An Optimization Framework for Anisotropic Simplex Mesh Adaptation: Application to Aerodynamic Flows

Citation: Yano, Masayuki, and David Darmofal. "An Optimization Framework for Anisotropic Simplex Mesh Adaptation: Application to Aerodynamic Flows." 50th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition (January 9, 2012).

As Published: http://dx.doi.org/10.2514/6.2012-79

Publisher: American Institute of Aeronautics and Astronautics

Persistent URL: http://hdl.handle.net/1721.1/96928

Version: Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

An Optimization Framework for Anisotropic Simplex Mesh Adaptation: Application to Aerodynamic Flows

Masayuki Yano* and David L. Darmofal†

Aerospace Computational Design Laboratory, Massachusetts Institute of Technology

We apply an optimization-based framework for anisotropic simplex mesh adaptation to high-order discontinuous Galerkin discretizations of two-dimensional, 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 without making *a priori* assumptions about the solution behavior. As a result, the method handles any discretization order, naturally incorporates both the primal and adjoint solution behaviors, and robustly treats irregular features. The numerical results demonstrate that the proposed method is at least as competitive as the previous method that relies on *a priori* assumption of the solution behavior, and, in many cases, outperforms the previous method by over an order of magnitude in terms of the output accuracy for a given number of degrees of freedom.

I. Introduction

For decades, significant research effort has been devoted to improve computational efficiency and reliability of aerodynamic simulations through automatic adaptation. The development of adaptive schemes has been motivated by the lack of solution accuracy that results from the reliance on best-practice guidelines for mesh generation. Mavriplis demonstrated deficiencies of the non-adaptive solvers even for a routinely-solved geometry such as the wing-only configuration.[1] The lack of solution accuracy and reliability severely limits the applicability of non-adaptive solvers to explore radically different designs, populate databases, or optimize configurations—the areas in which computational simulations should excel. Progress has been made in the development of adaptive solution strategies that overcome the shortcomings of the traditional, non-adaptive solvers. In particular, adjoint-based error estimation methods, such as the dual-weighted residual (DWR) method of Becker and Rannacher,[2,3] provide a mathematical framework for assessing the quality of the performance variables, such as lift and drag, in practical engineering simulations. The effectiveness of isotropic adaptation strategies that incorporate the DWR error estimate have been demonstrated in numerous early studies.[4–6]

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. The inability to drive anisotropic adaptation is a major limitation, especially for high Reynolds number aerodynamic flows, with many directional features such as boundary layers, wakes, and shocks. In order to overcome the lack of information necessary to make anisotropy decisions, Venditti and Darmofal combined the DWR technique with an anisotropy detection based on the Hessian of the Mach number to drive output-based adaptation for the compressible Navier-Stokes equations.[7] Fidkowski and Darmofal[8] and

*Doctoral candidate, AIAA student member, 77 Massachusetts Ave. 37-442, Cambridge, MA, 02139, myano@mit.edu

†Professor, AIAA associate fellow, 77 Massachusetts Ave. 37-451, Cambridge, MA, 02139, darmofal@mit.edu

Leicht and Hartmann[9] later generalized the method to higher-order discretizations by using the higher-order derivative of the Mach number to guide their anisotropy decision. While the method produces anisotropic meshes, the choice of the Mach number to drive the anisotropic decision is arbitrary even for controlling the global quality of the solution. In the context of output-based adaptation, the method does not account for the behavior of the adjoint solution, which can have a different anisotropy requirement from the primal solution.

This work applies our optimization-based framework for anisotropic simplex mesh adaptation[10] to two-dimensional, steady-state aerodynamic flows. Utilizing the mesh-metric duality proposed by Loseille and Alauzet,[11] we cast the mesh adaptation problem as a continuous constrained optimization problem of the field $\{\mathcal{M}(x)\}_{x\in\Omega}$ of the Riemannian metric tensors that encode the approximability of the elements. Namely, we seek the optimal solution, $\mathcal{M}^*$, to the tensor field optimization problem

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

where $\mathcal{E}(\cdot)$ is the error functional, $\mathcal{C}(\cdot)$ is the cost functional, and N is the maximum permissible cost. A similar idea was recently pursued by Loseille et al.[11–13] The idea can be thought of as a generalization of the mesh optimization formulation for isotropic h-adaptation presented by Brandt,[14] Rannacher,[15] and Section 5 of Becker and Rannacher[3] to anisotropic h-adaptation. Our optimization framework differs from the previous approaches in that the error functional is approximated from the *a posteriori* error estimates, instead of from the *a priori* assumptions of the error function behavior. In particular, we first perform a series of local solves to survey the behavior of the local error as a function of the tensor field. The information gathered by the local sampling is then synthesized using a tensor-error interpolation technique, which incorporates a novel tensor manipulation framework developed by Pennec et al.[16] Finally, the mesh is driven toward optimality by manipulating the tensor field based on the reconstructed error function.

The idea of using local solves to guide the anisotropy decision for the DWR-based adaptation has been previously explored. Both Ceze and Fidkowski[17] and Georgoulis et al.[18] used local solves to guide their anisotropy decisions on quadrilateral meshes. However, the perspective set forth in these works are that of steepest descent in the discrete space where the local solves are used to guide a sequence of anisotropic subdivision of elements. A similar discrete optimization based approach for simplex meshes was pursued by Park[19] and Sun[20] using a sequence of edge splits. In this work, we consider a continuous optimization problem, and local solves are used to estimate the gradient required to iterate toward optimality. In other words, while the previous approaches consider a finite dimensional discrete optimization problem where the dimensionality is governed by the complexity of the current mesh, this work employs local solves as a means to solve the infinite dimensional optimization problem on the metric tensor field.

Our optimization method combines the rigor of the DWR error estimate and the flexibility of the anisotropic simplex mesh adaptation. Because the method is based on *a posteriori* output-based error estimates from the local samples, it automatically accounts for the primal and dual solution behaviors and works for arbitrarily high-order discretizations. The simplex mesh adaptation permits the generation of anisotropic elements in arbitrary orientation, which is crucial for resolving anisotropic features whose directions are not known *a priori*, such as wakes, stagnation streamlines, and oblique shocks. Previously, we have verified that the method produces optimal meshes in the context of L^2 error control and have shown its effectiveness for canonical advection-diffusion problems.[10] This paper demonstrates the effectiveness of our method in simulating practical two-dimensional aerodynamic flows.

II. An Optimization Framework for Anisotropic h-Adaptation

This section describes our optimization framework for anisotropic h-adaptation. The overall information flow for the feedback-based algorithm is shown in Figure 1, where the problem definition consists of the geometry, the flow condition, and the output of interest. The output is the computed engineering quantity of interest and the associated error estimate.

II.A. Output Error Estimation and Localization

Let us define the output estimation problem for a general steady state conservation law on a domain $\Omega \subset \mathbb{R}^d$, where d is the physical dimension. The governing equation is given by

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

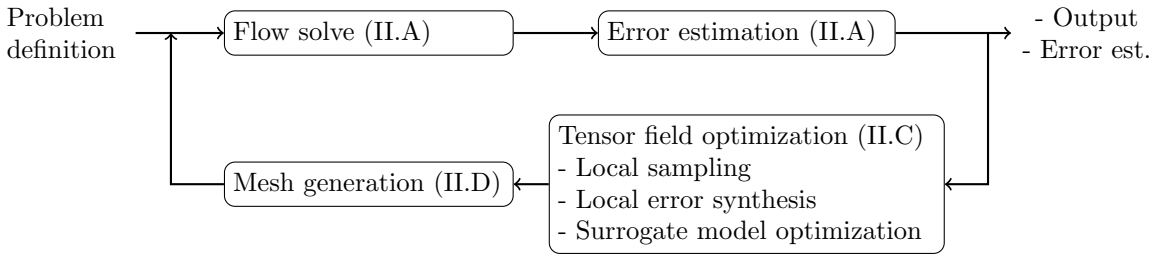


Figure 1. The information flow for the adaptive algorithm. The numbers in parenthesis correspond to the section numbers.

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

$$J = \mathcal{J}(u)$$

where $\mathcal{J}$ is the output functional of interest. For aerospace applications, $\mathcal{J}$ is often an integral quantity on surfaces (lift/drag estimation) or in the domain (sonic boom prediction).

An approximation to the desired output is obtained by discretizing the conservation law and evaluating the discrete output functional. In particular, this work employs a high-order discontinuous Galerkin (DG) finite element method, resulting in the 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 triangulation $\mathcal{T}_h$ of Ω, and $R_{h,p}(\cdot,\cdot) : V_{h,p} \times V_{h,p} \to \mathbb{R}$ is the semilinear form. This work uses Roe's approximate Riemann solver[21] for the inviscid flux, Bassi and Rebay's second discretization[22] (BR2) for the viscous flux, and Oliver's asymptotically dual-consistent discretization[23] for the source term. The nonlinear algebraic system resulting from Eq. (1) is solved using Newton's method with pseudo-time continuation and line search (see our recent work[24,25] for details). The linear system is solved using GMRES,[26] preconditioned with an in-place block-ILU(0) factorization[27] with minimum discarded fill ordering and $p = 0$ multigrid correction.[28] 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 the discrete functional that maintains dual consistency.[23,29,30]

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[2,3] to estimate the output error and to localize the error. For brevity, we omit the derivation of the method and state the main results; a recent review of the method applied to aerodynamic flows is provided by Fidkowski and Darmofal[31] and the references therein. 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

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

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

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 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 behavior of both the primal and adjoint solutions.

II.B. 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 $V_{h,p}$ and is often set by the available computational resource. 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.[11–13] We appeal to the fact that the metric field $\mathcal{M} = \{\mathcal{M}(x)\}_{x\in\Omega}$, consisting of symmetric positive definite (SPD) matrices $\mathcal{M}(x) \in Sym^+_d$, captures the approximability of the triangulation $\mathcal{T}_h$. With a metric-conforming mesh generator and a metric reconstruction scheme on the triangulation, we have the mesh-metric duality in the sense that

$$\mathcal{T}_h = \text{MeshGen}(\mathcal{M}) \quad \text{and} \quad \mathcal{M} \approx \text{MetricRecon}(\mathcal{T}_h),$$

where $\approx$ denotes that the two triangulations have similar approximability. Thus, we can cast a continuous relaxation of the discrete problem Eq. (4) as

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

For brevity, 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.

II.C. 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.[10]

II.C.1. Error Locality Assumption

In order to solve the error minimization problem Eq. (5), let us assume that the error functional results from a sum of the local contributions, i.e.

$$\mathcal{E}(\mathcal{M}) = \int_\Omega e(\mathcal{M}(x), x)dx,$$

where $e(\cdot,\cdot) : Sym_d^+ \times \mathbb{R}^d \to \mathbb{R}^+$ is the local error function that maps the local approximability described by $\mathcal{M}(x)$ to the local contribution of the output error. This locality assumption is formally only applicable to errors that only depend on local properties, e.g. L^2 projection errors. However, we have found that the algorithm developed based on the assumption works well in practice for output error control. Similarly, the cost functional takes the form

$$\mathcal{C}(\mathcal{M}) = \int_\Omega c(\mathcal{M}(x), x)dx,$$

where $c(\cdot,\cdot) : Sym_d^+ \times \mathbb{R}^d \to \mathbb{R}^+$ is the local cost function. In the view of metric-mesh duality,[11] the local cost function for DG methods is given by

$$c(\mathcal{M}(x), x) = c_p\sqrt{\det(\mathcal{M}(x))}, \tag{6}$$

where c_p is the number of degrees of freedom associated with a reference element normalized by its size. For example, a triangle, the elemental cost is $c_p = (2/\sqrt{3})(p+1)(p+2)$.

II.C.2. Tensor Manipulation Framework

Let us introduce a framework for manipulating a metric tensor, i.e. an SPD matrix. The framework we employ results from endowing the space of SPD matrices with an affine-invariant Riemannian metric introduced by Pennec et al.,[16] which produces a manifold structure suited for tensor manipulation. The change in the tensor is induced by the exponential map of a symmetric matrix $S \in Sym_d$, which we call the *step matrix*, in the tangent space about the current tensor, $\mathcal{M}_0$, i.e.

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

where $\exp(\cdot) : Sym_d \to Sym_d^+$ is the matrix exponential. Conversely, the logarithmic map of a tensor $\mathcal{M}$ to the tangent space about $\mathcal{M}_0$ provides the step matrix for the manipulation, i.e.

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

where $\log(\cdot) : Sym_d^+ \to Sym_d$ is the matrix logarithm. We note that, by construction, $\mathcal{M}(S)$ is SPD for any symmetric S; thus, no special constraint is necessary to maintain the positive definiteness.

To gain some insight, let us decompose the step matrix into the isotropic and tracefree parts, i.e.

$$S_\kappa = s_\kappa I + \tilde{S}_\kappa, \tag{7}$$

where $s_\kappa = \mathrm{tr}(S_\kappa)/d$ such that $\mathrm{tr}(\tilde{S}_\kappa) = 0$. 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 the volume.

II.C.3. Local Error Sampling

The goal of the local error sampling step is to probe the behavior of the local elemental error η_κ as a function of the local metric $\mathcal{M}_\kappa$. Here, we probe the functional dependency by directly monitoring the behavior of the *a posteriori* error estimate for several different configurations. The split configurations and the associated metrics are shown in Figure 2. For each configuration, we solve an element-wise local problem associated with κ_i. The local solution, $u_{h,p}^{\kappa_i} \in V_{h,p}(\kappa_i)$, satisfies

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

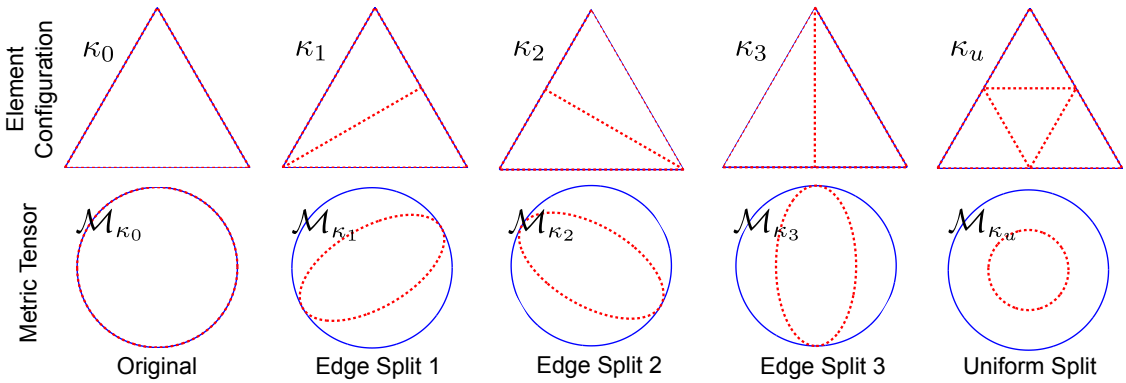


Figure 2. The original, edge split, and uniformly split configurations used to sample the local error function behavior. The metrics implied by the sampled configurations are shown in dashed lines.

where the local semilinear form, $\mathcal{R}^{\kappa_i}_{h,p}(\cdot,\cdot)$, sets the boundary fluxes on κ_i assuming the solution on the neighbor elements does not change. Then, we recompute the localized DWR error estimate corresponding to the subdivided mesh as

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

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

$$\eta_{\kappa_i} = |\mathcal{R}^{\kappa_i}_{h,p}(u^{\kappa_i}_{h,p}, (\psi_{h,\hat{p}} - \psi^{\kappa_i}_{h,p})|_{\kappa_0})|.$$

The 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 *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 $\{\mathcal{M}_{\kappa_i}, \eta_{\kappa_i}\}_{i=1}^{n_{\text{config}}}$.

II.C.4. Local Error Model Synthesis

The goal of the error model synthesis step is to construct an 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. Our interpolation framework builds on Pennec's affine invariant framework for tensor manipulation.[16]

As discussed in Section II.C.2, the logarithmic map of a metric about the original configuration $\mathcal{M}_{\kappa_0}$ provides a convenient means of characterizing the change in the approximability of the region over κ. Thus, we measure the changes in each configuration as

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

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

$$f_{\kappa_i} = \log\left(\eta_{\kappa_i}/\eta_{\kappa_0}\right), \quad i = 0, \ldots, n_{\text{config}}. \tag{8}$$

Again, the original error, η_{κ_0}, maps to zero by construction.

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,

$$f_\kappa(S_\kappa) = \text{tr}(R_\kappa S_\kappa). \tag{9}$$

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.

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

Thus, from Eq. (8) and (9), the local error model over the region covered by κ is given by

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

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

$$R_\kappa = r_\kappa I + \tilde{R}_\kappa, \tag{10}$$

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

II.C.5. Local Cost Function Model

We obtain the element-wise cost function model, ρ_κ, by directly integrating the continuous local cost function, Eq. (6), over an element, i.e.

$$\begin{aligned}\rho_\kappa(S_\kappa) &= \int_K c(\mathcal{M}(x), x)dx = \int_K c_p\sqrt{\det \mathcal{M}(x)}dx = \int_K c_p\sqrt{\det(\mathcal{M}_{K_0}^{1/2}\exp(S_K)\mathcal{M}_{K_0}^{1/2})}dx \\ &= \int_K c_p\sqrt{\det(\mathcal{M}_{K_0}^{1/2}\exp(s_K I + \tilde{S}_K)\mathcal{M}_{K_0}^{1/2})}dx = \rho_{K_0}\exp\left(\frac{1}{2}\mathrm{tr}(s_K d)\right).\end{aligned}$$

Note that the cost is only a function of s_K, which controls the scaling of the tensor.

II.C.6. 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.

$$\mathcal{M}_\nu(S_\nu) = \mathcal{M}_{\nu_0}^{1/2}\exp\left(S_\nu\right)\mathcal{M}_{\nu_0}^{1/2}. \tag{11}$$

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

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

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

$$\{S_\nu^*\}_{\nu\in\mathcal{V}} = \inf_{\{S_\nu\}_{\nu\in\mathcal{V}}}\mathcal{E}\left(\{S_\nu\}_{\nu\in\mathcal{V}}\right) \tag{12}$$

$$\text{s.t.} \quad \mathcal{C}\left(\{S_\nu\}_{\nu\in\mathcal{V}}\right) = N \tag{13}$$

$$|(S_\nu)_{ij}| \le \alpha, \quad i,j = 1,\ldots,d, \quad \forall\nu\in\mathcal{V}. \tag{14}$$

The last constraint, Eq. (14), 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[10] 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. (12)-(14). For simplicity, let us assume that the current configuration is sufficiently close to the optimal configuration such that the constraints Eq. (14) are inactive. The first order optimality condition is given by

$$\frac{\partial \mathcal{E}}{\partial s_\nu} + \lambda \frac{\partial \mathcal{C}}{\partial s_\nu} = 0, \quad \forall \nu \in \mathcal{V}, \tag{15}$$

$$\frac{\partial \mathcal{E}}{\partial \tilde{S}_\nu} = 0, \quad \forall \nu \in \mathcal{V}, \tag{16}$$

for some Lagrange multiplier λ, where $s_\nu = \mathrm{tr}(S_\nu)/d$ and $\tilde{S}_\nu$ is the trace-free part of S_ν. The first condition, Eq. (15), is a global condition for the size distribution. In particular, if we define the "local" Lagrange multiplier as

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

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. (16), 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

$$\tilde{R}_\kappa = 0, \quad \forall \kappa \in \mathcal{T}_h, \tag{17}$$

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 a $\tilde{S}_\nu$ 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.

II.D. High-Order Mesh Generation

From the new requested metric tensor field, an anisotropic mesh is generated using a metric-conforming mesh generator. In order to generate a mesh consisting of curved elements that capture high-order geometry information of curved surfaces, this work uses a two-step strategy developed by Oliver.[23] First, a linear anisotropic mesh (i.e. straight edged elements) is generated using BAMG.[32] Then, the linear mesh is globally curved using linear elasticity. A similar elasticity based approach was pursued by Persson et al. to generate high-order meshes.[33]

II.E. Properties of the Optimization Method

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

- The method takes into account both the primal and adjoint solution behaviors in manipulating the mesh. In particular, both the sizing decision and the anisotropy decision are driven directly by the behavior of the *a posteriori* error estimate, which automatically captures the behaviors of both the primal and dual solutions. The *a posteriori* error estimate also captures the behaviors of all the components of the state vector when applied to a system of equations.
- The method handles any discretization order.

- The method does not make any *a priori* assumption about the convergence behavior of the error, unlike the previous optimization based methods.[7,15,34] Instead, the 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 due to the presence of a singularity or a singular perturbation, and the actual convergence rate differs from the asymptotic convergence rate based on the polynomial interpolation theory.

- 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,[17,18] in which the allowable anisotropy is restricted by the topology of the initial mesh.

- The majority of the computational cost stems from the local, element-wise solves. The reconstruction of the local error function requires at least three local solves in two dimensions, and we currently use four local solves (three edge splits and uniform refinement). The cost increases to a minimum of six solves in three dimensions. However, local solves are perfectly scalable, and becomes a smaller fraction of the flow solve, which scales superlinearly, as the problem size grows.

The cases in the following result section are chosen to study the benefits of the above features in the context of aerodynamic simulations.

III. Results

III.A. The Assessment Procedure

We present numerical examples of applying the proposed optimization framework to aerodynamic problems. As a comparison, we also provide the results obtained using the method based on fixed-fraction marking and the Mach number-based anisotropy detection,[24] a modification of the algorithm developed by Fidkowski.[34] The fixed-fraction marking, which controls the size of the elements, is based on the DWR error indicator described in Section II.A. The anisotropy request is driven by the $(p+1)$-th derivative of the Mach number estimated by approximately solving the flow problem in the p-enriched space, $V_{h,p+1}$. Note that while the sizing decision accounts for the influence of the adjoint, the anisotropy decision is driven by a single scalar characterization of the primal solution. This approach will be referred to as the fixed-fraction Mach-anisotropy method, or FFMA, from here on.

We compare the output quality at several different numbers of degrees of freedom for each case. In particular, for each select number of degrees of freedom, we run several iterations of the optimization algorithm illustrated in Figure 1 and described in Section II until no further improvements in the error estimate is made. The number of iterations to reach stationarity depends on the quality of the initial mesh. For example, when a viscous problem is solved from an isotropic mesh only suited for Euler calculation, generating a mesh suitable for viscous calculation may take over 10 iterations. However, each of these iterations is cheap, since we perform the first optimization on a small problem. Once stationarity is achieved for the first select number of degrees of freedom, we scale the mesh to the next selection and repeat the optimization procedure. The optimization converges quickly for the second and all subsequent selection of degrees of freedom, as the initial mesh from the previous selection already captures the majority of the important flow features.

Throughout this section, we will assess the performance of the adaptive procedures by measuring the true output error rather than the error estimate. As the analytical solutions to the problems are not available, we approximate the true output by computing the solution in a space that is much richer than the solutions being compared by increasing the number of degrees of freedom and, in some cases, the polynomial order. To assess the quality of the reference solution, we first adaptively solve the problem of interest in the enriched space using both the proposed and the FFMA approach. If the error computed with respect to the two reference solutions is indistinguishable, then the reference solution is deemed accurate enough for the purpose of the assessment.

III.B. Laminar Flat Plate: $M_\infty = 0.2$, $Re_L = 10^5$

We first consider laminar flow over a flat plate. The problem is solved on a rectangular domain of size $[-0.5, 1.0] \times [0, 0.5]$ with the plate spanning from $x = 0.0$ to 1.0. The inflow Mach number is $M_\infty = 0.2$, the Reynolds number is $Re_L = 10^5$, and the adiabatic no-slip condition is specified along the plate. The output

of interest is the drag on the plate. This canonical problem tests ability of the optimization approach to produce anisotropic elements in the boundary layer and to control the effect of the leading edge singularity.

Figure 3 shows the convergence of the drag error for $p = 1$ and $p = 2$ DG discretizations adapted using the optimization framework and FFMA. The reference solution is obtained on an adapted $p = 3$, dof = 20,000 mesh. The convergence history shows that, for the $p = 1$ discretization, the optimization-based method produces four to five times smaller drag error than FFMA for a given problem size. Another interpretation is that the proposed method using 500 dof achieves a similar level of error as FFMA using 2,000 dof for $p = 1$. For the $p = 2$ discretization, the optimization framework performs significantly better than FFMA on a coarse mesh (e.g. dof = 250); the improvement means that the $p = 2$ discretization achieves the 1% error range ($\approx 10^{-5} c_d$) using less than half the dof for FFMA. Asymptotically, the $p = 2$ performance of the two adaptive schemes are similar.

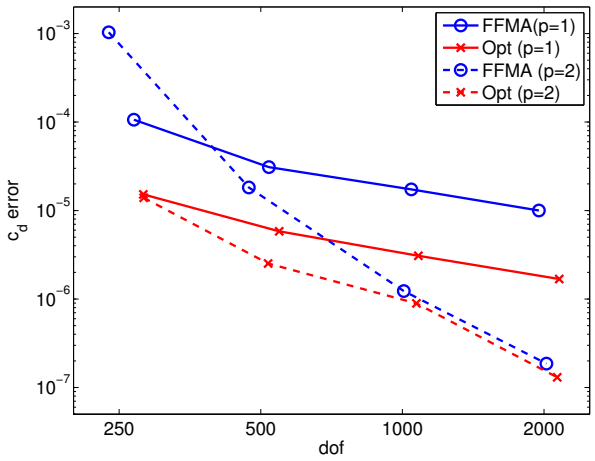
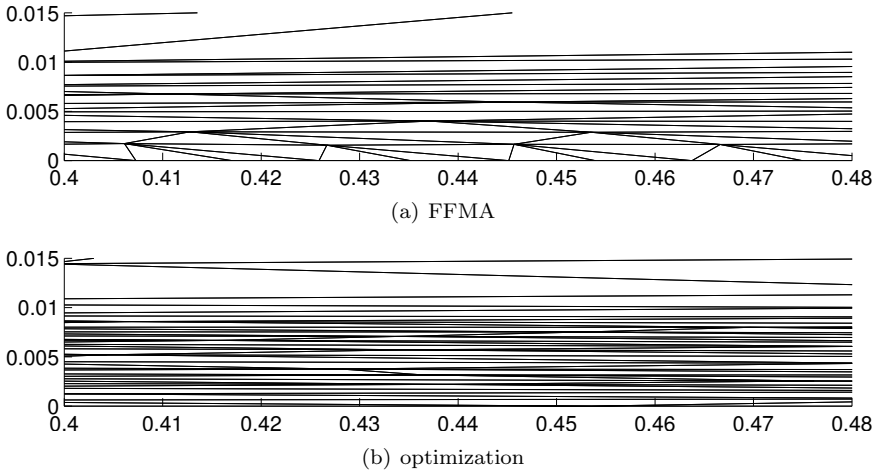


Figure 3. Drag error convergence for the laminar flat plate problem.



(a) FFMA

(b) optimization

Figure 4. Close views of the meshes for the laminar flat plate problem. ($p = 1$, dof $= 2,000$)

The difference in the output accuracy for the $p = 1$ case is due to the difference in the anisotropy of the elements used to resolve the boundary layer, as shown in Figure 4. When the Mach-based anisotropy detection is employed, the anisotropy of elements on the wall is limited, as the Mach profile has an inflection point at the wall. Having a vanishing second derivative, the Mach-anisotropy detection employs elements with relatively small aspect ratios on the wall. The proposed optimization approach employs elements with much higher aspect ratios, resulting in a smaller error for $p = 1$. The difference between the adaptation strategies is smaller for $p = 2$, as the third derivative of the Mach profile is large near the wall and FFMA employs highly anisotropic elements on the wall.

This simple case demonstrates a problem of using *a priori* knowledge of the solution behavior to control

anisotropy. While the Mach number has been found to be a good indicator for making the anisotropy decision in previous works,[7,34,35] there are instances where the indicator fails to capture the anisotropic behavior of the flow. The example also demonstrates that the ability of the Mach-anisotropy to produce the required anisotropy is dependent on the discretization order. In particular, while the inappropriate aspect ratio that results from the presence of inflection points in the Mach number is a known problem for second-order discretizations,[36] there could be instances where vanishing higher-order derivatives can lead to inappropriate aspect ratio for higher-order discretizations. The proposed optimization-based method driven by the *a posteriori* error estimates from the local solves automatically considers the behaviors of all state variables, providing robust anisotropy decisions for arbitrary-order discretization of system of equations.

III.C. RAE2822 Transonic RANS-SA: $M_\infty = 0.729$, $Re_c = 6.5 \times 10^6$, $\alpha = 2.31°$

We consider turbulent transonic flow over an RAE2822 airfoil. The Reynolds-averaged Navier-Stokes (RANS) equations with the Spalart-Allmaras (SA) equation[37] are solved in fully turbulent mode using the $p = 2$ DG discretization. The RANS-SA equation is modified for robustness following Oliver's work.[23] The shock capturing is provided by Barter's PDE-based artificial viscosity[38] with modifications for anisotropic meshes.[24] Each mesh consists of cubic ($q = 3$) elements representing the geometry, and the farfield is 200 chord lengths away. The output of interest is the drag on the airfoil. This standard RANS case requires accurate computation of the shock-boundary layer interaction and also exhibits multiple singular and singularly perturbed features.

Figure 5 shows the drag output convergence history. The reference solution is obtained on adapted $p = 3$, dof = 160,000 mesh. The convergence history shows that, combined with the high-order discretization, both anisotropic h-adaptation strategies achieve the drag error of fractions of a count using less than 40,000 degrees of freedom. The proposed optimization framework outperforms FFMA for all numbers of degrees of freedom considered.

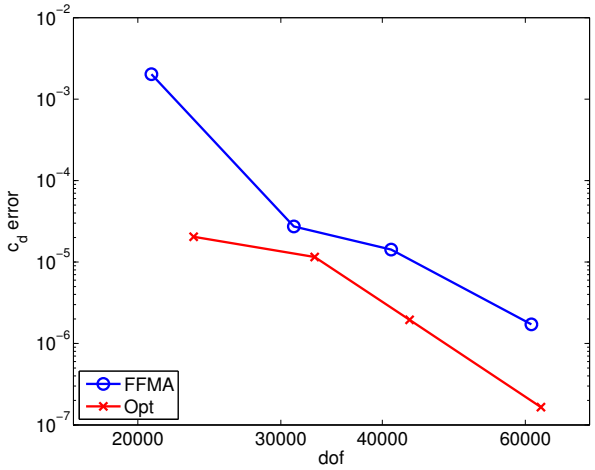
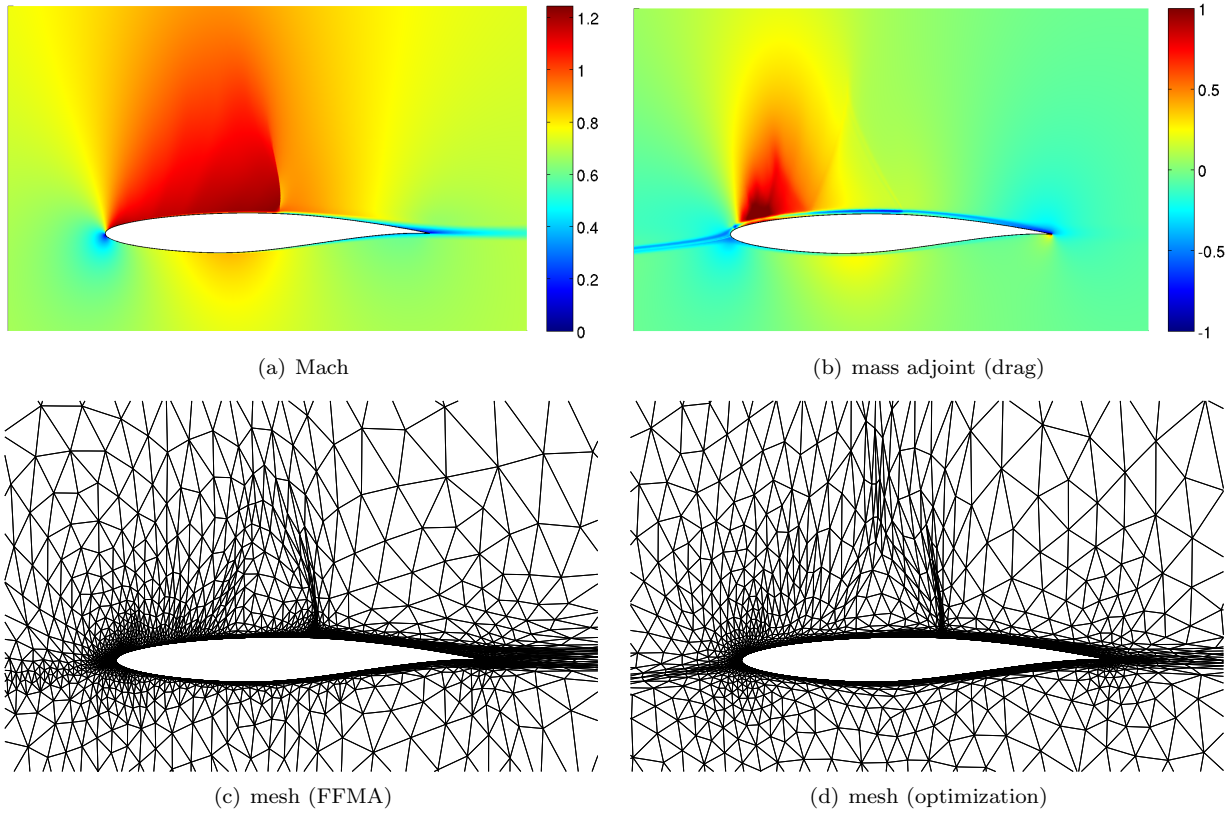


Figure 5. Drag error convergence for the RAE2822 Transonic RANS-SA problem. ($p = 2$)

The difference in the drag error convergence can be understood by comparing the meshes generated by the two anisotropic adaptation strategies, shown in Figure 6. In particular, recalling that the output error is a product of the primal residual and the dual error, we can compare the primal and the dual features targeted by the strategies. Both strategies target the boundary layers using highly anisotropic elements that have the aspect ratio approaching 10^3. Similarly, the shock is resolved using anisotropic elements. The key difference between the methods is the choice of elements used to resolve the stagnation streamline. Because the primal solution, and the Mach number in particular, does not exhibit anisotropic behavior along the stagnation streamline, Mach-anisotropy detection chooses isotropic elements along the stagnation streamline. However, the adjoint solution exhibits a wake-like feature along the stagnation streamline (of the primal solution), as shown in Figure 6(b). The proposed optimization approach employs anisotropic elements to resolve this feature, as the local *a posteriori* error estimates automatically accounts for both the primal and adjoint solution behaviors.



(a) Mach

(b) mass adjoint (drag)

(c) mesh (FFMA)

(d) mesh (optimization)

Figure 6. The Mach number, the mass adjoint, and the meshes for the RAE2822 Transonic RANS-SA problem. ($p = 2$, dof $= 60,000$)

III.D. NACA0006 Euler Supersonic Shock Propagation: $M_\infty = 2.0$, $\alpha = 0.0°$

We consider a problem of predicting the sonic boom generated by supersonic flow over a NACA0006 airfoil at zero degrees angle of attack. The Euler equations are solved using a $p = 2$ DG discretization, and the mesh is adapted for the pressure output 50 chord lengths below the airfoil. In particular, the output functional is given by

$$\mathcal{J}(u) = \int_{\Gamma_{\text{line}}} (p(u) - p_\infty)^2 ds,$$

where $p(u)$ is the pressure, p_∞ is the freestream pressure, and Γ_{line} is the line along which the pressure perturbation is measured. Meshes consist of cubic elements, and the farfield is 200 chord lengths away. This problem tests the ability of the adaptive schemes to propagate singular features over a long distance.

Figure 7 shows the convergence of the pressure line integral error. The reference solution is computed on an adaptive $p = 2$ discretization with 120,000 dof. The optimization approach shows approximately an order of magnitude improvement in the pressure line error compared to FFMA for the entire range of degrees of freedom considered.

To understand the difference in the pressure line errors, we compare the meshes obtained by the optimization method and FFMA, shown in Figure 8. As expected, both meshes employ highly anisotropic elements to resolve the shock formed in front of the airfoil and the shock emanating from the trailing edge.

The FFMA method uses anisotropic elements in the flow direction behind the trailing shock, which does not seem to be appropriate for this flow. These elements are generated due to negative interaction between the solver and adaptation algorithms. First, the numerical solution through the shock experiences $\mathcal{O}(h)$ noise, producing an artificial variation in the flow quantities along the shock direction.[38] Second, this variation is convected downstream with little dissipation due to the use of the high-order method, creating stream-wise streaks. Third, the anisotropy detection based on the higher order derivative of the Mach number captures these artificial streaks, requesting elements that are stretched in the stream-wise direction. Finally,

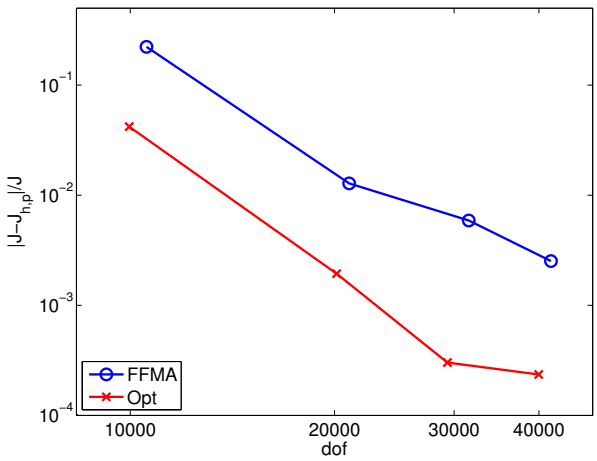


Figure 7. Pressure line error convergence for the NACA0006 Euler shock propagation problem.

the process worsens in the next adaptation iteration, as the stream-wise refinement of the elements results in generation of even smaller streaks. This case highlights the shortcomings of the anisotropy detection algorithm based on *a priori* convergence behavior of the solution, especially when a high-order discretization is applied to aerodynamic flows with low regularity.

Contrary to the FFMA method, the proposed optimization approach produces large, low aspect ratio elements behind the second shock. The method clearly wastes no degrees of freedom in this region. Driven by the anisotropy in the adjoint solution, the method employs highly anisotropic elements aligned with the shock direction in the region between the leading and trailing shocks. The close up of the mesh near the airfoil also shows that the method resolves complex adjoint features using anisotropic elements. Unlike the Mach-based anisotropy detection, the *a posteriori* error estimate based on local solves automatically captures the influence of the solution regularity to the local error. This in turn results in more robust assessment of required anisotropies and generation of more efficient meshes, when a high-order discretization is applied to flows with limited regularity.

III.E. Three-Element MDA High-Lift Airfoil RANS-SA: $M_\infty = 0.2$, $Re_c = 9 \times 10^6$, $\alpha = 8.1^\circ$

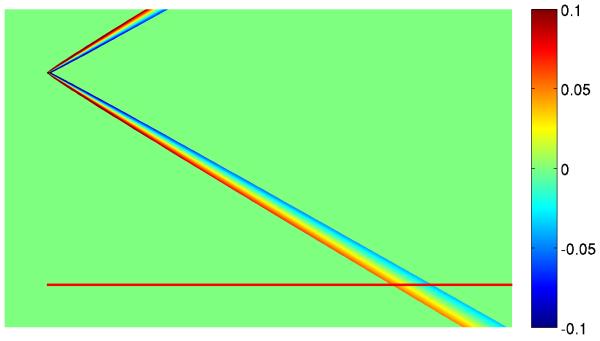
We now consider turbulent flow over a McDonnell Douglas Aerospace (MDA) three-element airfoil (30P-30N).[39] The RANS-SA equations are solved using the $p = 2$ DG discretization. Each mesh consists of cubic elements, and the farfield is 200 chord lengths away. The output of interest is the drag on the airfoil. This case requires accurate estimation of the interaction among the three elements of the airfoil.

Figure 9 shows the convergence of the drag error for the two adaptation strategies. The reference solution is obtained on an adaptive $p = 3$ discretization with 200,000 dof. The error levels achieved by the adaptation strategies are similar for each number of degrees of freedom. We conclude that the proposed optimization method is competitive to the FFMA method for this flow over the complex geometry.

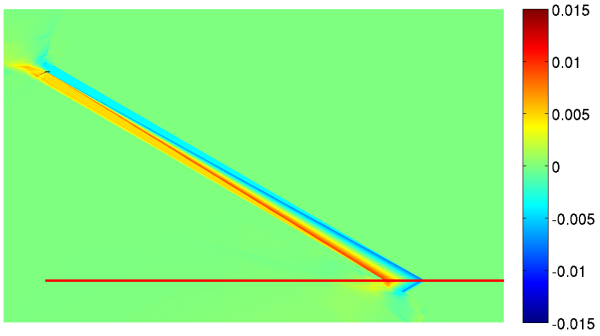
Instead of comparing the meshes generated by the two adaptation strategies, we focus on the mesh generated by the optimization approach for this complex flow shown in Figure 10. First, we note that all geometry surfaces are refined using highly anisotropic elements suited for resolving the boundary layers. Second, the wakes from the three elements are resolved using anisotropic elements aligned in the wake direction. Third, the stagnation streamlines are refined using anisotropic elements, as the adjoint solution exhibits wake-like features along the stagnation streamlines. In particular, the far-field view shows that the optimization method considers the upstream stagnation streamline to be at least as important as the downstream wake in computing the drag.

III.F. Multi-Element Supercritical 8 Transonic RANS-SA: $M_\infty = 0.775$, $Re_c = 2\times10^7$, $\alpha = -0.7^\circ$

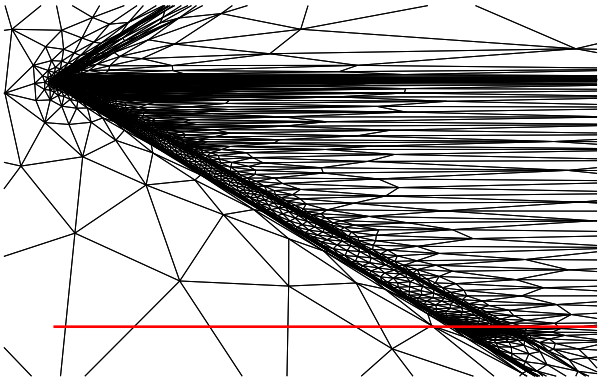
As the final example, we consider turbulent flow over a multi-element supercritical airfoil (MSC8). The original geometry with sharp trailing edges, provided by Drela,[40] is modified to have blunt trailing edges to facilitate adaptive meshing.[25] The farfield is 200 chord lengths away. The output of interest is the drag. The



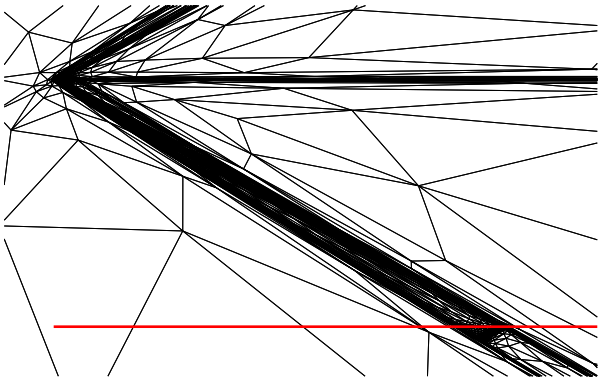
(a) pressure perturbation $(p - p_\infty)/p_\infty$



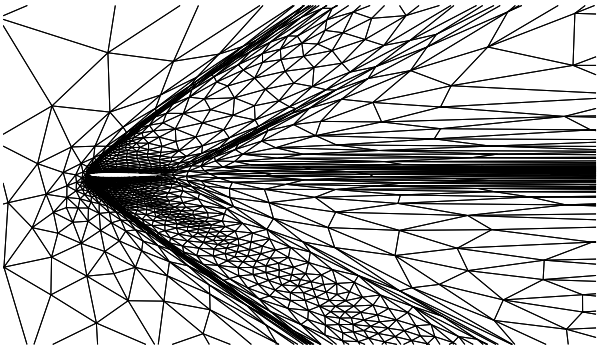
(b) mass adjoint (pressure line)



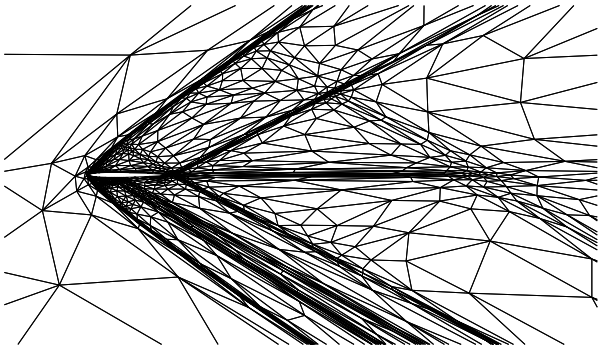
(c) mesh (FFMA)



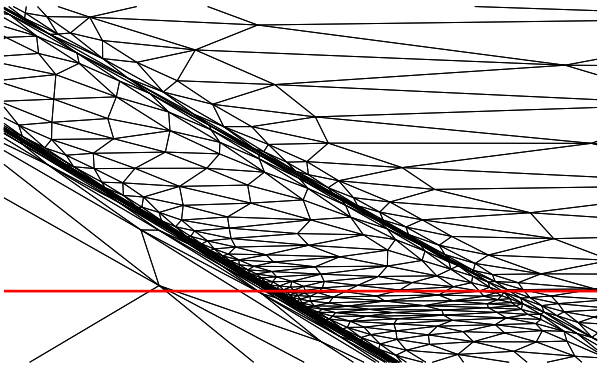
(d) mesh (optimization)



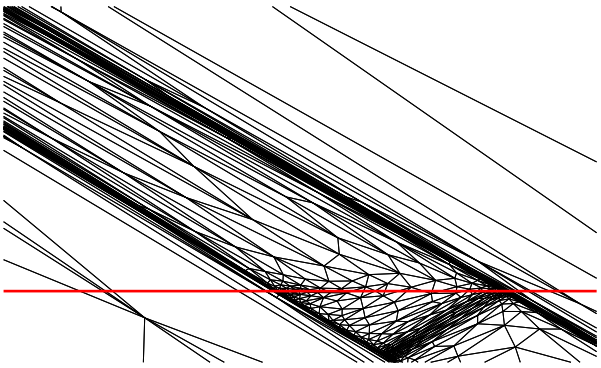
(e) mesh near field (FFMA)



(f) mesh near field (optimization)



(g) mesh line zoom (FFMA)



(h) mesh line zoom (optimization)

Figure 8. The pressure, the mass adjoint, and the meshes for the NACA0006 Euler supersonic shock propagation problem. ($p = 2$, dof $= 40,000$)

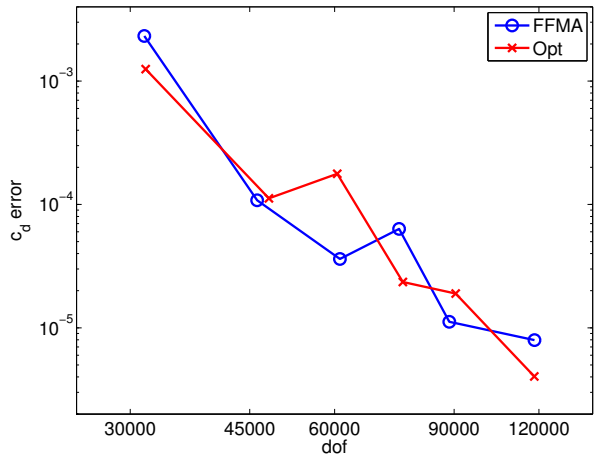
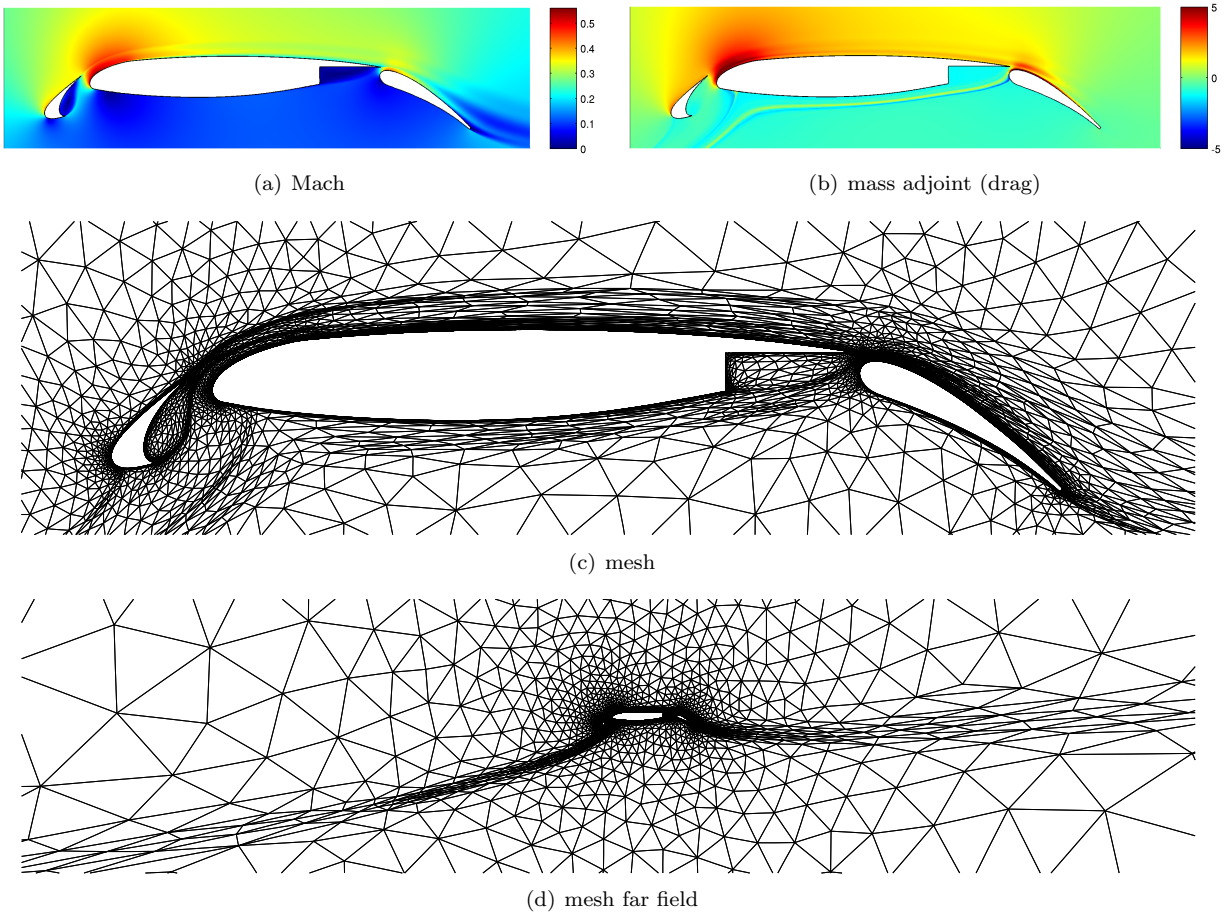


Figure 9. Drag error convergence for the three-element MDA airfoil RANS-SA problem.



(a) Mach

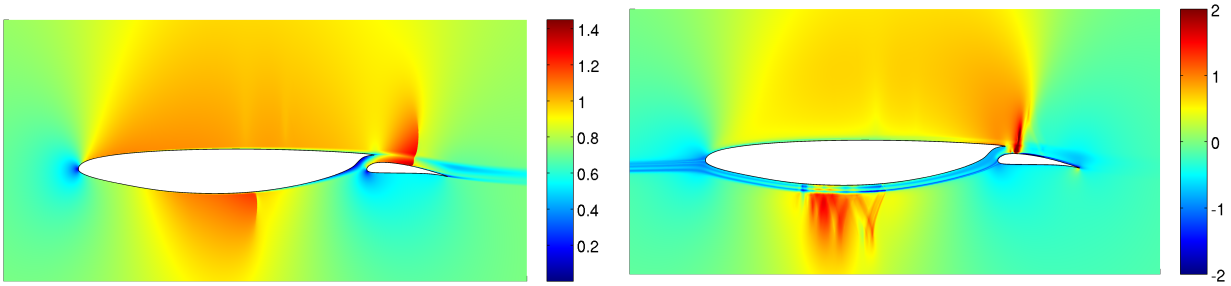
(b) mass adjoint (drag)

(c) mesh

(d) mesh far field

Figure 10. The Mach number, the mass adjoint, and the mesh for the three-element MDA airfoil RANS-SA problem. ($p = 2$, dof $= 120,000$)

solution to the problem is shown in Figure 11. Similar to the three-element airfoil case, this flow exhibits complex interactions between the main element and the flap; the presence of the two shocks introduces additional challenges in this case.

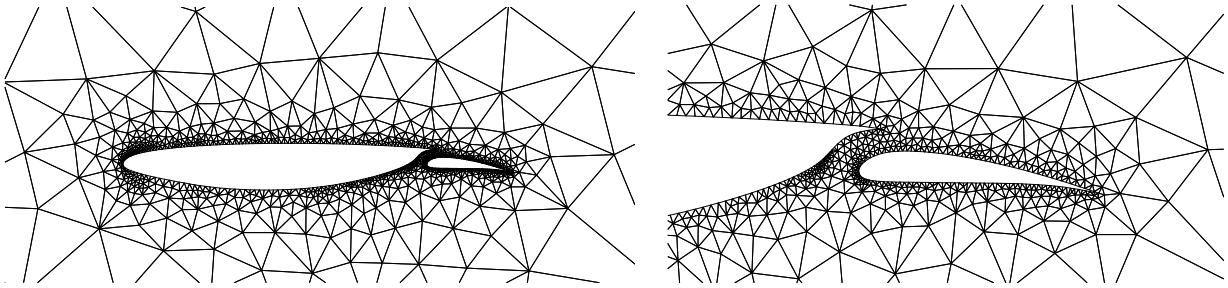


(a) Mach (b) mass adjoint (drag)

Figure 11. Solution to the MSC8 transonic RANS-SA problem.

III.F.1. The Initial Transition

Making the initial transition from an isotropic mesh, shown in Figure 12, to a mesh suitable for RANS calculation is particularly challenging for this problem. To illustrate the challenge, we consider the transition for the $p = 1$ discretization using 40,000 degrees of freedom.



(a) overview (b) flap zoom

Figure 12. The initial mesh for the MSC8 transonic RANS-SA problem.

The drag convergence histories for the transition stage are shown in Figure 13. The reference solution is computed on an adapted $p = 2$, dof $= 250,000$ mesh. The figure shows that the drag computed using FFMA does not approach the reference value, even though the optimization method shows that $40,000$ degrees of freedom is sufficient for the $p = 1$ discretization to approximate the drag to within 2 counts. To understand the cause of the failed adaptation, let us study the fifth mesh generated by the FFMA algorithm, shown in Figure 14, which is representative of the other meshes generated by the adaptive scheme. Because the flow is supersonic over the upper surface of the airfoil, any small non-smooth perturbation from the underresolved boundary layer can induce a shock over the upper surface. Note also that the boundary layer for this high Reynolds number flow is completely underresolved at this early stage of adaptation, and the presence of the boundary layer cannot be detected through the variation in the Mach number. As the artificial shocks are observable while the boundary layer is not, the FFMA scheme attempts to resolve the features above the upper surface using anisotropic elements aligned with the artificial shocks—elements aligned in the direction perpendicular to the boundary layer. Due to the use of inappropriate anisotropy, FFMA is unable to detect and resolve the boundary layer, and the transition to a RANS mesh fails even after 30 adaptation iterations.

Figure 13 shows that the optimization method makes a successful transition from the isotropic mesh to a RANS mesh, converging to the reference solution in about 15 adaptation iterations. The mesh obtained after five adaptation iterations is shown in Figure 15(a). Similar to the fifth FFMA-adapted mesh, the optimization approach also suffers from the presence of the artificial shocks on the suction side and uses shock-aligned anisotropic elements away from the boundary. However, right on the boundary, the method employs boundary-aligned anisotropic elements. With five more adaptation iterations, the optimization method generates a RANS mesh shown in Figure 15(b). The boundary layer is resolved using highly anisotropic

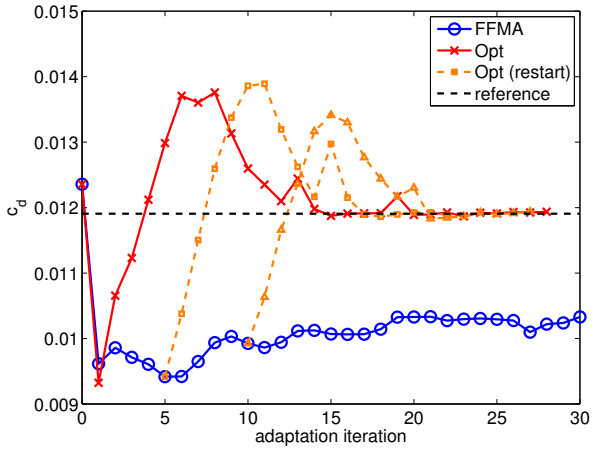
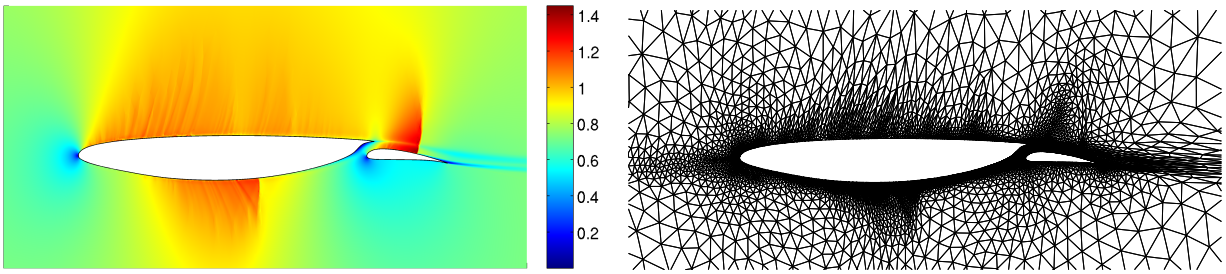
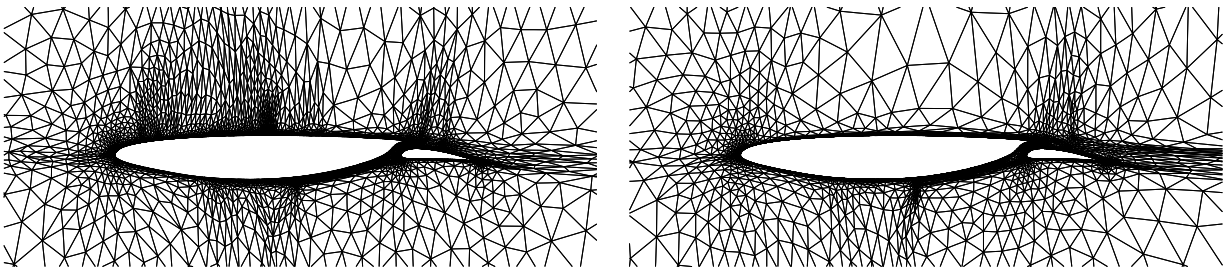


Figure 13. Drag adaptation histories for the $p = 1$, dof $= 40,000$ isotropic-to-RANS mesh transition test.



(a) Mach (b) mesh

Figure 14. The Mach number distribution and the mesh for the fifth adaptation iteration starting from the isotropic mesh in Figure 12 using FFMA ($p = 1$, dof $= 40,000$).



(a) 5th adapted mesh (b) 10th adapted mesh

Figure 15. The adapted meshes starting from the isotropic mesh in Figure 12 using the optimization method.

elements, the artificial shocks disappear, and the drag output rapidly converges to the reference value. To illustrate the reliability of this transition, the drag convergence histories starting from the fifth and tenth FFMA-adapted meshes are also shown in Figure 13. For both cases, the drag value converge to the reference value in about 15 iterations.

Our experience suggests that the optimization method can infer the presence of an anisotropic feature through DWR-based local sampling even if the feature is significantly underresolved. In other words, even on a coarse mesh for which the $p+1$ solution reconstruction—and the subsequent $p+1$-derivative-based anisotropy detection—is unreliable, the sampling-based anisotropy detection appears to behave correctly. Thus, the optimization method is more robust than FFMA in the presence of underresolved features.

III.F.2. $p = 2$ Convergence Results

Figure 16 shows the convergence of the drag error for FFMA and the optimization strategy. The reference solution is computed on an adapted $p = 2$, dof $= 250,000$ mesh. As FFMA is incapable of making an isotropic-to-RANS mesh transition, the initial RANS mesh for FFMA is constructed by performing several FFMA adaptation iterations starting from a RANS mesh prepared using the optimization method. The optimization method achieves about an order of magnitude lower error for a given number of degrees of freedom. Thus, the optimization strategy is not only more robust but also more efficient than FFMA for this complex, multi-element, multi-shock problem.

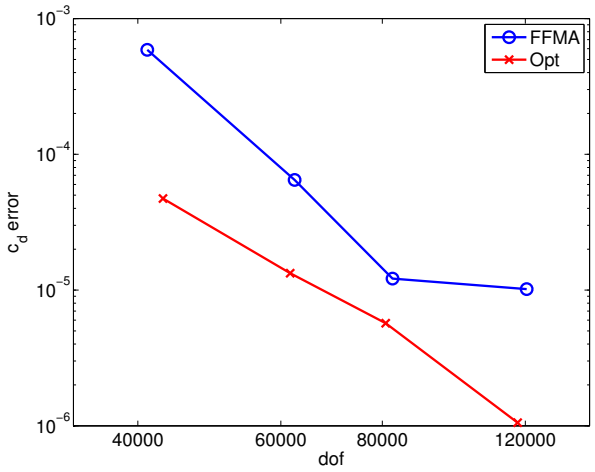


Figure 16. Drag error convergence for the MSC8 Transonic RANS-SA problem. ($p = 2$)

Figure 17 shows the $p = 2$, dof $= 80,000$ meshes generated by FFMA and the optimization method. The optimization strategy uses anisotropic elements along the stagnation streamlines emanating from the main element and the flap. The complex flow and adjoint features in the flap region can be inferred from the refinement pattern in the flap region.

III.G. Computational Cost

To analyze the computational cost, we decompose the time for a single adaptation cycle into:

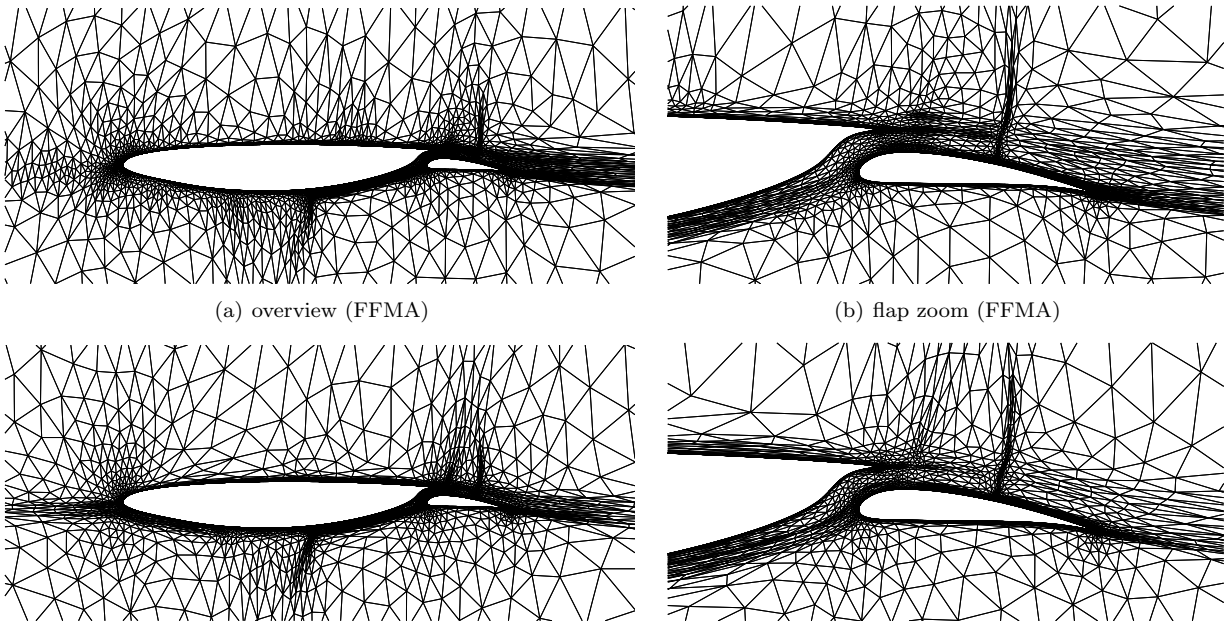
Primal solve: the time for solving the primal equation (i.e. the flow equation)

Dual solve: the time to obtain the $p+1$ degree surrogate solution to the dual problem

Adapt (FFMA): the time to perform 10 Newton iterations of $p+1$ degree primal solve to construct an approximate $p+1$ derivative

Adapt (Opt): the time to sample local errors, synthesize the errors, and optimize the surrogate error model

In the context of output error control, "primal solve" is the cost of computing the output, "dual solve" is the cost of endowing the output with an error estimate, and "adapt" is the cost of controlling and improving



(a) overview (FFMA) (b) flap zoom (FFMA)

(c) overview (optimization) (d) flap zoom (optimization)

Figure 17. The Mach number, the mass adjoint, and the meshes for the MSC8 Transonic RANS-SA problem. ($p = 2$, dof $= 80,000$)

the output error in the next solve. An effective adaptation algorithm must keep the cost of error estimation and control a fraction of the flow solve.

The first row of Table 1 shows a timing breakdown for the NACA0006 Euler shock propagation problem considered in Section III.D. Both FFMA and the optimization method use the same $p = 2$, dof $= 20,000$ mesh. For this case with small number of degrees of freedom, computing the dual surrogate solution and constructing an error estimate requires 17% of the flow solve time. For FFMA, the additional cost of constructing a $p + 1$ derivative approximation is 50% of the flow solve. For the optimization method, the additional adaptation cost is 43% of the flow solve, the majority of which stems from the local solves. For both FFMA and the optimization method, the additional cost for error estimation and control is a fraction of flow solve, even for this relatively small case. Moreover, the optimization method is not only more efficient than FFMA in terms of error-per-dof (as shown in Section III.D) but also is faster in terms of timing-per-dof.

Case	Primal (p)	Dual ($p+1$)	FFMA Adapt	Opt Adapt
NACA0006 Euler shock propagation ($p = 2$, dof $= 20000$)	1.000	0.174	0.495	0.431
RAE2822 transonic RANS ($p = 2$, dof $= 60000$)	1.000	0.092	0.157	0.072

Table 1. Timing breakdown normalized by the primal solve time.

As the second example, we consider a more complex flow: the RAE2822 transonic RANS case considered in Section III.C. The second row of Table 1 shows the timing breakdown on a $p = 2$, dof $= 60,000$ mesh. Due to the increased complexity of the problem, the nonlinear primal problem is harder to converge, and the relative cost of solving the dual problem, which is inherently linear, decreases to about 9% of the flow solve. Moreover, the local sampling cost for the optimization algorithm decreases to 7% of the flow solve cost. This decrease is attributed to two factors. First, even though RANS equations are highly nonlinear, each element-wise localized problem can still be solved in few Newton iterations for most of the cases. Second, the time for the local solves scales linearly with the number of elements, whereas the cost of the global linear solve scales superlinearly. As a result, the relative cost of the adaptation stage decreases with the problem complexity. We also note that the anisotropy detection by $p+1$ Mach derivative reconstruction requires 16% of the flow solve time, compared to the 7% of the optimization algorithm. Again, the optimization method is

not only more accurate (as shown in Section III.C) but also faster for a given number of degrees of freedom.

IV. Conclusions

In this work, we applied our optimization framework for anisotropic mesh adaptation to two-dimensional, steady-state aerodynamic flows. Some of the key features of the adaptation framework are: the use of *a posteriori* error estimates from fully scalable local solves; sizing and anisotropy decisions that account for primal and adjoint solution behaviors; unbiased treatment of all components of the state for a system of equations; applicability to arbitrarily high order discretizations; robust anisotropy decisions in the presence of low regularity features; and versatility of handing arbitrarily oriented solution features via simplex meshes. For the wide range of aerodynamic flows considered, the proposed optimization method was at least as competitive as the method based on fixed-fraction marking and Mach-based anisotropy detection, and in some cases produced over an order of magnitude improvement in the output error. In particular, as the method stems from the first principle of output error minimization and is guided by the *a posteriori* error behavior, it does not suffer from degradation of the performance when the flow includes features that violate *a priori* assumption of the error behavior. In terms of computational cost, the time spent on error estimation and adaptation is a small fraction of the flow solve time, and the relative cost decreases for complex problems requiring a larger number of degrees of freedom.

Acknowledgments

The authors would like to thank the ProjectX team for the many contributions and Dr. Steven Allmaras for insightful discussions during the course of this work. This work was supported by the Singapore-MIT Alliance Fellowship in Computational Engineering and The Boeing Company with technical monitor Dr. Mori Mani.

References

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

[2]Becker, R. and Rannacher, R., "A feed-back approach to error control in finite element methods: Basic analysis and examples," *East-West Journal of Numerical Mathematics*, Vol. 4, 1996, pp. 237–264.

[3]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.

[4]Venditti, D. A. and Darmofal, D. L., "Adjoint error estimation and grid adaptation for functional outputs: Application to quasi-one-dimensional flow," *Journal of Computational Physics*, Vol. 164, No. 1, 2000, pp. 204–227.

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

[6]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.

[7]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.

[8]Fidkowski, K. J. and Darmofal, D. L., "A triangular cut-cell adaptive method for higher-order discretizations of the compressible Navier-Stokes equations," *Journal of Computational Physics*, Vol. 225, 2007, pp. 1653–1672.

[9]Leicht, T. and Hartmann, R., "Anisotropic mesh refinement for discontinuous Galerkin methods in two-dimensional aerodynamic flow simulations," *International Journal for Numerical Methods in Fluids*, Vol. 56, 2008, pp. 2111–2138.

[10]Yano, M. and Darmofal, D., "An Optimization Framework for Anisotopic Simplex Mesh Adaptation," *Journal of Computational Physics*, submitted.

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

[12]Loseille, A. and Alauzet, F., "Optimal 3D highly anisotropic mesh adaptation based on the continuous mesh framework," *Proceedings of the 18th International Meshing Roundtable*, Springer Berlin Heidelberg, 2009, pp. 575–594.

[13]Loseille, A., Dervieux, A., and Alauzet, F., "On 3-D Goal-Oriented Anisotropic Mesh Adaptation Applied to Inviscid Flows in Aeronautics," AIAA 2010-1067, 2010.

[14]Brandt, A., "Multi-Level Adaptive Solutions to Boundary-Value Problems," *Mathematics of Computation*, Vol. 31, No. 138, 1977, pp. 333–390.

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

[16]Pennec, X., Fillard, P., and Ayache, N., "A Riemannian Framework for Tensor Computing," *International Journal of Computer Vision*, Vol. 66, No. 1, 2006, pp. 41–66.

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

[18]Georgoulis, E. H., Hall, E., and Houston, P., "Discontinuous Galerkin methods on *hp*-anisotropic meshes II: A posteriori error analysis and adaptivity," *Applied Numerical Mathematics*, Vol. 59, 2009, pp. 2179–2194.

[19]Park, M. A., *Anisotropic Output-Based Adaptation with Tetrahedral Cut Cells for Compressible Flows*, PhD thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, 2008.

[20]Sun, H., *Impact of triangle shapes using high-order discretizations and direct mesh adaptation for output error*, Masters thesis, Massachusetts Institute of Technology, Computation for Design and Optimization Program, 2009.

[21]Roe, P. L., "Approximate Riemann solvers, parameter vectors, and difference schemes," *Journal of Computational Physics*, Vol. 43, No. 2, 1981, pp. 357–372.

[22]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.

[23]Oliver, T. A., *A Higher-Order, Adaptive, Discontinuous Galerkin Finite Element Method for the Reynolds-averaged Navier-Stokes Equations*, PhD thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, June 2008.

[24]Yano, M., Modisette, J. M., and Darmofal, D., "The importance of mesh adaptation for higher-order discretizations of aerodynamic flows," AIAA 2011–3852, June 2011.

[25]Modisette, J. M., *An Automated Reliable Method for Two-Dimensional Reynolds-averaged Navier-Stokes Simulations*, PhD thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, 2011.

[26]Saad, Y. and Schultz, M. H., "GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems," *SIAM Journal on Scientific and Statistical Computing*, Vol. 7, No. 3, 1986, pp. 856–869.

[27]Diosady, L. T. and Darmofal, D. L., "Preconditioning methods for discontinuous Galerkin solutions of the Navier-Stokes equations," *Journal of Computational Physics*, 2009, pp. 3917–3935.

[28]Persson, P.-O. and Peraire, J., "Newton-GMRES preconditioning for Discontinuous Galerkin discretizations of the Navier-Stokes equations," *SIAM Journal on Scientific Computing*, Vol. 30, No. 6, 2008, pp. 2709–2722.

[29]Lu, J., *An a Posteriori Error Control Framework for Adaptive Precision Optimization Using Discontinuous Galerkin Finite Element Method*, Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, 2005.

[30]Hartmann, R., "Adjoint Consistency Analysis of Discontinuous Galerkin Discretizations," *SIAM Journal on Numerical Analysis*, Vol. 45, No. 6, 2007, pp. 2671–2696.

[31]Fidkowski, K. and Darmofal, D., "Review of output-based error estimation and mesh adaptation in computational fluid dynamics," Vol. 49, No. 4, 2011, pp. 673–694.

[32]Hecht, F., "BAMG: Bidimensional Anisotropic Mesh Generator," 1998,
`http://www-rocq1.inria.fr/gamma/cdrom/www/bamg/eng.htm`.

[33]Persson, P.-O. and Peraire, J., "Curved mesh generation and mesh refinement using Lagrangian solid mechanics," AIAA 2009-0949, 2009.

[34]Fidkowski, K. J., *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, June 2007.

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

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

[37]Spalart, P. R. and Allmaras, S. R., "A one-equation turbulence model for aerodynamics flows," *La Recherche Aérospatiale*, Vol. 1, 1994, pp. 5–21.

[38]Barter, G. E. and Darmofal, D. L., "Shock capturing with PDE-based artificial viscosity for DGFEM: Part I, Formulation," *Journal of Computational Physics*, Vol. 229, No. 5, 2010, pp. 1810–1827.

[39]Klausmeyer, S. M. and Lin, J. C., "Comparative results from a CFD challenge over a 2D three-element high-lift airfoil," NASA Technical Memorandum 112858, 1997.

[40]Drela, M., Personal Communication via email, Nov. 2010.