ANISOTROPIC SIMPLEX MESH ADAPTATION BY METRIC OPTIMIZATION FOR HIGHER-ORDER DG DISCRETIZATIONS OF 3D COMPRESSIBLE FLOWS

Masayuki Yano[1], David L. Darmofal[1]

[1] Aerospace Computational Design Laboratory, Massachusetts Institute of Technology
(myano@mit.edu, darmofal@mit.edu)

Abstract. *We extend our optimization-based framework for anisotropic simplex mesh adaptation to three dimensions and apply it to high-order discontinuous Galerkin discretizations of steady-state aerodynamic flows. The framework iterates toward a mesh that minimizes the output error for a given number of degrees of freedom by considering a continuous optimization problem of the Riemannian metric field. The adaptation procedure consists of three key steps: sampling of the anisotropic error behavior using element-wise local solves; synthesis of the local errors to construct a surrogate error model in the metric space; and optimization of the surrogate model to drive the mesh toward optimality. The anisotropic adaptation decisions are entirely driven by the behavior of the a posteriori error estimate. As a result, the method handles any discretization order, naturally incorporates both the primal and adjoint solution behaviors, and robustly treats irregular features. Numerical results demonstrate the effectiveness of the adaptive high-order discretization applied to the compressible Navier-Stokes equations in three dimensions.*

Keywords: *anisotropic adaptation, discontinuous Galerkin method, higher-order, dual-weighted residual, compressible Navier-Stokes*

1. INTRODUCTION

Despite ever-increasing computational power, a reliable simulation of complex three-dimensional flows remains one of the challenges in the computational fluid dynamics (CFD). For instance, in a careful study following the 3rd AIAA Drag Prediction Workshop, Mavriplis demonstrated that a combination of a second-order accurate discretization and *a priori* generated best practice meshes is insufficient for accurately predicting the drag, even for a routinely-solved wing-only configuration [16]. In order to improve the reliability and efficiency of CFD simulations, significant research effort have been devoted to the development of adaptive higher-order methods, including a notable multi-organizational initiative, ADIGMA [10]. A recent review of adaptive higher-order CFD technology is provided by, for example, Hartmann and Houston [8] and Fidkowski and Darmofal [5].

One popular approach to enable adaptive higher-order discretizations for complex aerodynamic flows is to combine the discontinuous Galerkin (DG) discretization with the dual-weighted residual (DWR) method [2]. While the DWR error estimate can readily drive isotropic

adaptation, it is insufficient for making anisotropy decisions as it only assigns a single scalar value that indicates the magnitude of the error to each element. In order to enable anisotropic adaptation necessary for high Reynolds number flows, several researchers have augmented the DG and DWR methods with a means of estimating the anisotropic behavior of the error. Notable anisotropy detection approaches for compressible flows include: 1) Fidkowski and Darmofal who incorporated the higher derivatives of the Mach number field [7]; 2) Leicht and Hartmann who used the jump in the primal solution across element interfaces [11]; 3) Ceze and Fidkowski who used element-wise local solves on tensor-product elements (i.e. quadrilaterals and hexahedrons) [3]; and 4) Leicht and Hartmann who devised an anisotropic error estimate for tensor-product elements [12]. While these methods produce anisotropic meshes, approaches 1) and 2) incorporate only the behavior of the primal solution and not the dual solution in determining the anisotropy, resulting in a suboptimal mesh for the purpose of output prediction. On the other hand, approaches 2), 3), and 4) are only applicable for tensor-product elements, which prevents anisotropic resolution of arbitrarily-oriented features not aligned with the initial mesh.

To overcome some of the limitations of the aforementioned anisotropic adaptation strategies, we have recently introduced a general optimization-based anisotropic simplex mesh adaptation framework [26]. The key ingredients of the mesh optimization framework are: 1) casting of the (discrete) mesh optimization problem as a continuous optimization problem of the metric field, using a higher-order extension of the original mesh-metric duality proposed by Loseille et al. [13]; 2) sampling of the local anisotropic error using local solves, inspired by the work of Houston et al. [9] and Ceze and Fidkowski [3]; 3) synthesis of the metric-error relationship and construction of an anisotropic error model using the affine-invariant (metric) tensor manipulation framework of Pennec et al. [18]; and 4) optimization of the surrogate error model. To date, we have successfully applied the framework to the two-dimensional advection-diffusion equation [26] and various two-dimensional aerodynamic flow problems governed by the Euler, Navier-Stokes, and Reynolds-averaged Navier-Stokes equations [27]. (We have also presented various results at the 1st International Workshop on High-Order Methods [24].) This work extends our framework to three dimensions and presents result of applying the framework to three-dimensional steady-state compressible Navier-Stokes equations.

2. AN OPTIMIZATION FRAMEWORK FOR ANISOTROPIC MESH ADAPTATION

This section describes our optimization framework for anisotropic h-adaptation. The overall information flow for the feedback-based algorithm is illustrated in Figure 1. The input to the PDE solver are the geometry, boundary conditions, and the output of interest; the solver outputs the engineering quantity of interest and the associated error estimate.

2.1. Output Error Estimation and Localization

Let us define the output estimation problem for a steady-state conservation law on a domain $\Omega \subset \mathbb{R}^d$, with $d = 3$. The conservation law is of the form

$$\nabla \cdot \mathcal{F}^{\text{inv}}(u, x) - \nabla \cdot \mathcal{F}^{\text{vis}}(u, \nabla u, x) = 0, \quad \forall x \in \Omega,$$

where $u(x) \in \mathbb{R}^m$ is the state vector, $\mathcal{F}^{\text{inv}}$ is the inviscid flux function, $\mathcal{F}^{\text{vis}}$ is the viscous flux function, and m denotes the number of components of the state. The output is given by
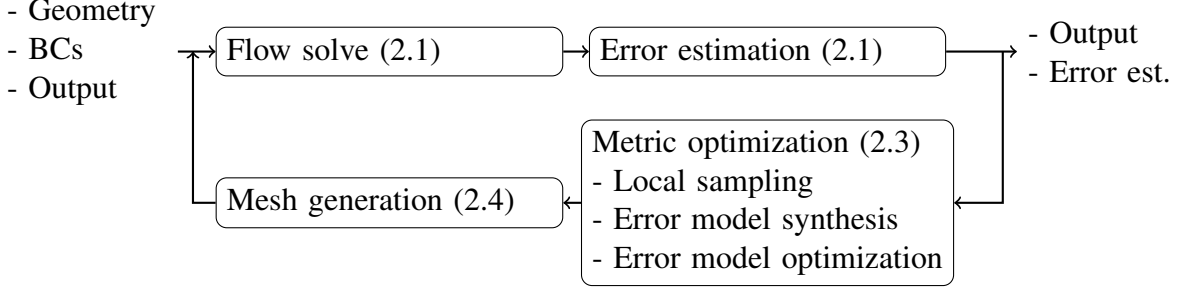
$$J = \mathcal{J}(u),$$

```
- Geometry      ┌─────────────────┐    ┌──────────────────────┐   - Output
- BCs        ──►│ Flow solve (2.1) │──►│ Error estimation (2.1)│──►- Error est.
- Output        └─────────────────┘    └──────────────────────┘
                         ▲              ┌──────────────────────────┐
                         │              │ Metric optimization (2.3)│
                ┌─────────────────────┐ │ - Local sampling         │
                │ Mesh generation (2.4)│◄─│ - Error model synthesis  │◄─
                └─────────────────────┘ │ - Error model optimization│
                                        └──────────────────────────┘
```

Figure 1. The information flow for the adaptive algorithm. The numbers in parenthesis correspond to the section numbers in this paper.

where $\mathcal{J}$ is the output functional of interest. For all applications considered in this work, $\mathcal{J}$ is a functional on the domain boundary.

This work employs a high-order discontinuous Galerkin (DG) method to approximate the output, resulting in a weak form: Find $u_{h,p} \in V_{h,p}$ such that

$$\mathcal{R}_{h,p}(u_{h,p}, v_{h,p}) = 0, \quad \forall v_{h,p} \in V_{h,p}, \tag{1}$$

where $V_{h,p}$ is the space of discontinuous, p-th order piecewise polynomial functions defined on the tessellation $\mathcal{T}_h$ of Ω, and $\mathcal{R}_{h,p}(\cdot, \cdot) : V_{h,p} \times V_{h,p} \to \mathbb{R}$ is the semilinear form. This work uses Roe's approximate Riemann solver [21] for the inviscid flux, and Bassi and Rebay's second discretization [1](BR2) for the viscous flux. The nonlinear algebraic system resulting from Eq. (1) is solved using Newton's method with pseudo-time continuation. The linear system is solved using GMRES [22], preconditioned with an in-place block-ILU(0) factorization [4] with minimum discarded fill ordering [19]. Once $u_{h,p} \in V_{h,p}$ is obtained, the desired output is estimated by

$$J_{h,p} = \mathcal{J}_{h,p}(u_{h,p}),$$

where $\mathcal{J}_{h,p} : V_{h,p} \to \mathbb{R}$ is a dual-consistent formulation of the output functional.

The objective of the functional error estimation is to approximate the true error,

$$\mathcal{E}_{\text{true}} \equiv J - J_{h,p} = \mathcal{J}(u) - \mathcal{J}_{h,p}(u_{h,p}),$$

and to identify the elements causing a large error for the purpose of adaptation. This work relies on the DWR method [2] to estimate the output error and to localize the error. For brevity, we omit the derivation of the method and state the main results. The DWR method requires the approximate adjoint $\psi_{h,\hat{p}} \in V_{h,\hat{p}}$ that satisfies

$$\mathcal{R}'_{h,p}[u_{h,p}](v_{h,\hat{p}}, \psi_{h,\hat{p}}) = \mathcal{J}'_{h,p}[u_{h,p}](v_{h,\hat{p}}), \quad \forall v_{h,\hat{p}} \in V_{h,\hat{p}}, \tag{2}$$

where $V_{h,\hat{p}} \supset V_{h,p}$ with $\hat{p} = p+1$ is the enriched space and $\mathcal{R}'_{h,p}[u_{h,p}](\cdot, \cdot)$ and $\mathcal{J}'_{h,p}[u_{h,p}](\cdot)$ denote the Fréchet derivative of $\mathcal{R}_{h,p}(\cdot, \cdot)$ and $\mathcal{J}_{h,p}(\cdot)$ with respect to the first argument evaluated about the finite element approximation $u_{h,p}$, respectively. The DWR error estimate is provided by

$$\mathcal{E}_{\text{true}} \approx -\mathcal{R}_{h,p}(u_{h,p}, \psi_{h,\hat{p}}).$$

We note that by Galerkin orthogonality we can rewrite the equation to arrive at another error representation

$$\mathcal{E}_{\text{true}} \approx \mathcal{R}'_{h,p}[u, u_{h,p}](u - u_{h,p}, \psi_{h,\hat{p}} - v_{h,p}), \quad \forall v_{h,p} \in V_{h,p} \tag{3}$$

where $\bar{\mathcal{R}}'_{h,p}[u, u_{h,p}](\cdot, \cdot)$ is the mean-value linearized semilinear form as defined in [2]. The expression states that the output error is a weighted product of the primal error, $u - u_{h,p}$, and the adjoint approximation error, $\psi - v_{h,p}$.

For the purpose of mesh adaptation, we define a more conservative error estimate that results from summing locally positive quantities, i.e.

$$\mathcal{E} \equiv \sum_{\kappa \in \mathcal{T}_h} \eta_\kappa,$$

where the element-wise localized error estimate η_κ is defined by

$$\eta_\kappa \equiv |\mathcal{R}_{h,p}(u_{h,p}, \psi_{h,\hat{p}}|_\kappa)|.$$

Assuming the primal solution $u_{h,p}$ to Eq. (1) and the adjoint solution $\psi_{h,\hat{p}}$ to Eq. (2) are unique for a given approximation space $V_{h,p}$, we can associate the (conservative) error estimate to $V_{h,p}$, i.e. $\mathcal{E} = \mathcal{E}(V_{h,p})$. Note that, because the output error is related to the primal and adjoint errors by Eq. (3), an effective control of the output error requires $V_{h,p}$ that accounts for the behaviors of both the primal and adjoint solutions.

2.2. Output Error Minimization Problem

The objective of the output-based adaptation is to find the space $V^*_{h,p}$ that minimizes the output error for a given dimension of $V_{h,p}$, i.e.

$$V^*_{h,p} = \arg \inf_{V_{h,p}} \mathcal{E}(V_{h,p}) \quad \text{s.t.} \quad \dim(V_{h,p}) \leq N,$$

where N is the maximum permissible dimension of $V_{h,p}$. In particular, if $V_{h,p}$ consists of elements with a constant polynomial order p, then $V_{h,p}$ is described by the triangulation $\mathcal{T}_h$ and the scalar p, i.e. $V_{h,p} = V_{h,p}(\mathcal{T}_h, p)$. Thus, for a fixed $p \in \mathbb{R}^+$, the optimization problem simplifies to that of finding the optimal triangulation $\mathcal{T}^*_h$ such that

$$\mathcal{T}^*_h = \arg \inf_{\mathcal{T}_h} \mathcal{E}(V_{h,p}(\mathcal{T}_h, p)) \quad \text{s.t.} \quad \dim(V_{h,p}(\mathcal{T}_h, p)) \leq N. \tag{4}$$

This is a discrete-continuous optimization problem, as the triangulation $\mathcal{T}_h$ is defined by the node locations and the connectivity of the nodes. In general, the problem is intractable.

In order to find an approximate solution to the problem Eq. (4), we consider a continuous relaxation of the discrete problem, following the approach pursued by Loseille et al.[14, 13, 15]. In particular, we invoke the concept of mesh-metric duality, which states that ability of an anisotropic simplex mesh to approximate a function is described by the metric field associated with the tessellation. Here, the metric field $\mathcal{M} = \{\mathcal{M}(x)\}_{x \in \Omega}$ consists of symmetric positive definite (SPD) matrices $\mathcal{M}(x) \in Sym_d^+$ such that all edges of the triangulation are close to unit length with respect to the metric, i.e.

$$\sqrt{e^T \mathcal{M}(x_c) e} \approx 1, \quad \forall e \in \text{Edges}(\mathcal{T}_h),$$

where x_c is the midpoint of the edge. The metric field provides an anisotropic element size specification that guides anisotropic mesh generation. Our extension of the original mesh-metric duality for piecewise linear functions by Loseille et al. [13] to higher-degree polynomials is described in details in [25]. As the metric field captures the approximation properties of the mesh, we can cast a continuous relaxation of the discrete problem Eq. (4) as

$$\mathcal{M}^* = \arg \inf_{\mathcal{M}} \mathcal{E}(V_{h,p}(\mathcal{M}, p)) \quad \text{s.t.} \quad \dim(V_{h,p}(\mathcal{M}, p)) \leq N.$$

For notational simplicity, we write the optimization problem as

$$\mathcal{M}^* = \arg\inf_{\mathcal{M}} \mathcal{E}(\mathcal{M}) \quad \text{s.t.} \quad \mathcal{C}(\mathcal{M}) \leq N, \tag{5}$$

where $\mathcal{E}$ and $\mathcal{C}$ are the error and cost functionals that map the metric tensor field to the error and cost, respectively. The expression assumes that the polynomial order, p, is fixed.

2.3. Metric Tensor Field Optimization Algorithm

This section describes our approach to approximately solving Eq. (5). The optimization algorithm consists of three steps: sampling of the local error behavior, synthesis of the local error model, and optimization of the surrogate model. The full derivation and rationales behind the choices are provided in our recent work [26].

2.3.1 Error Localization

In order to solve the error minimization problem, Eq. (5), we must estimate the behavior of the error functional $\mathcal{E}$ and the cost functional $\mathcal{C}$. In particular, assuming the error is localizable, we decompose the error and cost functionals as

$$\mathcal{E}(\mathcal{M}) = \sum_{\kappa \in \mathcal{T}_h} \eta_\kappa(\mathcal{M}_\kappa) \quad \text{and} \quad \mathcal{C}(\mathcal{M}) = \sum_{\kappa \in \mathcal{T}_h} \rho_\kappa(\mathcal{M}_\kappa).$$

Our goal then is to develop the local error models $\eta_\kappa : Sym_d^+ \to \mathbb{R}^+$, $\kappa \in \mathcal{T}_h$, and the local cost models $\rho_\kappa : Sym_d^+ \to \mathbb{R}^+$, $\kappa \in \mathcal{T}_h$. Here, $\mathcal{M}_\kappa$ should be understood as the "mean" metric associated with the region covered by κ; the exact interpretation of the "mean" will be presented shortly.

2.3.2 Local Error Sampling

The goal of the local error sampling step is to probe the anisotropic behavior of the local elemental error η_κ as a function of the local metric $\mathcal{M}_\kappa$. Here, we directly monitor the *a posteriori* error for several different configurations obtained by local mesh refinement. The six split configurations κ_i, $i = 1, \ldots, n_{\text{config}} = 6$, and the associated metrics in three dimensions are shown in Figure 2. The original, unsplit configuration is denoted by κ_0. For each configuration κ_i, we solve an element-wise local problem. The local solution, $u_{h,p}^{\kappa_i} \in V_{h,p}(\kappa_i)$, satisfies

$$\mathcal{R}_{h,p}^{\kappa_i}(u_{h,p}^{\kappa_i}, v_{h,p}^{\kappa_i}) = 0, \quad \forall v_{h,p}^{\kappa_i} \in V_{h,p}(\kappa_i),$$

where the local semilinear form, $\mathcal{R}_{h,p}^{\kappa_i}(\cdot, \cdot)$, sets the boundary fluxes on κ_i by freezing the solution on the neighbor elements. Then, we recompute the localized DWR error estimate corresponding to the split mesh as

$$\eta_{\kappa_i} \equiv |\mathcal{R}_{h,p}^{\kappa_i}(u_{h,p}^{\kappa_i}, \psi_{h,\hat{p}}|_{\kappa_0})|.$$

Due to the local Galerkin orthogonality of the DG scheme, we can rewrite the local error as

$$\eta_{\kappa_i} \equiv |\mathcal{R}_{h,p}^{\kappa_i}(u_{h,p}^{\kappa_i}, \psi_{h,\hat{p}}|_{\kappa_0})| = \inf_{v_{h,p}^{\kappa_i} \in V_{h,p}(\kappa_i)} |\mathcal{R}_{h,p}^{\kappa_i}(u_{h,p}^{\kappa_i}, (\psi_{h,\hat{p}} - v_{h,p}^{\kappa_i})|_{\kappa_0})|.$$

The second equality signifies that the local sampling procedure automatically accounts for the improvement in the adjoint approximability resulting from the local refinement even though the local adjoint problem is not explicitly solved. Thus, the local sampling technique based on the local solve and *a posteriori* error estimate automatically captures the behaviors of both primal and dual solutions. Finally, we compute the local metric associated with κ_i, $\mathcal{M}_{\kappa_i}$, to construct metric-error pairs $\{\mathcal{M}_{\kappa_i}, \eta_{\kappa_i}\}_{i=1}^{n_{\text{config}}}$.

2.3.3 Local Error Model Synthesis

The goal of the error model synthesis step is to construct an element-wise continuous metric-error function $\eta_\kappa(\cdot) : Sym_d^+ \to \mathbb{R}^+$ using the pairs $\{\mathcal{M}_{\kappa_i}, \eta_{\kappa_i}\}_{i=1}^{n_{\text{config}}}$ collected in the sampling stage. In order to build a continuous error model, we need means of interpolating metric tensors (i.e. SPD matrices). The Euclidean framework (i.e. entry-wise manipulation) is unsuited for manipulation of SPD matrices as SPD matrices define a convex cone and none-SPD matrices are finite distance away from SPD matrices. Our tensor interpolation framework builds on Pennec's affine-invariant Riemannian framework for tensor manipulation [18], which is briefly reviewed here.

In the affine-invariant framework, the difference between two tensors $\mathcal{M}$ and $\mathcal{M}_0$ are measured in terms of a symmetric matrix $S \in Sym_d$ given by a logarithmic map,

$$S(\mathcal{M}) = \log(\mathcal{M}_0^{-1/2} \mathcal{M} \mathcal{M}_0^{-1/2}),$$

where $\log(\cdot) : Sym_d^+ \to Sym_d$ is the matrix logarithm. We will refer to this matrix as the *step matrix*. Conversely, an exponential map of a step matrix, S, induces a change in a tensor $\mathcal{M}_0$,

$$\mathcal{M}(S) = \mathcal{M}_0^{1/2} \exp(S) \mathcal{M}_0^{1/2},$$

where $\exp(\cdot) : Sym_d \to Sym_d^+$ is the matrix exponential. To gain some insight, let us decompose the step matrix into the isotropic and tracefree parts, i.e.

$$S_\kappa = s_\kappa I + \tilde{S}_\kappa, \tag{6}$$

where $s_\kappa = \text{tr}(S_\kappa)/d$ such that $\text{tr}(\tilde{S}_\kappa) = 0$. The isotropic part, $s_\kappa I$, scales the tensor without changing its shape, and the tracefree part, $\tilde{S}_\kappa$, changes the shape of the tensor without changing its volume.

Applying the affine-invariant tensor interpolation framework to the metric tensor samples collected in the error sampling stage, we measure the changes in each configuration as

$$S_{\kappa_i} = \log\left(\mathcal{M}_{\kappa_0}^{-1/2} \mathcal{M}_{\kappa_i} \mathcal{M}_{\kappa_0}^{-1/2}\right), \quad i = 0, \dots, n_{\text{config}}.$$

Note that, by construction, the original configuration, $\mathcal{M}_{\kappa_0}$, maps to the origin, i.e. $S_{\kappa_0} = 0$. Similarly, we measure the associated changes in the errors as

$$f_{\kappa_i} = -|\log(\eta_{\kappa_i}/\eta_{\kappa_0})|, \quad i = 0, \dots, n_{\text{config}}. \tag{7}$$

Note that we take the negative of the absolute value of the change to ensure that the model predicts reduction in the error when the element is refined; without the change, inaccurate error estimate on very coarse meshes could produce a model which predicts increase in the error with refinement. Again, the original error, η_{κ_0}, maps to zero by construction.

(a) original   (b) edge split 1   (c) edge split 2   (d) edge split 3

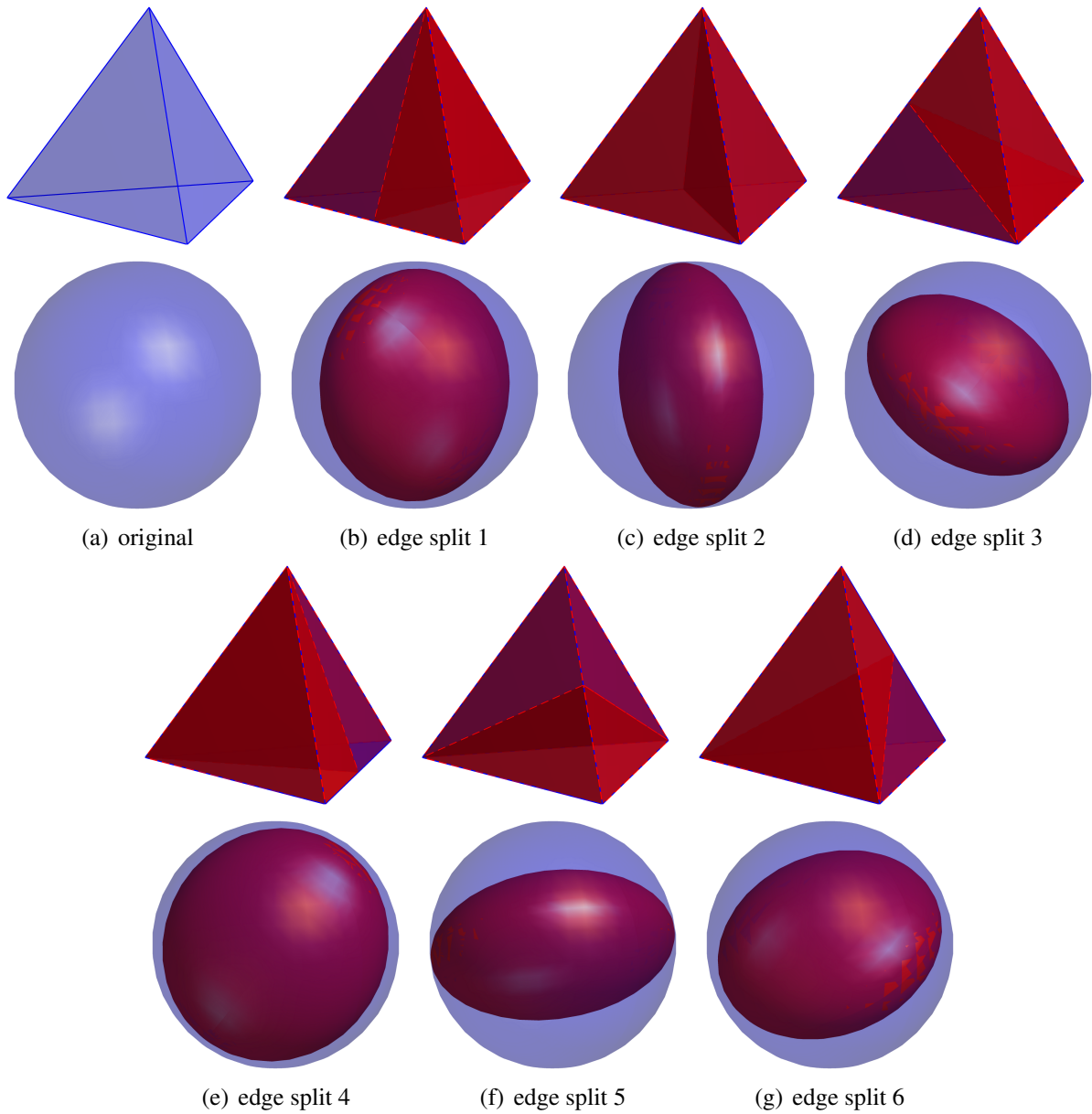(e) edge split 4   (f) edge split 5   (g) edge split 6

Figure 2. The original and edge split configurations for sampling the anisotropic error behavior in three dimensions. The metric implied by the sampled configurations are shown in red ellipsoids.

Once we have the pairs $\{S_{\kappa_i}, f_{\kappa_i}\}_{i=1}^{n_{\text{config}}}$ that characterize the change in the error as a function of the change in the configuration, our objective is to construct a continuous function $f_\kappa(\cdot) : Sym_d \to \mathbb{R}$. We choose to construct a linear function in the entries of S_κ,

$$f_\kappa(S_\kappa) = \text{tr}(R_\kappa S_\kappa). \tag{8}$$

To find an appropriate $d \times d$ symmetric matrix R_κ that governs the behavior of the linear function, we perform the least-squares regression of the known data, i.e.

$$R_\kappa = \arg\min_{Q \in Sym_d} \sum_{i=1}^{n_{\text{config}}} (f_{\kappa_i} - \text{tr}(QS_{\kappa_i}))^2 .$$

Thus, from Eq. (7) and (8), the local error model over the region covered by κ is given by

$$\eta_\kappa(S_\kappa) = \eta_{\kappa_0} \exp(\text{tr}(R_\kappa S_\kappa)).$$

The tensor, R_κ, can be thought of as a generalization of the convergence rate for isotropic scaling to anisotropic manipulation. If we consider the decomposition

$$R_\kappa = r_\kappa I + \tilde{R}_\kappa, \tag{9}$$

where $r_\kappa = \text{tr}(R_\kappa)/d$ and $\text{tr}(\tilde{R}_\kappa) = 0$, then r_κ and $\tilde{R}_\kappa$ capture the sensitivity of the error to the change in the element size and shape, respectively.

2.3.4 Local Cost Function Model

The element-wise cost function model, ρ_κ, is given in terms of the step matrix $S_\kappa = s_\kappa I + \tilde{S}_\kappa$ by

$$\rho_\kappa(S_\kappa) = \rho_{\kappa_0} \exp\left(\frac{1}{2}s_\kappa d\right),$$

where ρ_{κ_0} is the number of degrees of freedom associated with an element, e.g. $(p+1)(p+2)(p+3)/6$ for a tetrahedron. Note that the cost is only a function of s_κ, which controls the volume of the tensor, and not $\tilde{S}_\kappa$, which changes the shape of the tensor without changing the volume.

2.3.5 Optimization and Optimality Conditions

The final step of the adaptation algorithm is to optimize the metric field, $\mathcal{M}$, described by the vertex values $\{\mathcal{M}_\nu\}_{\nu \in \mathcal{V}}$. The vertex-based metric can then be used to generate a metric-conforming mesh using an anisotropic mesh generator. Starting from the original configuration $\{\mathcal{M}_{\nu_0}\}_{\nu \in \mathcal{V}}$ implied by the current mesh, we manipulate the metric tensor $\mathcal{M}_\nu$ at vertex ν using a step matrix $S_\nu \in Sym_d$, i.e.

$$\mathcal{M}_\nu(S_\nu) = \mathcal{M}_{\nu_0}^{1/2} \exp(S_\nu) \mathcal{M}_{\nu_0}^{1/2}. \tag{10}$$

Given $\{\mathcal{M}_{\nu_0}\}_{\nu \in \mathcal{V}}$, our objective is to choose the step matrices $\{S_\nu\}_{\nu \in \mathcal{V}}$ to reduce the error.

Using the surrogate error model and the cost model, we approximate the functionals appearing in the optimization problem Eq. (5) as

$$\mathcal{E}(\mathcal{M}) \approx \tilde{\mathcal{E}}(\{S_\nu\}_{\nu \in \mathcal{V}}) \equiv \sum_{\kappa \in \mathcal{T}_h} \eta_\kappa \left(\overline{\{S_\nu\}_{\nu \in \mathcal{V}(\kappa)}} \right)$$

$$\mathcal{C}(\mathcal{M}) \approx \tilde{\mathcal{C}}(\{S_\nu\}_{\nu \in \mathcal{V}}) \equiv \sum_{\kappa \in \mathcal{T}_h} \rho_\kappa \left(\overline{\{S_\nu\}_{\nu \in \mathcal{V}(\kappa)}} \right),$$

where the $\overline{\{S_\nu\}_{\nu \in \mathcal{V}(\kappa)}}$ is the approximate step matrix over the region covered by κ and is defined as the mean of the vertex step matrices on its vertices, $\mathcal{V}(\kappa)$, i.e.

$$\overline{\{S_\nu\}_{\nu \in \mathcal{V}(\kappa)}} \equiv \frac{1}{|\mathcal{V}(\kappa)|} \sum_{\nu \in \mathcal{V}(\kappa)} S_\nu.$$

Using the surrogate error and cost functions, we have turned our infinite dimensional optimization problem Eq. (5) into a finite dimensional optimization of vertex step matrices. The surrogate optimization problem for the optimal $\{S_\nu\}_{\nu \in \mathcal{V}}$ is

$$\{S_\nu^*\}_{\nu \in \mathcal{V}} = \underset{\{S_\nu\}_{\nu \in \mathcal{V}}}{\arg \inf} \mathcal{E}\left(\{S_\nu\}_{\nu \in \mathcal{V}}\right) \tag{11}$$

$$\text{s.t.} \quad \mathcal{C}\left(\{S_\nu\}_{\nu \in \mathcal{V}}\right) = N \tag{12}$$

$$|(S_\nu)_{ij}| \leq \alpha, \quad i, j = 1, \dots, d, \quad \forall \nu \in \mathcal{V}. \tag{13}$$

The last constraint, Eq. (13), is added to prevent a large charge in the metric field that would render our sampling-based error model inaccurate. Our procedure for solving the optimization problem is detailed in [26] and is omitted here for brevity.

Let us describe the features of the optimal mesh by appealing to the optimality conditions of the optimization problem Eq. (11)-(13). For simplicity, let us assume that the current configuration is sufficiently close to the optimal configuration such that the constraints Eq. (13) are inactive. The first order optimality condition is given by

$$\frac{\partial \mathcal{E}}{\partial s_\nu} + \lambda \frac{\partial \mathcal{C}}{\partial s_\nu} = 0, \quad \forall \nu \in \mathcal{V}, \tag{14}$$

$$\frac{\partial \mathcal{E}}{\partial \tilde{S}_\nu} = 0, \quad \forall \nu \in \mathcal{V}, \tag{15}$$

for some Lagrange multiplier λ, where $s_\nu = \text{tr}(S_\nu)/d$ and $\tilde{S}_\nu$ is the trace-free part of S_ν. The first condition, Eq. (14), is a global condition for the size distribution. In particular, if we define the "local" Lagrange multiplier as

$$\lambda_\nu \equiv \frac{\partial \mathcal{E}}{\partial s_\nu} \bigg/ \frac{\partial \mathcal{C}}{\partial s_\nu},$$

then we must have $\lambda_\nu = \lambda, \forall \nu \in \mathcal{V}$. The global coupling is provided by the (global) Lagrange multiplier, λ. The local Lagrange multiplier, λ_ν, is interpreted as the marginal improvement in the local error for a given investment in the local cost, which is the number of degrees of freedom in the context of mesh adaptation. The global condition states that, at optimality, the investment to any element results in the same marginal improvement in the error.

The second condition, Eq. (15), is a local condition that states that the error is stationary with respect to the shape change. Note that this second optimality condition is satisfied if

$$\tilde{R}_\kappa = 0, \quad \forall \kappa \in \mathcal{T}_h, \tag{16}$$

where $\tilde{R}_\kappa$ is the tracefree part of R_κ. The shape change, induced by $\tilde{S}_\nu$, does not affect the cost. Thus, if $\tilde{R}_\kappa \neq 0$, then we can reduce the error by choosing an $\tilde{S}_\nu$ such that $\mathrm{tr}\left(\tilde{R}_\kappa \overline{\{S_\nu\}_{\nu \in \mathcal{V}(\kappa)}}\right) < 0$ without affecting the cost. Thus, the stationarity with respect to the shape change is required at optimality.

2.4. High-Order Three-Dimensional Mesh Generation

From the new requested metric field, an anisotropic mesh is generated using a metric-conforming mesh generator. This work uses Edge Primitive Insertion and Collapse (EPIC) [17] developed by The Boeing Company to generate all meshes.

2.5. Properties of the Optimization Method

We summarize the key features of the proposed optimization method in the context of output-based adaptation:

- The method automatically takes into account all components of both the primal and adjoint solutions, as the *a posteriori* error estimate automatically captures the behaviors of both solutions.

- The method handles any discretization order.

- Unlike the previous optimization-based methods that rely on *a priori* assumption of the error behavior [20, 23, 6], the proposed method approximates the convergence rate based on the behavior of the local *a posteriori* error estimates. This makes the proposed method more robust when features are underresolved, e.g., due to the presence of a singularity, and the convergence rate is lower than the optimal *a priori* convergence rate for smooth functions.

- The method operates on simplex meshes, which allow for arbitrarily oriented anisotropic elements. This is in contrast to the anisotropic adaptation methods based on hierarchical subdivision of parent elements, e.g. [3, 12], in which the allowable anisotropy is restricted by the topology of the initial mesh.

- The majority of the computational cost stems from the six local element-wise solves. As the local solves are perfectly scalable while the global flow solve scales superlinearly, the adaptation cost becomes a small fraction of the flow solve with the problem size.

3. RESULTS

3.1. Assessment Procedure

We assess the effectiveness of the adaptation algorithm using two three-dimensional laminar flow cases considered by Leicht and Hartmann [12]. These two cases were originally designed for the ADIGMA project [10] and were also selected as test cases in the recent High-Order Workshop [24]. In particular, we compare the drag convergence obtained using our
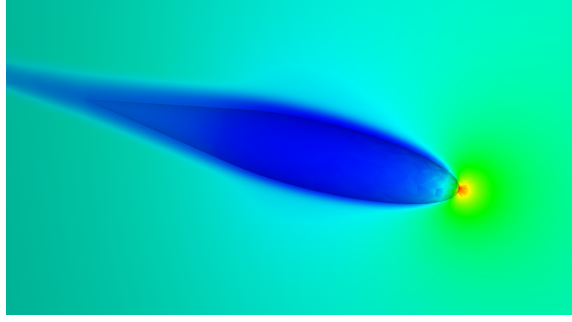
Figure 3. The normalized density, ρ/ρ_∞, for the streamlined body case. The color scale ranges from 0.938 (blue) to 1.154 (red).

adaptive algorithm and that obtained on the meshes prepared for the workshop. Note that the workshop meshes are designed for the particular flow conditions considered and can be thought of as representative of "best-practice" meshes. These workshop meshes consist of hexahedral elements with anisotropic boundary layer and wake refinement.

The convergence history for the adaptive algorithm is obtained by optimizing the mesh at several numbers of degrees of freedom. In particular, for each select number of degrees of freedom, we apply several iterations of the adaptation algorithm illustrated in Figure 1 until no further improvement in the error estimate is observed. The number of adaptation iterations required to produce an optimal mesh is dependent on the starting mesh; as we use the mesh optimized for a lower number of degrees of freedom as the starting mesh, few adaptation iterations are required to achieve optimality.

3.2. Streamlined Body

The first case we consider is flow over a three-dimensional streamlined body, considered by Leicht and Hartmann [12] and in Case 2.3 of the High-Order Workshop [24]. The precise geometry definition is provided on the workshop website [24]. The laminar flow has a freestream Mach number of $M_\infty = 0.5$, a length-based Reynolds number of $Re_L = 5000$, and an angle of attack of $\alpha = 1.0°$. The viscosity is assumed constant with a Prandtl number of $Pr = 0.72$. Adiabatic wall boundary condition is imposed on the body. The normalized density distribution on the body and the symmetry plane are shown in Figure 3. The reference drag coefficient value computed for the ADIGMA project is $C_D = 0.06317$ [12].

Figure 4 shows the drag error convergence results using the $p = 1$ and $p = 2$ DG discretizations. The results for the *a priori* generated workshop meshes are from the report by the University of Michigan group [24]. Even for this relatively simple case, output-based adaptation significantly improves the error-to-dof efficiency. For instance, the number of degrees of freedom required to meet the 0.1% drag error tolerance is reduced by an order of magnitude for the $p = 2$ discretization. The $p = 1$ discretization also benefits from adaptation, though the improvement is harder to quantify because of the noisy convergence behavior on the *a priori* generated meshes due to output error cancellation. Note that the difference between the $p = 1$ and $p = 2$ discretization is more pronounced on the adapted meshes than on the *a priori* generated meshes. This is partially due to the presence of a weak edge singularity along the trailing edge, which prevents the $p = 2$ discretization to converge at the optimal rate on *a priori* generated meshes. Thus, even for this simple flow, adaptive mesh refinement that controls the influence of the singularity is crucial to realize the full potential of the $p = 2$ discretization.
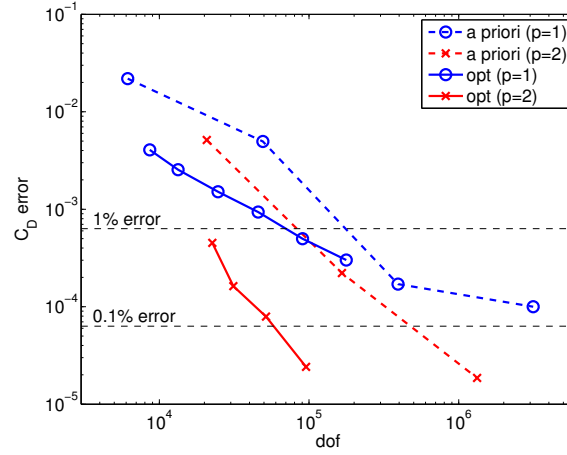
Figure 4. Drag error convergence for the streamlined body case. The *a priori* results are from the High-Order Workshop [24].

(a) $p = 1$, dof $= 90,000$, $\mathcal{E} \approx 5.0 \times 10^{-4}$

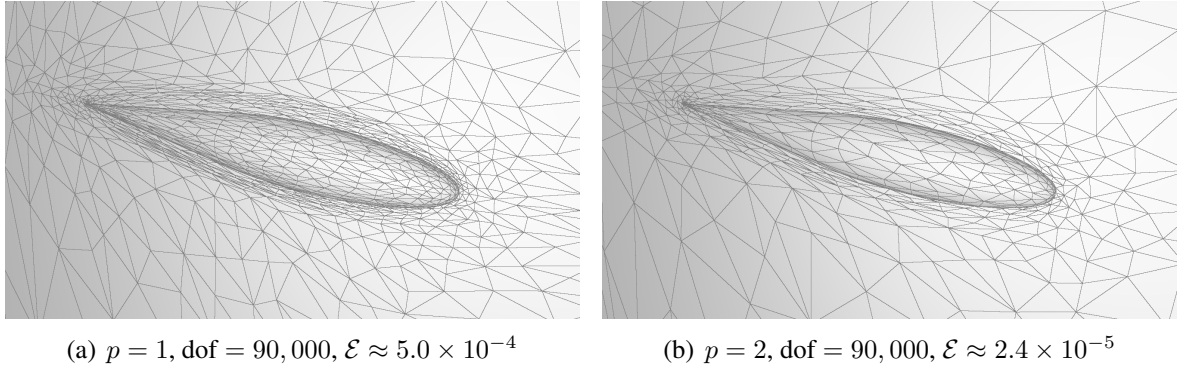(b) $p = 2$, dof $= 90,000$, $\mathcal{E} \approx 2.4 \times 10^{-5}$

Figure 5. Adapted meshes for the streamlined body case.

Figure 5(a) and 5(b) show $p = 1$ and $p = 2$ adapted meshes, each having approximately 90,000 degrees of freedom. Both $p = 1$ and $p = 2$ adapted meshes employ anisotropic elements in the boundary layer, providing the highest resolution in the wall-normal direction and the next higher resolution in the azimuthal direction. Note that the $p = 2$ mesh is significantly coarser than the $p = 1$ mesh particularly in the boundary layer region; the fact that the $p = 2$ discretization achieves 20 times lower error than the $p = 1$ discretization on this coarser mesh highlights the effectiveness of the $p = 2$ discretization in resolving smooth features such as the boundary layer.

3.3. Delta Wing

The second case we consider is laminar flow over a delta wing at a high angle of attack, the case originally considered by Leicht and Hartmann [12] and in Case 2.4 of the High-Order Workshop [24]. The delta wing has a sharp leading edge and a blunt trailing edge. The freestream Mach number is $M_\infty = 0.3$, the angle of attack is $\alpha = 12.5°$, and the Reynolds number based on the root chord is $Re_{c_r} = 4000$. The viscosity is assumed to be constant and the Prandtl number is set to 0.72. Isothermal no-slip boundary condition with the wall temperature equal to the freestream condition is imposed on the wing.

The Mach number distribution and streamlines of the flow around the delta wing are shown in Figure 6. The flow rolls up over the sharp leading edge and creates large vortices
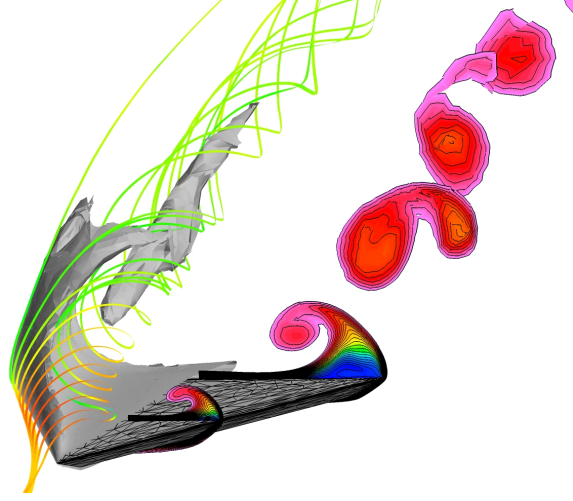
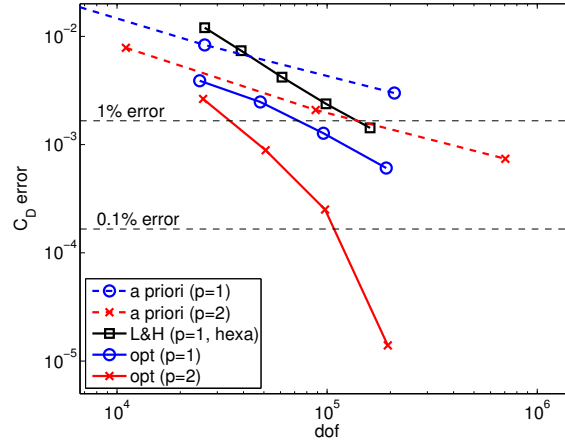Figure 6. The Mach number distribution for the delta wing case.

Figure 7. Drag error convergence for the flow over a delta wing. The *a priori* results are from the High-Order Workshop [24]. "L&H" corresponds to the result reported by Leicht and Hatrmann in [12].

on the upper surface of the wing. Both the singularity along the leading edge and the smooth vortices on the upper surface must be captured to accurately compute the drag. The reference value of the drag coefficient computed for the ADIGMA project is $C_D = 0.1658$ [12].

Figure 7 shows the convergence of the drag coefficient using several different methods. The results for the *a priori* generated workshop meshes are from the report by the University of Michigan group [24]. The label "L&H" corresponds to the result reported by Leicht and Hatrmann in [12] using their hexahedron-based, anisotropic hierarchical subdivision adaptation. The initial mesh for our adaptation consists of only 26 elements; as shown in Figure 8(a), the upper surface of the delta wing covered by a single face of a tetrahedron.

Due to the presence of multiple geometry-induced singularities, both the $p = 1$ and $p = 2$ discretizations achieve the same low convergence rate on *a priori* generated meshes. In particular, the benefit of higher-order discretization is not realized on this family of meshes. Our adaptive algorithm significantly improves the quality of the drag prediction. For the $p = 1$ discretization, our adaptive algorithm produces a family of meshes that are more efficient than the meshes generated *a priori* or those generated through hexahedron-based anisotropic
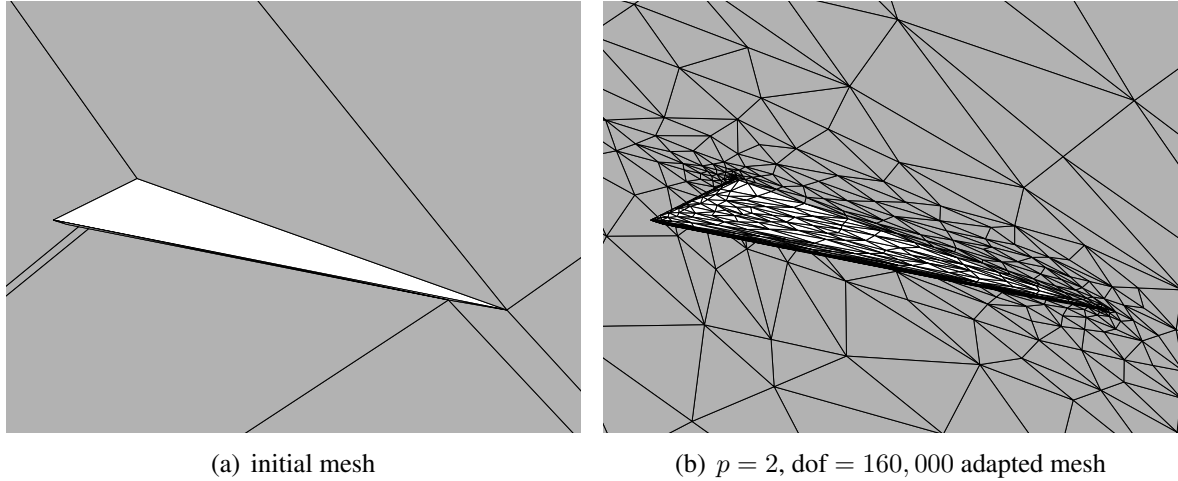
| (a) initial mesh | (b) $p = 2$, $\mathrm{dof} = 160,000$ adapted mesh |

Figure 8. Initial and adapted meshes for the delta wing case.

hierarchical subdivision.[1] The adaptation significantly improves the performance of the $p = 2$ discretization, reducing the number of degrees of freedom required to achieve 10 drag counts of error by over an order of magnitude. For a higher-fidelity simulation requiring a tighter error tolerance, the adaptation achieves more drastic improvement in the error-dof efficiency. The higher-efficiency is achieved through aggressive mesh refinement toward the geometric singularities, as shown in Figure 8(b).

4. Summary and Conclusion

We applied our output-based anisotropic simplex mesh adaptation framework to three-dimensional laminar flows over a streamlined body and a delta wing. For both cases considered, adapted meshes significantly improved the error-to-dof efficiency for drag prediction compared to the *a priori* generated workshop meshes. The difference is pronounced for the delta wing case with the leading edge singularity, whose resolution is crucial for an accurate prediction of the vortex roll up. The results suggest that a combination of a higher-order discretization and fully-automated mesh adaptation is a positive step forward in realizing reliable output prediction for three-dimensional aerodynamic flows.

Acknowledgments

5. REFERENCES

[1] F. Bassi, S. Rebay, GMRES discontinuous Galerkin solution of the compressible Navier-Stokes equations, in: K. Cockburn, Shu (eds.), Discontinuous Galerkin Methods: Theory, Computation and Applications, Springer, Berlin, 2000, pp. 197–208.

---

[1]Note that, unlike our simplex adaptation method, the efficiency of the hexahedron-based anisotropic hierarchical subdivision is dependent on the initial mesh. Thus, the L&H result could likely be improved (or worsened) using a different initial mesh.

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

[3] M. Ceze, K. J. Fidkowski, Output-driven anisotropic mesh adaptation for viscous flows using discrete choice optimization, AIAA 2010–170 (2010).

[4] L. T. Diosady, D. L. Darmofal, Preconditioning methods for discontinuous Galerkin solutions of the Navier-Stokes equations, J. Comput. Phys. 228 (2009) 3917–3935.

[5] K. Fidkowski, D. Darmofal, Review of output-based error estimation and mesh adaptation in computational fluid dynamics 49 (4) (2011) 673–694.

[6] K. J. Fidkowski, A simplex cut-cell adaptive method for high-order discretizations of the compressible Navier-Stokes equations, PhD thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics (Jun. 2007).

[7] K. J. Fidkowski, D. L. Darmofal, A triangular cut-cell adaptive method for higher-order discretizations of the compressible Navier-Stokes equations, J. Comput. Phys. 225 (2007) 1653–1672.

[8] R. Hartmann, P. Houston, Error estimation and adaptive mesh refinement for aerodynamic flows, in: H. Deconinck (ed.), VKI LS 2010-01: 36th CFD/ADIGMA course on hp-adaptive and hp-multigrid methods, Oct. 26-30, 2009, Von Karman Institute for Fluid Dynamics, Rhode Saint Genèse, Belgium, 2009.

[9] P. Houston, E. H. Georgoulis, E. Hall, Adaptivity and a posteriori error estimation for DG methods on anisotropic meshes, International Conference on Boundary and Interior Layers.

[10] N. Kroll, ADIGMA - a European project on the development of adaptive higher-order variational methods for aerospace applications, AIAA 2009-176 (2009).

[11] T. Leicht, R. Hartmann, Anisotropic mesh refinement for discontinuous Galerkin methods in two-dimensional aerodynamic flow simulations, Internat. J. Numer. Methods Fluids 56 (2008) 2111–2138.

[12] T. Leicht, R. Hartmann, Error estimation and anisotropic mesh refinement for 3d laminar aerodynamic flow simulations, J. Comput. Phys. 229 (2010) 7344–7360.

[13] A. Loseille, F. Alauzet, Continuous mesh model and well-posed continuous interpolation error estimation, INRIA RR-6846 (2009).

[14] A. Loseille, F. Alauzet, Optimal 3d highly anisotropic mesh adaptation based on the continuous mesh framework, in: Proceedings of the 18th International Meshing Roundtable, Springer Berlin Heidelberg, 2009.

[15] A. Loseille, A. Dervieux, F. Alauzet, On 3-d goal-oriented anisotropic mesh adaptation applied to inviscid flows in aeronautics, AIAA 2010-1067 (2010).

[16] D. J. Mavriplis, Results from the 3rd Drag Prediction Workshop using the NSU3D unstructured mesh solver, AIAA 2007-256 (2007).

[17] T. Michal, J. Krakos, Anisotropic mesh adaptation through edge primitive operations, AIAA 2012-159 (2012).

[18] X. Pennec, P. Fillard, N. Ayache, A Riemannian framework for tensor computing, Int. J. Comput. Vision 66 (1) (2006) 41–66.

[19] P.-O. Persson, J. Peraire, Newton-GMRES preconditioning for discontinuous Galerkin discretizations of the Navier-Stokes equations, SIAM J. Sci. Comput. 30 (6) (2008) 2709–2722.

[20] R. Rannacher, Adaptive Galerkin finite element methods for partial differential equations, Journal of Computational and Applied Mathematics 128 (2001) 205–233.

[21] P. L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, J. Comput. Phys. 43 (2) (1981) 357–372.

[22] Y. Saad, M. H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM Journal on Scientific and Statistical Computing 7 (3) (1986) 856–869.

[23] D. A. Venditti, D. L. Darmofal, Anisotropic grid adaptation for functional outputs: Application to two-dimensional viscous flows, J. Comput. Phys. 187 (1) (2003) 22–46.

[24] Z. J. Wang, 1st international workshop on high-order CFD methods, `http://zjw.public.iastate.edu/hiocfd.html`.

[25] M. Yano, An optimization framework for adaptive higher-order discretizations of partial differential equations on anisotropic simplex meshes, PhD thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics (Jun. 2012).

[26] M. Yano, D. Darmofal, An optimization framework for anisotopic simplex mesh adaptation, J. Comput. Phys. (submitted).

[27] M. Yano, D. Darmofal, An optimization framework for anisotropic simplex mesh adaptation: application to aerodynamic flows, AIAA 2012–0079 (Jan. 2012).