculate the elemental spatial error at each time step via

$$\mathcal{E}_e^{n,\text{space}} \equiv \mathcal{E}_e^n - \mathcal{E}_e^{n,\text{time}}. \tag{21}$$

The aggregated temporal and spatial errors are then given by the sum over all of the elements and time slabs:

$$\delta\bar{J}^{\text{time}} = \sum_{n=1}^{N_t}\sum_{e=1}^{N_e} \mathcal{E}_e^{n,\text{time}}, \quad \delta\bar{J}^{\text{space}} = \sum_{n=1}^{N_t}\sum_{e=1}^{N_e} \mathcal{E}_e^{n,\text{space}} = \delta\bar{J} - \delta\bar{J}^{\text{time}}. \tag{22}$$

Often in adaptation conservative error estimates of the localized errors are used in order to avoid noise caused by coincidental cancellation of errors. These expressions are similar to those in Eqn. 22, but with absolute values:

$$\epsilon^{\text{time}} \equiv \sum_{n=1}^{N_t}\left|\sum_{e=1}^{N_e} \mathcal{E}_e^{\text{n,time}}\right|, \quad \epsilon^{\text{space}} \equiv \sum_{e=1}^{N_e} \epsilon_e^{\text{space}}, \quad \epsilon_e^{\text{space}} = \left|\sum_{n=1}^{N_t} \mathcal{E}_e^{\text{n,space}}\right|. \tag{23}$$

B. Mesh Optimization

With the spatial and temporal error estimates known for a given mesh, the next step is deciding how to allocate the degrees of freedom based on those error estimates. The method used in this work is based on the unsteady mesh optimization procedure introduced in [30], which relies on the aggregated output-based error estimates discussed in the previous work, and a cost model based on the total number of space-time degrees of freedom. The relatively simple model uses a measure based only on the total number of space-time degrees of freedom. It equally treats the choices of increasing temporal degrees of freedom by adding more time steps or increasing the spatial degrees of freedom by increasing the mesh resolution, in as much as these affect the total number of degrees of freedom. For a given mesh, the total number of space-time degrees of freedom, C, is given by the product of both the temporal, C^{time}, and spatial, C^{spatial}, degrees of freedom. User-prescribed growth rates, f^{space} and f^{time}, were used for each adaptive iteration to govern the relative change in cost:

$$C^{\text{space}} = C_0^{\text{space}} f^{\text{space}}, \quad C^{\text{time}} = C_0^{\text{time}} f^{\text{time}}, \tag{24}$$

where the subscript 0 implies the degrees of freedom of the current mesh. The growth factors are chosen so that the marginal error to cost ratio in both space and time is relatively the same. This means that for each adaptive iteration, either the spatial or temporal degrees of freedom will be increased more. In addition, the number of degrees of freedom can decrease, leading to a coarsening of the mesh.

To enable a solution for both f^{space} and f^{time}, a constraint is imposed on the growth of the total degrees of freedom: $f^{\text{space}} f^{\text{time}} = f^{\text{tot}}$. In [30], f^{tot} was a user-prescribed growth rate. This leads to an increase in total degrees of freedom allocation for each adaptive iteration. This make a comparison of methods at the same degrees of freedom difficult if the number of degrees of freedom drifts during adaptation.

In this work, we use a user-prescribed degrees of freedom target in lieu of a user-prescribed rate. For each adaptive iteration, we calculate f^{tot} by taking the degrees of freedom target divided by the current number of degrees of freedom. This allows us to use the same expressions for both f^{space} and f^{time} in terms of f^{tot} presented in [30] to determine the cost allocation for each iteration. Whereas in that work this lead to a relative increase in degrees of freedom for each adaptive iteration, in this approach the total degrees of freedom always moves closer to the user-prescribed target. Given the cost allocation for temporal and spatial discretization, we use the approach outlined in our previous work [30] to optimize the mesh, which is based on the MOESS (mesh optimization through error sampling and synthesis) algorithm [32, 33]. It is worth noting that the entropy variables can be used in place of the output-based adjoint in the MOESS sampling

approach. This leads to a new sampling approach that uses entropy variables to determine the refinement of a particular element in the mesh. Both entropy-based and output-based sampling are considered in this work. For consistency purposes, only cases that use the output-based adjoint to govern refinement used output-based sampling. All other cases, all of which at least partially rely on adaption using entropy variables, used entropy-based sampling. While output-based sampling could be used in the combined approach, it makes more sense to use the entropy-based sampling as we will soon discuss.

V. Combined Indicator Approach

In the previous section, the error estimation and adaptation process using the unsteady output-based adjoint, $\mathbf{\Psi}$, was presented. As previously mentioned, the same general process is repeated for the entropy-based approach. The only difference is that the entropy variables, $\mathbf{v}$, in Eqn. 11 are substituted for the output-based adjoint. This leads to spatial and temporal error indicators based on the entropy variables. In this work entropy variables are used to calculate both conservative and non-conservative spatial and temporal error indicators. With the ability to calculate separate error indicators using either the output-based adjoint or entropy variables, the option now exists to combine them. In this paper, several ways of combining the indicators are considered.

A. Combination of Error Indicators using a Fine-Space Output-based Adjoint

In this approach, two sets of non-conservative, aggregated spatial and temporal error indicators are calculated via Eqn. 22. The first set is based on using the output-based adjoint $\mathbf{\Psi}$, while the second set is based on the entropy variables $\mathbf{v}$. The aggregated temporal error indicator based on the output-based adjoint, $\delta\bar{J}^{\text{time}}_{\mathbf{\Psi}}$, is obtained by a summation of $\mathcal{E}^{n,\text{time}}_{e,\mathbf{\Psi}}$ over all of elements and time steps via Eqn. 20. This equation is dependent on the fine-space output-based adjoint $\mathbf{\Psi}_h$. The aggregated temporal error indicator based on the entropy variables, $\delta\bar{J}^{\text{time}}_{\mathbf{v}}$, is found in a similar fashion. The only difference is that the fine-space entropy variables $\mathbf{v}_h$ are used in place of the fine-space output-based adjoint. This leads to two separate aggregated temporal error indicators.

Next, the output error based on the the output-based adjoint, $\delta\bar{J}_{\mathbf{\Psi}}$, is computed by a summation of error contributions of each element for each time step, $\mathcal{E}^{n}_{e,\mathbf{\Psi}}$, using Eqn. 19. Again, the computation of $\mathcal{E}^{n}_{e,\mathbf{\Psi}}$ relies on the output-based adjoint. The output error based on the entropy variables can be computed using a similar approach, the only difference being the use of the fine-space entropy variables in lieu of the output-based adjoint in Eqn. 19. This in turn leads to the computation of the elemental error contribution at each time step based on fine-space entropy variables, $\mathcal{E}^{n}_{e,\mathbf{v}}$.

The aggregated spatial error indicator based on the output-based adjoint, $\delta\bar{J}^{\text{space}}_{\mathbf{\Psi}}$, can be found by taking the difference between $\delta\bar{J}_{\mathbf{\Psi}}$ and $\delta\bar{J}^{\text{time}}_{\mathbf{\Psi}}$. The spatial error based on entropy variables, $\delta\bar{J}^{\text{space}}_{\mathbf{v}}$, can be computed in a similar fashion. At this point, we can compute our multiplicative combined version of the non-conservative temporal and spatial error indicators via:

$$\begin{aligned}\delta\bar{J}^{\text{space}}_{\text{comb}} &= \delta\bar{J}^{\text{space}}_{\mathbf{\Psi}} \cdot \delta\bar{J}^{\text{space}}_{\mathbf{v}},\\ \delta\bar{J}^{\text{time}}_{\text{comb}} &= \delta\bar{J}^{\text{time}}_{\mathbf{\Psi}} \cdot \delta\bar{J}^{\text{time}}_{\mathbf{v}}.\end{aligned} \tag{25}$$

If we are interested in a conservative error estimate, two separate sets of conservative, aggregated spatial and temporal error estimates are computed via

$$\begin{aligned}\epsilon^{\text{time}}_{\mathbf{\Psi}} &\equiv \sum_{n=1}^{N_t}\left|\sum_{e=1}^{N_e}\mathcal{E}^{\text{n,time}}_{e,\mathbf{\Psi}}\right|, \quad \epsilon^{\text{space}}_{\mathbf{\Psi}} \equiv \sum_{e=1}^{N_e}\epsilon^{\text{space}}_{e,\mathbf{\Psi}}, \quad \epsilon^{\text{space}}_{e,\mathbf{\Psi}} = \left|\sum_{n=1}^{N_t}\mathcal{E}^{\text{n,space}}_{e,\mathbf{\Psi}}\right|\\ \epsilon^{\text{time}}_{\mathbf{v}} &\equiv \sum_{n=1}^{N_t}\left|\sum_{e=1}^{N_e}\mathcal{E}^{\text{n,time}}_{e,\mathbf{v}}\right|, \quad \epsilon^{\text{space}}_{\mathbf{v}} \equiv \sum_{e=1}^{N_e}\epsilon^{\text{space}}_{e,\mathbf{v}}, \quad \epsilon^{\text{space}}_{e,\mathbf{v}} = \left|\sum_{n=1}^{N_t}\mathcal{E}^{\text{n,space}}_{e,\mathbf{v}}\right|.\end{aligned} \tag{26}$$

Combined versions are again obtained by simply multiplying the two sets together:

$$\begin{aligned}\epsilon^{\text{time}}_{\text{comb}} &= \epsilon^{\text{time}}_{\mathbf{\Psi}} \cdot \epsilon^{\text{time}}_{\mathbf{v}},\\ \epsilon^{\text{space}}_{\text{comb}} &= \epsilon^{\text{space}}_{\mathbf{\Psi}} \cdot \epsilon^{\text{space}}_{\mathbf{v}}.\end{aligned} \tag{27}$$

In addition to combining the aggregated spatial and temporal (conservative and non-conservative) error, the elemental spatial error indicator vectors for both the output-based and entropy-based approach are combined via elemental

multiplication. This can be expressed mathematically as

$$\epsilon^{\text{space}}_{e,\text{comb}} = \epsilon^{\text{space}}_{e,\boldsymbol{\Psi}} \cdot \epsilon^{\text{space}}_{e,\mathbf{v}}. \tag{28}$$

This combination is done after the aggregated errors are combined. The combined elemental spatial error indicator, $\epsilon^{\text{space}}_{e,\text{comb}}$ is not used in obtaining the aggregated total error contributions. It is only used to determine the error sampling of each element within MOESS.

B. Combination of Error Indicators using a Coarser Output-Based Adjoint

In Section IV, Eqn. 19 requires the computation of the fine-space output-based adjoint, of spatial order $p+1$, at every time step. Given that the solution of the fine-space output-based adjoint, $\boldsymbol{\Psi}_h$, is quite expensive, unsteady simulations only magnify this problem. To minimize the cost of this step, an additional combined approach was implemented in which no fine-space output-based adjoint is computed. Instead, an output-based adjoint at the current approximation space, $\boldsymbol{\Psi}_H$, is computed at each time step and used to calculate both the spatial and temporal errors. Computing an output-based adjoint at a lower spatial approximation space is much less computationally expensive, but it leads to much more inaccurate error estimations.

To mitigate the inaccuracies brought on by not using the fine-space output-based adjoint, we combine the spatial and temporal error indicators with those obtained using fine-space entropy variables, $\mathbf{v}_h$, since fine-space entropy variables are far less computationally expensive to obtain. By combining the less accurate error estimates obtained using the coarser output-based adjoint with those obtained using fine-space entropy variables, we should get results that compare favorably to the standard combined approach, but are far less expensive to compute.

Using a coarser output-based adjoint does not change the process outlined in Section IV significantly. The main change is that all of the equations that use the fine-space output-based adjoint, $\boldsymbol{\Psi}_h$, are now solved for at the current approximation space and use the output-based adjoint at the current-space, $\boldsymbol{\Psi}_H$. The output error based on the coarser output-based adjoint can now be written as

$$\delta \bar{J} \approx -\int_0^T \boldsymbol{\Psi}_H^{\mathrm{T}} \bar{\mathbf{R}}_H(\mathbf{U}_H^{\tilde{H}})dt = \sum_{n=1}^{N_t}\sum_{e=1}^{N_e} \underbrace{\int_{t^{n-1}}^{t^n} -\boldsymbol{\Psi}_{H,e}^{\mathrm{T}} \bar{\mathbf{R}}_{H,e}\left(\mathbf{U}_H^{\tilde{H}}\right) dt}_{\mathcal{E}^n_{e,\boldsymbol{\Psi}}}, \tag{29}$$

where $\mathbf{U}_H^{\tilde{H}}$ is the projection of the primal from the space H at order p, to space $\tilde{H}$ at order $p-1$, and then back up to space H at order p. This step is necessary since the unsteady residual $\bar{\mathbf{R}}_H$ of the current-space solution $\mathbf{U}_H$ is equal to zero. By projecting the state down to order $p-1$ and then back up, we obtain an approximation for the state error that allows us to solve Eqn. 29.Since the output error using entropy variables is still evaluated at the fine-space, it is necessary to make another copy of the primal state that is injected into the finer-space h and evaluate a subsequent residual. We can write this mathematically as

$$\delta \bar{J}_{\mathbf{v}} \approx -\int_0^T \mathbf{v}_h^{\mathrm{T}} \bar{\mathbf{R}}_h(\mathbf{U}_h^H)dt = \sum_{n=1}^{N_t}\sum_{e=1}^{N_e} \underbrace{\int_{t^{n-1}}^{t^n} -\mathbf{v}_{h,e}^{\mathrm{T}} \bar{\mathbf{R}}_{h,e}\left(\mathbf{U}_h^H\right) dt}_{\mathcal{E}^n_{e,\mathbf{v}}}. \tag{30}$$

Unlike the spatial error contribution which changes due to the use of the coarser output-based adjoint, the temporal error contribution, Eqn. 20, is obtained the same way. The only difference is that no spatial projection of the fine-space output-based adjoint is necessary since the adjoint is already at the current-space order p. The combination of the errors outlined in the previous subsection does not change since they are not dependent on what order the output-based adjoint or entropy variables were when the errors were accumulated.

C. Combination of Error Indicators using a Coarse Output-Based Adjoint and Mask

Building off of the previous approach, the final approach considered in this paper incorporates a mask on the elemental spatial error indicator vector based on the entropy variables, $\epsilon^{\text{space}}_{e,\mathbf{v}}$. This mask is applied to the indicator prior to obtaining the aggregated errors obtained in Eqn. 26. The rationale is so the aggregated spatial errors can be properly computed with the effect of the mask. This allows a link between the allocation of spatial and temporal degrees of freedom to the effect of the mask.

The purpose of the mask is to limit the weight of the entropy indicator within MOESS so that the output-based indicator carries more weight in the spatial optimization. This is accomplished by creating an element-based refinement indicator, i.e. a mask, based on the spatial adaptation indicator obtained using the coarser output-based adjoint. This mask, μ_e, is a vector that is exactly the same size as the output-based, elemental, spatial error indicator $\epsilon^{\text{space}}_{e,\mathbf{\Psi}}$. The mask is created by examining the relative magnitude of the indicator for each element. Note that this requires taking an absolute value of the indicator. Using a user-specified percentage, a subset of the output-based indicator is created. For example, if the user wishes to apply a 25% mask on the entropy-based indicator, μ_e is created from the output indicator: it consists of 1 for the 25% elements with the largest $\epsilon^{\text{space}}_{e,\mathbf{\Psi}}$, and 0 for all other elements.

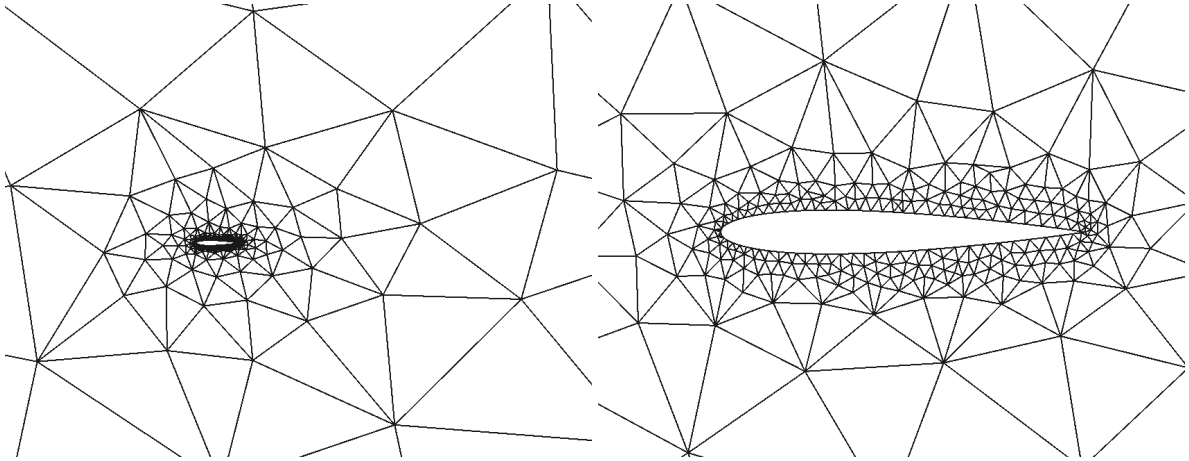
The mask is then multiplied by the entropy-based spatial error indicator in an element-wise fashion to obtain the final indicator that ultimately governs MOESS,

$$\epsilon^{\text{space}}_{e,\text{comb}} = \mu_e \cdot \epsilon^{\text{space}}_{e,\mathbf{v}}. \tag{31}$$

The reason for using a masking approach is the mask should completely eliminate elements targeted by the entropy-based indicator which have little influence on the desired engineering output, regardless of the magnitude of the entropy-based indicator for these elements.

VI. Results

A case that consists of an encounter between a NACA 0012 airfoil and a vertical gust in used in this work to test the combined output-based and entropy-based adjoint approach for unsteady simulations. The initial mesh, shown in Figure 1, is made up of 664 triangular elements with a farfield situated 100 chord lengths away from the airfoil. Curved elements are used to represent the airfoil geometry.

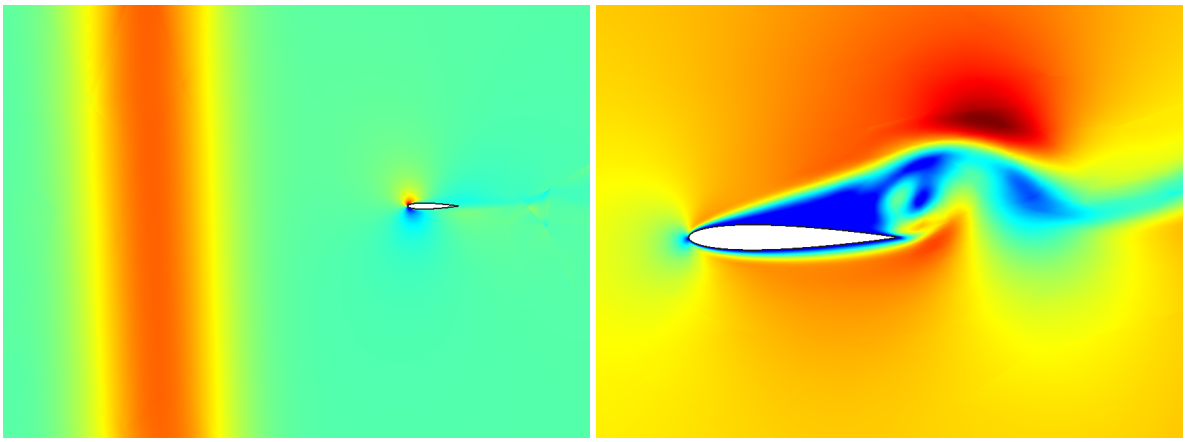


(a) Initial Mesh (b) Initial Mesh (Zoom)

Fig. 1 Gust simulation: initial mesh

The initial condition is a converged steady-state solution at $Re = 1000$, $M_\infty = 0.4$, $\alpha = 2°$, with adiabatic boundary conditions. The Prandtl number is set to $Pr = 0.72$ and viscosity is held constant. After the steady-state solution is obtained on the initial mesh, a vertical gust is superimposed on the solution by perturbing the vertical velocity ahead of the airfoil with a Gaussian function of amplitude $0.3U_\infty$, standard deviation of one chord length, centered five chords upstream of the leading edge, and offset to reach zero perturbation one chord length upstream of the leading edge. The location of the gust at $t = 0$ is shown in Figure 2(a). Convenient units in which the freestream density and velocity are set to unity are used for this simulation. The airfoil chord length is also set to $c = 1$. The simulation runs from $t = 0$ to $t = 15$, where one time unit is simply the unit chord divided by the freestream velocity. The initial number of time steps is $N_t = 15$. DIRK3 is used to advance the primal solution in time, while DIRK4 is used to march the adjoint backward in time. Mach number contours showing the gust inducing flow separation and vortex shedding are shown in Figure 2.

As previously mentioned in Section IV.B, it is desirable, for comparison purposes, that the degrees of freedom are kept relatively consistent among the various adaptation approach used. Thus, fixed degrees of freedom targets were



(a) Y-Momentum at $t = 0$ (b) Mach Number at $t = 7.5$ (0 to 0.6)

Fig. 2 Gust simulation: primal solution

used to drive the mesh optimization in lieu of a fixed growth rate. For this case five different targets were considered, starting with 240,000, and increasing by a factor of 2 until 3,840,000 degrees of freedom. At each degrees of freedom target, 15 adaptive iterations were considered in order for the cases to settle on the specified target. Averages of the desired output were taken over the last seven iterations, with the maximum and minimum values discarded to reduce noise. The computational time in order to run the 15 adaptive iterations was also recorded for each case to compare relative computation times. All cases were run in parallel with four cores on the high-performance computing cluster, Flux, at the University of Michigan. The core hardware is 2.5 to 2.8 GHz Intel Xeon processors all connected with Infiniband networking. Each core was assigned 4GB of RAM per job. The computational times shown in this paper are not exact as portions of the solver have not been fully optimized. The computational times are included in this work to show the relative performance of the various approaches. They are not meant to be taken as definitive.

The output of interest in this simulation is the weighted time-integral of the lift force,

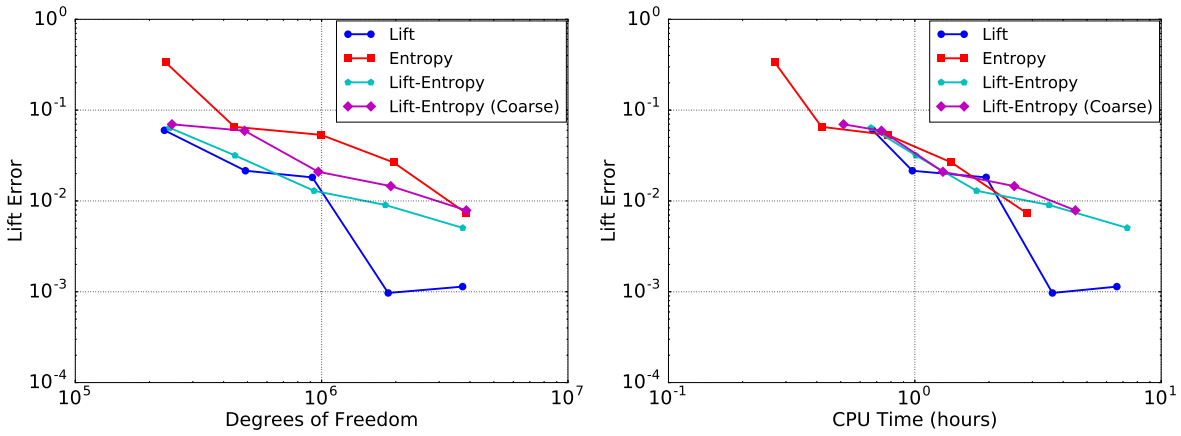
$$\bar{J} = \int_0^T w(t)L(t)dt, \tag{32}$$

where the temporal weighting function, $w(t) = \exp(-0.5(t-7.5)^2)$, is used to smooth the instantaneous lift force on the airfoil, $L(t)$, over time. The weighted time-integral of the lift force is the output of choice that guides the various adaptation strategies. The truth solution for this output is obtained by using an output-based refined mesh made up of 4,299 elements and running a simulation at $p = 3$ with 500 times steps of DIRK4.

A. Non-Conservative Results

Figure 3(a) presents the convergence of the output error versus total degrees of freedom for various strategies using non-conservative error estimates. As previously mentioned, the outputs were averaged over the last seven iterations, with the maximum and minimum output values over that range excluded from the averaging. The corresponding degrees of freedom were averaged as well. The four strategies considered in this convergence study are output-based, entropy-based, the standard combined approach, and the combined approach using the coarser output-based adjoint. At the same degrees of freedom target, the output-based approach significantly outperforms the others. The lift coefficient error at the last degrees of freedom target is almost an entire magnitude lower. The entropy variables approach yields the highest magnitude, hence the rationale for exploring the combined approach. The combined approach that uses the fine-space output-based adjoint performs slightly better than the combined approach with the coarser-space output-based adjoint as expected.

However, the benefit of the coarser combined approach is evident in Figure 3(b), where the same errors are plotted against the CPU time required to run the 15 iterations at each degrees of freedom target. The coarser combined approach is significantly cheaper than the fine-space combined approach. The reason is clearly due to how much cheaper it is



(a) Convergence versus total degrees of freedom

(b) Convergence versus CPU time

Fig. 3 Output error convergence using non-conservative error estimates

to solve for entropy-variables on the fine-space. From these results, we observe that it takes over double the amount of time to run the output-based adjoint approach compared to the entropy variables approach. Despite the fact the coarser combined approach is cheaper, it does not gain much lower error magnitudes relative to the fine-space combined approach. The issue is that the entropy variables do not provide nearly as accurate non-conservative error estimates.

One of the reasons the entropy-variables approach does not yield comparable output error results to the output-based approach is the emphasis on targeting spatial degrees of freedom over temporal. Figure 4 compares the number of spatial versus temporal degrees of freedom for the five targeted totals. For the highest degrees of freedom targets, the output-based approach has a much greater emphasis on temporal degrees of freedom compared to the other approaches. While the entropy-based approach initially targets more temporal refinement, by the final two adaptive targets it levels off and refines in space only. Based on comparing the relative performance in estimating the output, it is evident the entropy-variables approach is not yielding an accurate temporal error estimate. This negatively impacts the combined approaches, especially the coarse combined approach, where the preference of spatial over temporal refinement is even more drastic.

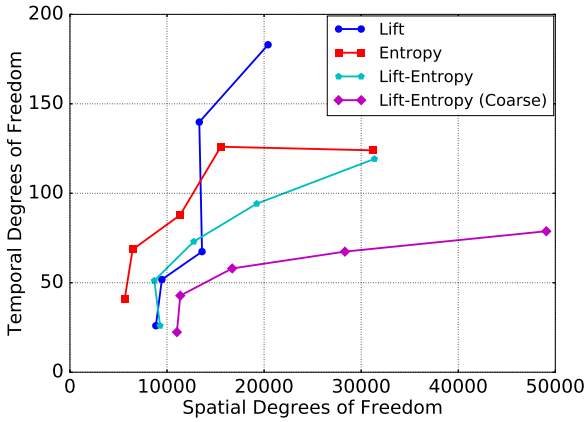


Fig. 4 Degrees of freedom using non-conservative error estimates

The final adapted mesh for the output-based, entropy-based, and coarse combined approaches are shown in Figure 5. From the previous figure, it was known that the output-based approach should have far fewer elements in the mesh. This conclusion can clearly be observed in the zoomed out view of the output-based mesh. There are fewer elements away from the airfoil compared to the other two approaches. The reason is that entropy is generated due to the gust in regions

not near the airfoil. This is the common issue with the entropy-based approach for adaptation we observed in previous findings [15] for steady flow problems. The combined approach slightly mitigates this issue as the mesh near the airfoil is more resolved than the mesh generated from the entropy-based approach. However, the combined approach still has far too much refinement above and below the airfoil that extends unnecessarily to the outer boundaries of the mesh. In addition, the significant lack of temporal degrees of freedom for this approach cancels out the benefit of better spatial refinement near the mesh. These results show that using the output-based adjoint garners far superior conservative error estimates compared to using entropy variables and subsequently the combined approach.

B. Conservative Results

In this section, we will analyze the same case, with similar approaches, the only difference being the use of conservative error estimates, Eqn. 23, to govern the allocation of temporal and spatial degrees of freedom. The reason for using conservative error estimates is to eliminate the excess noise caused by the cancellation of errors.

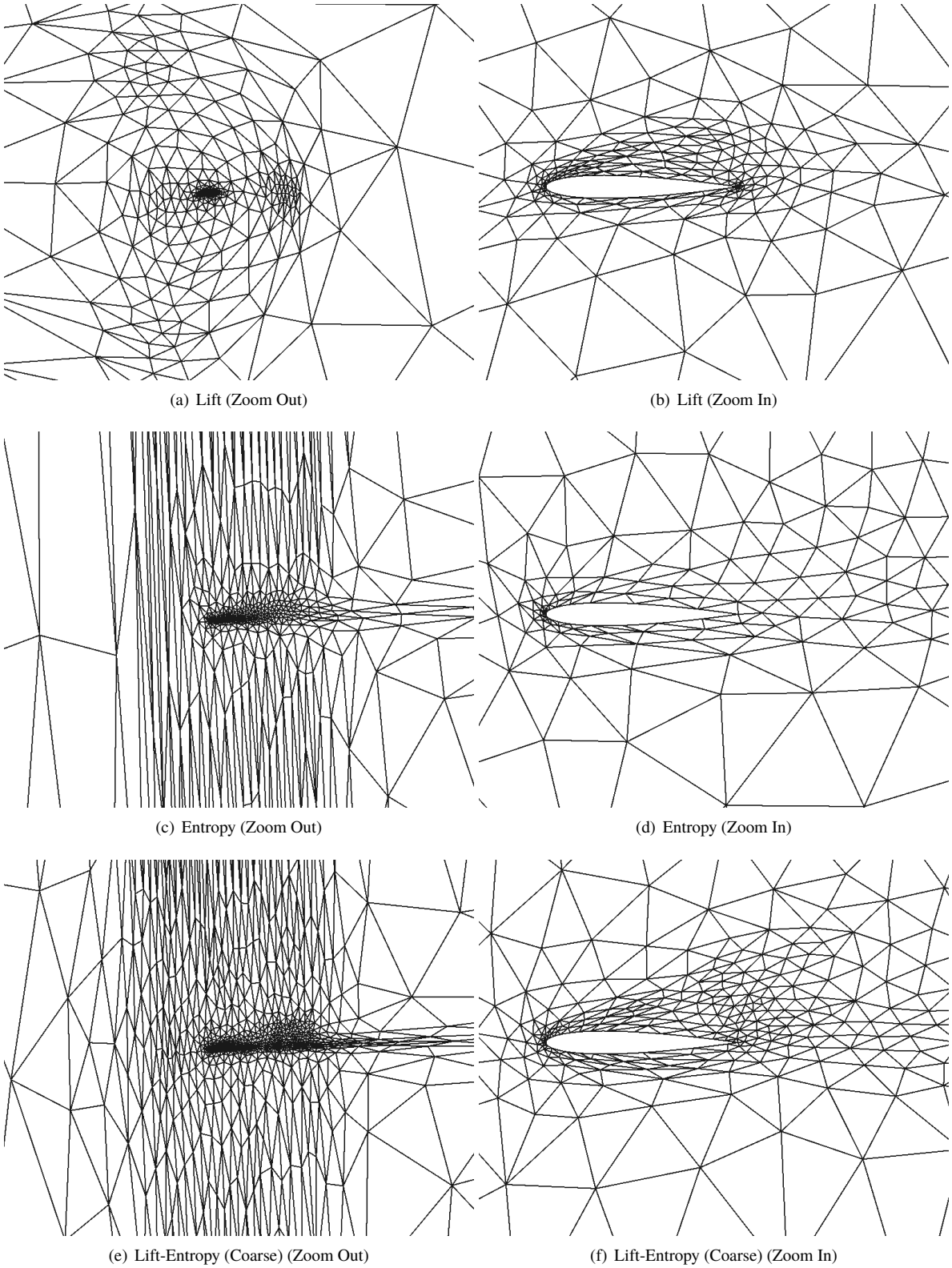
Figure 6(a) shows the integrated lift error convergence versus degrees of freedom at the same five degrees of freedom targets as before. The first major observation is how poor the output-based approach does relative to before when non-conservative error estimates were used. The entropy-based approach does slightly better compared to before, but still worse than the output-based approach. Since the output-based approach output error magnitudes are higher than before, the combined approach suffers as well. Its relative performance to the combined approach using non-conservative error estimates is far less severe compared to the output-based approach. This is because the entropy-based approach remains relatively unchanged. The most interesting finding from this result though is the fact that not only does the coarser combined approach improve with conservative error estimates, it actually yields comparatively better performance than all of the other approaches, including the output-based approach. The effect of the improved performance in the coarse combined approach is even more apparent Figure 6(a) since the coarse combined approach is so much less computationally expensive to run.

The reason the coarser combined approach outperforms the output-based approach is clearly evident in Figure 7. Unlike for the non-conservative error estimates, where the output-based approach preferred allocating more degrees of freedom to the temporal domain compared to the entropy-based approach, the opposite is true here. The average number of temporal degrees of freedom at the last degrees of freedom target for output-based case decreased by almost half, while the spatial degrees of freedom subsequently doubled. This leads to the combined approach that uses the fine-space output-based adjoint also allocating a substantial decrease in the total amount of temporal degrees of freedom. For the entropy-based approach, the number of temporal degrees continues to increase with each target, as opposed to before where it leveled off. This leads to the coarser combined approach yielding far more temporal degrees of freedom than before. The reason the coarser combined approach sees an increase in temporal degrees of freedom while the fine-space combined approach does not is that the effect of the errors due to the output-based adjoint is far less significant. It is quite apparent that the output-based adjoint approach is far more inaccurate using conservative variables, while the entropy variables approach is more accurate. This negatively impacts the fine-space combined approach, but positively impact the coarser combined approach since the more accurate errors due to the entropy variables are more dominant.

The final adapted mesh for the output-based approach in Figure 8 shows that the extra spatial degrees of freedom that were allocated to the temporal domain when using non-conservative variables are now used to refine the mesh downstream of the airfoil. Based on the poorer output error convergence, it is clear that the extra refinement there does not affect the integrated lift coefficient as much as extra temporal degrees of freedom do. The entropy-based approach that yields more temporal degrees of freedom, which in theory should lead to more accuracy, produces a much coarser mesh near the airfoil. Ideally, the mesh would have been coarsened far away from the airfoil. This explains why the accuracy did not improve despite the necessary increase in temporal degrees of freedom. The coarse combined approach though produced a relatively similar mesh near the airfoil compared to what was shown in Figure 5. With the increase in temporal degrees of freedom and a similar mesh near the airfoil, this leads to a much more accurate output prediction.

C. Effect of the Mask on the Combined Approach

In the previous section, we saw the potential of using the coarser combined approach with conservative error estimates. In this section that method will be further analyzed with the incorporation of the mask outlined in the previous section. The purpose of the mask, generated from the elemental output-based spatial error indicator, is to improve the same indicator obtained using entropy variables by eliminating areas of the mesh far away from the airfoil that do not affect the desired output. For this study, we will use the following masking percentages: 25%, 40%, 75%, and 90%. All results shown in this section use conservative error estimates.



(a) Lift (Zoom Out)

(b) Lift (Zoom In)

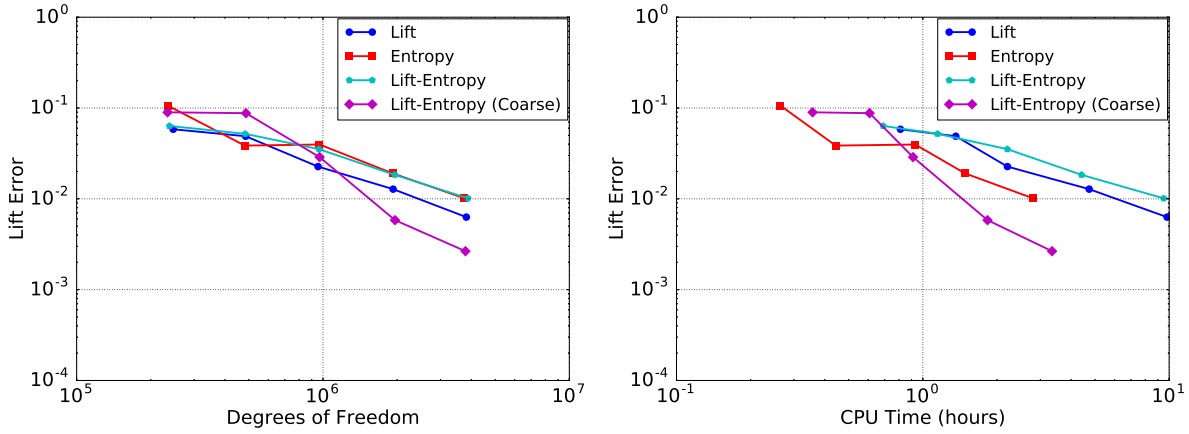
(c) Entropy (Zoom Out)

(d) Entropy (Zoom In)

(e) Lift-Entropy (Coarse) (Zoom Out)

(f) Lift-Entropy (Coarse) (Zoom In)

Fig. 5 Final adapted meshes at using non-conservative error estimates



(a) Convergence versus total degrees of freedom

(b) Convergence versus CPU time

Fig. 6 Output error convergence using conservative error estimates

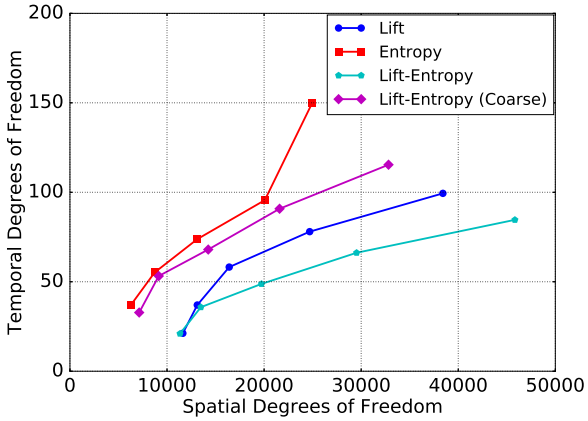
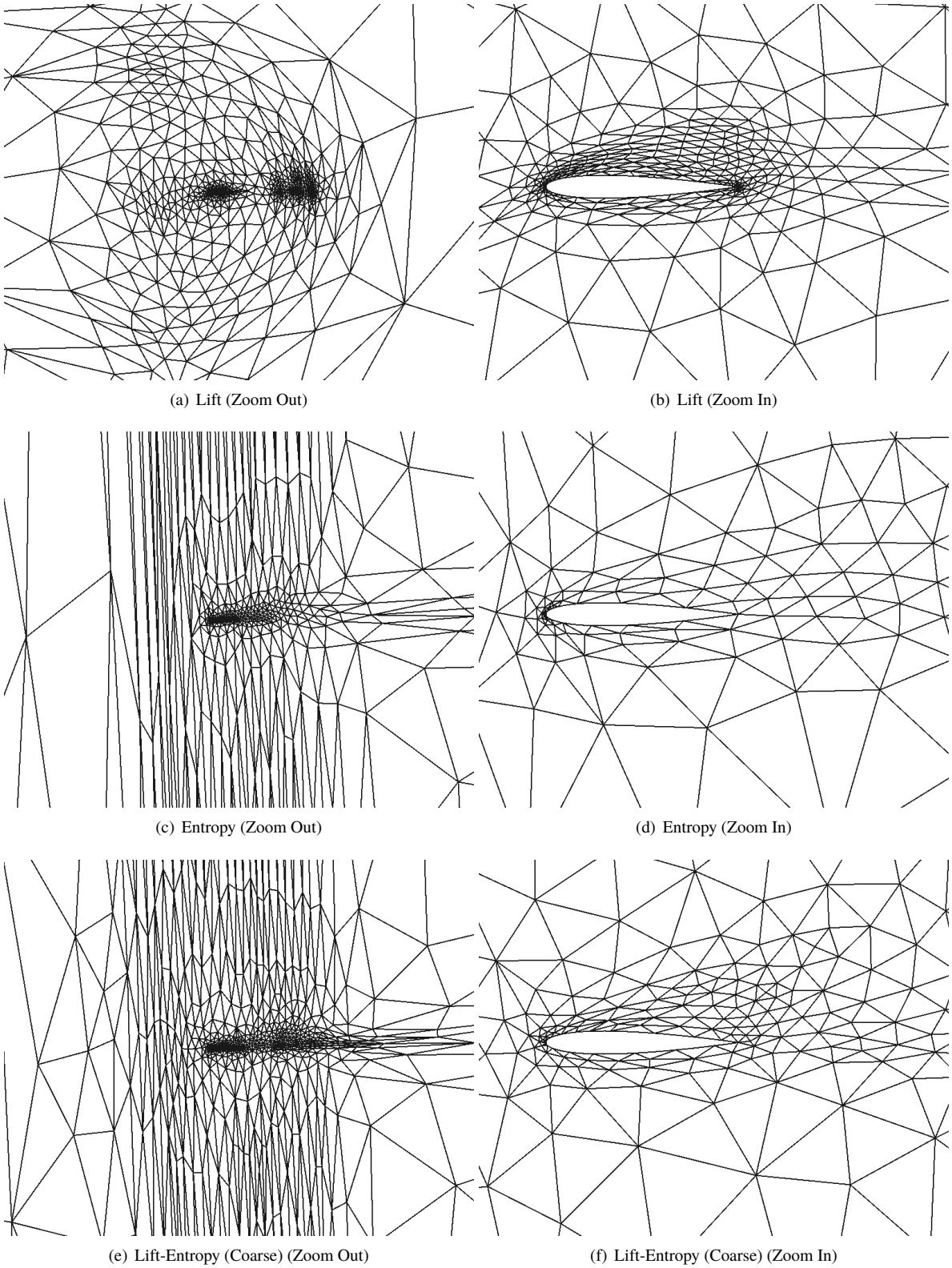


Fig. 7 Degrees of freedom using conservative error estimates



(a) Lift (Zoom Out)

(b) Lift (Zoom In)

(c) Entropy (Zoom Out)

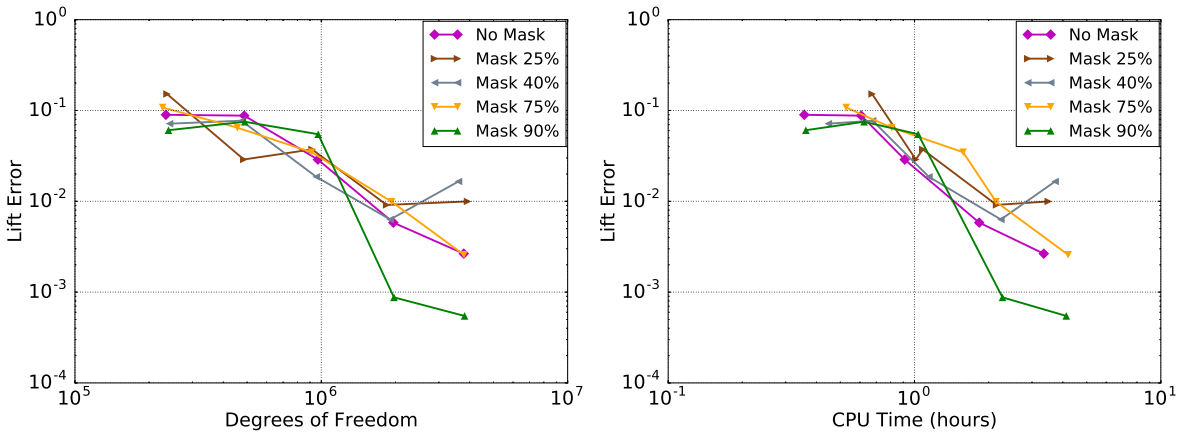
(d) Entropy (Zoom In)

(e) Lift-Entropy (Coarse) (Zoom Out)

(f) Lift-Entropy (Coarse) (Zoom In)

Fig. 8 Final adapted meshes using conservative error estimates

Figure 9 shows the output error convergence for the coarser combined approach, as well as the same approach with the four different masking values. For the finer degrees of freedom targets, the 90% masking case generates a significantly more accurate output than any of the other cases. The 75% masking case is almost directly comparable to not using a mask, while the 25% and 40% masks yield worse error estimates. Based on these results it appears that the lower the masking percentage, i.e. the more values in the output-based spatial error indicator that are set to zero, the less accurate the output is. Figure 9(b) shows an increase in CPU time due to the mask, which is to be expected since the masking algorithm is not free given the large size of the error indicators. However, the penalty is not sufficient enough to render this approach impractical. The benefit of using the 90% mask is still significant.



(a) Convergence versus total degrees of freedom

(b) Convergence versus CPU time

Fig. 9 Effect of masking the coarser combined approach on output error convergence

Figure 10 illustrates that the higher masking percentages yield more temporal degrees of freedom compared to not using a mask. This might explain why there was a benefit in using the 90% mask. As the masking percentages decrease, there generally is an increase in the allocation of spatial degrees of freedom, to the point where the 25% mask allocates more spatial degrees of freedom than without.

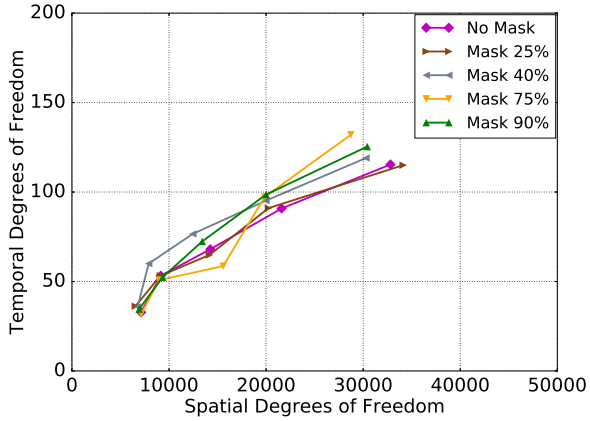
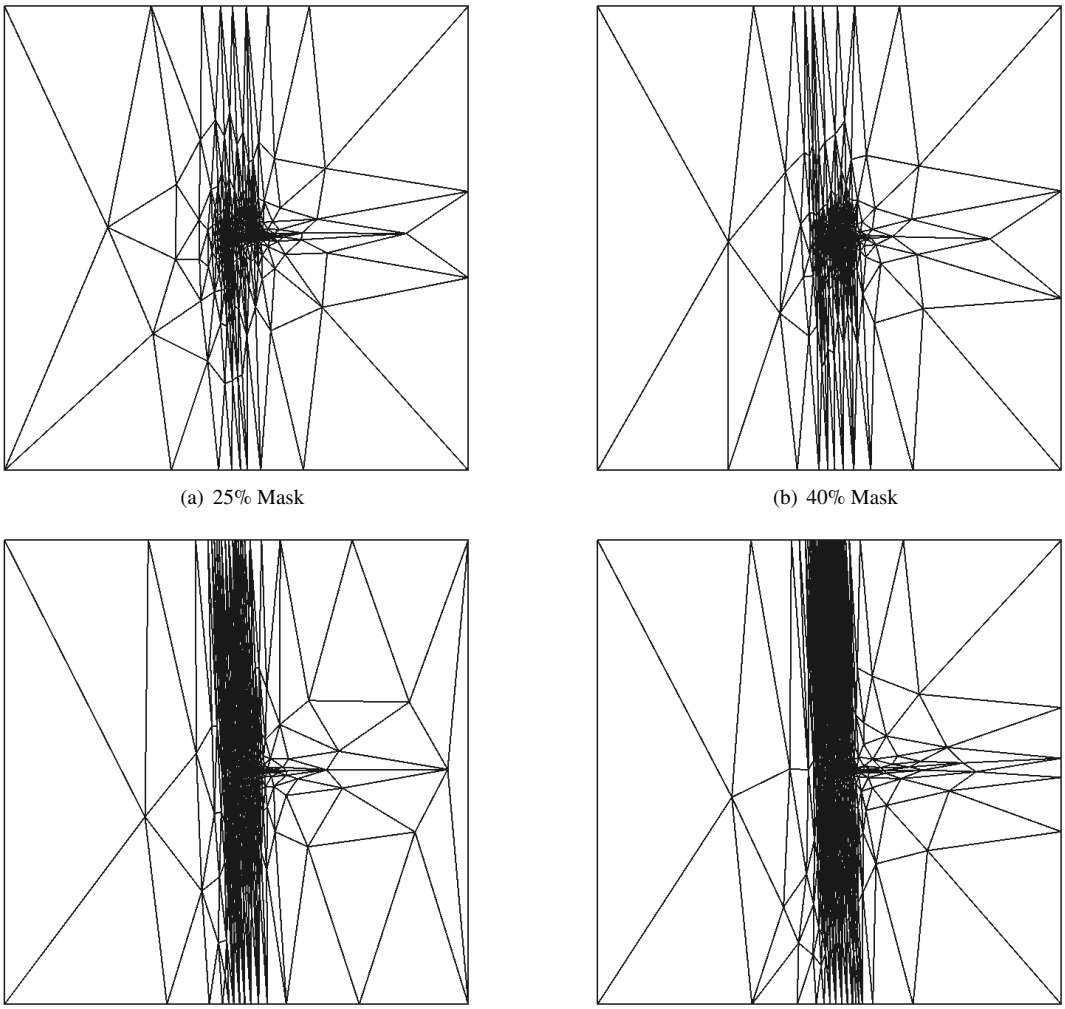


Fig. 10 Degrees of freedom using conservative error estimates with masks on coarser combined approach

In Section V we stated that the mask was applied prior to obtaining the aggregated spatial error estimates. Otherwise, there would be a disconnect between the spatial errors used to govern the allocation of degrees of freedom between the spatial and temporal domain, and the errors used to govern the relative mesh optimization with MOESS. The rationale for the mask is logical, as it effectively removes the influence of the spatial entropy-based error indicator in regions of

the domain that do not have as strong of an influence on the output error estimate. This is apparent in Figure 11 where the lower the masking percentage, the less refinement far away from the airfoil. This is because this refinement is only due to the entropy-based indicator. Unfortunately, these images might lead to the incorrect conclusion that the lower masking percentages should lead to more accurate output error estimates. Based on the output error convergence results, the exact opposite is true.

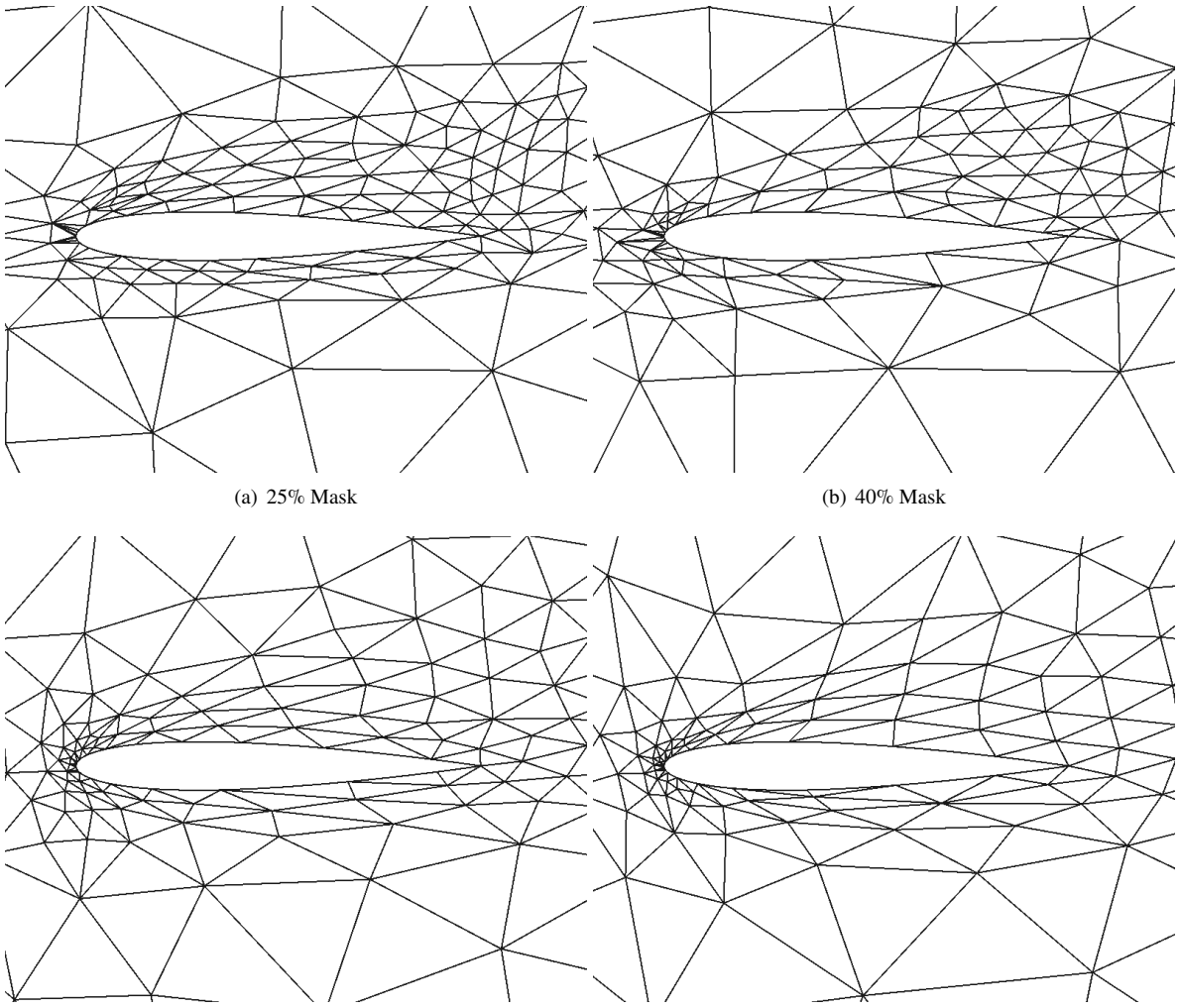


(a) 25% Mask (b) 40% Mask

(c) 75% Mask (d) 90% Mask

Fig. 11 Full view of adapted meshes using the masked coarser combined approach

One possible reason the lower percentage mask values lead to less accurate outputs is that the lower masking cases are not allocating enough degrees of freedom in the temporal domain. Perhaps using low masking percentages, which leads to a much strong effect in the spatial domain, leads to issues since no comparable mask is applied in the temporal domain. Another reason might be the lack of proper refinement near the nose of the airfoil for the lower masking percentage meshes, as evident in Figure 12. For the cases that use the 25% and 40% mask, the mesh ahead of the airfoil is quite coarse, with very anisotropic cells. This issue appears to get magnified as the masking percentage goes down. This might contribute to the fact that despite the mesh being less refined far away from the airfoil for these cases, the mesh near the airfoil does not improve. Overall, these results show that there can be a potential benefit to applying a mask on the entropy-based indicator. However, more work needs to be done regarding when the mask should be applied in the overall process, and should an additional mask in the temporal domain be considered as well.



(a) 25% Mask (b) 40% Mask

(c) 75% Mask (d) 90% Mask

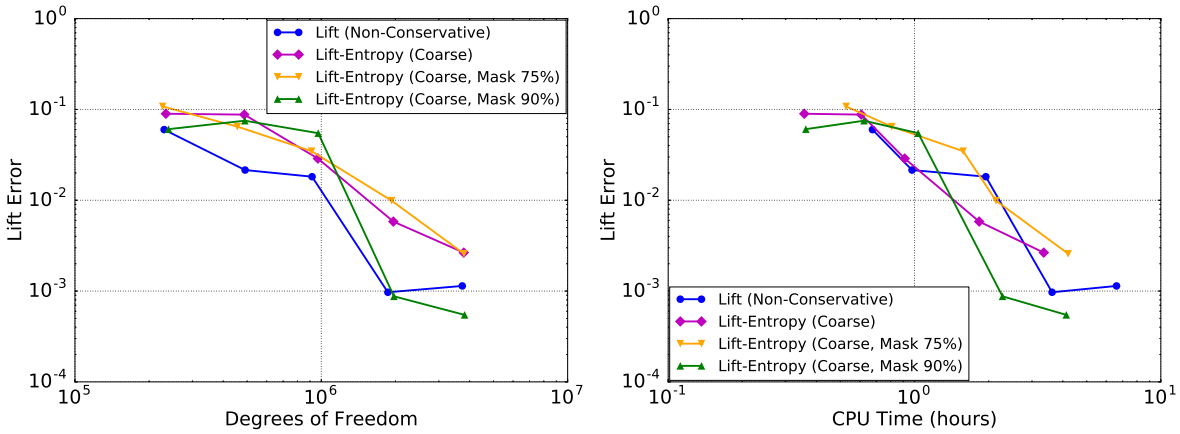
Fig. 12 Zoomed in view of final adapted meshes using the masked coarser combined approach

VII. Conclusions and Future Work

In this work, we demonstrated the potential of the combined output-based and entropy-based indicator approach for mesh refinement and error estimation of unsteady simulations. Indicators obtained from output-based adjoints are ideal since they specifically target spatial and temporal degrees of freedom that lead to the most accurate output error estimates. However, their expense, especially in unsteady simulations, hinders their effectiveness. The entropy-adjoint approach to error estimation is far cheaper, but not nearly as effective due to the tendency to target all regions of the spatial and temporal domains that exhibit high amounts of spurious entropy generation. The purpose of the combined approach is to mitigate the disadvantages of the two individual types of indicators. In this report, we observed that the combined approach often yields better integrated output error convergence than the entropy-based approach. In some cases, variations of the combined approach even produced better error convergence than the output-based approach.

Using non-conservative error estimates, it was clear that the output-based approach yielded superior output error estimates compared to both the entropy-based and combined approaches. Non-conservative error estimates obtained using entropy variables focus the allocation of degrees of freedom excessively in the spatial domain and not nearly enough in the temporal domain. This negatively impacts the coarser combined approach since the error estimates from the entropy variables dominate. However, the exact opposite is true when conservative variables are used. In this case, the output-based approach excessively refines the mesh while sacrificing too much accuracy in the temporal domain. Conversely, the entropy-based approach produces a much better balance between the allocation of spatial and temporal degrees of freedom. This leads to the coarser combined approach yielding a much more accurate output since the entropy-based error estimates are far more accurate. The addition of the mask to the coarser combined approach shows potential for some of the cases. If the masking percentage is high, which means only a small portion of the entropy-based spatial error indicator is affected, the error estimates show some substantial improvement. However, the mask can lead to comparatively worse performance if too much of the entropy-based spatial error indicator is affected.

Figure 13 compares the output error convergence for the output-based approach using non-conservative error estimates to several coarser output adjoint combined approaches using conservative error estimates. The outputs obtained from these selected combined approaches compare favorably to the output from the fine-space adjoint output-based approach when the overall CPU time is taken into regards. Since the combined approaches run faster than the output-based approach at the same degrees of freedom target, it can be argued that, especially for the combined approach with the 90% mask, the combined approaches produce more accurate results than the output-based approach. These results show the potential this new combined approach has due to its cheaper computational cost and sometimes more accurate error estimates.



(a) Convergence versus total degrees of freedom

(b) Convergence versus CPU time

Fig. 13 Output error convergence comparison

While this paper demonstrated the potential of the combined approach that uses coarser output-based adjoint, the results are limited to just one example. This approach needs to be thoroughly vetted on other unsteady Navier-Stokes simulations using both conservative and non-conservative error estimates to demonstrate whether the same conclusions from this case can be applied to others. Additionally, future work will focus on exploring and improving the masking of

the combined approach. This work highlighted the potential of the mask, but there are still some questions regarding what the ideal mask is and how it should be applied in the overall process. The combined approach using a mask has strong potential regarding the prediction of accurate output error estimates for unsteady simulation. However, this approach needs to be explored in many types of flow simulations so that it can be further understood and improved.

References

[1] Castro-Diaz, M. J., Hecht, F., Mohammadi, B., and Pironneau, O., "Anisotropic unstructured mesh adaptation for flow simulations," *International Journal for Numerical Methods in Fluids*, Vol. 25, 1997, pp. 475–491.

[2] Dompierre, J., Vallet, M.-G., Bourgault, Y., Fortin, M., and Habashi, W. G., "Anisotropic mesh adaptation: towards user-independent, mesh-independent and solver-independent CFD. Part III: Unstructured Meshes," *International Journal for Numerical Methods in Fluids*, Vol. 39, 2002, pp. 675–702.

[3] Hartmann, R., and Houston, P., "Adaptive discontinuous Galerkin finite element methods for the compressible Euler equations," *Journal of Computational Physics*, Vol. 183, No. 2, 2002, pp. 508–532.

[4] Venditti, D. A., and Darmofal, D. L., "Anisotropic grid adaptation for functional outputs: application to two-dimensional viscous flows," *Journal of Computational Physics*, Vol. 187, No. 1, 2003, pp. 22–46.

[5] Fidkowski, K. J., and Darmofal, D. L., "A triangular cut-cell adaptive method for high-order discretizations of the compressible Navier-Stokes equations," *Journal of Computational Physics*, Vol. 225, 2007, pp. 1653–1672. doi:10.1016/j.jcp.2007.02.007.

[6] Mavriplis, D. J., Vassberg, J. C., Tinoco, E. N., Mani, M., Brodersen, O. P., Eisfeld, B., Wahls, R. A., Morrison, J. H., Zickuhr, T., Levy, D., and Murayama, M., "Grid Quality and Resolution Issues from the Drag Prediction Workshop Series," AIAA Paper 2008-930, 2008.

[7] Nemec, M., Aftosmis, M. J., and Wintzer, M., "Adjoint-Based Adaptive Mesh Refinement for Complex Geometries," AIAA Paper 2008-0725, 2008.

[8] Pierce, N. A., and Giles, M. B., "Adjoint recovery of superconvergent functionals from PDE approximations," *SIAM Review*, Vol. 42, No. 2, 2000, pp. 247–264.

[9] Becker, R., and Rannacher, R., "An optimal control approach to a posteriori error estimation in finite element methods," *Acta Numerica*, edited by A. Iserles, Cambridge University Press, 2001, pp. 1–102.

[10] Fidkowski, K. J., and Darmofal, D. L., "Review of Output-Based Error Estimation and Mesh Adaptation in Computational Fluid Dynamics," *American Institute of Aeronautics and Astronautics Journal*, Vol. 49, No. 4, 2011, pp. 673–694. doi: 10.2514/1.J050073.

[11] Woopen, M., Balan, A., May, G., and Schütz, J., "A comparison of hybridized and standard DG methods for target-based hp-adaptive simulation of compressible flow," *Computers & Fluids*, Vol. 98, 2014, pp. 3–16.

[12] Fidkowski, K. J., and Darmofal, D. L., "Output-Based Error Estimation and Mesh Adaptation: Overview and Recent Results," AIAA Paper 2009-1303, 2009. doi:10.2514/6.2009-1303.

[13] Fidkowski, K. J., and Roe, P. L., "Entropy-based Mesh Refinement, I: The Entropy Adjoint Approach," AIAA Paper 2009-3790, 2009. doi:10.2514/6.2009-3790.

[14] Fidkowski, K. J., and Roe, P. L., "An Entropy Adjoint Approach to Mesh Refinement," *SIAM Journal on Scientific Computing*, Vol. 32, No. 3, 2010, pp. 1261–1287. doi:10.1137/090759057.

[15] Doetsch, K. T., and Fidkowski, K. J., "A Combined Entropy and Output-based Adjoint Approach for Mesh Refinement and Error Estimation," AIAA Paper 2018-0918, 2018.

[16] Giles, M., and Pierce, N., "Adjoint error correction for integral outputs," *Lecture Notes in Computational Science and Engineering: Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics*, Vol. 25, Springer, Berlin, 2002.

[17] Barth, T. J., "Space-Time Error Representation and Estimation in Navier-Stokes Calculations," *Complex Effects in Large Eddy Simulations*, edited by S. C. Kassinos, C. A. Langer, G. Iaccarino, and P. Moin, Springer Berlin Heidelberg, Lecture Notes in Computational Science and Engineering Vol 26, 2007, pp. 29–48.

[18] Besier, M., and Rannacher, R., "Goal-oriented space-time adaptivity in the finite element Galerkin method for the compution of nonstationary incompressible flow," *International Journal for Numerical Methods in Fluids*, Vol. 70, 2012, pp. 1139–1166.

[19] Schmich, M., and Vexler, B., "Adaptivity with Dynamic Meshes for Space-Time Finite Element Discretizations of Parabolic Equations," *SIAM Journal on Scientific Computing*, Vol. 30, No. 1, 2008, pp. 369–393.

[20] Mani, K., and Mavriplis, D. J., "Error Estimation and Adaptation for Functional Outputs in Time-Dependent Flow Problems," *Journal of Computational Physics*, Vol. 229, 2010, pp. 415–440.

[21] Fidkowski, K. J., and Luo, Y., "Output-based Space-Time Mesh Adaptation for the Compressible Navier-Stokes Equations," *Journal of Computational Physics*, Vol. 230, 2011, pp. 5753–5773. doi:10.1016/j.jcp.2011.03.059.

[22] Fidkowski, K. J., "An Output-Based Dynamic Order Refinement Strategy for Unsteady Aerodynamics," AIAA Paper 2012-77, 2012. doi:10.2514/6.2012-77.

[23] Kast, S. M., and Fidkowski, K. J., "Output-based Mesh Adaptation for High Order Navier-Stokes Simulations on Deformable Domains," *Journal of Computational Physics*, Vol. 252, No. 1, 2013, pp. 468–494. doi:10.1016/j.jcp.2013.06.007.

[24] 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, Massachusetts, 2005.

[25] Kast, S. M., Fidkowski, K. J., and Roe, P. L., "An Unsteady Entropy Adjoint Approach for Adaptive Solution of the Shallow-Water Equations," AIAA Paper 2011-3694, 2011. doi:10.2514/6.2011-3694.

[26] Barth, T. J., "Numerical Methods for Gasdynamic Systems on Unstructured Meshes," *An Introduction to Recent Developments in Theory and Numerics for Conservation Laws, Proceedings of the International School on Theory and Numerics for Conservation Laws, Berlin, Lecture Notes in Computational Science and Engineering*, edited by D. Kröner, M. Ohlberger, and C. Rhode, Springer-Verlag, 1999.

[27] Hughes, T. J. R., Franca, L. P., and Mallet, M., "A New Finite Element Formulation for Computational Fluid Dynamics: I. Symmetric Forms of the Compressible Euler and Navier-Stokes Equations and the Second Law of Thermodynamics." *Computer Methods in Applied Mechanics and Engineering*, Vol. 54, 1986, pp. 223–234.

[28] Roe, P. L., "Approximate Riemann solvers, parameter vectors, and difference schemes," *Journal of Computational Physics*, Vol. 43, 1981, pp. 357–372.

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

[30] Fidkowski, K. J., "Output-Based Space-Time Mesh Optimization for Unsteady Flows Using Continuous-in-Time Adjoints," *Journal of Computational Physics*, Vol. 341(15), 2017, pp. 258–277.

[31] Fidkowski, K. J., "Output error estimation strategies for discontinuous Galerkin discretizations of unsteady convection-dominated flows," *International Journal for Numerical Methods in Engineering*, Vol. 88, No. 12, 2011, pp. 1297–1322. doi:10.1002/nme.3224.

[32] Yano, M., "An Optimization Framework for Adaptive Higher-Order Discretizations of Partial Differential Equations on Anisotropic Simplex Meshes," Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, 2012.

[33] Fidkowski, K. J., "A Local Sampling Approach to Anisotropic Metric-Based Mesh Optimization," AIAA Paper SciTech, 2016.