Entropy-Based Drag Error Estimation and Mesh Adaptation in Two Dimensions

Krzysztof J. Fidkowski*, Marco A. Ceze†, and Philip L. Roe‡

University of Michigan, Ann Arbor, MI 48109

This paper presents a method for estimating the drag error in two-dimensional computational fluid dynamics simulations using a method that does not require auxiliary adjoint solutions. The error of interest is that caused by the numerical discretization, including effects of finite mesh size and approximation order. The target output is a drag calculation based on a farfield integration of the entropy. The error estimate is motivated by an interpretation of entropy variables as adjoint solutions to an entropy balance output. As entropy variables are obtained by a direct transformation of the state, no separate adjoint solution is required. The method is shown to be applicable not only to inviscid and laminar flows, but also to turbulent Reynolds-averaged Navier-Stokes flows, for which entropy variables generally lose their symmetrization property. Since the proposed drag error estimate is based on an integral of a weighted residual over the computational domain, it can be localized to provide an adaptive indicator over each element. Adaptive results for several flows of aerodynamic interest show that the error estimate is effective and that the method performs on par with adjoint-based methods.

*Assistant Professor, AIAA Senior Member

†Graduate Research Assistant

‡Professor, AIAA Fellow

Nomenclature

Ω	=	computational domain
S_{body}	=	airfoil surface
S_∞	=	farfield boundary
$\vec{n}$	=	normal pointing out of the computational domain
$\vec{V}$	=	velocity
u_∞	=	freestream speed
$\hat{x}$	=	unit vector aligned with the freestream velocity
p	=	pressure
ρ	=	density
E	=	total energy per unit mass
c_p	=	specific heat at constant pressure
c_v	=	specific heat at constant volume
R	=	gas constant
γ	=	ratio of specific heats
s	=	entropy
$\underline{\tau}$	=	viscous stress tensor
M_∞	=	freestream Mach number
c	=	airfoil chord
D	=	drag force
c_d	=	drag coefficient
$\mathbf{u}$	=	conservative state variables
$\vec{\mathbf{f}}$	=	conservative flux
$\mathbf{r}$	=	Navier-Stokes residual operator
$\mathbf{v}$	=	entropy variables
U	=	entropy function
$\vec{F}$	=	entropy flux
K	=	constant relating entropy integral output to drag coefficient
$\mathbf{u}_H$	=	approximate state vector solution
$\tilde{\nu}$	=	Spalart-Allmaras turbulence model working variable
$\mathbf{U}_H$	=	discrete solution vector
$\mathbf{R}_H$	=	discrete residual vector
$\mathbf{\Psi}_H$	=	discrete adjoint vector
κ	=	mesh element
ϵ	=	output error indicator

I. Introduction

Numerical error due to finite-dimensional discretization is present in virtually all practical computational fluid dynamics (CFD) solutions. In large, complex, simulations this error can be difficult to control, especially if computational resources are already taxed. Solution-based adaptive methods address this problem by allocating degrees of freedom to areas where they are deemed necessary based on some automated interrogation of the solution in order to produce numerically accurate solutions at reduced computational cost. Even when these methods are used, however, solution accuracy is not guaranteed. One complication is that accuracy can be measured in different ways, for example through norms of the residual or solution error, or through errors in scalar outputs. Yet very few of the adaptive methods used today provide bounds or estimates of any of these measures of accuracy. Indeed, many heuristic methods have been shown, at least in certain cases, to produce remarkably incorrect solutions under one or more relevant accuracy measures.

A notable exception is output-based adaptation, in which an error estimate is computed for a targeted scalar quantity during the adaptive process. These methods have been applied successfully to aerodynamic flows in two and three dimensions,[1–5] resulting in robust solution procedures. The output error estimates rely on adjoint solutions, which although not prohibitively expensive computationally, do require code-intrusive changes and are not yet widely available.

Recent work by the authors investigated an alternative adaptation method based on the "entropy adjoint".[6] This method appealed to the observation that the entropy state variables act as an adjoint solution for an output that expresses an entropy balance statement in the computational domain. The residual weighted by the entropy variables, which is just the entropy residual, then provides an error estimate for this output. That is, we can compute the numerical error in this particular output without calculating a separate adjoint solution, as the entropy variables are calculated directly from the conservative or primitive state vector.

Adaptation driven by the entropy-adjoint approach was demonstrated to produce "all-around" good solutions that competed with engineering output adjoint solutions in terms of output accuracy for several representative aerodynamic cases. However, the entropy-adjoint method did not provide error estimates for these engineering outputs. Without the exact solution, we do not know whether the adaptation converged to the right value and when the adaptation could be terminated. The only measure of error in the entropy-adjoint approach is the error in the entropy balance output,

which is generally not of direct engineering interest.

In this work we propose a relationship between the entropy-adjoint error estimate and an estimate of one engineering output of interest: drag, which is the freestream aligned force on an object in an external flow. Drag is critical for analysis and design of aerospace vehicles, but it is notoriously difficult to predict. It is the subject of a large body of previous work,[7–15] including an ongoing workshop by the American Institute of Aeronautics and Astronautics.[16–19] The present work relies on the observation that drag in CFD, as in experiments, can be measured in more than one way: in particular via a near-field and via a farfield integration. The latter can be expressed in terms of entropy through a formula due to Oswatitsch,[20] and it is this formula that provides the basis for our relationship between drag and the entropy balance output. Moreover we show that for the cases considered, entropy-variable adaptation produces meshes in which the near-field and farfield drag errors are comparable, and we attribute this property to the conservative discretization and to the targeting of areas of spurious entropy generation by the entropy-variable indicator.

The organization of this paper is as follows. Section II presents the near-field and farfield drag expressions, including Oswatitsch's formula. Section III discusses the numerical error estimation procedure using engineering output adjoints and using entropy variables. The implementation in an adaptive finite-element solver is presented in Section IV, and results for inviscid, laminar, and turbulent flows in two dimensions are given in Section V.

II. Drag Calculation in Computational Fluid Dynamics

The drag force on an airfoil in an external flow governed by the steady Navier-Stokes equations can be calculated using a direct integration of the stress on the body surface,

$$D_{\text{near}} = \int_{S_{\text{body}}} [p\vec{n} - \vec{n} \cdot \underline{\tau}] \cdot \hat{x} \, dS. \tag{1}$$

Figure 1 illustrates the setup schematically for an airfoil body. As the subscript on D indicates, this is a near-field calculation; i.e. one performed locally on the airfoil surface.

By conservation of momentum in steady state, the drag on the body can also be computed

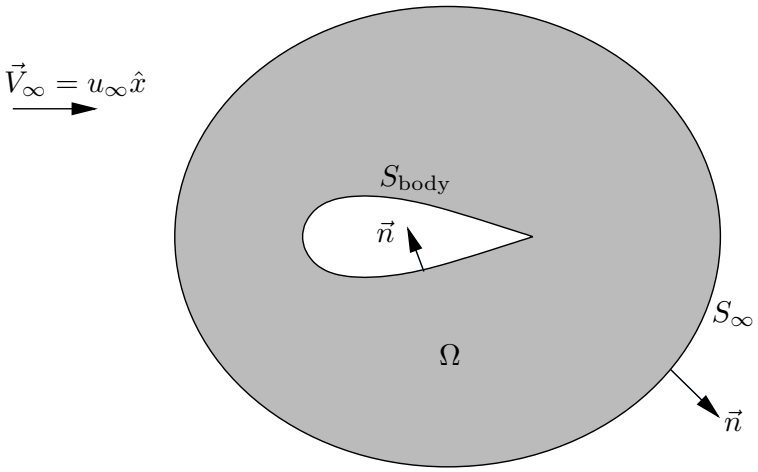


Figure 1. Schematic diagram for near- and farfield drag calculations.

without approximation via an integral on the farfield surface,

$$D_{\text{far}} = \int_{S_\infty} \left[-\rho(\vec{V} \cdot \vec{n})\vec{V} - p\vec{n} + \vec{n} \cdot \underline{\tau} \right] \cdot \hat{x} \, dS. \tag{2}$$

D_{near} and D_{far} will be identical for the exact solution and equivalent to machine precision for a discrete solution when using a conservative scheme in which the residual is converged to machine precision.

In the case of no trailing vortices, if S_∞ is sufficiently far from the body such that $p = p_\infty$ and the flow is parallel to the freestream, the farfield expression in Eqn. 2 can be re-written as,[11]

$$D_{\text{osw}} = \int_{S_\infty} u_\infty \left[1 - \sqrt{1 + \frac{2}{(\gamma - 1)M_\infty^2} \left(1 - e^{\Delta s / c_p} \right)} \right] \rho \vec{V} \cdot \vec{n} \, dS, \tag{3}$$

where u_∞ is the x-component of the freestream velocity. Note that we assume $\vec{V}_\infty = u_\infty \hat{x}$ in our case, as shown in Figure 1. The entropy change is measured relative to the freestream entropy,

$$\Delta s \equiv s - s_\infty, \quad s \equiv c_v \ln p - c_p \ln \rho.$$

Eqn. 3 will be referred to as the exact Oswatitsch expression.[20,21] It accounts for drag produced either by shocks or by boundary layers. When the farfield boundary is not sufficiently far from the body, correction terms can be added to account for "mid-field" changes in pressure and enthalpy.

However, these terms will not be considered in the present work.

Another difference between the drag values obtained using Eqns. 1 and 3 is due to numerical error. While in an exact solution the drag computed using Eqn. 3 is equal, up to the above assumptions, to that computed using Eqn. 1 or Eqn. 2, this thermodynamic equivalence does not necessarily hold for a discrete numerical solution. Specifically, most solution schemes do not conserve entropy, so that the entropy measured downstream of the body is polluted by spurious entropy generation. Eqn. 3 will then suffer the effects of this spurious entropy pollution.

Under the additional assumption that $\Delta s/R \ll 1$, a first-order Taylor-series expansion of Eqn. 3 yields the "approximate Oswatitsch" drag expression,[20]

$$D_{\text{osw}} \approx \frac{u_\infty}{\gamma R M_\infty^2} \int_{S_\infty} \Delta s \, \rho \vec{V} \cdot \vec{n} \, dS. \tag{4}$$

In this integral Δs can be replaced by s since the farfield surface is assumed to be closed and s_∞ is constant. Finally, in two dimensions, we consider the non-dimensional drag coefficient, defined as $c_d = D/\left(\frac{1}{2}\rho_\infty |\vec{V}_\infty|^2 c\right)$.

III. Drag Error Estimation

We are interested in estimating the drag error that results from solving the Navier-Stokes equations approximately, using computational fluid dynamics. The exact solution, $\mathbf{u}$, satisfies the Navier-Stokes equations, which we write as

$$\mathbf{r}(\mathbf{u}) \equiv \nabla \cdot \vec{\mathbf{f}}(\mathbf{u}) = 0. \tag{5}$$

Eqns. 1 and 2 can then be written as

$$\begin{aligned} D_{\text{near}}(\mathbf{u}) &= -\int_{S_{\text{body}}} \mathbf{e}_{\text{xmom}}^T \vec{\mathbf{f}}(\mathbf{u}) \cdot \vec{n} \, dS, \\ D_{\text{far}}(\mathbf{u}) &= \int_{S_\infty} \mathbf{e}_{\text{xmom}}^T \vec{\mathbf{f}}(\mathbf{u}) \cdot \vec{n} \, dS, \end{aligned}$$

where $\mathbf{e}_{\text{xmom}}$ is a vector of all zeros except for a 1 in the $x-$momentum state position. Again, we note that $D_{\text{near}}(\mathbf{u}) = D_{\text{far}}(\mathbf{u})$.

A discrete numerical solution, $\mathbf{u}_H$, obtained from computational fluid dynamics will not necessarily satisfy Eqn. 5 exactly, and it will not produce exact values for D_{near} and D_{far}. We consider two approaches to estimating the error in drag calculated using $\mathbf{u}_H$.

A. Output Adjoint Approach

The approximate state $\mathbf{u}_H$ satisfies a perturbed equation,

$$\mathbf{r}_H(\mathbf{u}_H) \equiv \mathbf{r}(\mathbf{u}_H) + \delta\mathbf{r} = 0, \tag{6}$$

for some $\delta\mathbf{r}$, where we assume small perturbations in the state and residual. The continuous adjoint $\boldsymbol{\psi}$ for a scalar output $J(\mathbf{u})$, e.g. drag, is the sensitivity of J to a residual source perturbation in the partial differential equation – i.e. to $\delta\mathbf{r}$ in Eqn. 6. In particular,

$$\delta J \equiv J(\mathbf{u}_H) - J(\mathbf{u}) = \int_\Omega \boldsymbol{\psi}^T \delta\mathbf{r}\, d\Omega, \tag{7}$$

where the superscript T denotes the transpose. $\boldsymbol{\psi}$ is obtained from an auxiliary linear equation that results from linearizing the governing equations and the output about the state.[5] Eqn. 7 with $\delta\mathbf{r}$ from Eqn. 6 provides the desired output error estimate.

In this work Eqn. 7 is applied to the near-field and Oswatitsch drag coefficient outputs. The respective adjoints are denoted by $\boldsymbol{\psi}_{\text{near}}$ and $\boldsymbol{\psi}_{\text{osw}}$. We note that by conservation of momentum, which holds even at the discrete level when a conservative scheme is used, the adjoint for the farfield drag output in Eqn. 2, $\boldsymbol{\psi}_{\text{far}}$, yields the same error estimate as the near-field drag adjoint. Specifically, the equivalence between the near-field and farfield drag adjoints can be derived by considering the effect of a residual perturbation $\delta\mathbf{r}$ on D_{near} and D_{far}. Integrating Eqn. 6 over the domain and using Eqn. 5 gives

$$\int_\Omega \delta\mathbf{r}\, d\Omega = -\int_\Omega \nabla\cdot\vec{\mathbf{f}}(\mathbf{u}_H)\, d\Omega = -\int_{S_{\text{body}}\cup S_\infty} \vec{\mathbf{f}}(\mathbf{u}_H)\cdot\vec{n}\, dS. \tag{8}$$

Taking the $x-$momentum component by dotting with $\mathbf{e}_{\text{xmom}}$ yields

$$\begin{aligned}
\int_\Omega \mathbf{e}_{\text{xmom}}^T \delta\mathbf{r}\, d\Omega &= -\int_{S_{\text{body}}} \mathbf{e}_{\text{xmom}}^T \vec{\mathbf{f}}(\mathbf{u}_H)\cdot\vec{n}\, dS - \int_{S_\infty} \mathbf{e}_{\text{xmom}}^T \vec{\mathbf{f}}(\mathbf{u}_H)\cdot\vec{n}\, dS \\
&= D_{\text{near}}(\mathbf{u}_H) - D_{\text{far}}(\mathbf{u}_H) \\
&= [D_{\text{near}}(\mathbf{u}_H) - D_{\text{near}}(\mathbf{u})] - [D_{\text{far}}(\mathbf{u}_H) - D_{\text{far}}(\mathbf{u})] \\
&= \delta D_{\text{near}} - \delta D_{\text{far}} \\
&= \int_\Omega \boldsymbol{\psi}_{\text{near}}^T \delta\mathbf{r}\, d\Omega - \int_\Omega \boldsymbol{\psi}_{\text{far}}^T \delta\mathbf{r}\, d\Omega.
\end{aligned}$$

Combining the final terms into one integral and requiring the expression to hold for arbirtrary $\delta\mathbf{r}$ gives $\boldsymbol{\psi}_{\text{far}} = \boldsymbol{\psi}_{\text{near}} - \mathbf{e}_{\text{xmom}}$, which states that the near-field and farfield drag adjoints are identical except for a constant unity offset in the $x-$momentum component. For an approximate solution $\mathbf{u}_H$ obtained from a conservative scheme, $\int_\Omega \delta\mathbf{r} d\Omega = 0$ by Eqn. 8, so that $\delta D_{\text{near}} = \delta D_{\text{far}}$ and the constant offset in the adjoint does not affect the error estimates. However, we note that the equivalence between error estimates obtained using $\boldsymbol{\psi}_{\text{far}}$ and $\boldsymbol{\psi}_{\text{near}}$ does not extend to the error estimate obtained using $\boldsymbol{\psi}_{h,\text{osw}}$ because entropy is generally not conserved at the discrete level.

B. Entropy Variable Approach

The Navier-Stokes equations admit an entropy function for which the corresponding entropy variables symmetrize both the inviscid and viscous terms.[22] This entropy function, unique up to additive and multiplicative constants, is

$$U = -\rho s/R, \qquad \text{where } s = c_v \ln p - c_p \ln \rho.$$

Differentiating with respect to the conservative state $\mathbf{u} = [\rho, \rho\vec{V}, \rho E]^T$ yields the entropy variables,

$$\mathbf{v} = U_{\mathbf{u}}^T = \left[\frac{\gamma}{\gamma-1} - \frac{s}{R} - \frac{1}{2}\frac{\rho V^2}{p},\ \frac{\rho\vec{V}}{p},\ -\frac{\rho}{p}\right]^T, \tag{9}$$

where $V^2 = \vec{V}\cdot\vec{V}$ is the square of the velocity magnitude. Note that the entropy variables are obtained via a nonlinear transformation of the conservative variables. The corresponding entropy

flux is $\vec{F} = \vec{V}U = -s\rho\vec{V}/R$.

As shown in,[6] for first-order conservation laws such as the Euler equations, the entropy variables $\mathbf{v}$ serve as an adjoint for the output

$$J^i \equiv \int_{\partial\Omega} \vec{F} \cdot \vec{n}\, dS,$$

which is zero for the exact solution, but non-zero for an approximate solution due to spurious entropy generation. For second-order conservation laws such as the Navier-Stokes equations, the output for which the entropy variables are an adjoint includes additional viscous dissipation terms. For general Reynolds-averaged Navier-Stokes (RANS) equations, the entropy variables do not necessarily serve as adjoint solutions. However, in all cases we can still write

$$\mathbf{v}^T \underbrace{\nabla \cdot \vec{\mathbf{f}}^i(\mathbf{u})}_{\mathbf{r}^i(\mathbf{u})} = \nabla \cdot \vec{F} \quad \Rightarrow \quad \int_\Omega \mathbf{v}^T \mathbf{r}^i(\mathbf{u})\, dS = \int_{\partial\Omega} \vec{F} \cdot \vec{n}\, dS = J^i, \tag{10}$$

where the superscript i denotes inviscid residuals or fluxes. We now make the observation that J^i, is directly related to the approximate form of Oswatitsch's drag formula, Eqn. 4,

$$D_{\text{osw}} \approx \frac{u_\infty}{\gamma R M_\infty^2} \int_{S_\infty} s\rho\vec{V} \cdot \vec{n}\, dS = -\frac{u_\infty}{\gamma M_\infty^2} \int_{S_\infty} \vec{F} \cdot \vec{n}\, dS = -\frac{u_\infty}{\gamma M_\infty^2} \int_{\partial\Omega} \vec{F} \cdot \vec{n}\, dS = -\frac{u_\infty}{\gamma M_\infty^2} J^i, \tag{11}$$

where in the second-to-last step the integral of $\vec{F} \cdot \vec{n}$ over the body, S_{body}, is zero due to no flow through the body. The drag coefficient is then approximated by

$$c_{d,\text{osw}} = \frac{D_{\text{osw}}}{\frac{1}{2}\rho u_\infty^2 c} \approx \underbrace{-\frac{2}{\rho u_\infty \gamma M_\infty^2 c}}_{K} J^i. \tag{12}$$

This is a direct relationship between J^i and an output of engineering interest: the drag coefficient measured using Oswatitsch's approximate formula. Note that the constant K depends only on freestream conditions. The error in the drag coefficient is then approximately

$$\delta c_{d,\text{osw}} \approx K\, \delta J^i = K \left(J^i(\mathbf{u}_H) - J^i(\mathbf{u}) \right) = K \int_\Omega \mathbf{v}^T \left(\mathbf{r}^i(\mathbf{u}_H) - \mathbf{r}^i(\mathbf{u}) \right) d\Omega. \tag{13}$$

In the inviscid case, $\mathbf{r}^i(\mathbf{u}) = 0$, and this formula becomes identical to the adjoint-weighted residual with $K\mathbf{v}$ in place of $\boldsymbol{\psi}$. That is, the output error obtained when using the entropy variables as adjoints corresponds approximately to $1/K$ times the output error in the drag coefficient computed using Oswatitsch's formula. Since the entropy variables are readily computable from the primal state, this error estimate has a key advantage that it does not require a separate adjoint solution.

In the general (viscous) case, $\mathbf{r}^i(\mathbf{u}) \neq 0$, and this quantity needs to be estimated in order to apply the error estimate in Eqn. 13. Specifically, the error estimate requires two inviscid residual evaluations, one with the approximate solution and one with the exact solution. In this work we investigate the performance of the error estimate when $\mathbf{u}$ is not known exactly since the error estimate becomes trivial when the exact state is available. The details are described in the next section.

We note that the present approach differs from previous entropy-residual approaches that decompose drag into different sources, one of them being drag due to spurious entropy generation.[11,12,15,23] The previous approaches rely on a subdivision of the domain into three regions: one in the vicinity of shocks, one in boundary layers or wakes, where viscous dissipation is active, and the rest of the domain where no entropy generation is expected. The spurious drag is then estimated via a volume integral over this latter portion of the domain. The success of this approach relies on accurate sensors for shocks and boundary layers/wakes. Furthermore spurious entropy generation in the shock or viscous regions is not measured by the decomposition method. This contrasts with the present approach which does not require subdivision of the computational domain and which estimates error in drag due to spurious entropy production throughout the domain.

IV. Discretization

A. Primal Solution

We consider the compressible Navier-Stokes equations, Reynolds-averaged using the Spalart-Allmaras turbulence model,

$$\nabla \cdot \mathbf{f}^i(\mathbf{u}) - \nabla \cdot \mathbf{f}^v(\mathbf{u}, \nabla\mathbf{u}) + \mathbf{s}^t(\mathbf{u}, \nabla\mathbf{u}) = 0, \tag{14}$$

where $\mathbf{u} = [\rho, \rho\vec{V}, \rho E, \rho\tilde{\nu}]^T$ is the conservative state vector, now augmented with the turbulent viscosity working variable, $\mathbf{f}^i/\mathbf{f}^v$ are the inviscid/viscous fluxes, and $\mathbf{s}^t$ contains the turbulent source terms. We discretize Eqn. 14 using a discontinuous Galerkin (DG) finite element method that employs the Roe approximate Riemann solver[24] for the inviscid flux, the second form of Bassi and Rebay for the viscous flux,[25] and turbulence model modifications due to Allmaras and Oliver.[26] The solution is obtained via a Newton-GMRES implicit solver with element-line Jacobi preconditioning and local pseudo-time stepping. While a DG finite element method was used in this work, the conclusions are not strictly tied to the discretization.

The result of the discretization is a system of nonlinear algebraic equations,

$$\mathbf{R}_H(\mathbf{U}_H) = 0, \tag{15}$$

where entries in the discrete vector $\mathbf{U}_H$ are coefficients of order p basis functions used to approximate the state field $\mathbf{u}_H$. Eqn. 15 will be referred to as the primal system.

B. Adjoint Solution

The output error estimation strategy relies on adjoint solutions, which are estimated using a discrete adjoint approach. That is, $\boldsymbol{\psi}_H$ corresponding to an output of interest $J(\mathbf{u}_H)$ is obtained by solving the linearized transpose system,

$$\left(\frac{\partial \mathbf{R}_H}{\partial \mathbf{U}_H}\right)^T \boldsymbol{\psi}_H + \left(\frac{\partial J}{\partial \mathbf{U}_H}\right)^T = 0, \tag{16}$$

where the linearizations are performed about $\mathbf{u}_H$. The same element-line Jacobi preconditioned GMRES solver used in the primal solve is used for the linear adjoint solve. When using entropy variables, the entropy variable corresponding to the turbulent working variable $\rho\tilde{\nu}$ is set to zero.

C. Error Estimation

The drag output error estimation formula in Eqn. 7 is based on an exact continuous adjoint solution, $\boldsymbol{\psi}$, which is generally unavailable. Similarly, when using entropy variables to weight the residual in Eqn. 10, the exact $\mathbf{v}$ is not known – if we knew it we would know the exact state $\mathbf{u}$. Thus,

to make the error estimation tractable, we resolve to compare the discrete output $J(\mathbf{u}_H)$ to the output calculated from a "fine" solution, $\mathbf{u}_h$, on a richer discretization space. In a discontinuous Galerkin discretization, the solution space can be enriched by increasing the approximation order p and/or by refining the elements, and in this work we consider both options.

For our Galerkin finite element discretization, the discrete analogue of Eqn 7 is[5]

$$\delta J \approx \mathbf{\Psi}_h^T \mathbf{R}_h(\mathbf{U}_h^H), \tag{17}$$

where $\mathbf{U}_h^H$ is the discrete coarse-space state vector (coefficients of basis functions) injected into the fine space, and $\mathbf{\Psi}_h$ is the discrete fine-space adjoint vector. When using entropy variables, the discrete analogue of Eqn. 13 is

$$\delta c_{d,\text{osw}} \approx K \mathbf{V}_h^T \left[\mathbf{R}_h^i(\mathbf{U}_h^H) - \mathbf{R}_h^i(\mathbf{U}_h) \right], \tag{18}$$

where the superscript i denotes contributions to the discrete residual from the inviscid terms, and $\mathbf{U}_h$ and $\mathbf{V}_h$ are the discrete fine-space conservative state and entropy variable vectors, respectively. These are related directly through Eqn. 9. In practice, a coarse-space adjoint or entropy variable solution, injected into the fine space, is subtracted from $\mathbf{\Psi}_h^T$ or respectively $\mathbf{V}_h^T$ in the above equations to reduce errors due to non-zero iterative convergence of the residual.[5]

Crucial to the error estimates in Eqn. 17 and 18 are the fine-space state and adjoint solutions, and these can be obtained in several ways. The fine space adjoint can be reconstructed via least-squares or solved approximately or exactly in the fine space. In this work we approximate the fine space adjoint with $\nu_{\text{fine}} = 20$ iterations of element-block Jacobi relaxation on the fine space after initialization with the coarse-space adjoint. Tests with exact adjoint solutions show nearly identical results for the cases of interest in the present work. When using entropy variables, we apply this relaxation to the fine-space primal problem. We note that a simpler approach of using the drag from the approximate fine-space solution to calculate the output error generally gives very erroneous results, in contrast to the above estimate that is based on a residual weighted by an adjoint interpolation error.[27]

The estimates in Eqn. 17 and 18 are not bounds, and on under-resolved meshes, the estimated

error often under-predicts the true output error. To address this under-prediction, we make the error estimate more conservative by introducing absolute value signs that prevent cancellation of error between different mesh elements and between different conservation equations. The modified error estimates, used to monitor adaptive convergence, are

$$\epsilon^{\boldsymbol{\psi}} \equiv \sum_{\kappa} \sum_{j} \left| \left[\boldsymbol{\Psi}_h^T \mathbf{R}_h(\mathbf{U}_h^H) \right]_{\kappa,j} \right| \qquad \text{(output adjoints)}, \tag{19}$$

$$\epsilon^{\mathbf{v}} \equiv \sum_{\kappa} \sum_{j} \left| \left[K \mathbf{V}_h^T \left(\mathbf{R}_h^i(\mathbf{U}_h^H) - \mathbf{R}_h^i(\mathbf{U}_h) \right) \right]_{\kappa,j} \right| \qquad \text{(entropy variables)}, \tag{20}$$

where j indexes the conservation equations and the notation $[\cdot]_{\kappa,j}$ refers to restriction to degrees of freedom associated with element κ and equation j. When element subdivision is used for the fine space, error indicators for each sub-element are summed together to obtain the error indicator for each original element.

D. Mesh Adaptation

The output error estimate drives an adaptive process in which the problem is solved multiple times on successively refined meshes. The iterative process begins with a primal solution, and if necessary an adjoint solution, on a coarse mesh. The drag error is estimated using Eqn. 19 or 20, and if the error is below a specified tolerance, the iterative process terminates. Otherwise, the drag error is localized to the elements and the mesh is refined as described below.

From Eqns. 19 and 20, the local contribution of element κ to the total error is

$$\epsilon_{\kappa}^{\boldsymbol{\psi}} = \sum_{j} \left| \left[\boldsymbol{\Psi}_h^T \mathbf{R}_h(\mathbf{U}_h^H) \right]_{\kappa,j} \right| \qquad \text{(output adjoints)}, \tag{21}$$

$$\epsilon_{\kappa}^{\mathbf{v}} = \sum_{j} \left| \left[K \mathbf{V}_h^T \left(\mathbf{R}_h^i(\mathbf{U}_h^H) - \mathbf{R}_h^i(\mathbf{U}_h) \right) \right]_{\kappa,j} \right| \qquad \text{(entropy variables)}. \tag{22}$$

Results will also be shown for an unweighted-residual indicator, which takes the same form as the above error estimates but without the adjoint or entropy-variable term.

$$\epsilon_{\kappa}^{\boldsymbol{r}} = \sum_{j} \left\| \left[\mathbf{R}_h(\mathbf{U}_h^H) \right]_{\kappa,j} \right\| \qquad \text{(unweighted residual)}. \tag{23}$$

In this expression, $\|\cdot\|$ denotes the L_1 norm: absolute value signs are placed on every component in the residual vector to prevent cancellation when the coarse space residual is zero.

The above element error indicators are used to drive a fixed-fraction, isotropic, hanging-node adaptation strategy, in which a fraction $f^{\text{adapt}} = 0.1$ of the elements with the largest adaptive indicators are marked for refinement. Marked elements are adapted isotropically, with a maximum difference of one level of refinement between adjacent elements.

The steps involved in each adaptation iteration can be summarized as follows:

1. Solve the primal problem on the current mesh at order p to obtain $\mathbf{u}_H$. If adapting using a drag adjoint, solve the adjoint problem to obtain $\boldsymbol{\psi}_H$.
2. Inject $\mathbf{u}_H$ into a fine space, denoted by the subscript h.
3. If adapting using a drag adjoint, inject $\boldsymbol{\psi}_H$ in to the fine space and iterate the fine-space adjoint problem using element-block relaxation. If adapting using entropy variables, iterate $\mathbf{u}_H$ instead and calculate $\mathbf{v}_h(\mathbf{u}_h)$ using Eqn. 9.
4. Calculate the adaptive indicator, ϵ_κ, for each element using either the output adjoint or the entropy variable approach.
5. Refine a fraction f^{adapt} of the elements with the largest indicator.
6. Initialize the solution on the adapted mesh with a projection of $\mathbf{u}_H$ and return to step 1.

Element subdivision in the fine space is used during calculation of the error indicator in RANS simulations for improved robustness. To avoid uniformly refining the entire mesh at once, the fine space is constructed individually for each element using local meshes that consist of the element and its immediate neighbors.[28] Fine-space residual evaluation is performed on this local mesh.

V. Results

The following results present three demonstrations of error estimation and adaptation using the following methods:

- *Unweighted residual*: adaptation on the indicator in Eqn. 23.
- *Uniform refinement*: uniform h-refinement of the mesh.
- *Near-field or Oswatitsch drag adjoint*: adjoint-based adaptation and error estimation for either the near-field or the Oswatitsch drag output, using Eqn 21.

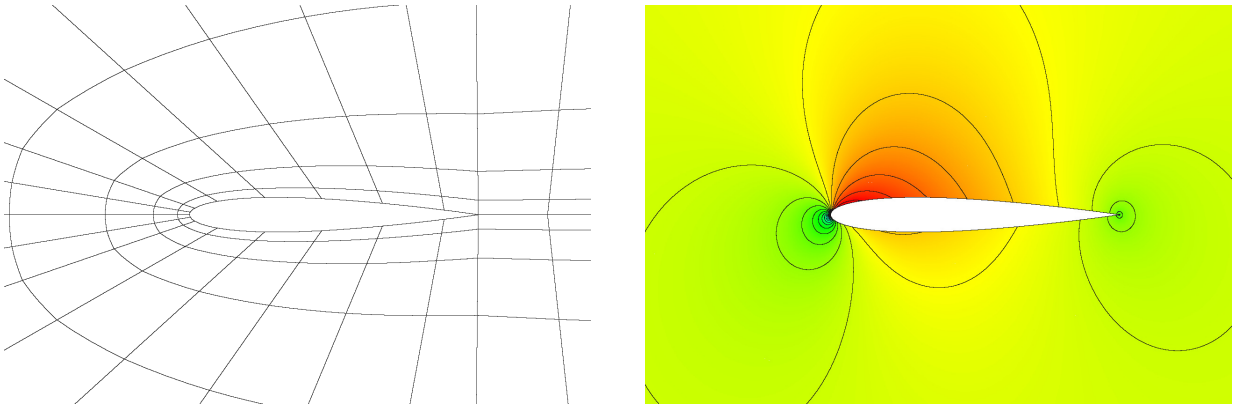
- *Entropy variables*: adaptation and error estimation using the inviscid residual error weighted by the entropy variables and the constant K, as in Eqn. 22.

The cases range from Euler to RANS simulations. Metrics of comparison include accuracy of the error estimate and efficiency of the adaptation. Accuracy of the error is measured relative to "exact" output values, which are obtained from solutions on uniformly-refined versions of the final output-adapted meshes, using order $p+1$ approximation. As the solver has not been tuned, efficiency is measured using degrees of freedom per equation instead of computational time. This favors the output-adjoint methods as the cost of the adjoint solve is not accounted for.

A. Inviscid Flow

The first example is inviscid, subcritical flow over a NACA 0012 airfoil. Although drag prediction for this flow is of no great engineering consequence, as we expect it to be zero, discretized simulations will yield some spurious drag and the accuracy to which the above methods predict the correct drag error is of interest in verifying the proposed approach.

The airfoil geometry for this example has a closed trailing edge, and the farfield is approximately 100 chord-lengths away from the airfoil. The initial mesh is illustrated in Figure 2 with Mach number contours in the near-field view. This mesh consists of quadrilaterals, with quartic ($q=4$) geometry representation. The initial structure of the mesh disappears with the first hanging-node adaptation iteration and the mesh storage is always fully unstructured. The following results use $p=2$ solution approximation and a fine-space of order $p+1$ for error estimation.

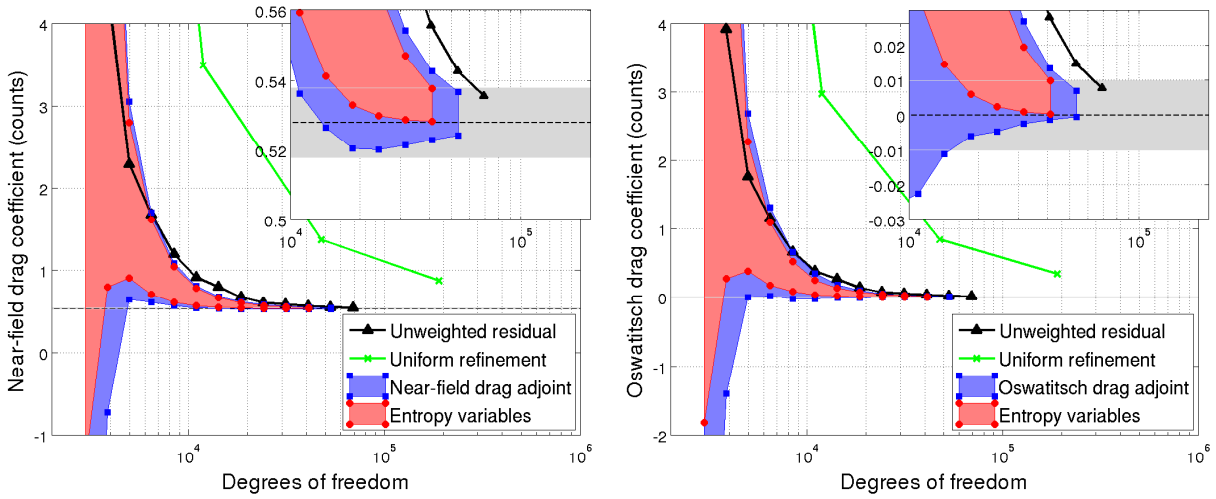


(a) Initial mesh: 330 elements

(b) Mach number contours (range is 0 to 0.73)

Figure 2. NACA 0012 $M=0.5$, $\alpha=2^o$, inviscid: initial mesh and Mach number contours.

The freestream Mach number for this case is $M_\infty = 0.5$, and the angle of attack is $\alpha = 2^o$. Stagnation quantities and flow direction are specified on the inflow boundary, and static pressure is specified on the outflow boundary. Figure 3 shows the convergence of the drag coefficient for the different adaptive strategies. For the adjoint- and entropy-variable-based methods, numerical error



(a) Near-field drag comparison

(b) Oswatitsch drag comparison

Figure 3. NACA 0012 $M = 0.5$, $\alpha = 2^o$, inviscid: convergence of near-field and Oswatitsch drag outputs for different adaptation schemes. Dashed lines indicate the true drag values and the gray shading corresponds to $\pm$ 0.01 drag count.

bars are included in the results as shaded regions that range $\pm\epsilon^{\psi}$ or $\pm\epsilon^{\mathbf{v}}$ around the computed outputs. The Oswatitsch drag asymptotes to zero as expected for this inviscid, subcritical, two-dimensional run. However, the near-field drag bottoms out at approximately 0.53 drag counts, and this is due to the finite distance of the farfield boundary from the airfoil. An output tolerance of 0.01 drag counts is illustrated by a gray shaded band around the exact output line.

For both drag measures, uniform refinement makes progress towards the exact drag value, but it does so quite slowly. One drag count error is not achieved until about 50,000 degrees of freedom for either drag measure. Next, the unweighted residual indicator converges at a rate that is only slightly worse than the weighted-residual methods. However, it does not come equipped with a drag error indicator. The entropy variable and adjoint-based weightings give nearly identical convergence results, with slightly less-conservative error estimates for the entropy variable method – neither estimate bounds the true error throughout the computation.

We note the similarity in cost for adaptive runs using the near-field and the Oswatitsch drag expressions in Figure 3. This is not intuitive since the farfield is quite distant from the airfoil,

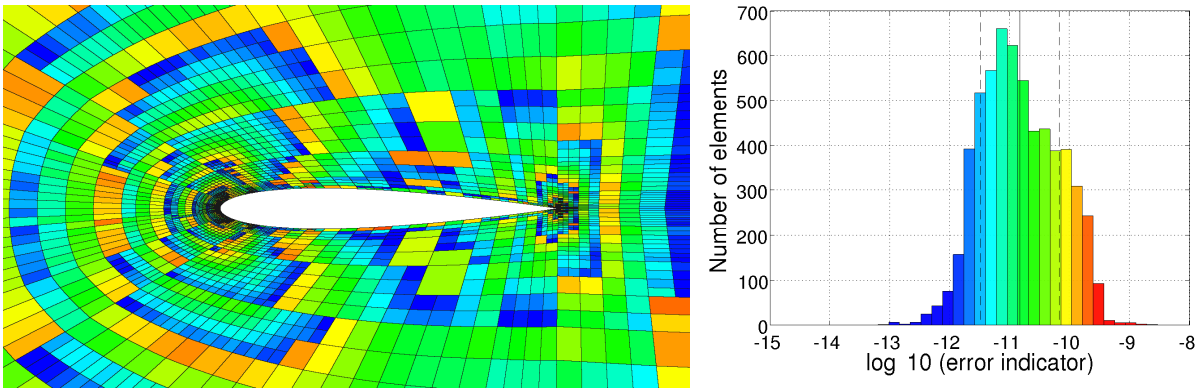
and keeping spurious entropy production low throughout the entire domain seems more challenging than keeping the solution accurate in the near-field. The results show that in this case, without a wake, it is not expensive to keep the spurious entropy production low throughout the domain, at least using a high-order DG method. The next examples will investigate cases with viscous wakes.

Figure 4 shows meshes, error maps, and error histograms for final meshes adapted using the different weighted-residual methods. These are the 11^{th} adaptive iteration for the adjoint methods and the 10^{th} adaptive iteration for the entropy variable method. The areas targeted for refinement are, as expected, the leading and trailing edges in all three methods. The error histograms and maps are also similar – we see that the error is fairly-well equidistributed throughout the computational domain.

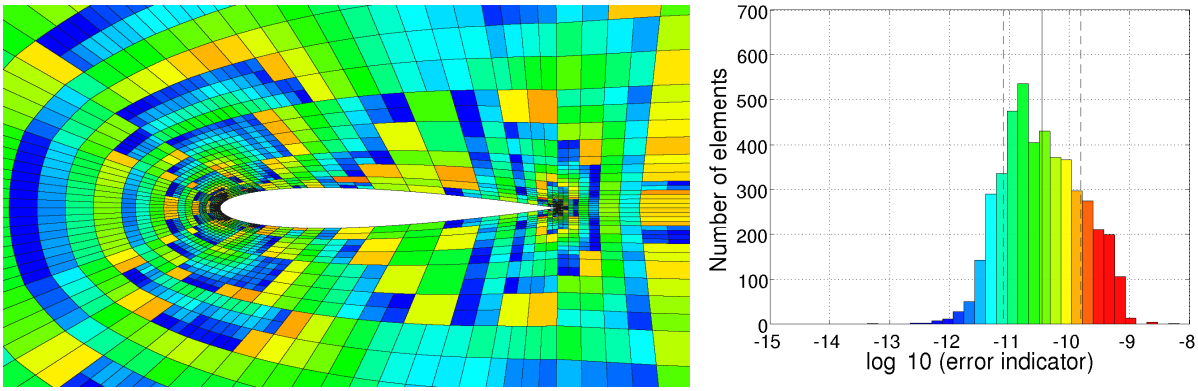
B. Laminar Flow

The second example consists of a NACA 0012 airfoil in viscous flow at $M_\infty = 0.5$, $Re = 5000$, and $\alpha = 2^o$. The initial mesh and Mach number distribution are illustrated in Figure 5. The farfield boundary for this case is located approximately 2000 chord-lengths away from the airfoil in order to assess the degree to which the Oswatitsch and entropy-variable indicators target the wake. The solution approximation order is $p = 2$, and the fine space for error estimation consists of an order increment to $p + 1 = 3$.

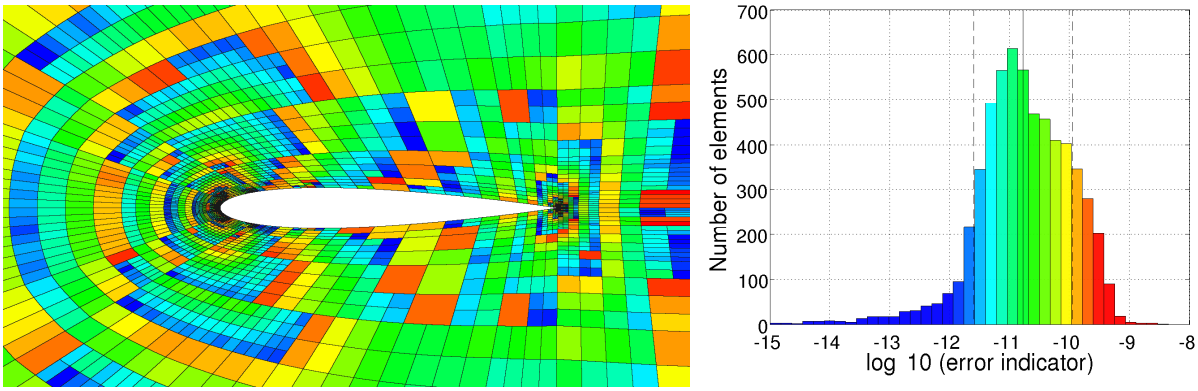
Figure 6 shows the convergence of the near-field and Oswatitsch drag coefficients calculated using the different adaptive strategies. The difference between the exact near-field and Oswatitsch drag values is about 0.4 counts. Considering first uniform refinement, we see that the near-field drag converges more quickly than the Oswatitsch drag. We attribute this effect to the poor wake resolution of the initial mesh. On the other hand, the adaptive methods perform comparably. In particular, the unweighted residual converges quickly, but again it does not offer an error estimate. It also exhibits oscillations when converging to the true Oswatitsch value. Indicators based on the Oswatitsch adjoint and on the entropy variables perform nearly identically in terms of convergence and error estimation. The near-field drag error estimates exhibit faster convergence than estimates based on the Oswatitsch adjoint and on the entropy variables. Specifically, the near-field drag error estimates are about half the Oswatitsch/entropy-variable error estimates for the same degrees of freedom. This effect is due to wake resolution requirements for the farfield entropy-based methods.



(a) Adapted using near-field drag adjoint, 5951 elements.

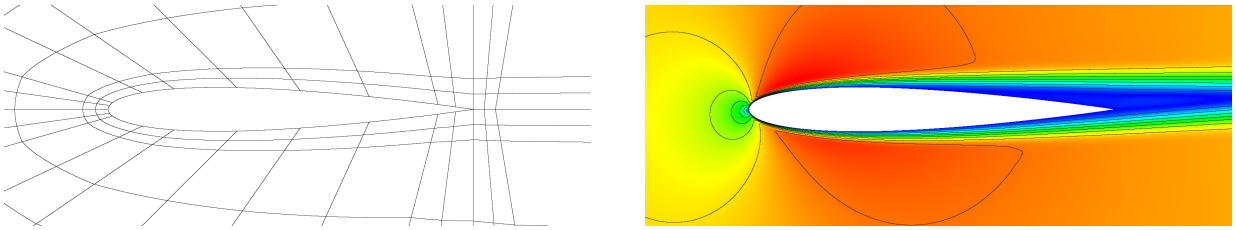


(b) Adapted using entropy variables, 4577 elements.



(c) Adapted using Oswatitsch drag adjoint, 5931 elements.

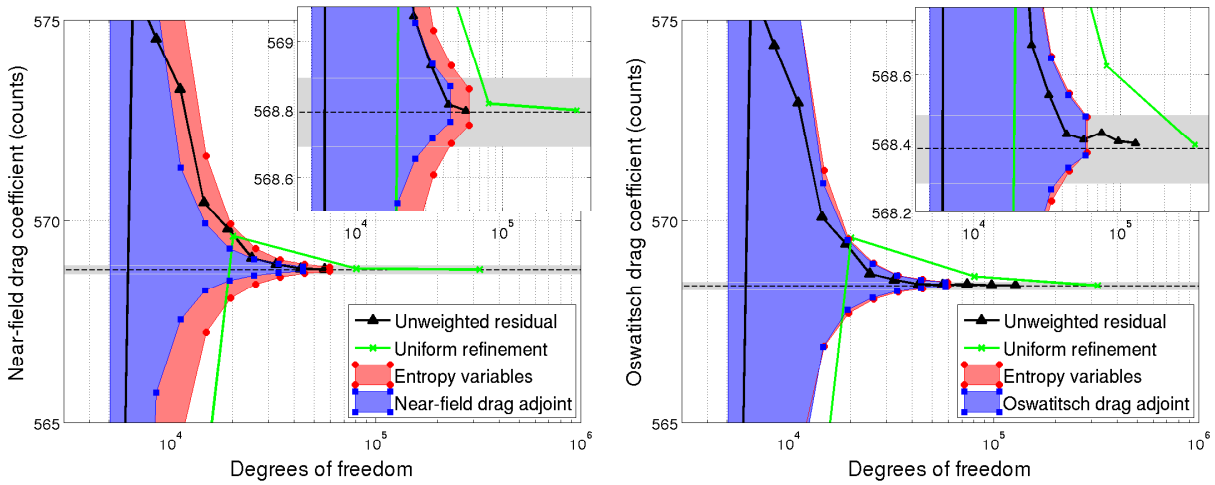
Figure 4. NACA 0012 $M = 0.5$, $\alpha = 2^o$, inviscid: final adapted meshes with error indicators and error histograms. The indicator color range is the same for all plots and consistent with the color on the histogram. In the error distribution, the solid line indicates the mean and the dashed lines indicate the standard deviation.



(a) Initial mesh: 560 elements (b) Mach number contours

Figure 5. NACA 0012 $M = 0.5$, $\alpha = 2^o$, $Re = 5,000$: initial mesh and Mach number contours.

However, the difference in performance is not very large considering that the farfield boundary is over 2000 chords away in this case.

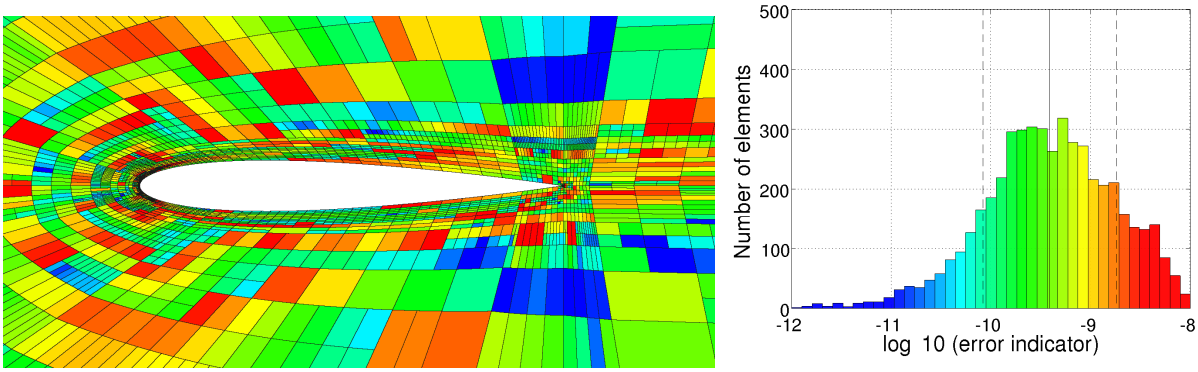


(a) Near-field drag comparison (b) Oswatitsch drag comparison

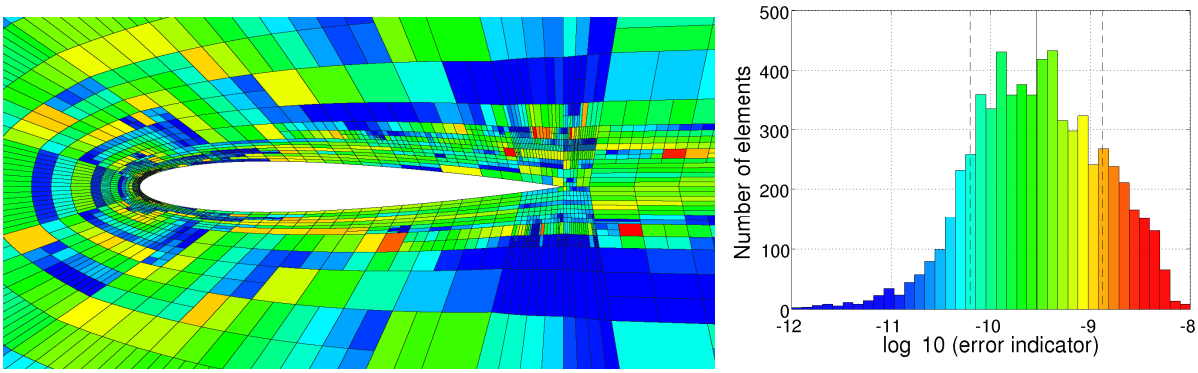
Figure 6. NACA 0012 $M = 0.5$, $\alpha = 2^o$, $Re = 5,000$: convergence of near-field and Oswatitsch drag outputs for different adaptation schemes. Dashed lines indicate the true drag values.

Final adapted meshes are shown in Figure 7. These are the 8^{th} iteration of the near-field drag adaptation, and the 9^{th} iterations of the Oswatitsch and entropy-variable adaptations. The areas targeted for refinement are similar between the three meshes: the leading and trailing edges, the edge of the boundary layer in the attached and separated regions, and the wake. The error histograms are also similar; the lower element counts in the near-field adaptation are due to a fewer total number of elements. The error maps are comparable between the near-field and the Oswatitsch adjoint-adapted meshes; the entropy-variable adapted mesh shows somewhat fewer high-error elements in the near-field.

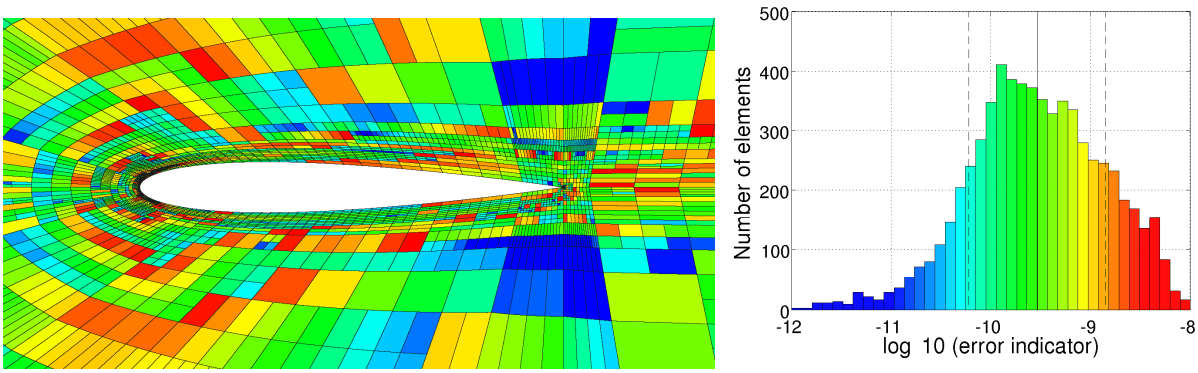
Figure 8 shows farfield views of the initial and adapted meshes. The entropy-variable and



(a) Adapted using near-field drag adjoint, 4893 elements.



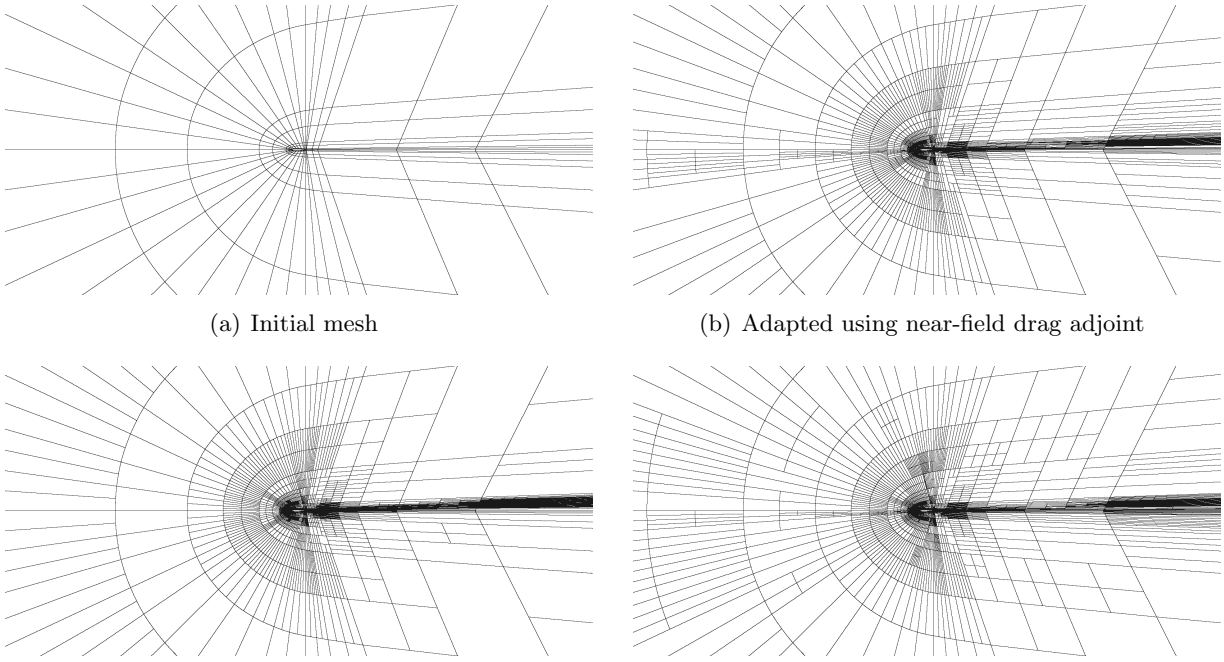
(b) Adapted using entropy variables, 6611 elements.



(c) Adapted using Oswatitsch drag adjoint, 6458 elements.

Figure 7. NACA 0012 $M = 0.5$, $\alpha = 2^o$, $Re = 5,000$**: final adapted meshes with error indicators and error histograms. The indicator color range is the same for all plots and consistent with the color on the histogram. In the error distribution, the solid line indicates the mean and the dashed lines indicate the standard deviation.**

Oswatitsch-adjoint approaches both target the wake, which is important for accurate entropy propagation to the farfield boundary. However, we note that the resolution of the wake does not preclude refinement near the airfoil, which is of interest as any spurious entropy production there would pollute the solution downstream. Both of the output adjoint strategies allocate some elements to the stagnation streamline region well in front of the airfoil, where the adjoint has a weak singularity,[29] whereas the entropy variable strategy leaves this area relatively coarse.



(a) Initial mesh

(b) Adapted using near-field drag adjoint

(c) Adapted using entropy variables

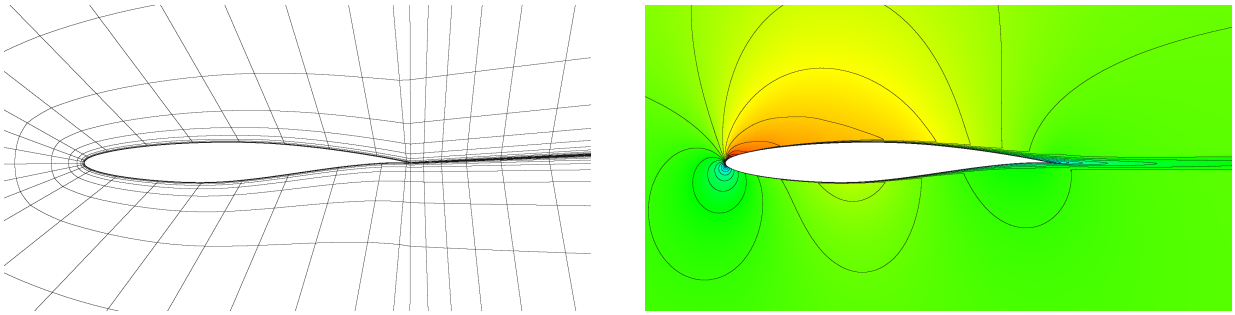
(d) Adapted using Oswatitsch drag adjoint

Figure 8. NACA 0012 $M = 0.5$**,** $\alpha = 2^o$**,** $Re = 5,000$**: farfield view of adapted meshes.**

C. Reynolds-Averaged Turbulent Flow

The final example consists of a RAE 2822 airfoil in viscous flow at $M_\infty = 0.6$, $Re = 6.3 \times 10^6$, and $\alpha = 2.57^o$. The initial mesh and Mach number distribution are illustrated in Figure 9. The farfield boundary for this case is located approximately 200 chord-lengths away from the airfoil. The solution approximation order is $p = 2$, and the fine space for error estimation consists of an order increment to $p+1 = 3$ combined with isotropic h-refinement of the elements when evaluating the residual.

Figure 10 shows the convergence of the near-field and Oswatitsch drag coefficients calculated using the different adaptive strategies. In this case, the difference between the exact near-field



(a) Initial mesh: 1610 elements

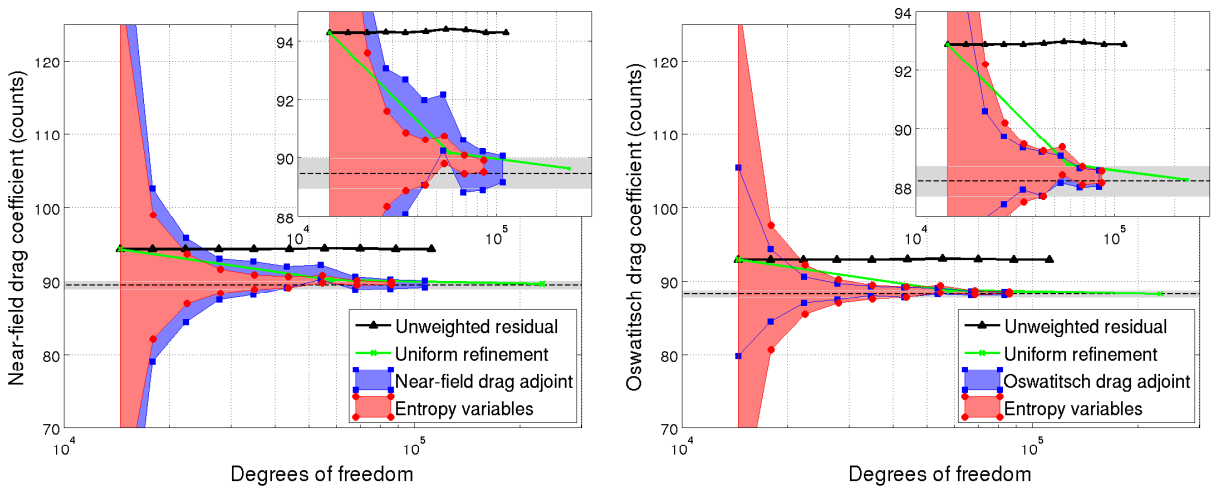
(b) Mach number contours (range is 0 to 1.11)

Figure 9. RAE 2822 $M = 0.6$, $\alpha = 2.57^o$, $Re = 6.3 \times 10^6$: initial mesh and Mach number contours.

and Oswatitsch drag values is about 1 count. Uniform refinement performs well, and this is due to a reasonably-well constructed initial mesh with high resolution in the boundary layer, wake, and trailing-edge regions. However, uniform refinement does not directly yield an error estimate; Richardson extrapolation could possibly be performed, but the fine mesh would be fairly expensive. Next, unweighted residual adaptation performs very poorly in this case. The indicator targets the wake and trailing-edge regions without touching the leading-edge, causing large errors in drag to remain during the adaptation.

On the other hand, the weighted residual methods perform much better. As in the previous example, error estimates obtained using the Oswatitsch drag adjoint are very close to those obtained using entropy variables. In this case, however, the Oswatitsch and entropy-variable adaptations exhibit faster error convergence than the near-field drag adjoint adaptation. We see that these error estimates are not always bounds; for example around 5.3×10^4 degrees of freedom, the near-field adjoint and entropy variable methods under-predict the near-field drag. Numerical tests via exact fine space solves show that this under-prediction is due to limited resolution of the fine space and not the approximate block-Jacobi relaxation solver on the fine space. Tests were also performed with pure $p + 1$ fine spaces, and the error under-prediction was worse in these cases. The combination of one level of h-refinement with $p + 1$ was found to be necessary for accurate error estimates.

Final adapted meshes and error histograms are shown in Figure 11. These are the 9^{th} adaptive iteration of the near-field drag adaptation, and the 8^{th} iterations of the Oswatitsch and entropy-variable adaptations. The areas targeted for refinement are similar between the three meshes: the supersonic flow region on the upper surface immediately after the leading edge, the outer edge of the boundary layers, the trailing edge, and the wake for at least several chord-lengths away from



(a) Near-field drag comparison

(b) Oswatitsch drag comparison

Figure 10. RAE 2822 $M = 0.6$, $\alpha = 2.57^o$, $Re = 6.3 \times 10^6$: convergence of near-field and Oswatitsch drag outputs for different adaptation schemes. Dashed lines indicate the true drag values.

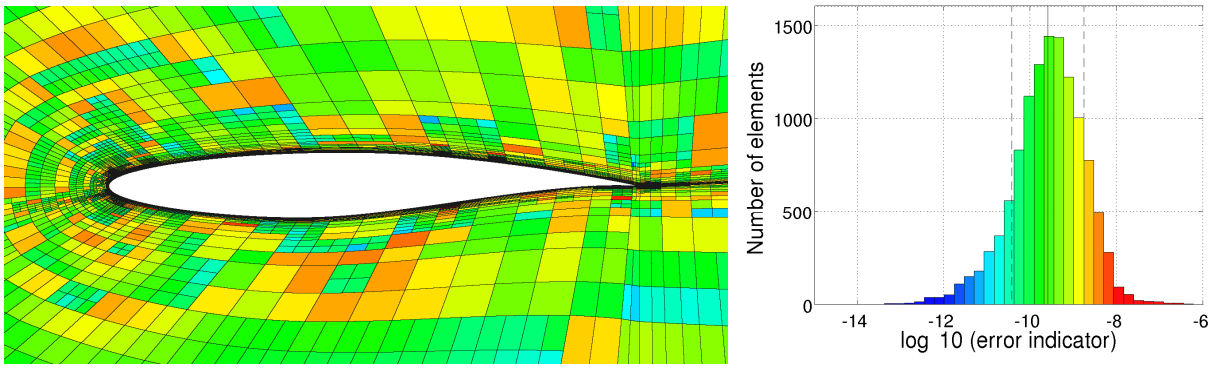
the airfoil. The error histograms and error maps are similar for the indicators based on the entropy-variables and the Oswatitsch adjoint. The latter has a fatter tail on the low error indicator side, causing a larger standard deviation. The histogram of the near-field drag adjoint indicator is of similar shape but shifted towards higher errors, even with a larger number of total elements. The error map indicates that this error is not confined to any particular region, but rather distributed throughout the domain.

Finally, Figure 12 shows zoomed-out views of the initial and adapted meshes. The wake is already fairly well-refined, but it is still targeted for refinement, most noticeably by the entropy-variable approach. As in the laminar example, we note that the resolution of the wake does not preclude refinement near the airfoil, which is similar for all three indicators. We see again that both of the output adjoint strategies refine the leading-edge stagnation streamline, while the entropy variable strategy leaves this area relatively coarse.

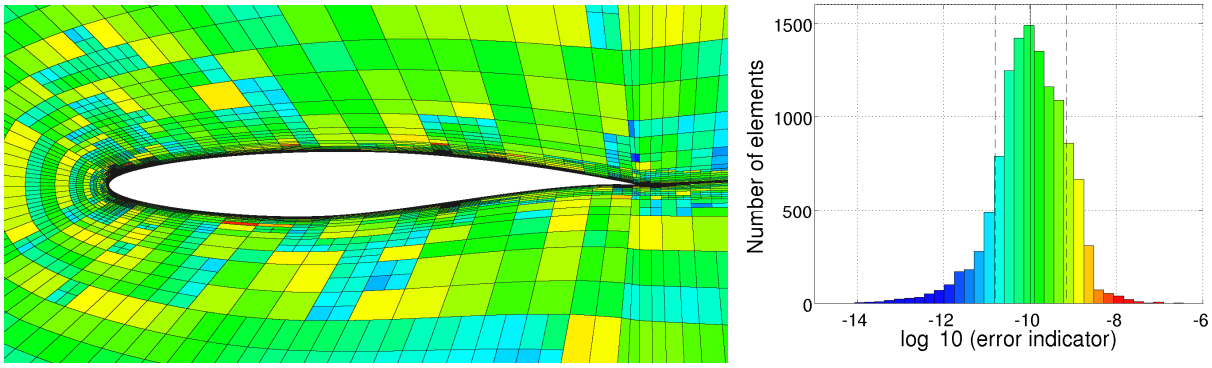
VI. Conclusions

A. Summary

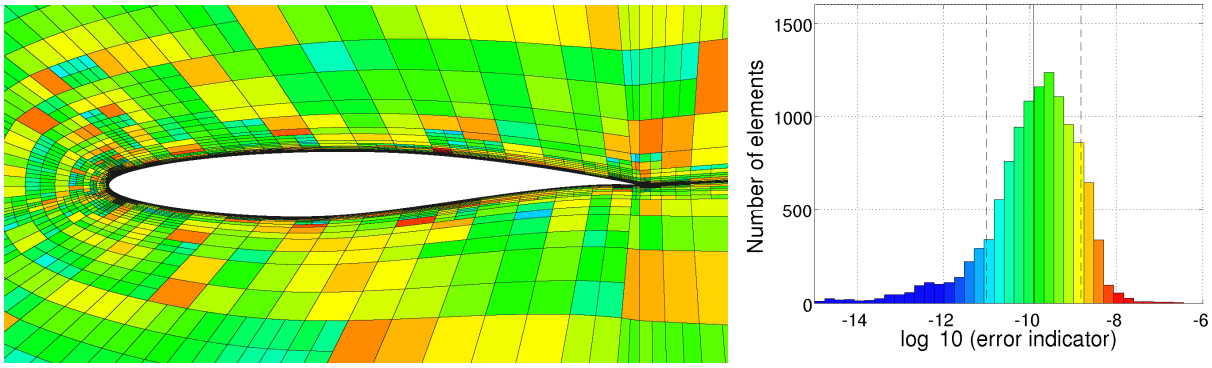
This paper presents a relationship between the error estimate computed using an entropy-variable weighted residual and the drag error in two-dimensional simulations of compressible Navier-Stokes



(a) Adapted using near-field drag adjoint, 11920 elements.

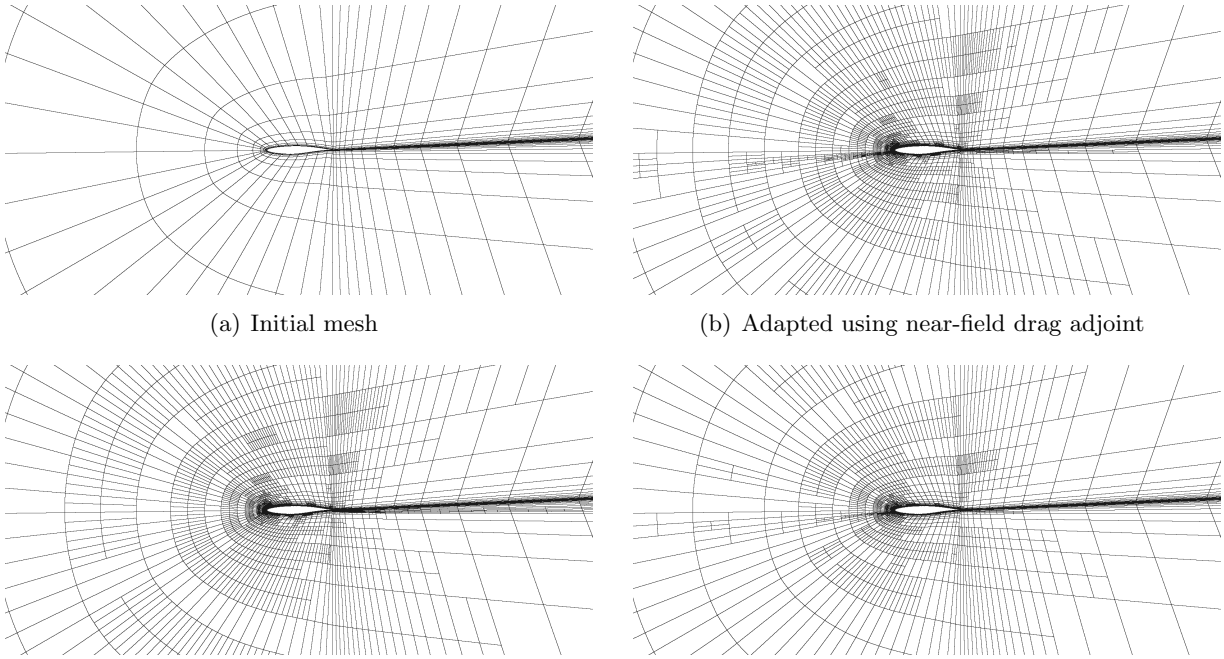


(b) Adapted using entropy variables, 9568 elements.



(c) Adapted using Oswatitsch drag adjoint, 9255 elements.

Figure 11. RAE 2822 $M = 0.6$, $\alpha = 2.57^o$, $Re = 6.3 \times 10^6$**: final adapted meshes with error indicators and error histograms. The indicator color range is the same for all plots and consistent with the color on the histogram. In the error distribution, the solid line indicates the mean and the dashed lines indicate the standard deviation.**



(a) Initial mesh

(b) Adapted using near-field drag adjoint

(c) Adapted using entropy variables

(d) Adapted using Oswatitsch drag adjoint

Figure 12. RAE 2822 $M = 0.6$, $\alpha = 2.57^o$, $Re = 6.3 \times 10^6$: farfield view of adapted meshes.

equations. The relationship holds for the particular case of farfield drag calculated via an integration of entropy, originally proposed by Oswatitsch. Specifically, this calculation casts the momentum integration on the farfield boundary in terms of the local entropy and freestream quantities. We relate the leading order term in this drag expression to the entropy balance output available from the connection between entropy variables and adjoints. In the inviscid case, the relationship is direct as the entropy balance output is an integral of the entropy flux through the domain boundary. In the viscous case, the inviscid residual term has to be treated separately at the cost of an extra fine-space residual evaluation during error estimation. This procedure holds for Reynolds-averaged Navier-Stokes systems, which are not generally symmetrizable, since the entropy flux output can still be written as an entropy-variable weighted residual.

B. Discussion

In the three cases tested, the accuracy of the drag error estimate derived from the entropy variable approach is on par with that from the output adjoints. Even though the drag estimate based on the entropy variables is associated with a farfield measurement, the estimate is accurate for near-field drag calculations. This observation can be reasoned by noting that the entropy variable approach

targets areas of spurious entropy production, and in a conservative scheme it is the spurious entropy production that is responsible for the difference in the near-field and Oswatitsch drag evaluations. Although the proposed approach uses a fine space state solution, $\mathbf{u}_h$, the drag error estimate is shown to perform well when $\mathbf{u}_h$ is calculated approximately, e.g. through inexpensive smoothing. Most notably, the adaptation and drag error estimation using the proposed approach do not require adjoint solution capability in the code, as the required entropy variables are computed directly from the conservative state vector.

C. Extensions

A natural question is whether the proposed error estimate and adaptive indicator extend to three dimensions. Farfield drag calculation in three dimensions is not as simple as in two dimensions due to the possibility of trailing vorticity. However, Oswatitsch's formula can still be applied via a first-order superposition estimate[8] in which the drag is decomposed as,

$$D_{\text{total}} = D_{\text{induced}} + \underbrace{D_{\text{entropy, physical}} + D_{\text{entropy, spurious}}}_{D_{\text{entropy}}\text{: estimated by Oswatitsch formula}}. \tag{24}$$

This formula neglects drag due to enthalpy changes, which are usually negligible for simulations without powered engines. D_{induced} is the drag due to the cross-flow kinetic energy, and it can be expressed on the midfield or farfield in terms of an integral of a vorticity-weighted stream function.[8] In viscous simulations with physical dissipation of vorticity, the value of D_{induced} depends on the downstream location of the measurement plane. Our error estimate will capture numerical errors in the entropy drag contribution to Eqn. 24. Specifically, the estimated error will be the spurious component, $D_{\text{entropy, spurious}}$. Some of the spurious drag originates in the wing near-field. However, a large portion of spurious drag is expected from "numerically-dissipated" trailing vorticity, by which we mean dissipation due to discretization errors that occurs at a rate that is faster than physical dissipation. Our indicator will see this as a contribution to the output error, $D_{\text{entropy, spurious}}$. This means that we expect to target the wake heavily when using this adaptive indicator, and in fact we have seen such refinement previously for inviscid flow over a wing.[6] The corresponding drag error estimate will then likely be more conservative than a near-field drag error estimate, as an entropy-based farfield drag measure may still be polluted by incorrectly-dissipated vorticity even when the

near-field drag is quite accurate.[13] Thus, the wake will be refined possibly very far from the body. However, the extra resolution required to preserve trailing vorticity for an accurate far-field drag error estimate could be justified on several accounts. First, accurate farfield drag calculations are useful for drag decomposition. Second, high-order methods combined with *hp* adaptation may make sufficient wake resolution possible and economical. Finally, the extra cost of wake adaptation has to be weighed against the costs of an adjoint solution when comparing to near-field adaptive methods.

An additional area of investigation is the applicability of the proposed method to flows with shocks. Oswatitsch's formula holds in the presence of shocks, but implementation choices related to stabilization can have a large effect on the numerical error estimate. Both shocks and extension of the method to three dimensions will be investigated in future works.

References

[1]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.

[2]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.

[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., "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.

[6]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.

[7]van Dam, C. and Nikfetrat, K., "Accurate prediction of drag using Euler methods," AIAA Paper 91-0038, 1991.

[8]Giles, M. B. and Cummings, R. M., "Wake integration for three-dimensional flowfield computations: Theoretical development," *Journal of Aircraft*, Vol. 36, No. 2, 1999, pp. 357–365.

[9]Giles, M. B. and Cummings, R. M., "Wake integration for three-dimensional flowfield computations: Applications," *Journal of Aircraft*, Vol. 36, No. 2, 1999, pp. 357–365.

[10]Chao, D. and van Dam, C., "Airfoil drag prediction and decomposition," *Journal of Aircraft*, Vol. 36, No. 4, 1999, pp. 675–681.

[11]D.Destarac, "Far-field/near-field drag balance and applications of drag extraction in CFD," *von Karman Institute for Fluid Dynamics, Lecture Series 2003-02*, edited by H. Deconinck, K. Sermeus, and C. V. Dam, Rhode-Saint-Genése, Belgium, Feb. 3-7, 2003.

[12]Paparone, L. and Tognaccini, R., "Computational fluid dynamics-based drag prediction and decomposition," *AIAA Journal*, Vol. 41, No. 9, 2003, pp. 1647–1657.

[13]Chao, D. and van Dam, C., "Wing drag prediction and decomposition," *Journal of Aircraft*, Vol. 43, No. 1, 2006, pp. 82–90.

[14]ZiQiang, Z., XiaoLu, W., Jie, L., and Zhou, L., "Comparison of predicting drag methods using computational fluid dynamics in 2D/3D viscous flow," *Science in China Series E: Technological Sciences*, Vol. 50, No. 5, 2007, pp. 534–549.

[15]Yamazaki, W., Matsushima, K., and Nakahashi, K., "Drag prediction, decomposition and visualization in unstructured mesh CFD solver of TAS-code," *International Journal for Numerical Methods in Fluids*, Vol. 57, 2008, pp. 417–436.

[16]Levy, D. W., Zickuhr, T., Vassberg, J., Agrawal, S., Wahls, R. A., Pirzadeh, S., and Hemsch, M. J., "Data summary from the first AIAA computational fluid dynamics drag prediction workshop," *Journal of Aircraft*, Vol. 40, No. 5, 2003, pp. 875–882.

[17]Laflin, K. R., Vassberg, J. C., Wahls, R. A., Morrison, J. H., Brodersen, O., Rakowitz, M., Tinoco, E. N., and Godard, J.-L., "Summary of data from the second AIAA CFD drag prediction workshop," AIAA Paper 2004-0555, 2004.

[18]Morrison, J. H. and Hemsch, M. J., "Statistical analysis of CFD solutions from the third AIAA drag prediction workshop," AIAA Paper 2007-254, 2007.

[19]Frink, N. T., "Test case presentations from the 4th AIAA drag prediction workshop," NASA Langley, 2010, `http://aaac.larc.nasa.gov/tsab/cfdlarc/aiaa-dpw/ Workshop4/presentations/DPW4_Presentations.htm`.

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

[21]Masson, C., Veilleux, C., and Paraschivoiu, I., "Airfoil wave-drag prediction using Euler solutions of transonic flows," *Journal of Aircraft*, Vol. 35, No. 5, 1998, pp. 748–753.

[22]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.

[23]Gariépy, M., Trépanier, J.-Y., and Masson, C., "Improvements in accuracy and efficiency for a far-field drag prediction and decomposition method," AIAA Paper 2010-4678, 2010.

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

[25]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.

[26]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.

[27]Fidkowski, K. J., Ceze, M., and Roe, P. L., "Drag output error estimation using the entropy adjoint approach," AIAA Paper 2011-3867, 2011.

[28]Ceze, M. A. and Fidkowski, K. J., "Output-driven anisotropic mesh adaptation for viscous flows using discrete choice optimization," AIAA Paper 2010-0170, 2010.

[29]Giles, M. B. and Pierce, N. A., "Adjoint equations in CFD: duality, boundary conditions and solution behavior," AIAA Paper 97-1850, 1997.