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

Kevin T. Doetsch[*] and Krzysztof J. Fidkowski[†]

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

This paper presents a strategy for mesh refinement driven by an indicator that combines two previously-investigated indicators: one based on a user-specified engineering output, and the other based on entropy variables. Using the entropy-variable indicator to adapt a mesh is computationally advantageous since it does not require the solution of an auxiliary adjoint equation. However, the entropy-variable indicator targets any region of the domain where spurious entropy is generated, regardless of whether or not this region affects an engineering output of interest. On the other hand, an indicator computed from an engineering output generally targets only those regions important for the chosen output, though it is more computationally taxing because of the required adjoint solution. Approximations in the adjoint calculation reduce this cost, at the expense of indicator accuracy. In combining these indicators, our objective is to maintain the low cost of approximate adjoint solutions while achieving improved indicator accuracy from the entropy adjoint. We demonstrate the potential for this method through several simulations governed by the compressible Navier-Stokes equations.

I. Introduction

A popular approach in Computational Fluid Dynamics (CFD) to obtain accurate solutions for problems that exhibit a wide range of spatial length scales, whose distributions are generally not a known ahead of time, is the practice of solution-based adaptive methods.[1–7] These methods use an indicator computed a posteriori from the solution to drive the adaptation of the computational mesh. In the literature, extensive work has been done studying various indicators with regard to their accuracy, robustness, and expense. Two previously-studied indicators relevant to our present work are indicators based on a user-specified engineering scalar output[4,8,9] and indicators based on entropy variables.[10,11] Both indicators have their strengths and weaknesses depending on the nature of a given problem. In this work, a new indicator is investigated that combines both the output and entropy-adjoint indicators. The main approach investigated in this paper is a combination that consists of a simple product of the two indicators, though various modifications to this approach are also considered.

Output-based adaptive methods are advantageous since they specifically target areas of the mesh that are critical to the prediction of the output of interest.[3,4,12,13] They are effective in accounting for propagation effects intrinsic to hyperbolic problems,[14] through the use of output-specific adjoint solutions, which function as the sensitivity of the output to local residuals. However, in certain cases, it is not always clear which scalar output should be chosen for adaptation, prior to obtaining solutions. Running cases with several output choices is an option, but that is often

computationally expensive. In fact, often just running one output-based adjoint solution is not cheap due to the need to solve the adjoint problem on a refined-space computational domain.

The entropy-adjoint approach alleviates both of the aforementioned issues with output-based adaptive methods. Unlike the engineering-output approach, entropy-variable refinement does not require the choice of a single output, which relieves the user of making this choice when an all-around good solution is sought. Additionally, the entropy variables are far less expensive to obtain, as they are computed from a direct variable transformation of the conservative state. Unfortunately, because entropy-variable indicators target all regions of the domain where there is production of spurious entropy, this strategy is not as globally discriminating as the output-based approach. However, combining both approaches together yields a new approach that encompasses the strengths of both methods, while limiting the downsides each method bears.

The outline of this paper is as follows. Section II reviews the concepts of using output-based adjoints for mesh refinement and error estimation. Section III reviews how the the entropy variables are defined and how they can be interpreted as adjoints for both inviscid and viscous conservation equations. Section IV compares the two mesh adaptation strategies used in this work, as well as a sequential breakdown of the various steps for each adaptive iteration. Section V reviews various modifications to obtaining the adaptation indicators that were investigated in this research. Section VI discusses the numerical implementation behind these adaptive strategies. Finally, Section VII outlines the results generated using these methods.

II. Adaptation Using Output Adjoints

Given a particular engineering output, output-based error estimation methods refine areas of the mesh important for an accurate prediction of the output. They account for error propagation effects that are inherent to convection-dominated flow simulations by targeting residuals to which the output is most sensitive. The resulting error estimates provide confidence levels to the output calculations and can be localized to elements to drive adaptation. These error estimations rely on the solution of an adjoint problem, which yields, in continuous form, the adjoint $\boldsymbol{\psi}(\vec{x})$. The adjoint is a Green's function that relates residual source perturbations to the output of interest, $J(\mathbf{u})$, where $\mathbf{u}$ denotes the state vector.

Consider a partial differential equation $\mathbf{r}(\mathbf{u}) = \mathbf{0}$, where $\mathbf{r}(\cdot)$ is a differential operator. The goal is to determine $\mathbf{u} \in \mathcal{V}$ such that

$$\mathcal{R}(\mathbf{u}, \mathbf{w}) = 0, \qquad \forall \mathbf{w} \in \mathcal{V}, \tag{1}$$

where $\mathcal{V}$ is the trial and test function space and $\mathcal{R}(\cdot, \cdot) : \mathcal{V} \times \mathcal{V} \to \mathbb{R}$ is the semilinear operator that represents the weak form of the differential equation. The adjoint $\boldsymbol{\psi} \in \mathcal{V}$ is the sensitivity of J to an infinitesimal source term, $\delta \mathbf{r}$, added to the governing equation,

$$\delta J = (\delta \mathbf{r}, \boldsymbol{\psi}), \tag{2}$$

where $(\cdot, \cdot)$ is an inner product over the computational domain, Ω. The adjoint satisfies the following weak statement:[8] determine $\boldsymbol{\psi} \in \mathcal{V}$ such that

$$\mathcal{R}'[\mathbf{u}](\mathbf{w}, \boldsymbol{\psi}) + J'[\mathbf{u}](\mathbf{w}) = 0, \qquad \forall \mathbf{w} \in \mathcal{V}, \tag{3}$$

where the primes denote Fréchét linearization. Using the solution of the adjoint equation, the output error can be estimated by the adjoint-weighted residual method, which is based on the following two observations:

- In general the approximate solution $\mathbf{u}_H$ in a finite-dimensional approximation space $\mathcal{V}_H$ will not satisfy the governing PDE. However, it will satisfy a PDE will a small perturbation, whose weak form reads: find $\mathbf{u}' \in \mathcal{V}$ such that:

$$\mathcal{R}(\mathbf{u}', \mathbf{w}) + (\delta\mathbf{r}, \mathbf{w}) = 0, \quad \forall \mathbf{w} \in \mathcal{V} \qquad \text{where } (\delta\mathbf{r}, \mathbf{w}) = -\mathcal{R}(\mathbf{u}_H, \mathbf{w}).$$

- The adjoint translates the residual perturbation to an output perturbation via Eqn. 2:

$$\delta J = (\delta\mathbf{r}, \boldsymbol{\psi}) = -\mathcal{R}(\mathbf{u}_H, \boldsymbol{\psi}). \tag{4}$$

Using a weighted residual approximation of the solution, the above expression quantifies the output's numerical error. For non-infinitesimal perturbations and nonlinear outputs or PDEs, the numerical error calculation is not exact.

Eqn. 4 is only computable if the continuous adjoint is approximated. This is achieved by solving the adjoint equation on a finer finite-dimensional space, $\mathcal{V}_h \supset \mathcal{V}_H$, either directly or iteratively.[15–17] Depending on the scope of the problem, this step can be computationally expensive. Solving the adjoint equation yields a fine-space adjoint $\boldsymbol{\psi}_h$ that can be used to obtain an adaptive indicator whose purpose is to relate the relative contribution of each element to the total output error. Eqn. 4 can be approximated as

$$\delta J \approx -\sum_{\kappa_H \in T_H} \mathcal{R}_h\left(\mathbf{u}_H, \boldsymbol{\psi}_h|_{\kappa_H}\right), \tag{5}$$

where $|_{\kappa_H}$ denotes the restriction of an interpolated function to element κ_H of the triangulation T_H. For the sake of simplicity, $\mathcal{V}_h$ is obtained from $\mathcal{V}_H$ by increasing the approximation order, while keeping the triangulation between the fine space and the coarse space the same. The adaptive indicator is obtained by taking the absolute value of the elemental contributions in Eqn. 5,[3, 12, 18, 19]

$$\eta_{\kappa_H} = \left|\mathcal{R}\left(\mathbf{u}_H, \boldsymbol{\psi}_h|_{\kappa_h}\right)\right|. \tag{6}$$

This indicator is computed separately, with absolute values, for each equation in a system of equations and then summed together, possibly yielding a magnitude greater than the actual output error estimate. However, such an indicator is still not a bound for the actual error estimate due to previously made approximations.

III.   Adaptation Using Entropy Variables

The previous section presented output-based error estimation, in which a user prescribes an engineering scalar output to drive mesh adaptation. This sections reviews the entropy-based adjoint indicator, which instead uses entropy variables to drive the adaptation. Regions in the mesh that exhibit high net production of *spurious* entropy are targeted for refinement. The subsequent review of the formulation of the entropy-adjoint approach follows our previous work,[11] to which we refer the reader for more details.

A.   Inviscid Conservation Laws

Consider a steady-state set of conservation laws in quasi-linear form combined with a scalar entropy conservation law,

$$\mathbf{A}_i \partial_i \mathbf{u} = \mathbf{0}, \qquad \partial_i F_i = \mathbf{0},$$

where i is the spatial index, $\mathbf{A}_i$ is the inviscid flux Jacobian, $\mathbf{u}$ is the state vector, and $F_i(\mathbf{u})$ is the entropy flux associated with an entropy function $U(\mathbf{u})$. 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 $\mathbf{v} = U_{\mathbf{u}}^T$, and they symmetrize the conservation laws in the sense that:[20, 21] the transformation Jacobian matrix, $\mathbf{u_v}$, is symmetric, positive definite; and $\mathbf{A}_i\mathbf{u_v}$ is symmetric. Using these properties the quasi-linear form of the conservation law can be manipulated via:

$$\mathbf{0} = \mathbf{A}_i\partial_i\mathbf{u} = \mathbf{A}_i\mathbf{u_v}\partial_i\mathbf{v} = \mathbf{u_v}\mathbf{A}_i^T\partial_i\mathbf{v} \quad \Rightarrow \quad \mathbf{A}_i^T\partial_i\mathbf{v} = \mathbf{0}. \tag{7}$$

The above equation, with the transpose of the inviscid flux Jacobian, represents the continuous adjoint equation, which indicates that the entropy variables are adjoint solutions. The output associated with this adjoint equation is a measure of the net entropy transport through the domain boundary,[11]

$$J = \int_{\partial\Omega} F_i n_i ds. \tag{8}$$

Therefore the entropy variables serve as the adjoint solution to an output that corresponds to the net entropy flow out of the domain. This adjoint can be used in Eqn. 6 with entropy variables to yield an indicator driven by areas of the mesh with spurious entropy generation, i.e. those that are important for the prediction of the net entropy outflow.

B. Viscous Conservation Laws

For a set of viscous conservation laws written in quasi-linear form,

$$\mathbf{A}_i\partial_i\mathbf{u} - \partial_i\left(\mathbf{K}_{ij}\partial_j\mathbf{u}\right) = \mathbf{0}, \tag{9}$$

with $\mathbf{K}_{ij}$ the diffusivity tensor, the entropy variables must also symmetrize $\mathbf{K}_{ij}$, in the sense that $\widetilde{\mathbf{K}}_{ij} = \widetilde{\mathbf{K}}_{ji}^T$, where $\widetilde{\mathbf{K}}_{ij} = \mathbf{K}_{ij}\mathbf{u_v}$.[21] Substituting $\partial_i\mathbf{u} = \mathbf{u_v}\partial_i\mathbf{v}$ into Eqn. 9 and taking the transpose yields the following equation for the entropy variables,

$$\partial_i\mathbf{v}^T\mathbf{A}_i\mathbf{u_v} - \partial_i\left(\partial_j\mathbf{v}^T\widetilde{\mathbf{K}}_{ji}\right) = \mathbf{0}. \tag{10}$$

The entropy variables still represent the sensitivity to residual perturbations of a specific output, and in the viscous case, this output is[11]

$$J = \int_{\partial\Omega} F_i n_i ds + \int_{\Omega} \partial_i\mathbf{v}^T\widetilde{\mathbf{K}}_{ij}\partial_j\mathbf{v}d\Omega - \int_{\partial\Omega} \mathbf{v}^T\widetilde{\mathbf{K}}_{ij}\partial_j\mathbf{v}n_i ds. \tag{11}$$

Each of the terms in Eqn. 11 has its own physical meaning outlined in previous works.[10, 11] Adapting on J using the entropy variables defined in the next section as adjoints is the driver for the entropy-adjoint based approach to mesh refinement. Without the need for a separate system solve to obtain the entropy adjoint, this approach is computationally cheaper than the output-based approach. However, this indicator does not disregard areas of spurious entropy generation that have no effect on a particular engineering output and hence may lead to over-refinement.

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,[21]

$$U = -\rho S/\left(\gamma - 1\right), \qquad S = \ln p - \gamma \ln \rho, \tag{12}$$

American Institute of Aeronautics and Astronautics

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

$$\mathbf{v} = U_{\mathbf{u}}^T = \left[\frac{\gamma - S}{\gamma - 1} - \frac{1}{2}\frac{\rho V^2}{p}, \ \frac{\rho u_i}{p}, \ -\frac{\rho}{p} \right]^T, \tag{13}$$

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. It is worth noting that the entropy variables are obtained via a nonlinear transformation of the conservative variables. The corresponding entropy flux function is $F_i = u_i U$.

IV. Adaptive Mechanics

For this research two different strategies are used to drive the mesh adaptation for each adaptive iteration. The first is a fixed-fraction hanging-node adaptation strategy. The driver of this approach is the elemental adaptive indicator, η_{κ_H}, which is calculated using $\boldsymbol{\psi}_h$ if the adaptation depends on an engineering output or $\mathbf{v}_h$ if the adaptation depends on entropy variables. However, in the combined approach, the output-based and entropy-based indicators are sequentially obtained and then combined to yield a new indicator to drive the adaptation. The term fixed-fraction means a certain fraction, f^{adapt}, of the elements with the largest adaptive indicators is marked for refinement. Marked elements are adapted uniformly, creating a series of hanging nodes. If two neighboring elements vary by more than one level of refinement, additional elements are flagged for refinement. Note that there is no element coarsening performed while using hanging-node adaptation in this research.

The second mesh adaptation approach is a variation of mesh optimization via error sampling and synthesis (MOESS).[22, 23] The distinguishing feature of MOESS is an error sampling approach for determining the convergence rate tensor of the error on a single element. This local sampling approach determines the optimal local refinement for that element by computing an error indicator $\Delta \mathcal{E}_{ei}$ for a finite number of refinement options i of the element. These indicators estimate the error between the coarse-space solution and that on a given refinement option i. Calculating $\Delta \mathcal{E}_{ei}$ requires projecting the fine-space adjoint, $\boldsymbol{\psi}_{hi}^{p+1}|_{\Omega_e}$ or $\mathbf{v}_{hi}^{p+1}|_{\Omega_e}$ depending on whether the adjoint is output-based or entropy-based, down to the space of the refinement option i and order p, and then back up to order $p+1$ on the original element. Adjoint-weighted residuals are then evaluated on the original element. For the output-based adjoint, mathematically this can be expressed as

$$\Delta \mathcal{E}_{ei} \equiv \left| \mathcal{R}_h^{p+1}\left(\mathbf{u}_h^p, \tilde{\boldsymbol{\psi}}_{hi}^p|_{\Omega_e} \right) \right|, \tag{14}$$

where $\tilde{\boldsymbol{\psi}}_{hi}^p$ is the fine-space adjoint $\boldsymbol{\psi}_h^{p+1}$ projected from $p+1$ to order p on refinement option i, and then back to order $p+1$ on the original element. The form of Eqn. 14 that corresponds to the entropy adjoint is identical except that $\mathbf{v}$ replaces $\boldsymbol{\psi}$. The refinement of each element based on local sampling creates anisotropy and leads to better mesh resolution, and thus better error estimation.

The following is a sequential breakdown of the various steps implemented for each adaptive iteration:

1. Solve the primal problem on the current mesh at order p to obtain $\mathbf{u}_H$. If the adaptation depends on an engineering output, solve the adjoint problem to obtain $\boldsymbol{\psi}_H$.
2. Inject $\mathbf{u}_H$ into a fine space of order $p+1$.
3. If the adaptation depends on an engineering output, solve or iterate the fine-space adjoint problem to obtain $\boldsymbol{\psi}_h$. If the adaptation depends on entropy variables, compute $\mathbf{v}_h\left(\mathbf{u}_h \right)$ using Eqn. 13 by either solving the fine-space problem exactly or iteratively smoothing ν_{fine} times to obtain $\mathbf{u}_h$.

American Institute of Aeronautics and Astronautics

4. On each element calculate the adaptive indicator, η_{κ_H}, using Eqn. 6 with $\boldsymbol{\psi}_h$ if the adaptation depends on an engineering output or $\mathbf{v}_h$ if the adaptation depends on entropy variables. For the combined approach, compute indicators based on both the output-based adjoint and the entropy-based adjoint, and then combine them, e.g. via an element-wise multiplication, to obtain a new indicator.

5. Refine the mesh on the elements with the largest indicator using either fixed-fraction hanging node adaptation or MOESS. If MOESS is driving the mesh adaptation, then calculate local sampling errors for each element using Eqn. 14.

6. Initialize the solution on the adapted mesh with the solution from the previous mesh and return to step 1.

V. Adaptive Indicator

The previous sections have illustrated the standard way for obtaining the mesh refinement indicator using the output-based adjoint, the entropy-based adjoint, and the combined approach in which both the output-based and entropy-based indicators are obtained separately and then combined through direct, elemental multiplication. However, various modification were tested for obtaining both the output-based and entropy-based indicators, as well their combination. These modifications were made in the interest of improving adaptive efficacy and reducing the computational cost.

A. Combined Indicator using a Coarse-Space Adjoint

In Section II, Eqn. 4 requires the continuous adjoint on a finer finite-dimensional space. This is an expensive step, especially for large or unsteady problems. To minimize the cost of this step, a new approach to combining the indicators was implemented in which no fine-space output-based adjoint is computed. Instead, both the adjoint and state at the current approximation order are projected down one order prior to obtaining the output-based indicator. Therefore, no new fine-space adjoint needs to be obtained since the fine-space is now at the original approximation order, where an output-based adjoint already exists. The coarse-space output-based adjoint is then subtracted from this adjoint to eliminate pollution from p-dependence of the weak form and non-converged residuals. Since the output-based indicator should still target similar general regions of the domain where more refinement is necessary, this less expensive approach should ideally not lead to significant deterioration of the adaptive performance.

B. Combined Indicator with a Mask

Building off of the previous approach, a subsequent approach was implemented with the addition of a mask on the entropy-based adjoint. The purpose of the mask is to limit the weight of the entropy indicator on the mesh adaptation so that the output-based indicator carries more weight. This is accomplished by creating an element-based refinement indicator, i.e. a mask, based on the output-based adaptation indicator. This mask is an array made up of zeros and ones that indicate which elements are selected for refinement. The percentage of elements selected for the mask is based on a user input. This mask is then multiplied by the entropy-based indicator in an element-wise fashion to obtain the final indicator that ultimately governs the mesh adaptation. The reason for this approach is that the mask should completely eliminate elements targeted by the entropy-based indicator that have no influence on the desired engineering output, regardless of the magnitude of the entropy-based indicator for these elements.

American Institute of Aeronautics and Astronautics

C. Combined Indicator with Output-based Local Sampling in MOESS

The next modification is specific to cases using MOESS as the mesh adaptation strategy. As mentioned in Section IV, in MOESS the adjoint weighted residual is used not only in computing the elemental errors, but also in determining the refinement error samples within each element. Since the sampling of elements within the boundary layer is especially important for correctly assessing the need for anisotropy, an alternative combined approach was implemented in which the combined output-based and entropy-based indicator still yields the total elemental error, but in which just the output-based indicator governs the local sampling of each element. This in theory should lead to more output-tailored anisotropy, especially in elements in the boundary layer.

D. Modified Entropy-Based Adjoint for RANS

The final modification is applicable for cases using RANS. Previous work[24] indicates that for RANS cases, the optimal drag coefficient error estimate using an entropy adjoint requires two inviscid residual evaluations, one with the approximate solution and one with the exact solution. This is motivated by relating the output J from Eqn. 8 directly to one output of interest, the drag coefficient, through Oswatitsch's formula[25] via

$$\delta c_{d,osw} \approx K \delta J^i = K \left(J^i \left(\mathbf{u}_H \right) - J^i \left(\mathbf{u} \right) \right) = K \int_{\Omega} \mathbf{v}^T \left(\mathbf{r}^i \left(\mathbf{u}_H \right) - \mathbf{r}^i \left(\mathbf{u} \right) \right) d\Omega, \tag{15}$$

where the superscript i denotes inviscid residuals or fluxes. For RANS cases, the term $\mathbf{r}^i \left(\mathbf{u} \right)$ needs to be approximated. When using entropy variables, the discrete analogue of Eqn. 15 is

$$\delta c_{d,osw} \approx K \mathbf{V}_h^T \left[\mathbf{R}_h^i \left(\mathbf{U}_h^H \right) - \mathbf{R}_h^i \left(\mathbf{U}_h \right) \right], \tag{16}$$

where $\mathbf{U}_h^H$ is the discrete coarse-space state vector injected into the fine space, and $\mathbf{U}_h$ is the discrete fine-space state. The residual operators $\mathbf{R}$ in Eqn. 16 include only the convective terms of the RANS equations.

VI. Numerical Implementation

A discontinous Galerkin (DG) finite-element code was used to drive both mesh adaptation strategies. The DG discretization of the compressible Navier-Stokes equations uses the Roe approximate Riemann solver[26] for the inviscid fluxes and the second form of Bassi and Rebay for the viscous flux.[27] A Newton-GMRES implicit solver with element-line-Jacobi preconditioning is used to obtain the steady-state solution. While a DG finite element method was used in this work, the practice of using output-based and entropy-variable adjoints to refine and adapt meshes is applicable to general finite element and finite volume methods.

When the adjoint solution depends on an output of interest, a discrete adjoint solution is obtained with the same element-line-Jacobi preconditioned GMRES solver used for the primal solve. Adjoint consistency is achieved by paying close attention to various discretization and output calculation terms.[28–30] The fine approximation space, $\mathcal{V}_h$, required for the adjoint solution $\boldsymbol{\psi}_h$, is obtained by reusing the same mesh while increasing the interpolation order from p to $p+1$. The fine space is then used for solving both the primal and adjoint problems to minimize error. However, when using the first modification to the combined approach in Section V, the output-based adjoint is only solved on the order p space. This reduces the computational burden for the combined approach. Additionally, for some cases, such as those in three dimensions, where computations are quite expensive, an iterative method was chosen for the fine-space primal and adjoint problems in

lieu of a an exact solve due to the excessive expense and previous experimental work illustrating that several smoothing iterations on $\mathcal{V}_h$ yield similar results to an exact solve. For this work $\nu_{\text{fine}} = 5$ element-block-Jacobi iterations were used for both the primal and the adjoint problems. Finally, the error indicator based on an engineering output is found using Eqn. 6, which in a discrete setting reduces to an inner product between the discrete adjoint and residual vectors on the fine space, with absolute values around the contribution from each equation in the system. Whereas the solution to the adjoint equation requires a potentially expensive solve, the entropy variables can be calculated directly from the (approximate) fine-space solution $\mathbf{u}_h$ on each element using a least-squares projection onto $\mathcal{V}_h$.

VII. Results

The results in this section compare adaptation using output-based adjoints, the entropy variables, and various applications of the combined approach for both two-dimensional and three-dimensional cases using various geometries. Some of the geometries, initial meshes, and solution parameters are similar to those used in previous works,[10,11,23] while other cases were designed specially for this research.

A.   NACA 0012 in Two-dimensional Inviscid Flow: $M_\infty = 0.95$, $\alpha = 0°$

The first case is two-dimensional, inviscid flow over the NACA 0012 airfoil with a closed trailing edge and a farfield approximately 40 chord lengths away. The freestream flow is transonic, which produces a substantial fishtail shock at the trailing edge, as shown in Figure 1. This strong fishtail shock makes it necessary to add artificial viscosity to stabilize the flow in the shock's immediate vicinity.
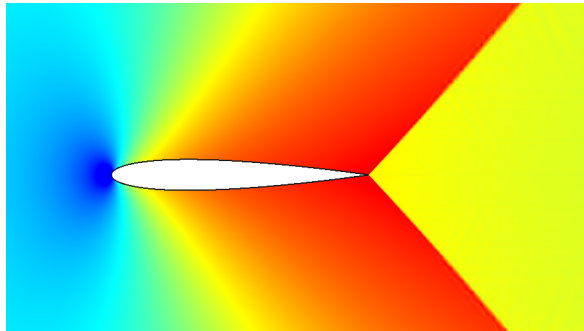

Figure 1. NACA 0012 $M_\infty = 0.95$, $\alpha = 0°$: Mach number contours (range: 0.4-1.6).

The initial mesh of 572 elements is shown in Figure 2. It is an unstructured (initially structured but not after adaptation), quadrilateral mesh with quartic, $q = 4$, curved elements representing the airfoil geometry. The quadratic solution interpolation order for all of the cases is $p = 2$ and the adaptation fraction is set to $f^{\text{adapt}} = 0.1$ for each of the seven adaptation iterations. This means that for each adaptive iteration, only 10% of the elements are refined.

For this case, only one engineering output, the drag coefficient, was analyzed. Integrals of the inviscid momentum flux, i.e. the pressure on the airfoil surface, were used to obtain the aforementioned output. Adaptation was driven using adjoints associated with the drag coefficient, as well as the entropy adjoint. In addition, a combined output-based and entropy-based adjoint solution was generated.

The presence of shocks in inviscid flows leads to the creation of entropy in the regions where
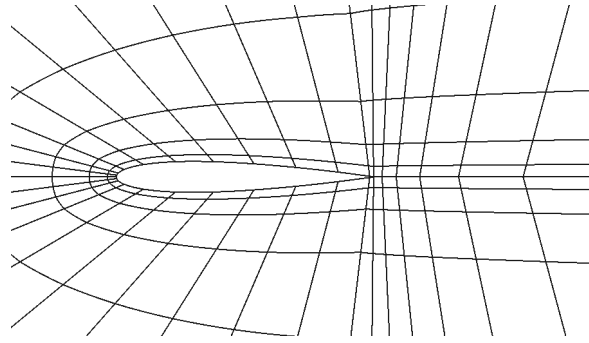
Figure 2. Initial NACA 0012 airfoil mesh.

shocks are present. This creates a problem for the entropy adjoint indicator since it will target regions of the mesh where spurious entropy is generated. The shocks may cause the entropy indicator to continue to refine regions of the mesh to which an engineering output function $J(\mathbf{u})$ may be insensitive. This not only leads to unnecessary mesh refinement in these regions, but also prevents regions of the mesh near the geometry to which $J(\mathbf{u})$ is sensitive from being refined, since only a certain percentage of cells are refined in each adaptive iteration. By combining the output-based adjoint indicator with the entropy adjoint indicator through direct, elemental multiplication, this issue can be alleviated because the output-based adjoint does not unnecessarily target regions of spurious entropy generation that are independent of $J(\mathbf{u})$.

Figure 3 shows the convergence of the drag coefficient for various adaptive indicators and uniform refinement. The truth drag coefficient was obtained from a case at $p = 3$ on a mesh obtained by uniformly refining the finest drag coefficient adapted mesh. Clearly, the creation of entropy due to the fishtail shocks prevents the entropy-based adaptation from properly refining the mesh to produce a more accurate drag coefficient. However, the combined adjoint does show error convergence, as the mesh is refined, since it is not as susceptible to unnecessary refinement in regions that do not affect the drag coefficient. In this particular case the combined-approach does not perform as well as the output-based adjoint approach alone. However, this is not a universal truth for this case since various adjustments made to the shock capturing approach yield results for both the output-based and combined indicator approaches that vary from those presented in this paper by an order of magnitude. Despite the changes to the shock capturing methods, the drag coefficient error convergence using the entropy-based indicator never improved.

Figure 4 compares the meshes after seven adaptation iterations for the error indicator strategies in Figure 3. The drag coefficient adaptation does not target the fishtail shock, outside of the region directly near the trailing edge, whereas the entropy adaptation focuses solely on the fishtail shock and does not refine the mesh anywhere else. This explains why in Figure 3 there is no change in drag coefficient error with each adapted mesh for the entropy-based indicator case. Since the mesh is barely refined near the airfoil surface, the drag coefficient error never improves. The combined drag and entropy adjoint approach yields a much improved mesh compared to just the entropy adjoint alone. However, there is still some unnecessary refinement due to the fishtail shock, because of how strong the entropy-based indicator is in that region. This is likely the reason why the combined approach does not perform as well as the output-based alone approach. One option to remedy this issue is to use the combined approach that masks the entropy indicator so that its effect is not as pronounced. This option will be investigated for a later result.
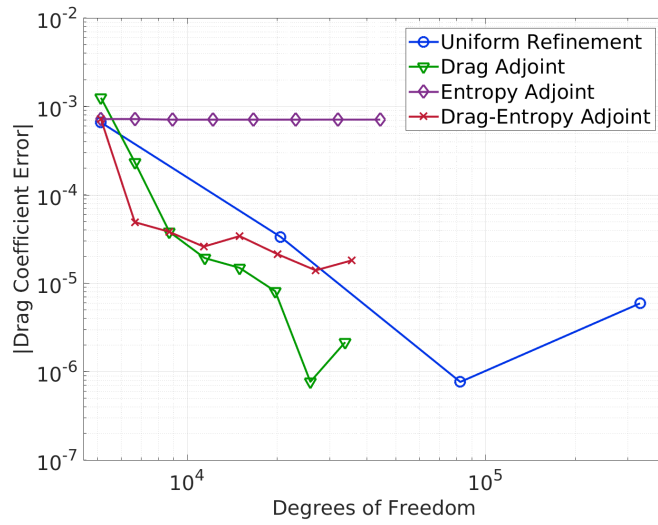
American Institute of Aeronautics and Astronautics

Figure 3. NACA 0012 $M_\infty = 0.95$, $\alpha = 0°$: **Comparison of drag coefficient convergence histories for various adaptation methods.**
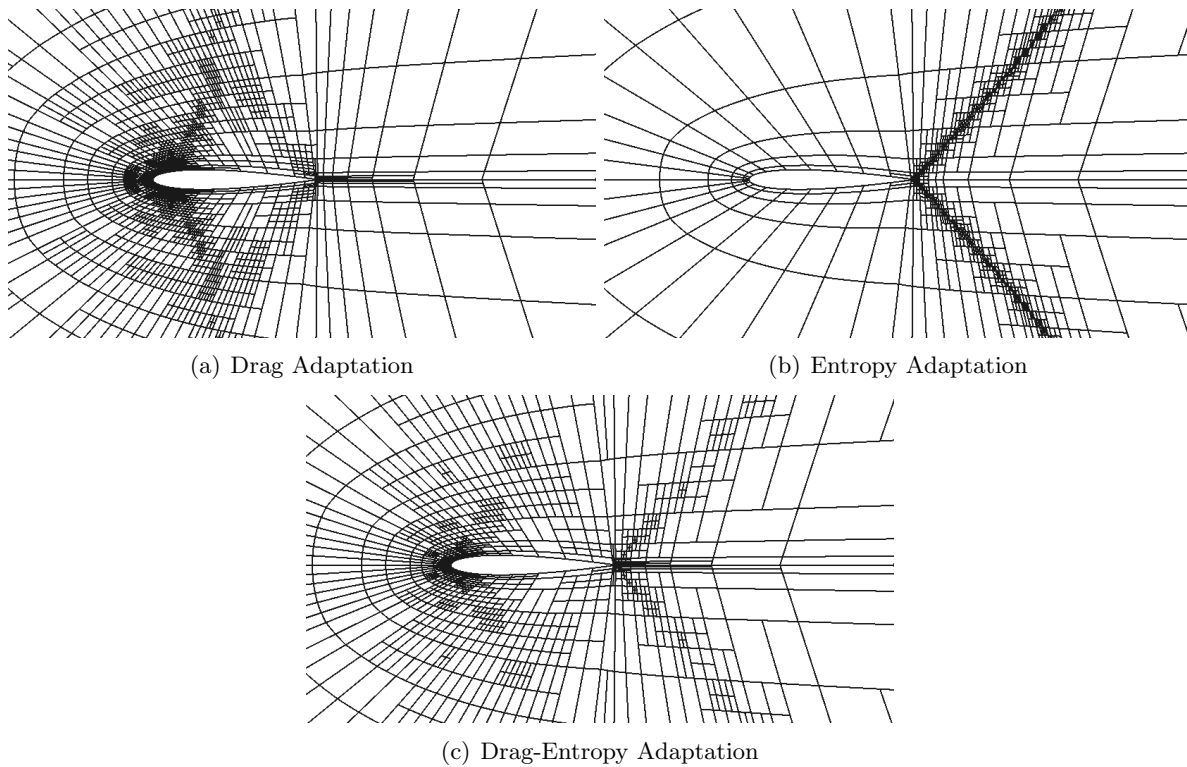

(a) Drag Adaptation

(b) Entropy Adaptation

(c) Drag-Entropy Adaptation

Figure 4. NACA 0012 $M_\infty = 0.95$, $\alpha = 0°$: **Meshes after seven adaptation iterations for various error indicators.**

American Institute of Aeronautics and Astronautics

B. NACA 0012 Wing in Inviscid Flow: $M_\infty = 0.4$, $\alpha = 3°$

The next case is a NACA 0012, untapered, untwisted wing with a closed trailing edge and a rounded wing tip. The wing has an aspect ratio of 10 and is inside a domain with a farfield 40 chord lengths away. The unstructured mesh is composed of cubic, $q = 3$, hexahedral elements that curve to match the wing geometry. Quadratic solution interpolation, $p = 2$, was used in the discretization, but for these runs the adaptation fixed fraction was lowered to $f^{\mathrm{adapt}} = 0.05$. Artificial viscosity stabilization was added to the solution to provide stability in the vicinity of the trailing-edge vortices.

The two engineering outputs that are considered for this case are drag and lift coefficient. Adjoint solutions associated with these two outputs were used to drive separate adaptation runs. Additionally, the entropy adjoint indicator, as well as the combined entropy and output-based adjoint indicator runs were tested. Figure 5 shows the results of these adaptation runs, along with a uniform refinement run for comparison. No truth solutions were obtained due to the computational expense of such runs. Consequently, what is shown in these figures is not error but the actual lift and drag coefficients. For the drag coefficient, the drag adjoint and combined drag and entropy adjoint behave comparably and converge to a very similar value, assuming this value is close to the exact solution. The entropy adjoint by itself does not converge to the same degree and particularly underperforms for the coarser meshes. Similarly to what was observed for the drag coefficient, the lift adjoint and combined lift and entropy adjoint perform comparably regarding their ability to predict the lift coefficient output. The entropy adjoint indicator by itself is much more erratic and does not perform nearly as well.

(a) Drag Output   (b) Lift Output
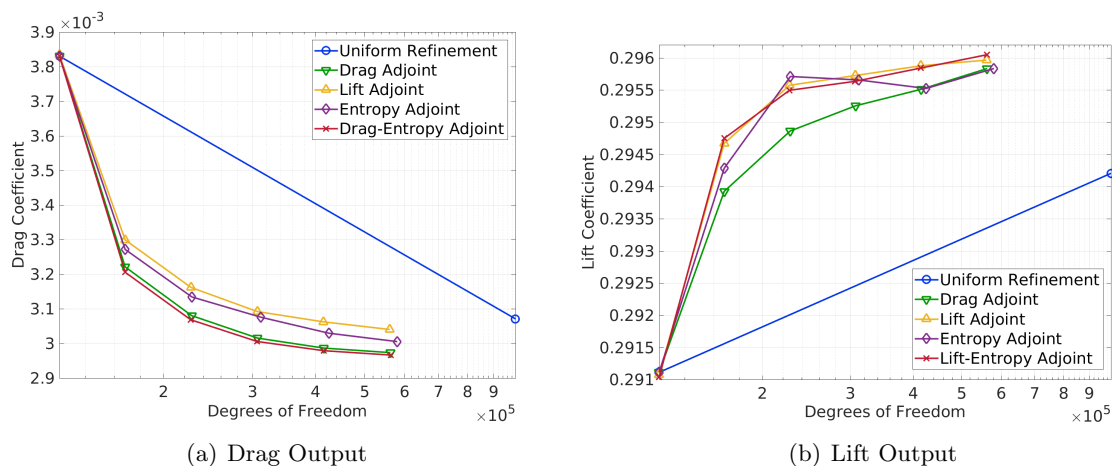
Figure 5. NACA 0012 Wing $M_\infty = 0.4$, $\alpha = 3°$: Comparison of output convergence histories for various adaptation methods.

An explanation for why the entropy adjoint indicator does not perform as well is its propensity to refine the mesh in areas consumed by the wing tip vortex. From experimental work[31] it is known that wakes form behind the trailing edge. This occurs due to the natural extension of the boundary layer from the wing surface, as well as streamwise vorticity that is shed due to lift distribution variation. This case is inviscid so there is no boundary layer. However, there still is shedding of streamwise vorticity, particularly at the wing tip. The numerical results presented in this paper show a very concentrated vortex emanating from the wing tip. Since the under-resolved wing tip vortex leads to spurious entropy generation, the entropy adjoint indicator targets areas of the mesh through which the vortex travels.

Figure 6 shows the initial mesh, as well as the final meshes, after five adaptive iterations for the various adaptive strategies. The cut plane in these images is not far behind the wing trailing edge. The output-based indicators refine the mesh near the wing surface and do not significantly refine the mesh downstream of the wing trailing edge. However, the entropy adjoint indicator refines the regions of the mesh engulfed by the wake, especially where the wing tip vortex is. While areas of the mesh, particularly those near the leading edge, are also refined, the entropy adjoint indicator sacrifices refinement on the upper surface of the wing due to the excessive refinement in regions of the mesh where the wing tip vortex is. The combined output-based and entropy-based adjoint indicators have some refinement due to the wing tip vortex, but it is not nearly excessive enough to lead to poor adaptation performance, as illustrated by the results in Figure 5.

The impact of the wing tip vortex can be further examined via the results in Figure 7, which show cuts of the mesh at the wing tip. Again, the output-based adjoint indicators focus all of the refinement near the wing tip surface, while the entropy adjoint indicator focuses refinement downstream of the wing trailing edge due to the wing tip vortex. The combined drag and entropy adjoint indicator refines the region of the mesh consumed by the wing tip vortex near the wing tip surface, but does not excessively refine the mesh further downstream. It is reasonable to surmise that some refinement due to the wing tip vortex is necessary to accurately predict the drag coefficient, but not nearly to the level the entropy adjoint indicator indicates. The combined drag and entropy adjoint indicator does a much better job balancing refinement between regions in the wake and near the airfoil surface. This is even more evident for the combined lift and entropy adjoint indicator case, where the refinement due to the wing tip vortex is even less pronounced since the wing tip vortex under-resolution does not contribute to the lift coefficient error to the same degree as to the drag coefficient error.

C.   Diamond Airfoil in Two-Dimensional Inviscid Supersonic Flow: $M_\infty = 1.5$, $\alpha = 2°$

The next case is two-dimensional, inviscid flow over a thin diamond airfoil with a thickness ratio of 0.05. The grid is a square with a side length of 10 chord lengths. The freestream flow is supersonic, which produces shocks emanating from both the top and bottom surfaces of the airfoil, as visible in Figure 8. However, because the airfoil is thin enough to produce weak shocks, no artificial viscosity was added to the solution. This is ideal since it is desirable to not have artificial viscosity present in a simulation since it might adversely affect the entropy-based indicator.

The airfoil is situated slightly off from the center of the domain so that the shocks emanating from the airfoil do not directly impact the corner of the mesh domain. Figure 9 shows the initial, unstructured, quadrilateral mesh made up of linear, $q = 1$, elements. As evident from the figure, the initial mesh is quite coarse. Just as with the inviscid NACA 0012 case, the solution approximation order for all of the cases is $p = 2$ and the adaptation fraction was set to $f^{\text{adapt}} = 0.1$ for all of the adaptation iterations. In addition, for these runs a new parameter, $f^{\text{error}} = 0.95$ was set. This parameter, called the adaptation fixed error fraction, indicates the fraction of total error (sum of indicators) that will be targeted for refinement during adaptation. However, the actual fraction of elements adapted will still be limited by f^{adapt}.

For this case the drag coefficient and the lift coefficient are again used as outputs to govern the mesh adaptation. Adaptation runs using adjoints based on these two outputs, as well as runs using both the entropy-based adjoint and the combined entropy and output-based adjoint were generated. In addition, three modified forms of the combined approach were considered for this case. The first modified approach, described in Section V, entails not solving for the fine-space output-based adjoint. Instead, the adjoint and the state are projected down one order, to $p = 1$ for this case, so that the output-based indicator can be obtained at the $p = 2$ space. The second and third modified approaches build off of the first modified approach by adding a mask on the

American Institute of Aeronautics and Astronautics

(a) Initial Mesh

(b) Drag Adaptation

(c) Lift Adaptation

(d) Entropy Adaptation

(e) Drag-Entropy Adaptation

(f) Lift-Entropy Adaptation
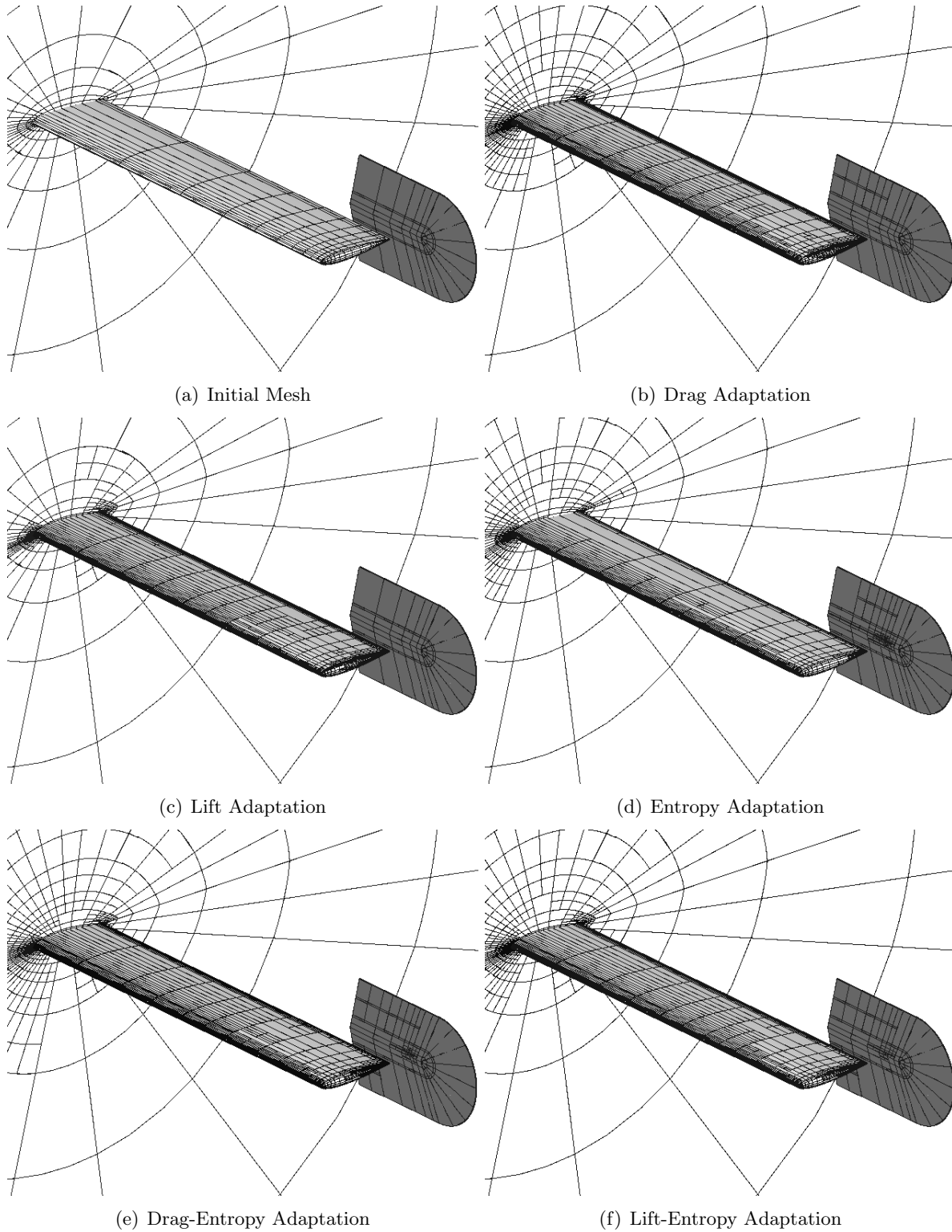
Figure 6. NACA 0012 Wing $M_\infty = 0.4$, $\alpha = 3°$: **Initial mesh and meshes after five adaptation iterations for various adaptive strategies.**
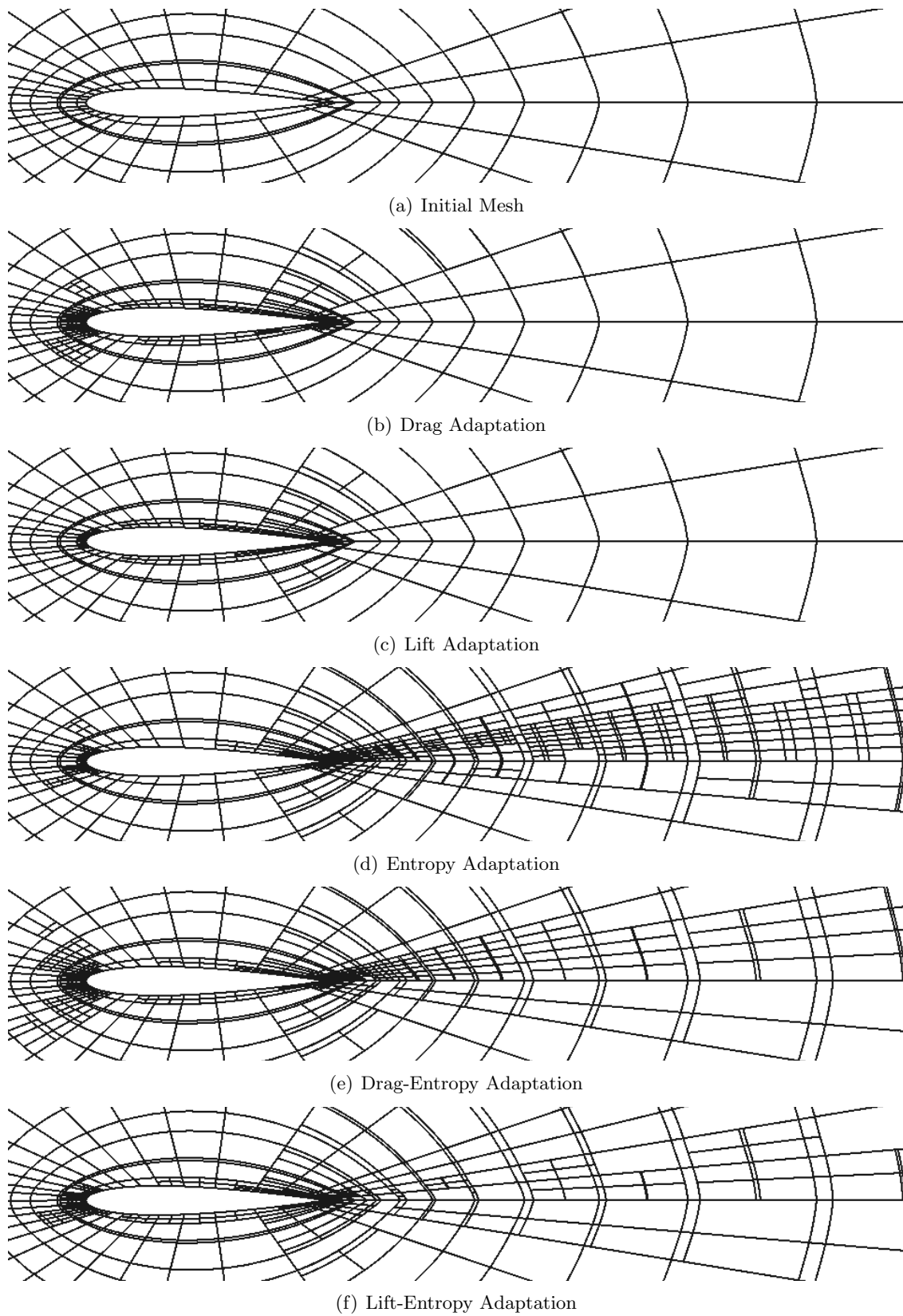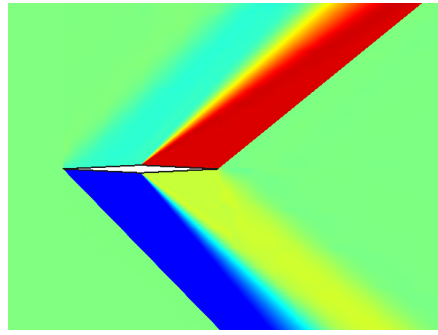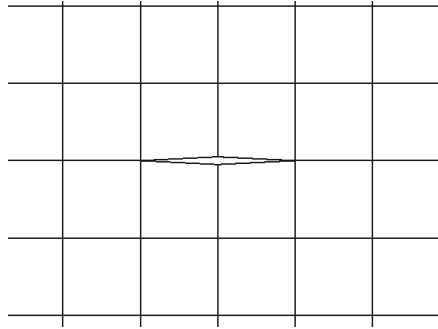
(a) Initial Mesh

(b) Drag Adaptation

(c) Lift Adaptation

(d) Entropy Adaptation

(e) Drag-Entropy Adaptation

(f) Lift-Entropy Adaptation

Figure 7. NACA 0012 Wing $M_\infty = 0.4$, $\alpha = 3°$: Cut-plane at wing tip from initial mesh and meshes after five adaptation iterations for various error indicator strategies.

Figure 8. Diamond $M_\infty = 1.5$, $\alpha = 2°$: Mach number contours (range: 1.3-1.7).

Figure 9. Initial diamond airfoil mesh.

entropy-based indicator using different masking percentages, in this case 10% and 25%.

Figure 10 shows the drag and lift coefficient convergence results for all of the various error indicator strategies. The number of adaptation iterations changes for each strategy because the adaptation fixed error fraction and the masking alter how many elements get refined in each iteration. The truth outputs for both lift and drag coefficient were obtained from a case at $p = 3$ on a mesh obtained by uniformly refining the finest output-adapted mesh. The modified combined approach, in which no fine space output-based adjoints were solved for and the fine-space primal problem was approximated with an iterative solver, shows slightly downgraded performance, as expected. However, the performance of this modified approach improves significantly with the addition of the mask on the entropy-based indicator. Unfortunately, no one particular masking percentage yields superior results for both lift and drag error estimation. The 10% mask, which requires many more adaptation iterations, since few elements get targeted in each adaptation iteration, is superior in terms of measuring drag coefficient. However, for lift coefficient error estimation, this strategy's error convergence eventually levels out and the 25% mask overtakes it.

Figure 11 presents the meshes after the final adaptation iteration for the various error indicator strategies that, outside of the entropy-based adjoint alone case, all include indicators based off the drag-based adjoint. The entropy-based adjoint indicator mostly focuses on the discontinuities emanating from the leading edge and trailing edge of the diamond. This refinement extends unnecessarily all the way to the farfield boundaries. This accounts for why the error convergence of the entropy-based approach is the worst performer. The drag-based adjoint indicator does not suffer from this issue, instead producing regions of refinement that when examined as a whole, resemble the shape of a diamond. The regions of refinement that originate near the airfoil and then travel downstream are necessary for good error estimation. However, by examining the four
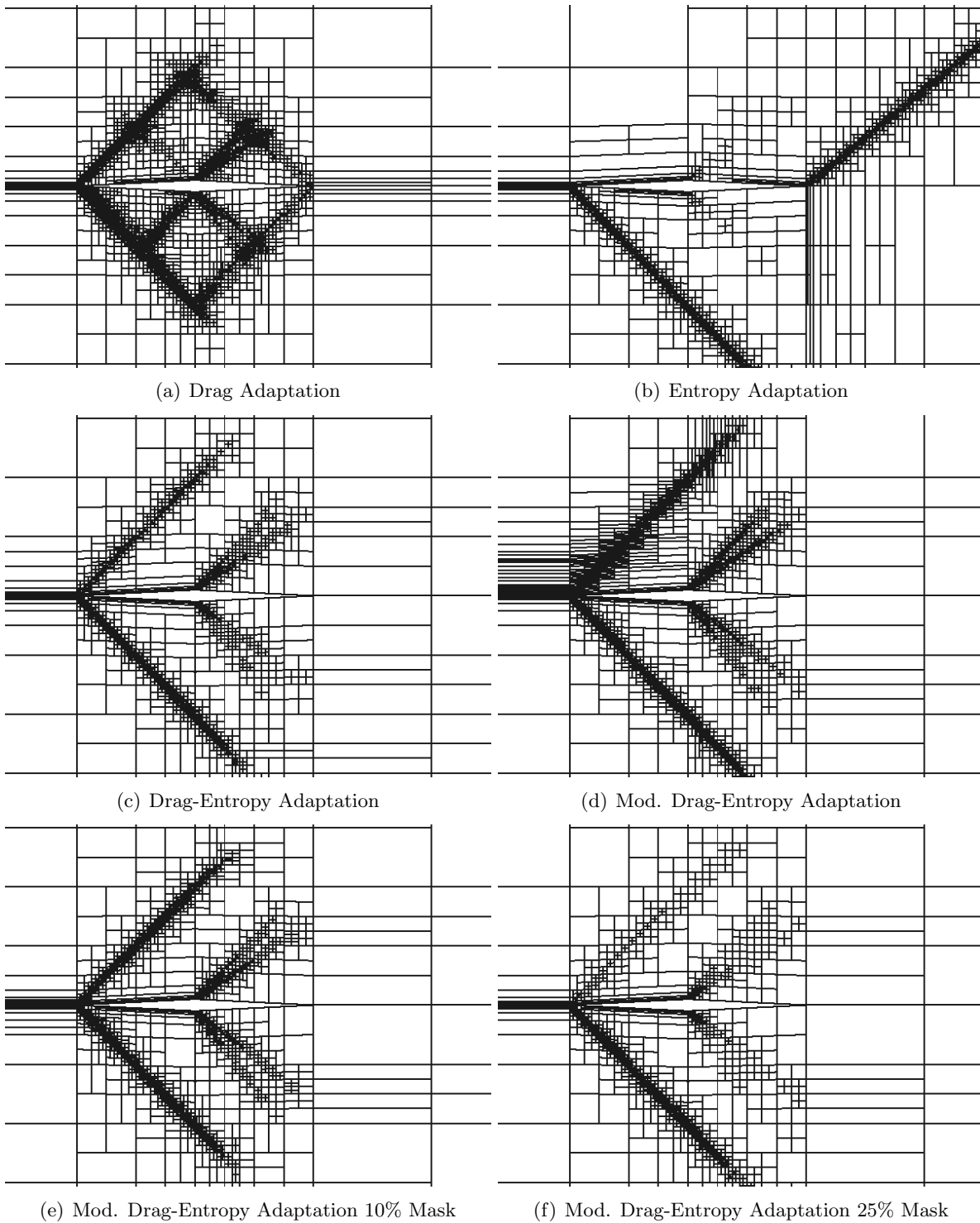
American Institute of Aeronautics and Astronautics

(a) Drag Output   (b) Lift Output
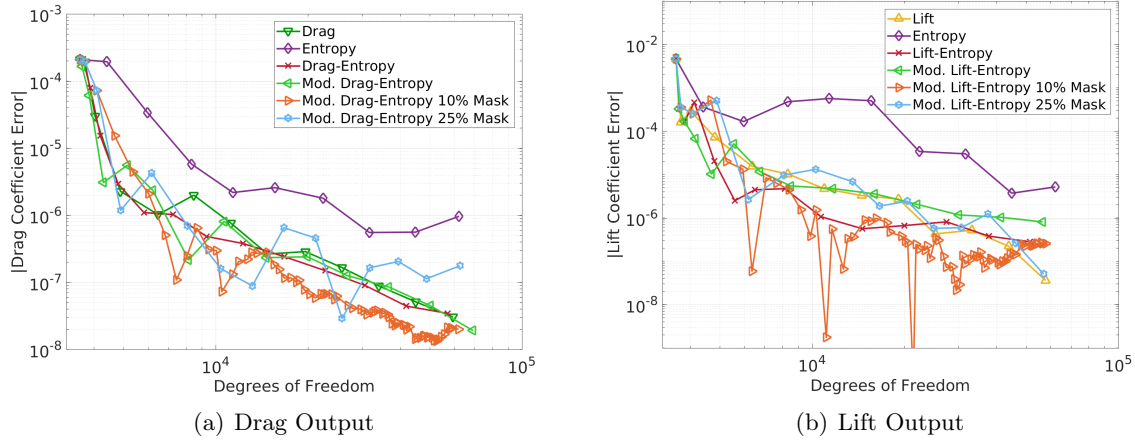
Figure 10. Diamond $M_\infty = 1.5$, $\alpha = 2°$: Comparison of output convergence histories for various adaptation methods.

meshes obtained using a variation of the combined approach, it is reasonable to conclude that the regions of refinement that travel upstream may not be necessary for accurate error estimation. The combined approach that uses a 25% mask on the entropy indicator produces significantly poorer drag coefficient convergence compared to the other three combined indicator approaches. Examining the mesh clearly shows a lack of refinement in the region of the domain above the leading tip of the diamond. This is because the entropy-based indicator does not target this region at all. Subsequently, a logical conclusion is that for this case, having a small masking percentage is ideal.

Figure 12 includes meshes after the final adaptation iteration for the various error indicator strategies that rely on the lift-based adjoint. While the combined approach using the 10% mask on the entropy indicator yields the best drag coefficient error convergence, the combined approach using the 25% masked yields a superior result with regard to estimating the lift coefficient. Inspection of the final mesh for this case leads to the incorrect conclusion that this case would not perform well since the mesh is relatively coarse above the diamond. However, the issues with the combined approaches that do not have the mask is that many of the elements, particularly those in the region above the airfoil, have high aspect ratios. Since hanging node adaptation requires certain ratios between cell dimensions, the very small grid cells near the top point and leading tip of the diamond airfoil lead to the propagation of large aspect ratio cells above the airfoil. These cells do a poor job of capturing the very thin discontinuity. The high aspect ratio cells are not as prevalent in the case with the 25% mask, hence its improvement in the estimation of the lift coefficient.

It is important to note that many masking percentages, as well as various values for f^{adapt} and f^{error}, were studied using this case. Additionally, masking was attempted on the output-based indicator as opposed to the entropy-based indicator. The authors have chosen to include combinations of inputs that yielded the best results for this particular case. However, it is clear that no one particular combination of inputs always yields the best error convergence. Part of the issue in determining the optimal approach is the inherent limitations of the hanging node adaptation discussed in the previous paragraph. To eliminate these limitations, the subsequent results will focus on adaptation using MOESS.

(a) Drag Adaptation

(b) Entropy Adaptation

(c) Drag-Entropy Adaptation

(d) Mod. Drag-Entropy Adaptation

(e) Mod. Drag-Entropy Adaptation 10% Mask

(f) Mod. Drag-Entropy Adaptation 25% Mask

Figure 11. Diamond $M_\infty = 1.5$, $\alpha = 2°$: Final mesh after adaptation iterations for various error indicator strategies, most of which require the drag coefficient adjoint.

(a) Lift Adaptation

(b) Entropy Adaptation

(c) Lift-Entropy Adaptation

(d) Mod. Lift-Entropy Adaptation

(e) Mod. Lift-Entropy Adaptation 10% Mask

(f) Mod. Lift-Entropy Adaptation 25% Mask
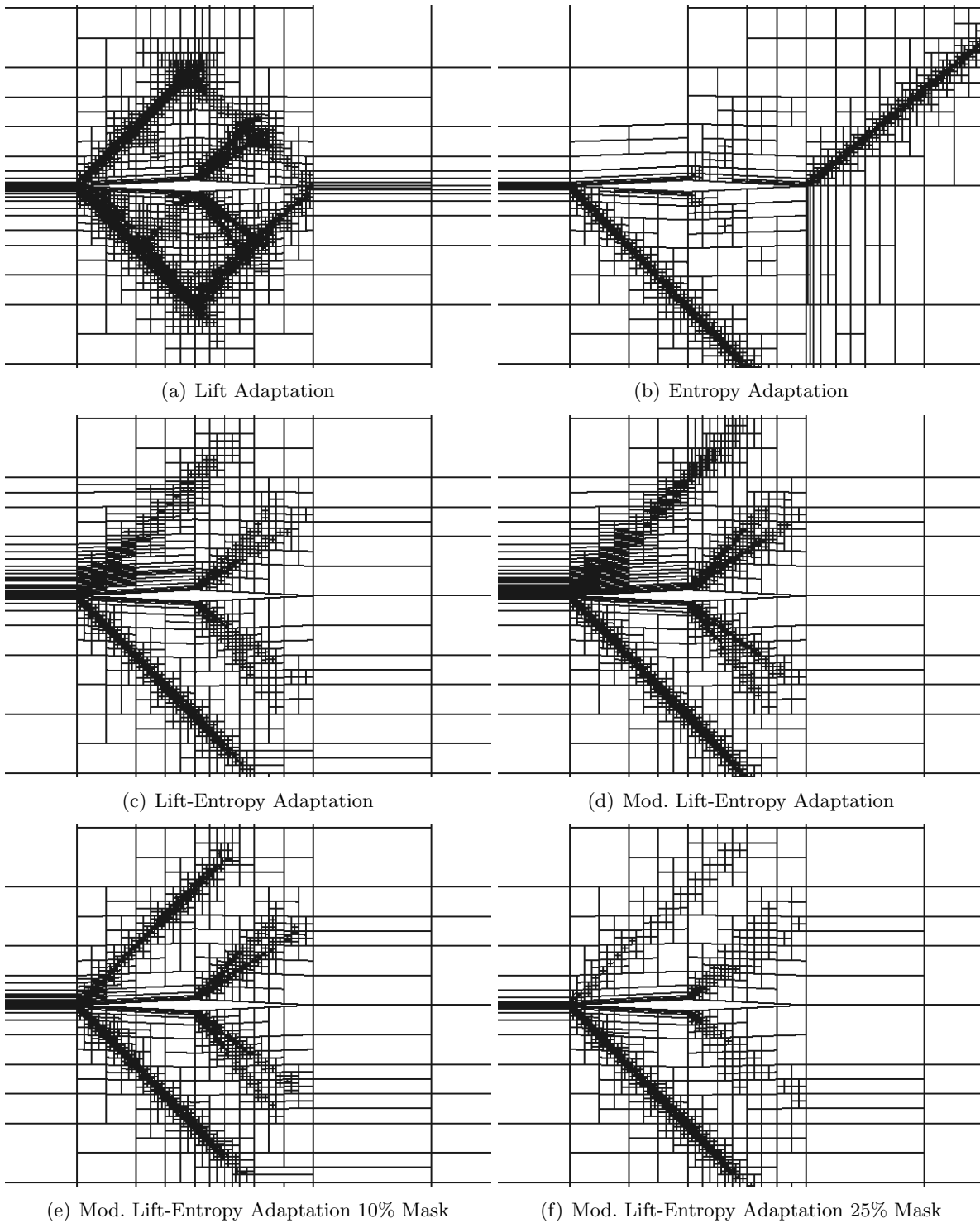
Figure 12. Diamond $M_\infty = 1.5$, $\alpha = 2°$: Final mesh after adaptation iterations for various error indicator strategies, most of which require the lift coefficient adjoint.

D.   NACA 0012 in Two-Dimensional Viscous Flow: $M_\infty = 0.5$, $\alpha = 5°$, $Re = 5000$

To illustrate how using MOESS adaptation compares to hanging node adaptation, we consider two-dimensional, viscous flow over the NACA 0012 airfoil with a closed trailing edge. The initial meshes are shown in Figure 13. The initial mesh used with hanging node adaptation is a quadrilateral mesh with quartic, $q = 4$, curved elements representing the geometry. The mesh is made up of 818 elements and the farfield is approximately 40 chord lengths away. The extra refinement near the surface is necessary to enable a solution on the coarsest mesh. The adaptation fraction was set to $f^{\text{adapt}} = 0.1$ and the adaptation fixed error fraction was set to $f^{\text{error}} = 1$ for each of the nine adaptation iterations. Finally, the solution was set to $p = 2$ for all cases.

The initial mesh used with MOESS is an unstructured, triangular mesh with 356 elements. The farfield is over 2000 chord-lengths away and the curved elements of geometry order $q = 4$ are used to represent the airfoil. The cases are run with a solution interpolation order of $p = 2$ with the following six degrees of freedom targets: 2000, 4000, 8000, 16000, 32000, and 64000. At each `dof` target, fifteen solution iterations are performed before moving to the next `dof` target.

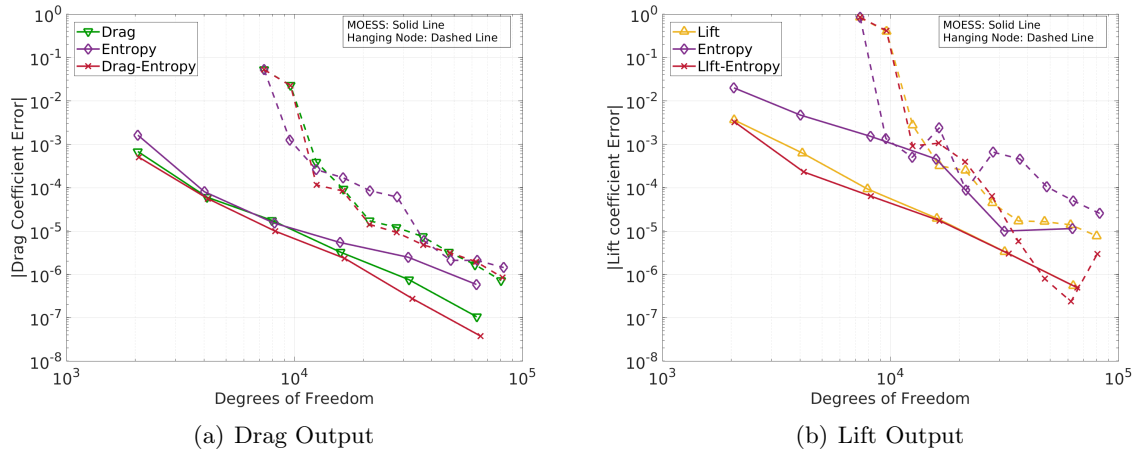(a) Hanging Node   (b) MOESS
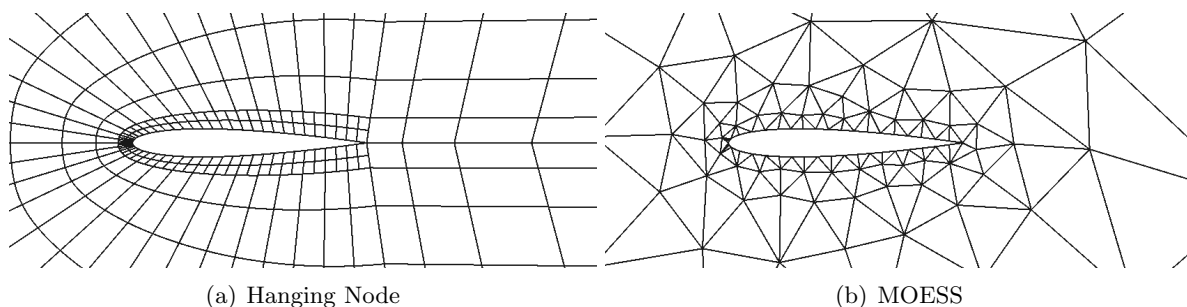
Figure 13. Initial viscous NACA 0012 airfoil meshes

For both sets of adaptation strategies, the drag coefficient and lift coefficients were used as outputs to drive the mesh adaptation. Adaptation runs for these outputs, the entropy adjoint alone, and the combined entropy and output-based adjoint were considered. None of the previously detailed modifications were made to the combined approach in this case. Figure 14 shows the convergence results for all of the error indicators using both sets of mesh adaptation strategies. The solid lines represent solutions using MOESS, while the dashed lines represent solutions using hanging node adaptation. The output and `dof` values reported from the cases incorporating MOESS are averages over the last 5 solution iterations at each target `dof`. The truth outputs for both the lift and drag coefficients were obtained by refining the final output-adapted mesh and running at $p = 3$.

Unsurprisingly the MOESS adapted cases for the most part perform much better than the hanging node adapted cases since MOESS has the ability to coarsen regions of the mesh that do not affect the output error estimation. When implementing hanging node refinement, the three error indicator strategies' drag coefficient error convergence does not vary significantly. However, for MOESS a clear distinction is present. The entropy adjoint based adaptation solution is quite poor, while the combined approach yields better performance than the drag coefficient based adaptation solution. For the lift coefficient error convergence, the combined approach using hanging node adaptation is superior to output-based hanging node adaptation. However, for MOESS there is not much difference between the combined approach and output-based approach.

Figure 15 presents the meshes at 64000 `dof` for the cases that used MOESS adaptation. The individual output-based approaches target regions of the domain near the airfoil, as well as the portions of the wake near the airfoil and areas near the stagnation streamline. In fact there is

American Institute of Aeronautics and Astronautics

(a) Drag Output (b) Lift Output

Figure 14. NACA 0012 $M_\infty = 0.5$, $\alpha = 5°$, $Re = 5000$: **Comparison of output convergence histories for various adaptation methods using both hanging node and MOESS adaptation.**

substantial refinement ahead of the stagnation streamline that is multiple chord lengths long. The entropy-based adjoint approach does not target regions of the mesh well ahead of the stagnation point. Instead it targets the entire wake, all the way to the farfield boundary since there is substantial spurious entropy generation in an under-resolved wake. The entropy indicator also leaves the aft portion of the upper surface of the airfoil relatively unrefined. The combined approach achieves a balance between refining ahead of the airfoil and in the wake. The meshes obtained from these approaches are refined in the wake, but not to the excessive extent present in the mesh produced from the entropy-based approach. In addition, these meshes have refinement ahead of the stagnation streamline, but the refinement does not extend well ahead of the airfoil as it does for the output-based approaches. This balance in refinement for this particular case clearly shows the benefit of using a combined approach to govern the mesh adaptation.

E. RAE 2822 in two-dimensional RANS flow: $M_\infty = 0.25$, $\alpha = 5°$, $Re = 6.5 \times 10^6$

The final case is an RAE 2822 airfoil in subsonic, turbulent flow with a modified Spalart-Allmaras turbulence model.[32] A low freestream Mach number of $M = 0.25$ was chosen so that the Mach number in the domain never approaches transonic conditions. The rationale for this is the concern that the present shock capturing techniques, through artificial viscosity stabilization, would affect the entropy adjoint. Figure 16 shows the initial coarse mesh for this case. It is an unstructured mesh made up of 1428 triangular elements. Curved elements of geometry order $q = 3$ were used to represent the airfoil geometry and the farfield boundary is 2000 chord-lengths away. The cases were run with a solution approximation order of $p = 2$ with the following five degrees of freedom targets: 5000, 10000, 20000, 40000, and 80000. Just as with the previous MOESS simulations, for each `dof` target, fifteen solution iterations were performed before moving to the next `dof` target.

As before, the drag coefficient and lift coefficients were used as outputs to govern the mesh adaptation. Adaptation runs for these outputs, the entropy adjoint alone, and the combined entropy and output-based adjoint solution were generated. The last two modifications presented in Section V were also investigated for this case. The first is a modification to the combined approach in which only the output-based indicator is used to target which elements get refined while using MOESS. Therefore the output-based indicator alone governs the local sampling. The second modification involves using a modified residual made up of only the convective terms to obtain the entropy-based
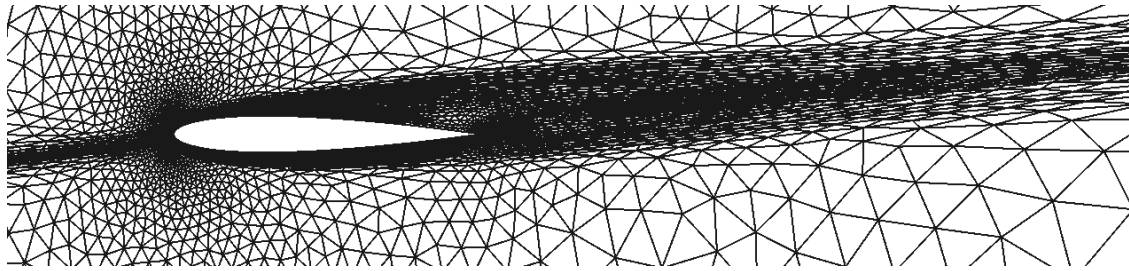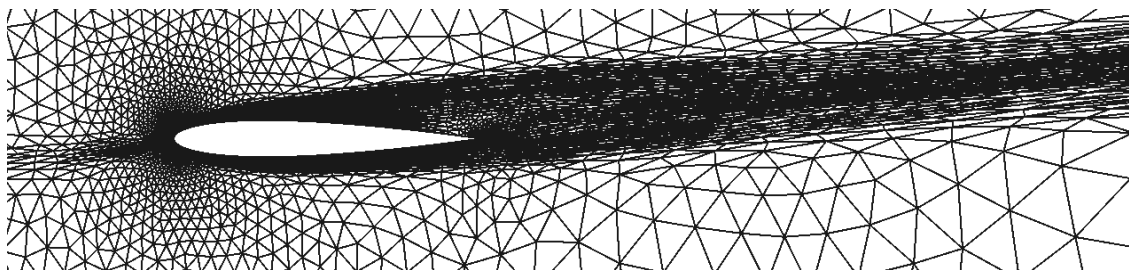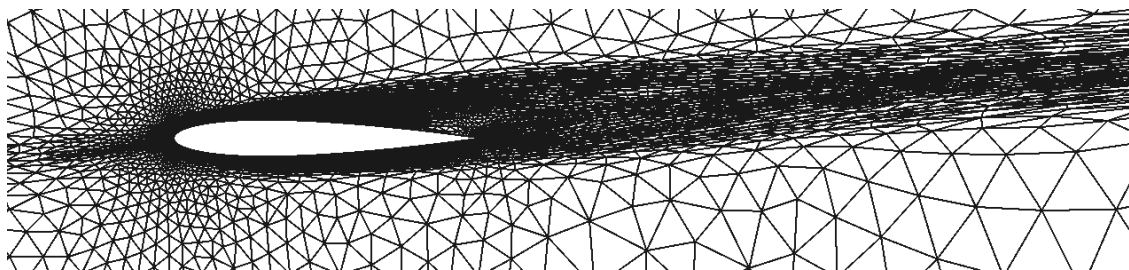
(a) Drag Adaptation
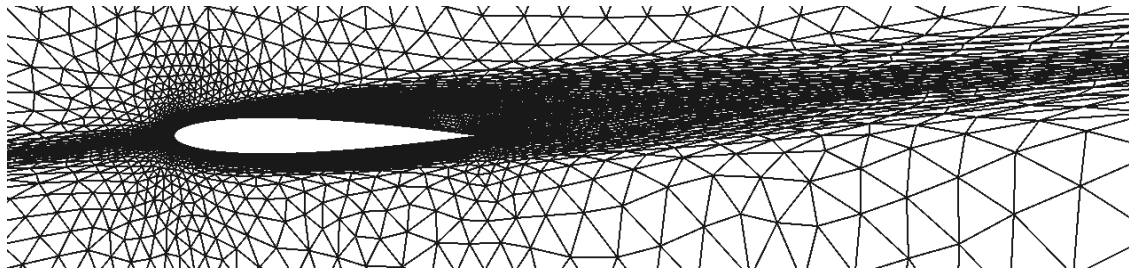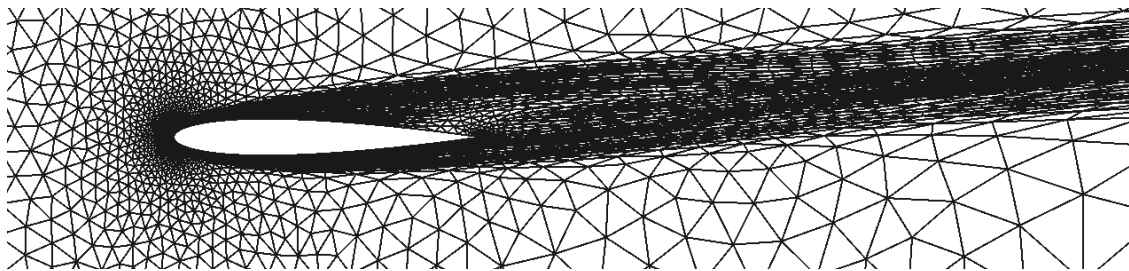
(b) Lift Adaptation
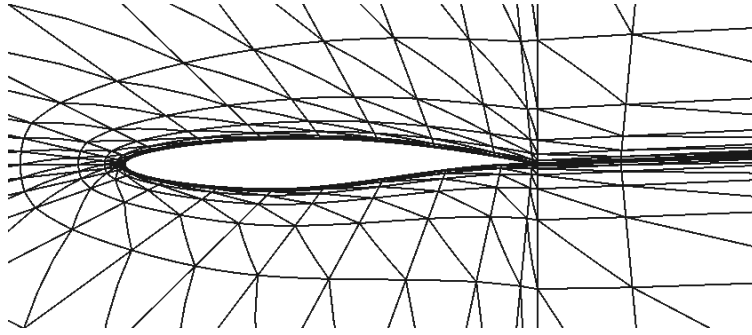

(c) Entropy Adaptation

(d) Drag-Entropy Adaptation

(e) Lift-Entropy Adaptation

Figure 15. NACA 0012 $M_\infty = 0.5$, $\alpha = 5°$, $Re = 5000$: **Meshes after final adaptation iteration (approximately 64000 `dof`) for various error indicator strategies.**

Figure 16. Initial RAE 2822 airfoil mesh.

adjoint. Since this is specifically applicable only to the entropy-based adjoint, modified versions of the entropy-based and combined approach using this modified entropy-adjoint were investigated.

Figure 17 shows the convergence results for all of the error indicator strategies. Again, the output and `dof` values reported from these results are averages over the last 5 solution iterations at each target `dof`. The truth outputs for both the drag and lift coefficients were obtained by refining the final output-adapted mesh and running at $p = 3$. Each strategy that requires the use of the entropy-based adjoint has a corresponding dashed line that corresponds to the results that incorporate the modified residual approach to obtain the entropy-based indicator. While the various approaches yield varying error convergence results for the first few DOF targets, the error values at the final `dof` targets are all pretty similar. Since there is no substantial difference in the output error convergence for the output-based approach versus the entropy-based approach, the combined adjoint approach does not yield any improvement in the error convergence. In addition, the alternate combined approach, in which the output-based indicator governs the sampling within MOESS, does not show a significant improvement, nor does the modified residual approach to obtaining the entropy-based indicator.

Figure 18 presents the meshes at 80000 `dof` for output-based adjoint approaches, the entropy-based approach, and the combined approaches with no modifications. Although the drag and lift coefficient errors at this `dof` are not that different among all of the adaptation indicator strategies, the meshes do show visible differences consistent with those in Figure 15. Clearly the output-based adjoint approaches focus refinement ahead of the airfoil and less in the wake, while the entropy-based approach focuses significant refinement in the wake. Yet the output error levels do not vary significantly among the different strategies. In fact the magnitudes of the drag and lift error are quite high. This phenomenon is not specific to this case, as other cases with different free-stream conditions and geometries produce similar conclusions. Future work will look into why using the various adaptation indicators strategies lead to noticeably different meshes, but not to different output error levels.

VIII.   Conclusions

By combining the output-based adjoint indicator with the entropy variable adjoint indicator, an appreciable benefit over the entropy variable adjoint approach was found in several examples of aerodynamic interest. The entropy variable adjoint targets areas of the mesh where spurious entropy is generated, even though those areas may not significantly affect the engineering output of interest. This sometimes prevents refinement in more critical regions of the mesh. By combining the entropy adjoint indicator with the output-based adjoint indicator, this problem is marginalized since output-based adjoint methods do not target these unnecessary areas of the mesh for refinement. In

American Institute of Aeronautics and Astronautics

| | (a) Drag Output | (b) Lift Output |
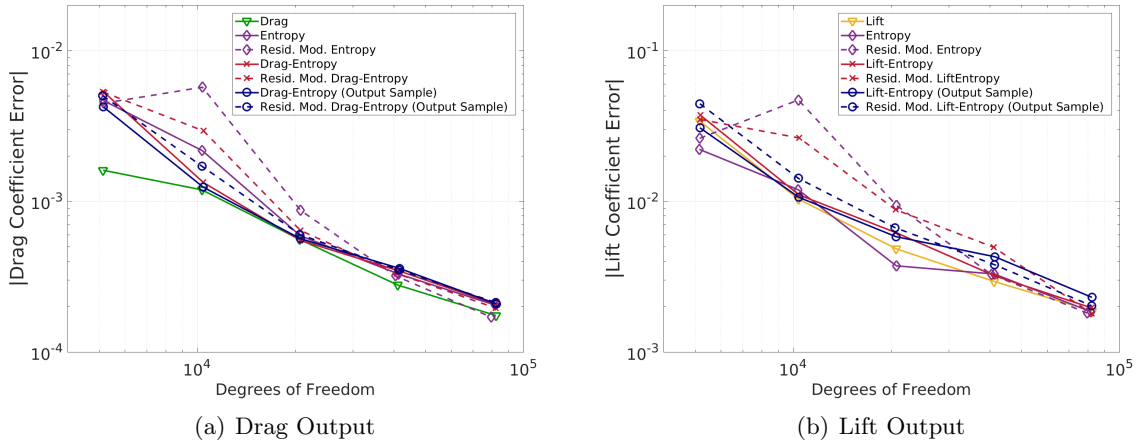
Figure 17. RAE 2822 $M_\infty = 0.25$, $\alpha = 5°$, $Re = 6.5 \times 10^6$: **Comparison of output convergence histories for various adaptation methods.**

addition, the combined indicator approach yields improved performance compared to the standard output-based indicator approach for some cases. Since the output-based indicator is derived from approximate adjoint solutions, which may be under-resolved and noisy, these methods may end up targeting areas of the domain where additional refinement is not necessary. This is particularly the case with the flow ahead of the body of interest, near the stagnation streamline. Combining the indicators limits excessive refinement both upstream of and downstream from the body.

When using hanging node mesh adaptation, it was evident that applying a mask on the entropy-based indicator can yield superior error convergence when just direct, elemental combination of indicators is not sufficient. However, the optimal mask is not always clear for a specific case, as changing what percentage of elements are masked can yield varying results.

The benefits to using the combined indicator approach are not exclusive to hanging node adaptation, as benefits to the combined approach were shown for cases using MOESS. However, while the combined approach often shows an appreciable benefit for inviscid and viscous cases, no benefit was currently found for fully RANS cases. The various adaptation indicator strategies produced different meshes, but the actual output error levels did not produce any appreciable variance. This phenomenon will be investigated in future work.

As previously mentioned, obtaining the fine-space output-based adjoint can be quite expensive, especially for unsteady cases. Future work will focus on applying the combined indicator approach to unsteady problems. To mitigate the cost associated with obtaining the output-based adjoint indicator, the indicator will be obtained on a coarser space without an exact solve. When combined with the entropy-based indicator, this should lead to results that are superior to those obtained using just entropy-based adaptation and not significantly more computationally expensive. The combined approach will always be more computationally expensive than using the entropy-based approach since it requires an output-based indicator, but because of the limitations with the entropy-based indicator, should often be more accurate. In addition, the combined approach using a coarse space approximate output-based indicator will be less expensive than an output-based approach using a fine-space adjoint for unsteady problems. These observations suggest that the benefits of the combined-approach will be much more pronounced for unsteady problems.
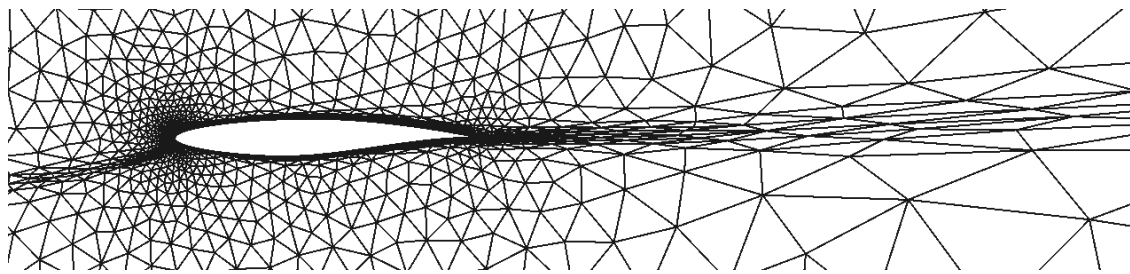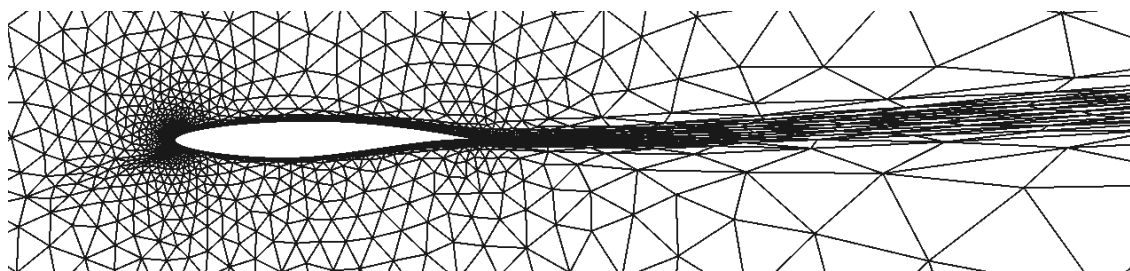
American Institute of Aeronautics and Astronautics
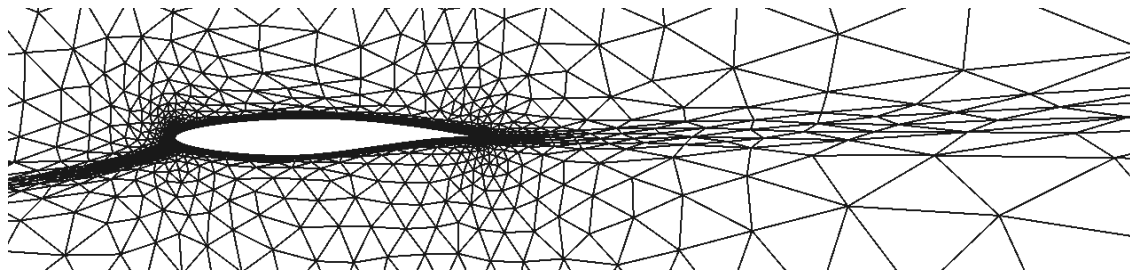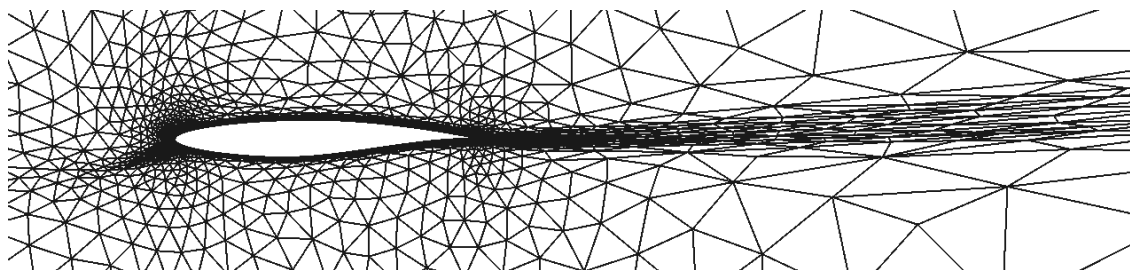
(a) Drag Adaptation
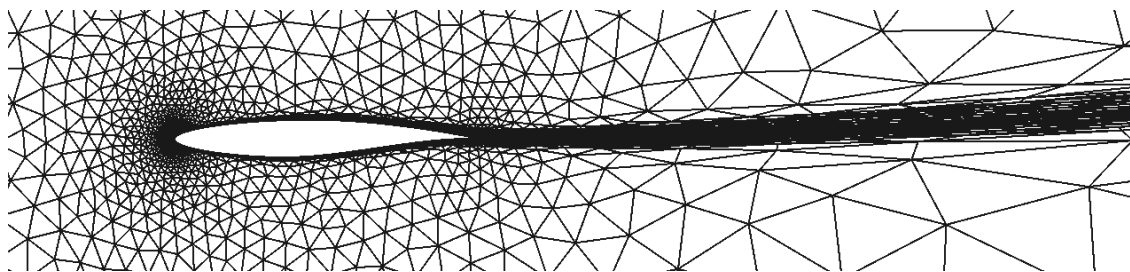
(b) Lift Adaptation

(c) Entropy Adaptation

(d) Drag-Entropy Adaptation

(e) Lift-Entropy Adaptation

Figure 18. RAE 2822 $M_\infty = 0.25$, $\alpha = 5°$, $Re = 6.5 \times 10^6$: **Meshes after final adaptation iteration (approximately 80000 DOF) for various error indicator strategies.**

American Institute of Aeronautics and Astronautics

Acknowledgements

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.

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

[9]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.

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

[11]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.

[12]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.

[13]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.

[14]Houston, P. and Süli, E., "Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics," *Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics*, edited by T. Barth and H. Deconinck, Springer-Verlag, Heidelberg, Lecture Notes in Computational Science and Engineering Vol 25, 2002, pp. 269–344.

[15]Rannacher, R., "Adaptive Galerkin finite element methods for partial differential equations," *Journal of Computational and Applied Mathematics*, Vol. 128, 2001, pp. 205–233.

[16]Barth, T. and Larson, M., "A posteriori error estimates for higher order Godunov finite volume methods on unstructured meshes," *Finite Volumes for Complex Applications III*, edited by R. Herban and D. Kröner, Hermes Penton, London, 2002, pp. 41–63.

[17]Solín, P. and Demkowicz, L., "Goal-oriented hp-adaptivity for elliptic problems," *Computer Methods in Applied Mechanics and Engineering*, Vol. 193, 2004, pp. 449–468.

[18]Venditti, D. A. and Darmofal, D. L., "Grid adaptation for functional outputs: application to two-dimensional inviscid flows," *Journal of Computational Physics*, Vol. 176, No. 1, 2002, pp. 40–69.

[19]Giles, M. B. and Süli, E., "Adjoint methods for PDEs: a posteriori error analysis and postprocessing by duality," *Acta Numerica*, Vol. 11, 2002, pp. 145–236.

[20]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.

[21]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.

[22]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.

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

[24]Fidkowski, K. J., Ceze, M. A., and Roe, P. L., "Entropy-Based Drag Error Estimation and Mesh Adaptation in Two Dimensions," *AIAA Journal of Aircraft*, Vol. 49, No. 5, September-October 2012, pp. 1485–1496.

[25]Oswatitsch, K., *Gas Dynamics*, Academic, New York, 1956.

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

[27]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.

[28]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.

[29]Hartmann, R., "Adjoint consistency analysis of Discontinuous Galerkin discretizations," *SIAM Journal on Numerical Analysis*, Vol. 45, No. 6, 2007, pp. 2671–2696.

[30]Oliver, T. A., *A High–order, Adaptive, Discontinuous Galerkin Finite Elemenet Method for the Reynolds-Averaged Navier-Stokes Equations*, Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, 2008.

[31]Spalart, P. R., "Airplane Trailing Vortices," *Annual Review of Fluid Mechanics*, Vol. 30, 1998, pp. 107–138.

[32]Allmaras, S., Johnson, F., and Spalart, P., "Modifications and Clarifications for the Implementation of the Spalart-Allmaras Turbulence Model," Seventh International Conference on Computational Fluid Dynamics (ICCFD7) 1902, 2012.