NASA/TP–2004–213003

Using High Resolution Design Spaces for Aerodynamic Shape Optimization Under Uncertainty

Wu Li and Sharon Padula
Langley Research Center, Hampton, Virginia

March 2004

NASA/TP–2004–213003

Using High Resolution Design Spaces for Aerodynamic Shape Optimization Under Uncertainty

Wu Li and Sharon Padula
Langley Research Center, Hampton, Virginia

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23681–2199

March 2004

Acknowledgments

We would like to thank Eric Nielsen for his assistance with using FUN2D to generate the simulation results.

The use of trademarks or names of manufacturers in this report is for accurate reporting and does not constitute an offical endorsement, either expressed or implied, of such products or manufacturers by the National Aeronautics and Space Administration.

Available from:

NASA Center for AeroSpace Information (CASI)
7121 Standard Drive
Hanover, MD 21076–1320
(301) 621–0390

National Technical Information Service (NTIS)
5285 Port Royal Road
Springfield, VA 22161–2171
(703) 605–6000

Abstract

This paper explains why high resolution design spaces encourage traditional airfoil optimization algorithms to generate noisy shape modifications, which lead to inaccurate linear predictions of aerodynamic coefficients and potential failure of descent methods. By using auxiliary drag constraints for a simultaneous drag reduction at all design points and the least shape distortion to achieve the targeted drag reduction, an improved algorithm generates relatively smooth optimal airfoils with no severe off-design performance degradation over a range of flight conditions, in high resolution design spaces parameterized by cubic B-spline functions. Simulation results using FUN2D in Euler flows are included to show the capability of the robust aerodynamic shape optimization method over a range of flight conditions.

Nomenclature

c	chord length of airfoil
c_d	drag coefficient
$\frac{\partial c_d}{\partial D}$	gradient of c_d with respect to D
$\frac{\partial c_d}{\partial \alpha}$	derivative of c_d with respect to α
c_l	lift coefficient
$\frac{\partial c_l}{\partial D}$	gradient of c_l with respect to D
$\frac{\partial c_l}{\partial \alpha}$	derivative of c_l with respect to α
c_l^*	target lift coefficient
D	design vector
$E(\cdot)$	mean of random variable
$\mathcal{F}$	feasible set for the design vector D
m	number of design conditions considered
M	free-stream Mach number
n	number of design variables
$p(u)$	probability density function of uncertainty parameter u
r	number of design conditions
x, y	coordinates of points on plane
u	uncertainty parameter
w	weighting coefficient
α	angle of attack
γ_{ave}	average rate of drag reduction at all design points
γ_{min}	minimum rate of drag reduction at all design conditions
ΔD	change in design vector
$\delta_{k,i}$, ρ_k	scalars defining the trust region
$\sigma^2(\cdot)$	variance of random variable
Ω	a given Mach range
$\langle \cdot, \cdot \rangle$	inner product in Euclidean space
$\|\| \cdot \|\|$	2-norm in Euclidean space

Subscripts and Superscripts

i	index for design condition
j	index for component of design vector
k	index for iteration or iterate

1 Introduction

Aerodynamic shape optimization refers to improvement in the performance of an aerodynamic surface under geometric shape constraints (such as minimum thickness of an airfoil) and performance constraints (such as the target lift requirements). Aerodynamic shape optimization involves computational fluid dynamics (CFD) analysis, sensitivity analysis, and nonlinear/nonconvex optimization. Because the performance measures are calculated by solving a system of partial differential equations (PDEs), aerodynamic optimization problems are PDE-constrained optimization problems. For surveys on aerodynamic shape optimization and related CFD issues, see references [1]– [4]. In references [1] and [3], Jameson examined the use of CFD as a tool for aircraft design. In reference [2], Jameson discussed ways to exploit computational simulations more effectively in the overall design process, with the primary focus on aerodynamic design, while recognizing the need for an integrated multidisciplinary design optimization (MDO). A general overview of industrial applications of aerodynamic shape optimization was given by Jameson and Vassberg in reference [4].

As computational resources get faster and CFD algorithms are improved, CFD analysis and shape optimization will inevitably play a larger role in all phases of the design process. These computational procedures will not take the place of wind tunnel and flight-testing but can maximize the benefits of expensive experimentation.

One major obstacle to increased use of high resolution design spaces in aerodynamic shape optimization is the existence of many (local) optimal solutions. Jameson et al. (ref. [5]) pointed out that "if one considers drag minimization of airfoils in two-dimensional inviscid transonic flow, then every shock-free airfoil produces zero drag, and thus optimization based solely on drag has a highly non-unique solution. Different shock-free airfoils can be obtained by starting from different initial profiles." Designers would use the existence of multiple solutions to tailor the design to the need of a specific application by using their experiences and knowledge of flow physics. On the other hand, automatic numerical algorithms might be confused by the existence of multiple local optimal solutions. As a result, different optimization methods lead to different optimal solutions due to the different search paths used in the optimization process. Choices of trust-region and search directions can all have noticeable impacts on the final optimal solution. From an aircraft designer's point of view, this dependency of the optimal solution on non-physical numerical tuning parameters is undesirable since the purpose of optimization is to use the underlying flow physics to search for a practical design. The dependency of the optimal solution on details of the optimization process makes designers question whether the optimization code is guided by the physics in the mathematical model or by other unknown factors. One way to avoid these questions is to choose a low resolution design space such that the optimal solution in that design space is likely unique. In the past, low resolution design space has been constructed so that the optimization procedure mimics the manual design process. That is to say, no matter what optimization method is used, the optimal solution preferred by designers will be recovered. Such a method has limited usefulness since experienced designers can readily produce comparable designs without using optimization.

Another obstacle to the use of aerodynamic shape optimization is the need to postprocess the resulting design. It is known that a high resolution design space tends to generate high frequency noise during the airfoil shape modification. Reuther et al. (ref. [6]) mentioned that "the use of every surface grid point as an independent design variable also has the disadvantage of introducing high-frequency noise into the gradient, that can cause simple descent methods to fail." Therefore, a common practice in airfoil optimization is to filter out high frequency components in the search direction during the optimization process (refs. [7]–[12]). Both explicit and implicit smoothing procedures for the steepest descent direction were used by Szmelter (ref. [13]) for optimization of transonic wings with three design conditions. The design principle behind using smoothed search directions is that efficient aerodynamic shapes are known to be smooth, as pointed out by Jameson and Vassberg (ref. [4]). In practice, an acceptable aerodynamic design must have characteristics that smoothly vary with small perturbations in shape and flow conditions.

To explain why search directions that induce high frequency oscillations must be avoided, we will establish a necessary and sufficient condition under which a linear prediction by the first-order Taylor expansion is accurate. A mathematical analysis of the gradients for the lift and drag shows that a linear prediction (using the linear Taylor expansion) is only accurate for smooth variations of the geometry. Such an analysis can explain why optimization methods succeed in a low resolution design space. More significantly, the analysis suggests a quantitative criterion for an optimization code to check whether a search direction is acceptable or not. This analysis leads to various methods for generating reliable descent directions even if the design space has very high resolution.

Another potential problem of using high resolution design space for lift-constrained drag minimization is unacceptable off-design performance degradation (ref. [14]). It is well-known that lift-constrained drag minimization with a single design point can generate an optimal design that is sensitive to variations in the design condition (see, e.g., ref. [15]). A weighted average of objective functions corresponding to various design conditions is substituted as the objective function to alleviate off-design performance degradation. This substitution leads to the standard multipoint optimization model. Unfortunately, the utility of multipoint optimization decreases with an increase in the number of design variables. For example, with 6 design conditions and 23 design variables (which are the coefficients of sinusoidal basis functions), Drela (ref. [14]) discovered that an optimization code tends to exploit insignificant physical scales during optimization iterations, which leads to off-design performance degradation and bumpy optimal airfoil shapes. In other words, given enough resolution in the design space, an optimization code might trade an insignificant improvement of aerodynamic performance at the design conditions for a severe off-design performance degradation (see ref. [16] for a mathematical explanation of how such a phenomenon is related to the resolution of the design space). This behavior is not the fault of the optimization code since the off-design performance is not included in the multipoint objective function.

Off-design performance degradation can be avoided with two approaches: (i) use a well-constructed design space so that it is impossible for an optimization code to "over-optimize" at the design points, or (ii) treat all design conditions as uncertainties in the design process and recast the aerodynamic optimization as an optimization problem under uncertainty (refs. [16] and [17]). By treating the design condition as an uncertainty parameter, one could explicitly make an optimization code aware of the need to optimize over a range of design conditions and help the optimization code to find robust optimal solutions (refs. [16]–[18]).

Finally, an interesting question in optimization research is how to use aerodynamic knowledge embedded in flow models to design better aerospace vehicles (that are beyond human intuition). To tackle this problem, we must make optimization codes "more intelligent." An optimization code can be improved in at least three ways: (i) use high-fidelity flow models and/or a more detailed configuration of an aerodynamic vehicle to reduce the error in the knowledge extracted from the mathematical models; (ii) use MDO models to make the optimization code aware that the actual aerodynamic performance depends on multidisciplinary aspects of the design, such as manufacturing tolerances and wing flexibility; and (iii) use optimization under uncertainty to make the optimization code aware that the aerodynamic design must adapt to a variety of operating conditions (e.g., high or low speeds, with and without flap deflections, fully loaded or empty). Progress in any of these research areas will bring MDO for aerospace vehicles closer to reality.

The main objective of this paper is to show how uncertainty modeling can make an optimization code more intelligent in searching for a robust optimal design in a very high resolution design space. In section 2, we discuss the importance of smooth airfoil changes and the effect of nonsmooth shape modifications on optimization algorithms. In section 3, we discuss a multiobjective optimization formulation for robust optimization with drag minimization as an example. In section 4, we combine the robust optimization strategies used in both references [16] and [17] to formulate a random multipoint optimization method with auxiliary drag constraints, which provides users with more options for finding a relatively smooth optimal airfoil with no off-design performance degradation in a high resolution design space. The numerical results in section 5 will show how uncertainty modeling, use of derivative information, parameterization of the design space, and choice of optimization procedures can all have significant impacts on the final optimal aerodynamic design. Concluding remarks are given in section 6.

2 Accuracy of Linear Prediction of Aerodynamic Coefficients

2.1 Accuracy of Sensitivity Derivatives of Aerodynamic Coefficients

The success of any gradient-based optimization procedure is directly related to the calculation of accurate sensitivity derivatives. The current state-of-the-art approaches for sensitivity analysis for a large number of design variables are based on either discrete or continuous adjoint methods. For example, Jameson applied the control theory approach to aerodynamic shape optimization that led to adjoint equations for sensitivity calculation derived from the potential and Euler equations (ref. [19]). Later, the method was extended to Navier-Stokes equations (ref. [20]). The control theory might be applied directly to the discrete flow equations that result from the numerical approximation of the flow equations by finite element, finite volume, or finite difference methods. This application leads directly to a set of discrete adjoint equations with a matrix that is the transpose of the Jacobian matrix of the full set of discrete nonlinear flow equations (ref. [19]). In theory, if the maximum length of the edges in the grid approaches zero, then the difference between the numerical derivatives obtained by continuous and discrete adjoint methods also converges to zero (see ref. [11] for some numerical confirmation of this hypothesis). Both continuous and discrete adjoint methods can generate quite accurate numerical derivatives for aerodynamic shape optimization. See references [11] and [12] for a summary on the formulation and development of adjoint methods for sensitivity calculations. Adjoint methods are widely used for sensitivity analysis in aerodynamic shape optimization (refs. [6, 7, 13], and [19]– [38]).

A sensitivity derivative code is usually verified by comparing its results with the ones generated by a finite difference method or a more reliable method based on complex variables (see refs. [39] and [40]) for a standard baseline configuration (such as NACA-0012 or RAE2822) with a particular parameterization of the boundary curve (such as a B-spline curve or a shape function based on Hicks-Henne bump functions). However, we believe that a sensitivity derivative code is truly verified when the predicted reduction of the objective function based on gradient information is consistently confirmed by the actual reduction of the objective function during the optimization process. Our current and previous numerical experiments (refs. [16] and [18]) indicate that the derivative information (with respect to shape parameters) at the current airfoil shape given by FUN2D (refs. [21], [41], and [23]) in Euler flows can be used reliably to predict the lift and drag coefficients of a slightly modified airfoil shape from iteration to iteration.

To use a gradient-based optimization method to minimize an aerodynamic quantity $\mathcal{L}$, we must first understand the conditions under which the derivatives can be used to provide an accurate prediction of the value of $\mathcal{L}$ at a new aerodynamic configuration. Mathematically, variational analysis is done for shape functions in an appropriate function space (such as a Sobolev space). The adjoint equations for derivative calculation derived by Jameson were formulated in terms of the variation of the derivative of the boundary transformation that maps the boundary of the flow domain to a circle (in two dimensions) or a closed surface (in three dimensions) (ref. [19]). Jameson's formulation of the adjoint equation implies that a linear prediction of an aerodynamic quantity is accurate only if the variation of the derivative of the boundary transformation is small.

Suppose that an aerodynamic quantity $\mathcal{L}$ (such as the lift or drag coefficient) is of the following form:

$$\mathcal{L}(\Gamma) = \int_{\Gamma} f_{\Gamma}(v)\ ds,$$

where Γ is the boundary of the domain for the flow solver (such as the airfoil or wing surface), $f_{\Gamma}(v)$ is defined by the solution of the flow equation, which depends on Γ, v is the vector of coordinates of a point, and the integration (ds) is done with respect to the surface area. For simplicity, we only consider the case when Γ is a two-dimensional curve and has the parametric form $v = g(t)$ for $0 \leq t \leq 1$. Then $\mathcal{L}(\Gamma)$ can be rewritten as

$$\mathcal{L}(\Gamma) = \int_0^1 f_{\Gamma}\Big(g(t)\Big) \cdot \|g'(t)\|\ dt,$$

where $g'(t)$ is the derivative of g with respect to t and $\|g'(t)\|$ denotes the length of $g'(t)$.

Let $(\Gamma + \Delta\Gamma)$ be a perturbation of the boundary curve with a parametric representation $v = g(t) + \Delta g(t)$, $0 \leq t \leq 1$. Then the variation in $\mathcal{L}$ with respect to $\Delta\Gamma$ (or Δg) can be calculated as follows:

$$\begin{aligned}\langle \delta_{\Gamma}\mathcal{L}, \Delta\Gamma\rangle &= \int_0^1 \Big[\langle \delta_{\Gamma} f_{\Gamma}, \Delta\Gamma\rangle + \langle \delta_v f_{\Gamma}, \Delta g\rangle\Big] \cdot \|g'(t)\|\ dt \\ &\quad + \int_0^1 f_{\Gamma}(g(t)) \cdot \langle g'(t), (\Delta g)'(t)\rangle \cdot \|g'(t)\|^{-1}\ dt, \end{aligned} \tag{1}$$

where $\langle A, B\rangle$ denotes the value of a linear operator A applied to an element B in an appropriate space. For example, if A and B are vectors in a finite-dimensional Euclidean space, then $\langle A, B\rangle$ denotes the dot product of A and B. In general, we have the following error estimate for the first-order sensitivity analysis:

$$\mathcal{L}(\Gamma + \Delta\Gamma) - \mathcal{L}(\Gamma) = \langle \delta_{\Gamma}\mathcal{L}, \Delta\Gamma\rangle + o\left(\int_0^1 \Big(\|(\Delta g)(t)\| + \|(\Delta g)'(t)\|\Big) dt\right). \tag{2}$$

From equation (1) we know that the variation of $\mathcal{L}$ with respect to Γ depends not only on the variation $\Delta\Gamma$ of Γ but also on its derivative $(\Delta g)'$. According to equation (2), the linear prediction of $\mathcal{L}$ in a neighborhood of Γ is accurate only if both Δg and $(\Delta g)'$ are small. Note that $(\Delta g)'$ is small if and only if the length of $\Delta\Gamma$ is small.

2.2 Importance of Smooth Shape Modification

The previous analysis sheds new light on the importance of using smooth shape modifications during the optimization process. For example, consider a parameterization of the boundary curve based on some shape functions:

$$v = \varphi_0(t) + \sum_{i=1}^{m} a_i \varphi_i(t) \quad \text{for } 0 \leq t \leq 1,$$

where $\varphi_0(t)$ represents the baseline shape and $\varphi_i(t)$ are shape functions. By using the chain rule, we can get the gradient $\nabla_a \mathcal{L}$ of $\mathcal{L}$ with respect to $a = (a_1, \ldots, a_m)$. The linear expansion of $\mathcal{L}$ at the point a can be written as

$$\mathcal{L}(\Gamma + \Delta\Gamma) - \mathcal{L}(\Gamma) = \langle \nabla_a \mathcal{L}, \Delta a \rangle + o(\|\Delta a\|) \quad \text{as } \|\Delta a\| \to 0.$$

However, such an estimate does not provide as much information as equation (2), especially when the number of shape functions m is relatively large.

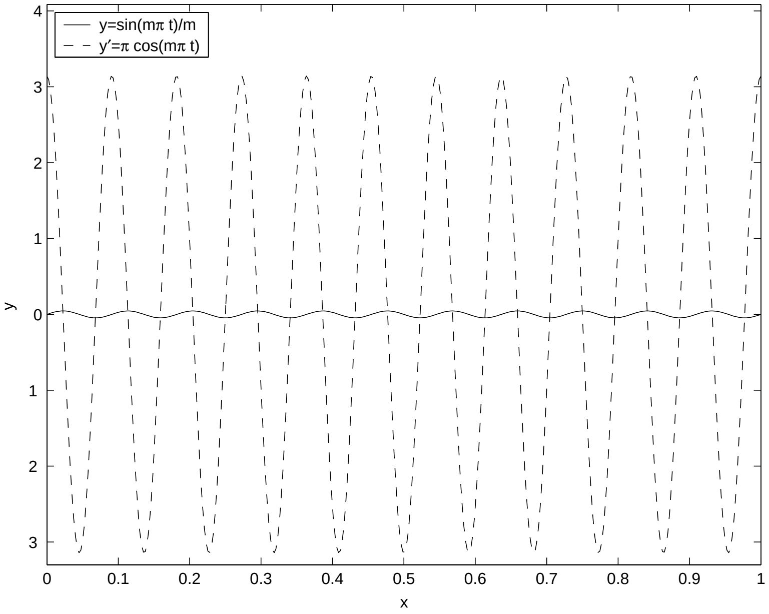

Figure 1. Curves of sinusoidal function (with $m = 22$) and its derivative.

Note that $(\Delta g)(t) = \sum_{i=1}^{m} \Delta a_i \cdot \varphi_i(t)$ and $(\Delta g)'(t) = \sum_{i=1}^{m} \Delta a_i \cdot \varphi_i'(t)$. If $\varphi_i'(t)$ is very large when compared with $\varphi_i(t)$, then a small variation Δg does not necessarily mean a small $(\Delta g)'$. This behavior could happen if the functions φ_i are either the sinusoidal basis functions or the B-spline basis functions. When m is large, the corresponding sinusoidal basis functions are very oscillatory. The oscillation leads to large values of φ_i' (see fig. 1). For the B-spline basis functions, a large value of m means that each basis function is positive on a small interval (called the support) and zero everywhere else. This correlation makes a B-spline basis function behave like a δ-function with a fast-rising peak near the center of its support, which also leads to a large derivative function (see fig. 2).

Let us use the following numerical example to demonstrate the potential adverse impact of the magnitude of $(\Delta g)'$ on the accuracy of linear predictions of lift and drag. We use an airfoil obtained after 50 optimization iterations in reference [18] as the baseline shape

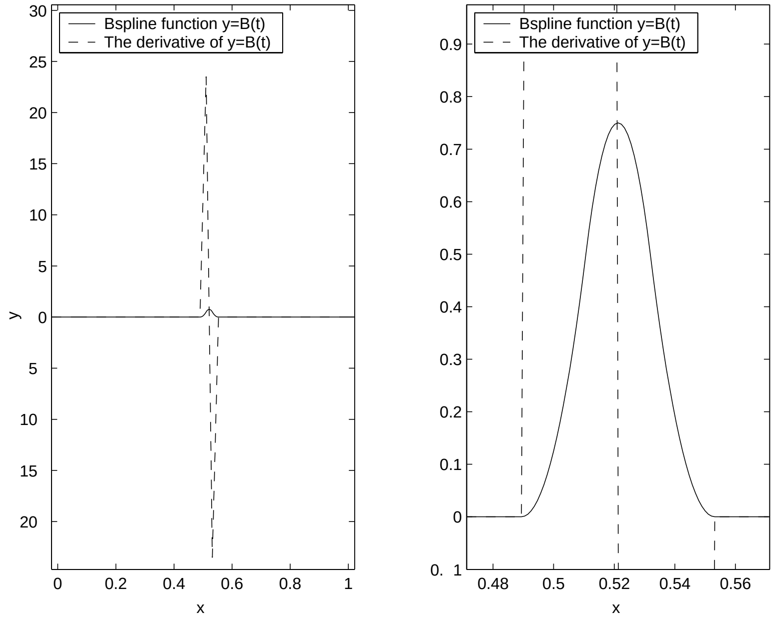

Figure 2. Curves of quadratic B-spline function (with 48 knots equally spaced on $[0, 1]$) and its derivative.

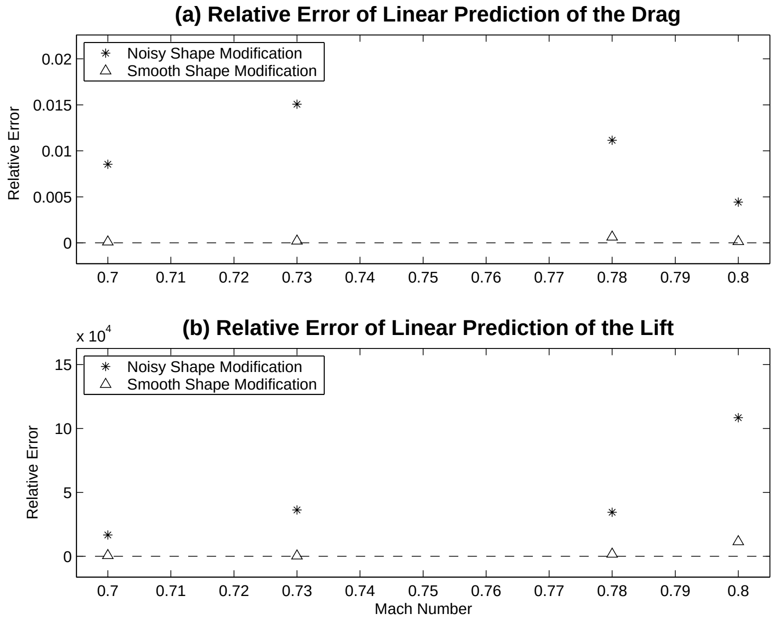

Figure 3. Accuracy of first-order Taylor expansion with respect to smooth and noisy shape modifications.

because an adverse impact of a large $(\Delta g)'$ can be clearly seen. The airfoil is represented by B-splines (as given in fig. 6) with 50 control points. Three control points (one at the leading edge and two at the trailing edge) are fixed. The y-coordinates of the remaining 47 control points, denoted by D, and the angle of attack are used as the independent variables. A new descent direction ΔD is calculated with $||\Delta D||_\infty \approx 10^{-3}$. Then we artificially generate a smooth shape modification ΔD_{smooth} and a noisy shape modification ΔD_{noisy} as follows:

$$(\Delta D_{\text{smooth}})_i = |(\Delta D)_i| \quad \text{and} \quad (\Delta D_{\text{noisy}})_i = (-1)^i |(\Delta D)_i| \quad \text{for } 1 \le i \le 47.$$

Note that ΔD_{smooth} and ΔD_{noisy} are identical in terms of the size of their components, but ΔD_{noisy} represents an oscillatory modification of the airfoil shape and ΔD_{smooth} represents a smooth modification of the airfoil shape. The relative errors of the linear prediction of the drag at D are

$$\epsilon_i\Big(\Delta \bar{D}\Big) = \frac{\left| c_d\Big(D + \Delta\bar{D}, \alpha_i + \Delta\alpha_i, M_i\Big) - c_d\Big(D, \alpha_i, M_i\Big) - \langle \frac{\partial c_d}{\partial D}, \Delta\bar{D}\rangle - \Delta\alpha_i \frac{\partial c_d}{\partial \alpha_i}\right|}{c_d\Big(D, \alpha_i, M_i\Big)},$$

where α_i is the angle of attack corresponding to the Mach number M_i. Figure 3(a) shows $\epsilon_i\Big(\Delta D_{\text{noisy}}\Big)$ and $\epsilon_i\Big(\Delta D_{\text{smooth}}\Big)$ for the linear prediction of drag coefficients. Similarly, figure 3(b) shows the relative errors of the linear prediction of the lift at D. In both cases, the linear prediction of the drag or lift with respect to noisy shape modification is significantly less accurate than the linear prediction with respect to smooth shape modification.

2.3 Conditions for Accurate Linear Prediction

The previous example can be used to explain why a simple descent method might fail for aerodynamic shape optimization with a high resolution design space. Note that if the design space has very high resolution, then a numerical approximation of the gradient will contain high frequency components. High frequency components have two possible sources: (i) numerical approximation errors and (ii) characteristics of the underlying high resolution design space. A simple descent method uses a constant multiple of the gradient to modify the original airfoil (or wing) shape. Therefore, if $\Delta\Gamma$ contains high frequency components inherited from the numerical gradient, then $(\Delta g)'$ will be much larger than Δg. As a consequence, the linear prediction of the value of the objective function at the next iterate might be totally wrong (see the residual term in eq. (2)), which could lead to failure of the descent method (i.e., the objective function value at the new iterate is actually worse than its value at the current iterate). However, if a smoothed gradient is used, then Δg does not have high frequency noise and $(\Delta g)'$ is as small as Δg. Thus, the linear prediction will match the actual objective function value at the next iterate, and the descent method will operate correctly.

To demonstrate the importance of using smooth corrections of the airfoil shape during the optimization process, we offer the following example. If a noisy airfoil is unintentionally generated for some iteration step, it will be very difficult for the optimization code to recover from such a mistake. To produce a smooth optimal airfoil and undo the noisy modification made in the previous step, Δg must be noisy in an equal and opposite direction; however, a noisy Δg recommended by the linear prediction model might not be reliable. In this case, a standard gradient-based optimization method will use the unreliable linear prediction to make a shape modification and, as a consequence, either the optimization process fails to make any progress or random numerical errors (due to unreliable linear predictions) will contaminate the final optimal solution.

A small $(\Delta g)'$ could also be achieved by using a smoothing procedure for the search direction to ensure smooth shape modifications during the optimization process (refs. [7]–[13]). Intuitively, larger values of φ_i' require a smaller change in coefficients Δa_i to keep $(\Delta g)'$ at a given level where the linear prediction model is accurate. This requirement gives an incentive for using small step size in gradient-based optimization methods for aerodynamic optimization (refs. [11] and [12]). For airfoil parameterization based on B-spline functions, we can use a constraint on the variance of $\Delta a_1, \ldots, \Delta a_m$ to achieve smooth modifications of airfoils. In fact, for B-spline basis functions, we have $\sum_{i=1}^{m} \varphi_i(t) = 1$ (which is called the partition of unit property of the B-spline basis functions (ref. [42])). Thus, $\sum_{i=1}^{m} \varphi_i'(t) = 0$. As a consequence, a small variance in the data sequence $\{\Delta a_1, \ldots, \Delta a_m\}$ implies a small derivative of Δg. Note that one can get a smoothed descent direction by using other data smoothing techniques (such as fitting a set of the data by curves with a given magnitude of derivatives (ref. [43])). However, we can make the optimization method more intelligent by explicitly imposing bound constraints on $(\Delta g)'$ in the trust region formulation for the search direction. For example, one could use the following condition to ensure a smooth shape modification:

$$\int_0^1 \|(\Delta g)'(t)\|^2 \, dt \leq \lambda \int_0^1 \|\Delta g(t)\|^2 \, dt$$

or

$$\max_{0 \leq t \leq 1} \|(\Delta g)'(t)\|_\infty \leq \lambda \max_{0 \leq t \leq 1} \|(\Delta g)(t)\|_\infty,$$

where $\|v\|_\infty$ is the supremum norm of a vector v. We can also use the ratio $\|(\Delta g)'\|/\|\Delta g\|$ as a quantitative measurement of the quality of the descent direction Δg. In general, a larger $\|(\Delta g)'\|/\|\Delta g\|$ means a less reliable descent direction.

Later in this paper, we will show that incorporating uncertainty modeling and design principles into the optimization procedure can make the procedure smart enough to generate relatively smooth descent directions automatically. This approach avoids inaccurate linear prediction of aerodynamic coefficients and leads to stable reduction of the objective function from iteration to iteration until a local optimal solution is found.

3 Multiobjective Optimization Formulation for Drag Minimization Under Uncertainty

The main purpose of this section is to provide a multiobjective optimization formulation for aerodynamic shape optimization under uncertainty. Mathematically, this formulation solves a more difficult problem than the traditional multipoint formulation, but it does not require an accurate approximation of the mean and variance of the objective function and constraints for its solutions.

First we would like to point out that optimization under uncertainty is a very broad topic, including reliability-based optimization in engineering (ref. [46]), robust optimization in engineering design processes (refs. [47]– [49]), and financial optimization (refs. [50] and [51]). Recently, optimization under uncertainty was applied in MDO and aerodynamic shape optimization (refs. [16]– [18, 44, 45], and [52]– [55]).

Some problems in optimization under uncertainty can be formulated as deterministic optimization problems. Such formulations seek the best solutions that optimize the objective function in the worst case scenario (for the uncertainty parameters) and have been applied in truss topology design (refs. [47] and [48]). However, these deterministic formulations are known to be NP-hard problems that are computationally intractable (refs. [47] and [48]), except the cases when the uncertainty parameters enter as linear terms in the objective function or the constraint functions. Since aerodynamic shape optimization problems are nonlinear and nonconvex in terms of Mach number M and target lift coefficient c_l^*, such a deterministic formulation renders the problem computationally intractable.

For probabilistic methods to solve optimization problems under uncertainty, the key is in the calculation of statistical quantities, such as the mean and variance, and their derivatives. The classical robust optimization approach is to minimize both the mean and variance, which is a multiobjective optimization problem with the mean and variance as the objective function. In the following we restrict our discussion to a robust optimization formulation for aerodynamic shape optimization applications.

Assume that a vector of uncertainty parameters u is specified for aerodynamic shape optimization. The case of $u = M$ (the Mach number) was discussed in references [16]– [18,44], and [45]. Other candidates for uncertainty parameters are Reynold's number, required lift c_l^*, maximum pitching moment and flap setting. Then, the lift-constrained drag minimization under uncertainty can be formulated as the following robust optimization problem:

$$\min_{D,\alpha(u)} \Big(E(c_d), \sigma(c_d)\Big) \tag{3}$$

subject to

$$D \in \mathcal{F} \quad \text{and} \quad c_l(D, \alpha(u), u) = c_l^*(u) \quad \text{for } u \in \Omega. \tag{4}$$

Here $c_l^*(u)$ is the target lift requirement for the flight condition corresponding to u, $\mathcal{F}$ is a given feasible set for geometric design variables (that could be defined by geometry constraints such as the maximum thickness constraint), and $\alpha(u)$ is the angle of attack corresponding to the uncertainty vector u. The drag and lift coefficients are c_d and c_l, respectively. The mean and variance of c_d are defined as

$$\left.\begin{aligned} E(c_d) &= \int_\Omega c_d(D, \alpha(u), u) \cdot p(u)\ du, \\ \sigma^2(c_d) &= \int_\Omega \left[c_d(D, \alpha(u), u) - E(c_d)\right]^2 p(u)\ du, \end{aligned}\right\} \tag{5}$$

where $p(u)$ is a probability density function of u and Ω is the domain of the uncertainty vector.

Unfortunately, equation (3) is only useful to satisfy mathematical curiosity. It is quite expensive to compute c_l and c_d for any given parameter u, and only crude approximations of $E(c_d)$ and $\sigma(c_d)$ can be calculated in practice. It is well-known in the aerodynamic optimization community that multiple design conditions can be used in aerodynamic shape optimization to produce a shape that is appropriate for a variety of flight conditions. The standard approach is to replace Ω by a finite sample set and then to find a solution to the following multipoint optimization problem:

$$\min_{D,\alpha_1,\ldots,\alpha_r} \sum_{i=1}^{r} w_i \cdot c_d(D, \alpha_i, u_i) \tag{6}$$

subject to

$$D \in \mathcal{F} \quad \text{and} \quad c_l(D, \alpha_i, u_i) = c_{l,i}^* \quad \text{for } 1 \le i \le r, \tag{7}$$

where $\{u_1, \ldots, u_r\}$ is a set of sample points in Ω and w_i are weighting factors that may include the effect of $p(u)$. Note that this approach can also be used to solve inverse design optimization by using given pressure distributions to guide the optimization method (ref. [56]).

The multipoint optimization approach uses a few flight conditions to deal with off-design performance degradation. However, aggressive drag minimization at the design conditions could lead to severe off-design performance degradation. In a special case of equation (6) where the uncertainty parameter is the Mach number M (i.e., $u = M$ is a scalar), Drela (ref. [14]) discovered that if r is much smaller than the number of free design variables, then the optimal solution of equation (6) could have severe off-design performance degradation and the optimal airfoil shape could have visible bumps corresponding to the design conditions. Drela called this phenomenon point-optimization behavior. A mathematical argument was given in reference [16] to explain why it is necessary to use more design points than the number of free design variables to avoid undesirable trade-off of marginal improvement at the design points and severe off-design performance degradation. However, since computation of the aerodynamic coefficients and their derivatives is very expensive, it is not practical to use many design points.

A compromise approach for aerodynamic shape optimization for a range of flight conditions is to use a few design conditions during the optimization process (so it is computationally tractable) and to find a Pareto solution of a multiobjective optimization problem while trying to avoid off-design performance degradation (refs. [16, 18], and [45]). For example, to overcome trading marginal improvement at the design points with severe degradation of off-design performance when only a few design conditions are used, Huyse and Lewis (ref. [17]) and Huyse et al. (ref. [45]) incorporate a random sampling of design points for each of the optimization iterations. The motivation for random sampling of design points during optimization is to reduce the bias of estimated mean value of the drag due to lack of sample size (i.e., the number of design points) and make the optimal solution independent of particular set of design points. Li, Huyse, and Padula (ref. [16]) use the profile optimization method for drag reduction over the whole range of Mach numbers. The profile optimization method solves a sequence of subproblems based on design heuristics. Three design heuristics are incorporated in the profile optimization method: (i) use a given linear prediction of drag reduction rate to determine the trust region size, (ii) force a simultaneous drag reduction at all design points, and (iii) use the least shape distortion to achieve the targeted drag reduction. The key idea behind the profile optimization method is to reduce the drag at every design point proportionally, which could implicitly reduce the mean and variance of the drag over the Mach range. Padula and Li (ref. [18]) provide a unified framework to combine these two methods, which leads to a hybrid method including randomly sampled design points and the strategies used in the profile optimization method to give the user more options of finding a robust airfoil design with respect to the uncertainty in the Mach number.

One unconventional feature of the hybrid method by Padula and Li (ref. [18]) is that different objective functions are minimized for each subproblem. This method tries to find a locally Pareto optimal solution of the following multiobjective optimization problem:

$$\min_{D,\alpha(u)} \quad c_d(D, \alpha(u), u) \text{ over } u \in \Omega \tag{8}$$

subject to the constraints given in equation (4).

Recall that $(D^*, \alpha^*(u))$ is locally Pareto optimal for (8) if $(D^*, \alpha^*(u))$ satisfies (4) and there is no $(D, \alpha(u))$ in a neighborhood of $(D^*, \alpha^*(u)$ such that $(D, \alpha(u))$ satisfies (4) and $c_d(D, \alpha(u), u) \leq c_d(D^*, \alpha^*(u), u)$ for all $u \in \Omega$ with $c_d(D, \alpha(u), u) < c_d(D^*, \alpha^*(u), u)$ for at least one u in Ω. If D^* is a strict local optimal solution to (6), let $\alpha^*(u)$ be the angle of attack such that $c_l(D^*, \alpha^*(u), u) = c_l^*(u)$ for $u \in \Omega$, then $(D^*, \alpha^*(u))$ is also locally Pareto optimal for (8). Note that by changing the objective function during optimization iterations, we are not solving a particular single objective optimization problem. Instead, such an approach is designed to find a local Pareto optimal solution of (8). See the next section for more details.

4 The Randomized Multipoint Optimization Method With Auxiliary Drag Constraints

In this section, we introduce the constrained multipoint optimization method for finding a Pareto optimal solution of equation (8). To facilitate our presentation, we divide this section into two subsections: the first one contains the description of our main algorithm and the second one provides a detailed explanation of the technical background for each step in the algorithm.

4.1 Algorithm Description

Our main algorithm is based on a randomized multipoint optimization formulation with auxiliary drag constraints. This algorithm uses a random sampling of design points and smart descent directions to find an approximate solution of equation (3).

The Randomized Multipoint Optimization Method (RMOM) With Auxiliary Drag Constraints. Let D^0 be a given initial design vector, $\epsilon > 0$ a tolerance control for termination, $\tau_k \geq 0$ random sampling control parameters, $\gamma_{\min}$ the minimum reduction rate for each design point, $\gamma_{\text{ave}} (\geq \gamma_{\min})$ the reduction rate for the weighted average of the drag at all design points, and $k = 0$. Then construct a sequence of design vectors as follows:

1. Generate random samples of design points. Find r design vectors $u_{1,k}, \ldots, u_{r,k}$ (when $k > 2$) in the uncertainty region such that

$$\left\| u_{i,k} - u_{i,k-1} \right\| \leq \tau_k \quad \text{for } 1 \leq i \leq r.$$

2. Compute feasible angles of attack. Find $\alpha_{1,k}, \alpha_{2,k}, \ldots, \alpha_{r,k}$ such that

$$c_l(D^k, \alpha_{i,k}, u_{i,k}) = c_{l,i}^* \quad \text{for } 1 \leq i \leq r.$$

3. Perform adaptive adjustment of the integration scheme. Let $\sum_{i=1}^{r} w_{i,k} \cdot c_d(D, \alpha_i, u_{i,k})$ be a numerical approximation of the expected drag $E(c_d)$ given in equation (5), where $w_{1,k}, w_{2,k}, \ldots, w_{r,k}$ are positive weights that include the effect of the probability density function $p(u)$.

4. Check the termination criterion. Consider the following two conditions:

$$\left\| \sum_{i=1}^{r} w_{i,k} \cdot \frac{\partial \hat{c}_d}{\partial D}(D^k, \alpha_{i,k}, u_{i,k}) \right\| \leq \epsilon \tag{9}$$

and

$$\min_{\lambda_i \geq 0, \sum_{i=1}^{r} \lambda_i = 1} \left\| \sum_{i=1}^{r} \lambda_i \cdot \frac{\partial \hat{c}_d}{\partial D}(D^k, \alpha_{i,k}, u_{i,k}) \right\| \leq \frac{\gamma_{\min}}{\gamma_{\text{ave}}} \cdot \epsilon, \tag{10}$$

where

$$\frac{\partial \hat{c}_d}{\partial D}(D^k, \alpha_{i,k}, u_{i,k}) = \left[\frac{\partial c_d}{\partial D} - \frac{\left(\frac{\partial c_d}{\partial \alpha_i} \right)}{\left(\frac{\partial c_l}{\partial \alpha_i} \right)} \cdot \frac{\partial c_l}{\partial D} \right] (D^k, \alpha_{i,k}, u_{i,k}).$$

If either equation (9) or equation (10) is satisfied, then output D^k as an optimal design vector and terminate the algorithm.

5. Solve a trust region subproblem. Let $c_{d,i,k}$ and $c_{l,i,k}$ be the linear approximations of the drag and lift at $(D^k, \alpha_{i,k})$:

$$c_{l,i,k}(\Delta D, \Delta \alpha_i) = c_l(D^k, \alpha_{i,k}, u_{i,k}) + \left\langle \frac{\partial c_l}{\partial D}, \Delta D \right\rangle + \frac{\partial c_l}{\partial \alpha} \Delta \alpha_i,$$

$$c_{d,i,k}(\Delta D, \Delta \alpha_i) = c_d(D^k, \alpha_{i,k}, u_{i,k}) + \left\langle \frac{\partial c_d}{\partial D}, \Delta D \right\rangle + \frac{\partial c_d}{\partial \alpha} \Delta \alpha_i,$$

where the derivatives are evaluated at $(D^k, \alpha_{i,k}, u_{i,k})$. Consider the following trust region subproblem:

$$\min_{\Delta D, \Delta\alpha_i} \sum_{i=1}^{r} w_{i,k} \cdot c_{d,i,k}(\Delta D, \Delta\alpha_i) \qquad \text{subject to} \tag{11}$$

$$\begin{aligned}
&-\gamma_{i,k}\delta_k \le \Delta D_i \le \gamma_{i,k}\delta_k \quad \text{for } 1 \le i \le m,\\
&-\alpha_{i,k} \le \Delta\alpha_i \le \alpha_{\max} - \alpha_{i,k} \quad \text{for } 1 \le i \le r,\\
&c_{l,i,k}(\Delta D, \Delta\alpha_i) = c_l^* \quad \text{for } 1 \le i \le r,\\
&c_{d,i,k}(\Delta D, \Delta\alpha_i) \le (1-\gamma_{\min}) \cdot c_d(D^k, \alpha_{i,k}, u_{i,k}) \quad \text{for } 1 \le i \le r,
\end{aligned}$$

where $\gamma_{i,k} \ge 0$ are some scaling factors. Choose $\delta_k > 0$ such that equation (11) is feasible and the least norm solution $(\Delta D^k, \Delta\alpha_{1,k}, \ldots, \Delta\alpha_{r,k})$ of equation (11) satisfies the following condition:

$$\sum_{i=1}^{r} w_{i,k} \cdot c_{d,i,k}(\Delta D^k, \Delta\alpha_{i,k}) = (1-\gamma_{\text{ave}}) \sum_{i=1}^{r} w_{i,k} \cdot c_d(D^k, \alpha_{i,k}, u_{i,k}).$$

6. Generate the new iterate. Let $\alpha_{i,k+1} = \alpha_{i,k} + \Delta\alpha_{i,k}$ for $1 \le i \le r$ and $D^{k+1} = D^k + \Delta D^k$.

7. Start a new iteration. Update k by $k+1$ and go back to step 1.

4.2 Technical Discussion of Algorithm

The previously described algorithm has a few unconventional strategies adopted from references [16] and [45] for finding robust optimal solutions. The algorithm is based on robust design heuristics for aerodynamic optimization. To fully understand how the algorithm finds a robust optimal solution, we need to explain what happens at each step in the algorithm.

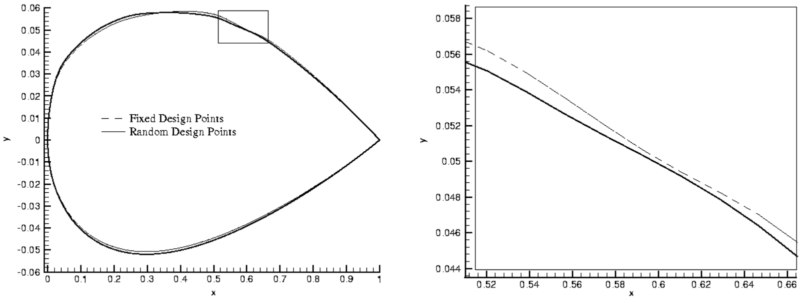

Figure 4. Two optimal airfoils generated by the RMOM with fixed and random design points.

Step 1 implements random sampling of the design points from iteration to iteration. This sampling allows an optimization code to modify the airfoil shape at various locations on the airfoil that influence performance to various design conditions and, thus, eliminate the point-optimization behavior. To illustrate, we plot the optimal airfoils generated by the RMOM with fixed design points and random design points, respectively, in figure 4. Note that the random sampling of the design points makes the optimal solution smoother. Figure 4(b) shows a close-up view of the boxed area in figure 4(a). This example shows some advantage of having the option of using random sampling of the design points in step 1 to make the optimal solution less dependent on the choice of design points. The random sampling control parameter τ_k can change from iteration to iteration or be set to zero if fixed design points are preferred.

In step 2 we find the feasible angles of attack. When the target lift is less than the maximum achievable lift, the lift coefficient is always a monotone increasing function of the angle of attack in a neighborhood of $\alpha_{i,k}$. Therefore, the optimal choice of the angle of attack is always the smallest one that satisfies the lift constraint. Thus, the inequality constraint for the lift in equation (3) can be replaced by an equality constraint to generate the same optimal solution. Kim et al. (ref. [34]) modified a standard flow solver to adjust the angle of attack during the iteration process so that the flow solver could calculate the angle of attack for a *given* target lift with a computational time only marginally more than the calculation of the lift and drag for a *given* angle of attack.

In step 3, one can choose arbitrary weights to define the objective function. In terms of multiobjective optimization for the drag at the given design points, the weights determine a particular weak optimal solution on the weak efficient front of the Pareto optimization problem. However, the choice of weights based on the probability density function $p(u)$ and a numerical integration formula for the expected drag makes the approach related to the minimization of the expected value of the drag.

In step 4, the termination criterion given is of theoretical nature since it is impossible in practice to determine what should be an appropriate value of ϵ. However, this criterion can serve as an assessment tool to measure the quality of the iterates generated by the algorithm.

The two termination conditions can be derived by using the implicit differentiation rule. Here we are assuming that, for any flight condition and a feasible target lift, one can adjust the angle of attack to achieve the target lift. For each design vector D, we can find the smallest $\alpha_i(D)$ such that

$$c_l(D, \alpha_i(D), u_i) = c^*_{l,i}. \tag{12}$$

Assuming that $\mathcal{F}$ is the whole space, we can rewrite equation (6) as

$$\min_D I(D) \tag{13}$$

where

$$I(D) = \sum_{i=1}^{r} w_i \cdot c_d(D, \alpha_i(D), u_i).$$

Using the implicit differentiation, we obtain

$$\frac{\partial I}{\partial D} = \sum_{i=1}^{r} w_i \cdot \left(\frac{\partial c_d}{\partial D} + \frac{\partial c_d}{\partial \alpha_i} \cdot \frac{\partial \alpha_i}{\partial D} \right) = \sum_{i=1}^{r} w_i \cdot \left(\frac{\partial c_d}{\partial D} - \frac{\left(\frac{\partial c_d}{\partial \alpha_i}\right)}{\left(\frac{\partial c_l}{\partial \alpha_i}\right)} \cdot \frac{\partial c_l}{\partial D} \right). \tag{14}$$

Note that equation (14) was discussed by Reuther et al. (ref. [6]) using variational analysis approach and later incorporated by Kim et al. (ref. [34]) into optimization algorithms for solving multipoint optimization problems. Thus, the first-order optimality condition for the multipoint optimization problem is

$$\left\| \sum_{i=1}^{r} w_i \cdot \left(\frac{\partial c_d}{\partial D} - \frac{\left(\frac{\partial c_d}{\partial \alpha_i}\right)}{\left(\frac{\partial c_l}{\partial \alpha_i}\right)} \cdot \frac{\partial c_l}{\partial D} \right) \right\| = 0. \tag{15}$$

This condition explains why we use equation (9) as a stopping criterion.

The first-order optimality condition for a minimax version of the multipoint optimization problem is

$$\min_{\lambda_i \geq 0, \sum_{i=1}^{r} \lambda_i = 1} \left\| \sum_{i=1}^{r} \lambda_i \cdot \left(\frac{\partial c_d}{\partial D} - \frac{\left(\frac{\partial c_d}{\partial \alpha_i}\right)}{\left(\frac{\partial c_l}{\partial \alpha_i}\right)} \cdot \frac{\partial c_l}{\partial D} \right) \right\| = 0. \tag{16}$$

Note that if equation (16) is satisfied, then it is impossible to find a descent direction that will reduce the drag at all design points *simultaneously.*

If there is no requirement on a minimum reduction of the drag at each design point (i.e., $\gamma_{\text{min}} \leq 0$), then equation (10) never holds and the RMOM is essentially searching for a multipoint optimal solution with respect to dynamically allocated design points. If $\gamma_{\text{min}} = \gamma_{\text{ave}}$, then equation (9) implies equation (10) and the RMOM will be terminated by equation (10). In this case, the RMOM generates an optimal solution in a minimax sense. In general, the ratio $\frac{\gamma_{\text{min}}}{\gamma_{\text{ave}}}$ provides an option for choosing a combination of the weighted average and the minimax strategies.

In step 5, the linear subproblem formulated is similar to the linear subproblem for a trust region method based on sequential linear programming, except for the auxiliary drag constraints. The auxiliary constraints ensure a simultaneous predicted drag reduction at all design points, when $\gamma_{\text{min}} > 0$. If γ_{min} is a large negative number (say, -1000) and the design points are fixed from iteration to iteration, then the RMOM solves the multipoint optimization problem because the drag constraints in equation (11) become superfluous and are always satisfied. Similarly, if γ_{min} is negative but the design points are randomly sampled from iteration to iteration, then the algorithm becomes a variation of the expected value optimization method (refs. [17] and [45]). If we choose $\gamma_{\text{min}} = \gamma_{\text{ave}}$ and fix the design points from iteration to iteration, then the RMOM is reduced to the profile optimization method proposed by Li, Huyse, and Padula (ref. [16]). Therefore, by using different values of γ_{min} and sampling schemes in the RMOM, we can get various hybrids of the multipoint optimization method, the expected value optimization method, and the profile optimization method.

In step 6, the new iterate generated is chosen to increase reliability of the linear prediction and to avoid random shape distortions. For example, let S be the orthogonal complement of the linear space generated by the gradients of the lift and the drag with respect to the design variables; that is, if

$$S = \left\{ v : \quad \left\langle v, \frac{\partial c_d}{\partial D}(D^k, \alpha_{i,k}, u_{i,k}) \right\rangle = 0 \text{ and} \right.$$
$$\left. \left\langle v, \frac{\partial c_l}{\partial D}(D^k, \alpha_{i,k}, u_{i,k}) \right\rangle = 0 \text{ for } 1 \leq i \leq r \right\},$$

then vectors in S do not change the linear prediction of the lift and drag at the design points. Thus, if $\Delta \overline{D}^k$ is in S, then for any real number t, $(\Delta D^k + t\Delta \overline{D}^k)$ is also a solution of the trust region subproblem as long as $(\Delta D^k + t\overline{\Delta D}^k)$ is in the trust region. In the case when the number of design variables is greater than $2r$, S is the set of solutions of $2r$ linear equations with more than $2r$ variables. Thus, the dimension of S is at least one and the linear subproblem might have multiple solutions. The simplex method finds a vertex of the solution region, and the interior point method finds the analytical center of the solution region. Therefore, different solvers of the linear subproblem might generate different values of ΔD^k, which lead to different iterates. These differences explain the impact a solver of the linear subproblem could have on the intermediate iterates. If a quasi-Newton method is used to find ΔD^k, then the solution of the trust region subproblem is unique. In this case, the S-components in the search direction ΔD^k are not introduced randomly. Instead, they reflect the (random) numerical error of the quadratic approximation of the objective function. Note that any shape change in S is unnecessary and could be considered as a random perturbation in the descent direction, which might lead to random shape distortions during the optimization iterations. From previous studies (e.g., refs. [7]– [13]), we know that smoothing of the airfoil shape is a quite common procedure added in an airfoil optimization process to filter out high frequency components in the airfoil shape to prevent noises in airfoils. To avoid randomly adding elements in S to the descent direction ΔD, we force ΔD to be the least norm solution of the trust region subproblem, which has two advantages: (i) it generates a solution ΔD^k that is orthogonal to vectors in S, which prevents random additions of those vectors in ΔD^k, and (ii) the descent direction ΔD^k has the smallest norm among all solutions of the trust region subproblem, which helps to increase the reliability of the linear prediction of the lift and drag.

The new iterate generated in step 6 is influenced by the values of drag reduction rate parameters $\gamma_{\min}$ and γ_{ave}. When $\gamma_{\min} > 0$, the purpose of the auxiliary drag constraints is to find a common descent direction for the drag at the design points. To demonstrate the implication of such a strategy, we eliminate α_i in equation (11) and rewrite the auxiliary drag constraints as follows:

$$\left\langle \Delta D^k, \frac{\partial \hat{c}_d}{\partial D}(D^k, \alpha_{i,k}, u_{i,k}) \right\rangle \leq -\gamma_{\min} c_d(D^k, \alpha_{i,k}, u_{i,k}) \quad \text{for } 1 \leq i \leq r.$$

Multiplying both sides of the above inequality by $\lambda_i \geq 0$ and adding all the resulting inequalities together, we obtain

$$\left\langle \Delta D^k, \sum_{i=1}^{r} \lambda_i \frac{\partial \hat{c}_d}{\partial D}(D^k, \alpha_{i,k}, u_{i,k}) \right\rangle \leq -\left(\sum_{i=1}^{r} \lambda_i \gamma_{\min} c_d(D^k, \alpha_{i,k}, u_{i,k}) \right) < 0$$

for $\lambda_i \geq 0$ and $\sum_{i=1}^{r} \lambda_i > 0$. If the gradient $\frac{\partial \hat{c}_d}{\partial D}$ of the drag at an off-design point u is a nonnegative combination of the gradients of the drag at the design points, then ΔD^k is also a descent direction for the drag at u. For example, one can prove that ΔD^k is a descent direction for the drag at u whenever $\frac{\partial \hat{c}_d}{\partial D}$ is in a neighborhood V of the convex cone generated by the gradients of the drag at the design points. In general, a larger $\gamma_{\min}$ implies a larger neighborhood V; that is, increasing $\gamma_{\min}$ reduces the risk of trading a drag reduction at the design points with off-design drag increment.

In the special case when $\gamma_{\min} = \gamma_{\text{ave}}$, our algorithm is actually solving a discrete version of the classical robust optimization problem given by equation (3). In fact, let $\hat{E}(c_d)$ (or $\bar{E}(c_d)$) and $\hat{\sigma}^2(c_d)$ (or $\bar{\sigma}^2(c_d)$) denote the discrete version of the expected value and variance of c_d at $D^k + \Delta D^k$ (or D^k) based on the sample points $u_{i,k}$; that is,

$$\hat{E}(c_d) = \sum_{i=1}^{r} w_{i,k} \cdot c_{d,i,k}(\Delta D^k, \Delta\alpha_{i,k}),$$

$$\hat{\sigma}^2(c_d) = \sum_{i=1}^{r} w_{i,k} \left(c_{d,i,k}(\Delta D^k, \Delta\alpha_{i,k}) - \hat{E}(c_d) \right)^2,$$

$$\bar{E}(c_d) = \sum_{i=1}^{r} w_{i,k} \cdot c_d(D^k, \alpha_{i,k}, u_{i,k}),$$

$$\bar{\sigma}^2(c_d) = \sum_{i=1}^{r} w_{i,k} \left(c_d(D^k, \alpha_{i,k}, u_{i,k}) - \bar{E}(c_d) \right)^2.$$

In most circumstances, if the design space has enough resolution, then one could find a descent direction such that the equalities hold for the auxiliary drag constraints in equation

(11). In this case, we have

$$\hat{E}(c_d) = (1 - \gamma_{\min})\bar{E}(c_d) \quad \text{and} \quad \hat{\sigma}^2(c_d) = (1 - \gamma_{\min})^2\bar{\sigma}^2(c_d). \tag{17}$$

That is, the predicted mean and variance of the drag at the design points are reduced by the given rate $\gamma_{\min}$. If the linear prediction is accurate, then the algorithm actually minimizes both estimates of the mean and variance of the drag computed at the design points. Therefore, the algorithm generates a solution of a discrete version of the classical robust optimization problem given by equation (3).

Finally, note that the change of each design variable is proportional to the scaling factors $\gamma_{i,k}$. One common choice for $\gamma_{i,k}$ is the absolute value $|D_i^k|$ of the ith component of D^k. However, other choices of $\gamma_{i,k}$ might be necessary for particular applications. Scaling factors $\gamma_{i,k}$ have a significant impact on how the optimization code will work. In general, a given percentage of predicted drag reduction can be achieved in many ways. By using various values of $\gamma_{i,k}$, one can force the optimization code to focus on modifying certain parts of the aerodynamic shape to achieve the predicted drag reduction.

Theoretically, we could use the standard trust region strategy to check the ratio of the actual drag reduction and the predicted drag reduction and update $\gamma_{\min}$ and γ_{ave} accordingly. We conjecture that the behavior of such an algorithm is expected to be similar to traditional trust region algorithms: an actual sufficient reduction of the objective function from iteration to iteration and any cluster point of the iterates being a stationary point of either a multipoint optimization problem or a minimax optimization problem.

Note that any local strict Pareto optimal solution of the following multiobjective optimization problem,

$$\min_D \; c_d(D, \alpha_i, u_i) \text{ over } 1 \leq i \leq m,$$

is also a Pareto optimal solution of equation (8). Thus, a Pareto optimal solution of equation (8) can be found by using a few design points. In practice, we usually choose γ_{ave} to achieve a meaningful drag reduction (such as 1–4 percent). The choice of $\gamma_{\min}$ is based on robustness requirement as mentioned previously. To avoid technical complications, we do not include the adjustment of γ_{ave} and $\gamma_{\min}$ in the algorithm, but readers could experiment with different values of γ_{ave} and $\gamma_{\min}$ for different iterations.

In summary, the RMOM is aimed at finding a Pareto optimal solution, while minimizing an approximation of the expected drag and maintaining a level of simultaneous drag reduction at all design points from iteration to iteration. The descent direction is selected in such a way that a given predicted drag reduction rate is achieved with a minimum amount of shape modification. This strategy helps to eliminate high frequency components in the descent direction, and reduces the need for airfoil smoothing during the optimization iterations (even if the airfoil is represented by B-splines with many free design variables).

5 Numerical Simulation Results

All our numerical results indicate that the RMOM generates relatively smooth airfoils with no severe off-design performance degradation, even when the design space consists of B-spline curves with over 50 control points. In this section, we only include a few sample results to illustrate how the sampling of design points, resolution of the design space, and the choice of minimum drag reduction rate (i.e., the value of $\gamma_{\min}$) affect the optimal solution generated by the RMOM.

5.1 Technical Information for Simulation Setting

For a given airfoil, the lift and drag coefficients c_l and c_d are obtained by solving Euler equations using FUN2D (refs. [21], [41], and [23]). For numerical results presented in this

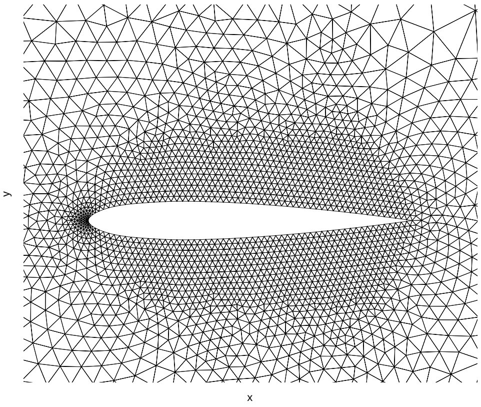

Figure 5. Grid used to solve Euler equations by FUN2D.

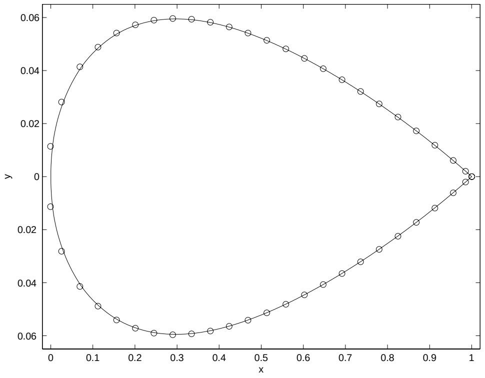

Figure 6. Parameterization of NACA-0012 by cubic B-splines with 50 control points.

paper, an unstructured grid with 2872 points and 5744 elements is used (see fig. 5 for the grid near the airfoil). The derivatives of the lift and drag are obtained by solving the adjoint equations of the discrete flow equations (i.e., by using the discrete adjoint method). With the given grid, the flow solver and the adjoint solver are terminated when the 2-norms of the residual of the density equations and its adjoint counterpart are reduced by at least 5 orders of magnitude. The initial airfoil is NACA-0012. The airfoil is parameterized by using B-splines with either 23 or 50 control points. See figure 6 for a parameterization of the NACA-0012 airfoil with 50 control points. The x-coordinates of all the control points are fixed during optimization. However, the maximum thickness of the airfoil is not constrained. The y-coordinates of the first and last control points are fixed such that the trailing edge of the airfoil is fixed during optimization. The y-coordinate of the leading edge of the airfoil is also fixed such that the chord length of the airfoil remains constant during optimization. The number of free design variables is the total number of control points minus 3. The weights $\gamma_{i,k}$ in the RMOM are chosen to be $|D_i^k|$ (i.e., proportional to the size of design variables). For simplicity, we consider only the lift-constrained drag minimization with uncertain flight speed (i.e., $u = M$) and $c_{l,i} = c_l^* = 0.4$ for all i. For each iterate D^k, we find angles of attack $\alpha_{1,k}, \ldots, \alpha_{m,k}$ such that $|c_l(D^k, \alpha_{i,k}, u_{i,k}) - c_{l,i}^*|/c_{l,i}^* \leq 0.001$. (Note that Kim et al. (ref. [34]) replaced the equality lift constraint $c_l = c_l^*$ by $0 \leq (c_l - c_l^*)/c_l^* \leq 0.003$.) The probability density function $p(u)$ is the uniform distribution over the Mach range $[0.7, 0.8]$.

The results presented in this section are selected from the following three sets of test cases:

1. Four equally spaced design points in $[0.7, 0.8]$, 47 free design variables, $\gamma_{\text{ave}} = 3$ percent, and $\gamma_{\text{min}} = -100000$, 1, 2, or 3 percent

2. Four randomly distributed design points in $[0.7, 0.8]$, 47 free design variables, $\gamma_{\text{ave}} = 3$ percent, and $\gamma_{\text{min}} = -100000$, 1, 2, or 3 percent

3. Four equally spaced design points in $[0.7, 0.8]$, 20 free design variables, $\gamma_{\text{ave}} = 3$ percent, and $\gamma_{\text{min}} = -100000$ or 3 percent

All the optimization runs are terminated after either 50 or 100 iterations, which allows us to compare different test cases easily. In this paper, we did not use the derived stopping criteria to terminate the optimization runs.

5.2 Optimal Airfoil Shapes

The optimal airfoils generated by the RMOM are quite smooth and do not have four bumps corresponding to four design conditions (i.e., Mach numbers). This is different from the numerical simulation results obtained earlier by Drela on multipoint airfoil optimization when sinusoidal basis functions were used to parameterized the airfoil shape. Figure 7 is a typical optimal airfoil generated by the RMOM after 50 iterations. All the optimal airfoils have shape patterns that are surprisingly similar to the airfoils generated by a two-point optimization method ($M_1 = 0.8, c^*_{l,1} = 0.3438$ and $M_2 = 0.82, c_{l,2} = 0.3348$), starting from NACA-0012 (see figs. 2 and 6 in ref. [26]). Interestingly, the RMOM actually finds such a shape pattern to be optimal among all possible airfoil shapes represented by B-splines with 47 free design variables. In comparison, the airfoil generated by Elliott and Peraire (ref. [26]) was given in the form of $z(x) = z_0(x) + \sum_{i=1}^{4} \beta_i h_i(x)$, where $h_i(x)$ are specific Hicks-Henne bump functions that allow the airfoil to deform to a highly aft-loaded supercritical airfoil. Thus, we contend that the RMOM has a potential of finding relatively realistic airfoil designs without intervention from aerodynamic experts.

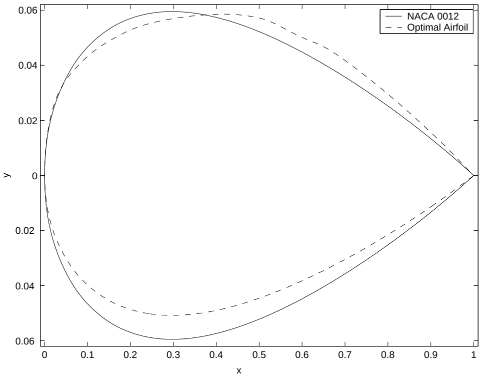

Figure 7. Typical airfoil generated by the RMOM after 50 iterations starting from baseline airfoil NACA-0012.

5.3 Impact of Sampling of Design Points

Our numerical results show that random sampling of design points can be an effective means to prevent the optimizer from excessively modifying any particular location on the airfoil surface to reduce the shock corresponding to a particular flight condition. This sampling avoids specific shock bumps on the airfoil surface corresponding to design conditions, which could occur during the standard multipoint airfoil shape optimization (ref. [14]). As a consequence, one might get smoother airfoil shapes by using randomized design points (see fig. 4). In terms of drag reduction, whether we use random design points or fixed design points, the drag profiles after 50 iterations are quite similar (see fig. 8). Figure 9 shows the complete history of sampled design points for 50 iterations. One can see that the variations of design points from iteration to iteration are quite substantial.

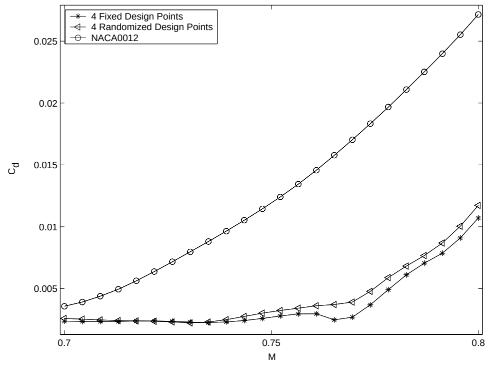

Figure 8. Drag curves versus Mach numbers for 50th iterates generated by the RMOM with $\gamma_{\min} = -100000$ percent, $\gamma_{\text{ave}} = 3$ percent, and 47 free design variables.

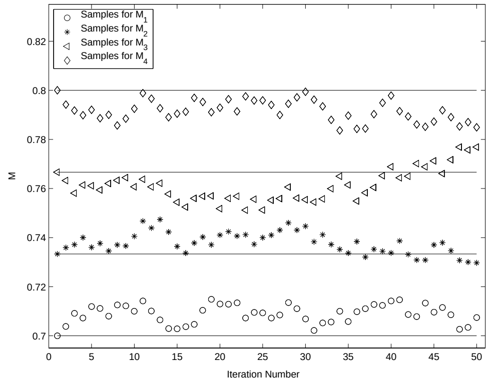

Figure 9. Complete history of sampled design points for 50 iterations of the RMOM with $\gamma_{\min} = -100000$ percent, $\gamma_{\text{ave}} = 3$ percent, and 47 free design variables.

From a theoretical point of view, unless the size of random perturbation of design points (i.e., γ_k) is sufficiently small, we can not ensure that the new iterate is indeed an "improvement" of the old iterate because changing design points means changing the objective function. Practically, changing design points could increase computational expenses. In general, for fixed design points, we do not have to adjust the angles of attack to satisfy the lift constraints because the linear predictions of the lift are accurate enough for the new iterate to satisfy the lift constraints. In contrast, if randomized design points are used, then the angles of attack must be adjusted to satisfy the lift constraints in each iteration. For the three sets of test cases reported here, each iteration requires 4 function calls and 8 derivative calls to FUN2D for the RMOM with fixed design points, while the RMOM with randomized design points requires approximately 8 additional function calls to find the feasible angles of attack at the 4 new design points. However, if the flow solver has an option of producing the drag and the required angle of attack for a given lift input, then the computational cost of using randomized design points is the same as using fixed design points.

5.4 Impact of Auxiliary Drag Constraints

Auxiliary constraints only become meaningful if $\gamma_{\min} > 0$, which allows one to enforce a certain percent of predicted drag reduction at each design point. By varying $\gamma_{\min}$ from $-\infty$ to γ_{ave}, we hope to achieve a dynamic balance between the most conservative optimization strategy (with $\gamma_{\min} = \gamma_{\text{ave}}$) and the most aggressive optimization strategy (with $\gamma_{\min} = -\infty$). Figure 10 includes the drag curves for the solutions generated by the RMOM after 50 iterations using various values of $\gamma_{\min}$. All these iterates were generated with $\gamma_{\text{ave}} = 3$ percent and 47 free design variables. In general, the drag curve for the solution has a smaller slope if $\gamma_{\min}$ is larger. Using the mean and standard deviation of the drag at 24 equally spaced sample Mach numbers in the Mach range [0.7,0.8] as approximations of the actual mean and standard deviation of the drag on $[0.7, 0.8]$, we attained the results presented in table 1. Table 1 clearly shows that $\gamma_{\min}$ provides an option for trade-off between the mean and standard deviation of the drag for the optimal solution over the given Mach range.

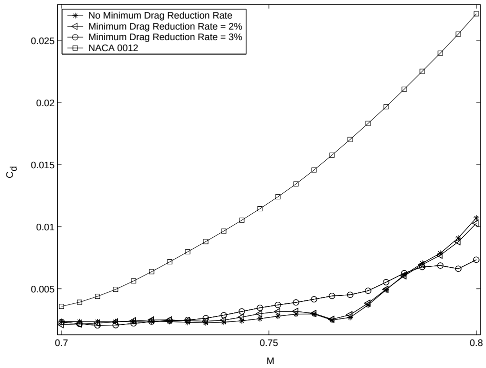

Figure 10. Drag curves versus Mach numbers for solutions generated by the RMOM after 50 iterations using various values of $\gamma_{\min}$.

Table 1. Estimated Mean and Standard Deviation of Drag in Mach Range $[0.7, 0.8]$

	Estimated Mean	Estimated Standard Deviation
$\gamma_{\min} = -\infty$	0.0038	0.0025
$\gamma_{\min} = 2$ percent	0.0039	0.0023
$\gamma_{\min} = 3$ percent	0.0040	0.0018

5.5 Impact of Resolution of Design Space

Figure 11 shows the the drag curves versus Mach numbers for the 100th iterate generated by the RMOM with $\gamma_{\min} = 3$ percent and $\gamma_{\text{ave}} = 3$ percent for 20 and 47 free design variables, respectively. For Mach numbers near 0.8, the drag for the 100th iterate using 20 free design variables is twice as large as the drag for the 100th iterate using 47 free design variables. By checking the optimization history, we find that the drag coefficients start to oscillate after 50 iterations when using 20 free design variables. Figure 12 shows the drag coefficients for 4 fixed design points from the 51st to the 100th iteration. In other words, if the initial baseline is the 50th iterate, then the RMOM could not improve the initial design with the most conservative optimization strategy (using $\gamma_{\min} = \gamma_{\text{ave}}$) because the design space does not have enough resolution to provide a highly restrictive search direction for drag reduction at each of the design points. However, if a different optimization strategy is employed (with $\gamma_{\min} = -100000$ percent $<< \gamma_{\text{ave}} = 3$ percent), then it is much easier to find a less restrictive search direction for an average drag reduction at the given design points. As a consequence, the drag curve of the 100th iterate generated by using 20 free design variables is not much different from that of the 100th iterate generated by using 47 free design variables (see fig. 13).

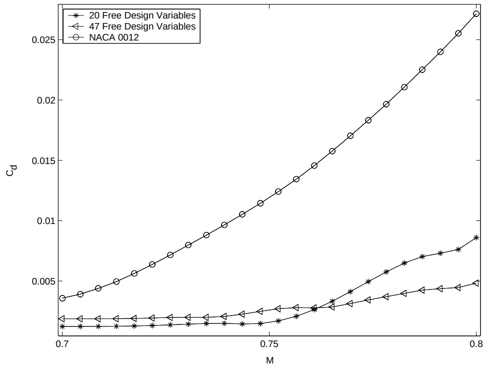

Figure 11. Drag curves versus Mach numbers for 100th iterates generated by the RMOM with $\gamma_{\min} = 3$ percent and $\gamma_{\text{ave}} = 3$ percent.

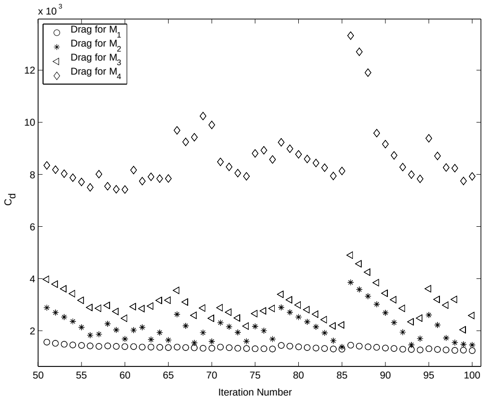

Figure 12. Complete history of drag values at 4 fixed design points for iterations 51–100.

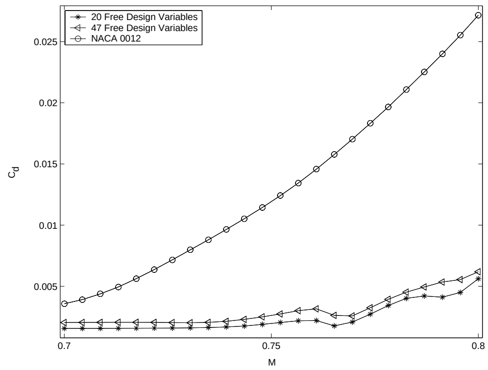

Figure 13. Drag curves versus Mach numbers for 100th iterates generated by the RMOM with $\gamma_{\text{min}} = -100000$ percent and $\gamma_{\text{ave}} = 3$ percent.

6 Concluding Remarks

This research is motivated by the needs of conceptual and preliminary design teams. These teams desire smooth airfoil shapes that are similar to the baseline design but have improved drag performance over a range of flight conditions. The resulting airfoil optimization method modifies a large number of design variables to search for the true optimal solution, yet it avoids any severe off-design performance degradation that is characteristic of single-point methods.

New versions of commercial aircraft are variants of baseline designs with modified operating conditions such as required lift or cruise speed. Given a good initial design, this new optimization method improves the performance without making drastic shape changes. Small reductions in the expected value of drag translate into large reductions in fuel consumption. Moreover, the design team gains valuable information by examining the details of the flow field around the airfoil before and after optimization.

As a demonstration of the optimization algorithm, we studied a lift-constrained drag minimization problem for a two-dimensional airfoil in Euler flow with 47 B-spline coefficients as free design variables. An unstructured grid computational fluid dynamics code, FUN2D, predicts the lift and drag values and their gradients with respect to changes in airfoil shape and changes in angle-of-attack. We were able to generate relatively smooth airfoils with no severe off-design performance degradation over the range of Mach numbers, even when the number of design points was as small as four.

This work further demonstrates that it is possible to avoid off-design performance degradation by using robust airfoil shape optimization methods, even if a large number of free-form design variables are used. Post-optimization analysis shows that the optimized airfoil also has better performance than the original airfoil at 24 off-design conditions. Results shown in the paper are typical of all numerical experiments performed by the authors. It is important to note that the solution of each optimization subproblem is a relatively smooth airfoil and that each airfoil has improved performance over the previous one. Termination criteria are provided and these criteria help the user judge when to stop because further improvement is unlikely.

To understand the importance of using smooth shape modification for aerodynamic shape optimization, we investigate the linear Taylor approximation error of an integral of the pressure around the airfoil. Such an error study shows that assuming the exact gradient information is given, the accuracy of the linear Taylor approximation of the lift or drag coefficient depends not only on the magnitude of shape change, but also on the magnitude of the derivative of shape change. As a consequence, with the same order of magnitude of shape modification, noisier shape modifications (i.e., larger derivatives of shape change) lead to larger disparities between the predicted reduction and the actual reduction of the drag. Thus, noisy shape modifications for a descent method may increase the drag for lift-constrained drag minimization problems.

Even though we do not use airfoil smoothing during the optimization, the least norm solution of the trust region subproblem is implemented to reduce random airfoil shape distortions during the optimization process, which forces a minimal shape change to achieve a predetermined rate of reduction for the given objective function.

Note that randomized multipoint optimization method (RMOM) does not solve an optimization problem with a single objective function. Using different objective functions for formulating optimization subproblems allows us to enforce a robust optimization policy without any information on the mean and variance of the drag. RMOM attempts to find a local Pareto optimal solution of the multiobjective optimization problem (8) that behaves like an optimal solution to the robust optimization problem (3).

The algorithm presented in our paper allows us to choose a combination of the weighted average and minimax strategies, which corresponds to a trade-off of performance and robustness during the optimization process, to find better aerodynamic shapes in a high resolution design space.

References

1. Jameson, A.: *Essential Elements of Computational Algorithms for Aerodynamic Analysis and Design.* NASA/CR-97-206268. ICASE Report No. 97-68, 1997.

2. Jameson, A.: Re-Engineering the Design Process Through Computation. *J. Aircr.*, vol. 36, 1999, pp. 36–50.

3. Jameson, A.: A Perspective on Computational Algorithms for Aerodynamic Analysis and Design. *Progress in Aerosp. Sci.*, vol. 37, no. 2, 2001, pp. 197–243.

4. Jameson, A.; and Vassberg, J.: Computational Fluid Dynamics for Aerodynamic Design: Its Current and Future Impact. AIAA-2001-0538, Jan. 2001.

5. Jameson, A.; Alonso, J.; Reuther, J.; Martinelli, L.; and Vassberg, J.: Aerodynamic Shape Optimization Techniques Based on Control Theory. *29th AIAA Fluid Dynamics Conference*, AIAA, June 15-18, 1998. (Also available as AIAA-98-2538.)

6. Reuther, J.; Jameson, A.; Alonso, J.; Rimlinger, M.; and Saunders, D.: Constrained Multipoint Aerodynamic Shape Optimization Using an Adjoint Formulation and Parallel Computers. *J. Aircr.*, vol. 36, 1999, pp. 51–74.

7. Jameson, A.; Optimum Aerodynamic Design via Boundary Control. *Optimum Design Methods in Aerodynamics*, AGARD-VKI Lecture Series, AGARD Rept. 803, April 1994, pp. 3-1–3-33.

8. Jameson, A.: Optimum Aerodynamic Design Using Control Theory. *Computational Fluid Dynamics Review 1995*, M. Hafez, ed., John Wiley & Sons, 1995, pp. 495–528.

9. Kuruvila, G.; Taásan, S.; and Salas, M.: Airfoil Optimization by the One-Shot Method. *Optimum Design Methods in Aerodynamics*, AGARD-VKI Lecture Series, AGARD Rept. 803, April 1994, pp. 7-1–7-21.

10. Lorentzen, L.: Aerodynamic Shape Optimization Based on Control Theory. Aeronautical Research Institute of Sweden, FFA TN 1997-42, June 1997.

11. Nadarajah, S.; and Jameson, A.: A Comparison of the Continuous and Discrete Adjoint Approach to Automatic Aerodynamic Optimization. AIAA-2000-0667, Jan. 2000.

12. Nadarajah, S.; and Jameson, A.: Studies of the Continuous and Discrete Adjoint Approaches to Viscous Automatic Aerodynamic Shape Optimization. AIAA-2001-2530, June 2001.

13. Szmelter, J.: Multipoint Aerodynamic Wing Optimization in Viscous Flow. *J. Aircr.*, vol. 38, 2001, pp. 860–867.

14. Drela, M.: Pros and Cons of Airfoil Optimization. *Frontiers of Computational Fluid Dynamics 1998*, D.A. Caughey and M.M. Hafez, eds., World Scientific, 1998.

15. Hicks, R.; and Vanderplaats, G.: Application of Numerical Optimization to the Design of Supercritical Airfoils Without Drag-Creep. SAE Paper 770440, Business Aircraft Meeting, Wichita, Kansas, March 29-April 1, 1977.

16. Li, W.; Huyse, L.; and Padula, S.: Robust Airfoil Optimization to Achieve Drag Reduction Over a Range of Mach Numbers, *Struc. Optim.*, vol. 24, no. 1, 2001, pp. 38–50.

17. Huyse, L.; and Lewis, R.: *Aerodynamic Shape Optimization of Two-Dimensional Airfoils Under Uncertain Operating Conditions.* NASA/CR-2001-210648. ICASE Report No. 2001-1, 2001.

18. Padula, S.; and Li, W.: Options for Robust Airfoil Optimization Under Uncertainty. AIAA-2002-5602, Sep. 2002.

19. Jameson, A.: Aerodynamic Optimization via Control Theory. *J. Sci. Comp.*, vol. 3, 1988, pp. 233–260.

20. Jameson, A.; Pierce, N.; and Martinelli, L.: Optimum Aerodynamic Design Using the Navier-Stokes Equations. AIAA-97-0101, Jan. 1997.

21. Nielsen, E.; and Anderson, W.: Aerodynamic Design Optimization on Unstructured Meshes Using the Navier-Stokes Equations. AIAA-98-4809, Sep. 1998. (AIAA J., vol. 37, no. 1, 1999, pp. 1411–1419.)

22. Nielsen, E.; and Anderson, W.: Recent Improvements in Aerodynamic Design Optimization on Unstructured Meshes. AIAA-2001-0596, Jan. 2001.

23. Anderson, W.; and Venkatakrishnan, V.: Aerodynamic Design Optimization on Unstructured Grids With a Continuous Adjoint Formulation. AIAA-97-0643, Jan. 1997.

24. Baysal, O.; and Eleshaky, M.: Aerodynamic Design Optimization Using Sensitivity Analysis and Computational Fluid Dynamics. *AIAA J.*, vol. 30, 1992, pp. 718–725.

25. Baysal, O.; and Ghayour, K.: Continuous Adjoint Sensitivities for Optimization With General Cost Functionals on Unstructured Meshes. *AIAA J.*, vol. 39, 2001, pp. 48–55.

26. Elliott, J.; and Peraire, J.: Constrained, Multipoint Shape Optimization for Complex 3-D Configurations. *Aeronaut. J.*, vol. 102, 1998, pp. 365–376.

27. Gallman, J.; Reuther, J.; Pfeiffer, N.; Forrest, W.; and Bernstorf, D.: Business Jet Wing Design Using Aerodynamic Shape Optimization. *34th Aerospace Sciences Meeting and Exhibit*, AIAA, 1996. (Also available as AIAA-96-0554.)

28. Iollo, A.; and Salas, M.: Contribution to the Optimal Shape Design of Two-Dimensional Internal Flows With Embedded Shocks. ICASE Report No. 95-28, 1995.

29. Jameson, A.: Aerodynamic Shape Optimization of Wing and Wing-Body Configuration Using Control Theory. AIAA-95-0123, Jan. 1995.

30. Jameson, A.: Optimum Aerodynamic Design Using CFD and Control Theory. AIAA-95-1729, June 1995.

31. Jameson, A.; and Reuther, J.: Control Theory Based Airfoil Design Using the Euler Equations. AIAA-94-4272, Jan. 1994.

32. Kim, S.; Alonso, J.; and Jameson, A.: A Gradient Accuracy Study for the Adjoint-Based Navier-Stokes Design Method. AIAA-99-0299, Jan. 1999.

33. Kim, S.; Obayashi, S.; and Nakahashi, K.: Flap-Deflection Optimization for Transonic Cruise Performance Improvement of Supersonic Transport Wing. *J. Aircr.*, vol. 38, 2001, pp. 709–717.

34. Kim, S.; Sasaki, D.; Obayashi, S.; and Nakahashi, K.: Aerodynamic Optimization of Supersonic Transport Wing Using Unstructured Adjoint Method. *AIAA J.*, vol. 39, 2001, pp. 1011–1020.

35. Newman, J.; Taylor, A.; Barnwell, R.; Newman, P.; and Hou, G.: Overview of Sensitivity Analysis and Shape Optimization for Complex Aerodynamic Configuration. *J. Aircr.*, vol. 36, 1999, pp. 87–96.

36. Newman, J.; Taylor, A.; and Burgreen, G.: An Unstructured Grid Approach to Sensitivity Analysis and Shape Optimization Using Euler Equations. AIAA-95-1646, 1995.

37. Pironneau, O.: Optimal Shape Design for Aerodynamics. *Optimum Design Methods in Aerodynamics*, AGARD-VKI Lecture Series, AGARD Rept. 803, April 1994, pp. 6-1–6-40.

38. Reuther, J.; Jameson, A.; Farmer, J.; Martinelli, J.; and Saunders, D.: Aerodynamic Shape Optimization of Complex Aircraft Configurations via an Adjoint Formulation. AIAA-96-0094, Jan. 1996.

39. Newman, J. C. III; Anderson, W. K.; and Whitfield, D. L.: Multidisciplinary Sensitivity Derivatives Using Complex Variables. MSSU-COE-ERC-98-08, Mississippi State University, MS 39762, July 1998.

40. Anderson, W. K.; Newman, J. C.; Whitfield, D. L.; and Nielsen, E.: Sensitivity Analysis for Navier-Stokes Equations on Unstructured Meshes Using Complex Variables. AIAA J., vol. 39, no. 1, 2001, pp. 56–63.

41. Anderson, W. K.; and Bonhaus, D. L.: An Implicit Upwind Algorithm for Computing Turbulent Flows on Unstructured Grids. Computers and Fluids, vol. 30, 1994, pp. 1–21.

42. Schumaker, L.: *Spline Functions: Basic Theory.* Wiley Interscience, 1980. (Reprinted by Krieger, Malabar, 1993.)

43. Li, W.; Naik, D.; and Swetits, J.: A Data Smoothing Technique for Piecewise Convex/Concave Curves. *SIAM J. Sci. Comput.*, vol. 17, 1996, pp. 517–537.

44. Huyse, L.: *Free-form Airfoil Shape Optimization Under Uncertainty Using Maximum Expected Value and Second-Order Second-Moment Strategies.* NASA/CR-2001-211020. ICASE Report No. 2001-18, 2001.

45. Huyse, L.; Lewis, M.; Padula, S.; and Li, W.: A Probabilistic Approach to Free-Form Airfoil Shape Optimization Under Uncertainty. *AIAA J.*, vol. 40, no. 9, 2002, pp. 1764–1772.

46. Rao, S.: *Reliability-Based Design.* McGraw-Hill, 1992.

47. Ben-Tal, A.; El Ghaoui, L.; and Nemirovski, A.: Chapter 6. Robustness. *Handbook on Semidefinite Programming*, H. Wolkowicz, R. Saigal, and L. Vandenberghe, eds., Kluwer Academic Publishers, 2000.

48. Ben-Tal, A.; and Nemirovski, A.: Robust Convex Optimization. *Math. Oper. Res.*, vol. 23, 1998, pp. 769–805.

49. Fowlkes, W.; and Creveling, C.: *Engineering Methods for Robust Product Design Using Taguchi Methods in Technology and Product Development.* Addison-Wesley, 1995.

50. Pardalos, P.; and Tsitsiringos, V.: *Financial Engineering, e-Commerce and Supply Chain.* Kluwer Academic Publishers, 2002.

51. Uryasev, S.: *Probabilistic Constrained Optimization: Methodology and Applications.* Kluwer Academic Publishers, 2000.

52. Gumbert, C.; Newman, P.; and Hou, G.: Effect of Random Geometric Uncertainty on the Computational Design of a 3-D Flexible Wing. *AIAA 20th Applied Aerodynamics Conference*, St. Louis, MO, June 24-28, 2002.

53. Garzon, V.; and Darmofal, D.: Using Computational Fluid Mechanics in Probabilistic Engineering Design. AIAA-2001-2528, June 2001.

54. Putko, M.; Taylor, A.; Newman, P.; and Green, L.: Approach for Input Uncertainty Propagation and Robust Design in CFD Using Sensitivity Derivatives. *J. Fluids Eng.*, vol. 124, 2002, pp. 60–69.

55. Zang, T.; Hemsch, M.; Hilburger, M.; Kenny, S.; Luckring, J.; Maghami, P.; Padula, S.; Stroud, W.: *Needs and Opportunities for Uncertainty-Based Multi-Disciplinary Design Methods for Aerospace Vehicles.* NASA/TM-2002-211462, April, 2002.

56. Kim, H.-J.; Kim, C.; and Rho, O.-H.: Multipoint Inverse Design Method for Transonic Wings. *J. Aircr.*, vol. 36, 1999, pp. 941–947.

57. Mangasarian, O.: Normal Solutions of Linear Programs. *Math. Prog. Study*, vol. 22, 1984, pp. 206–216.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)*
01-03-2004

2. REPORT TYPE
Technical Publication

3. DATES COVERED *(From - To)*
10/2000–9/2002

4. TITLE AND SUBTITLE
Using High Resolution Design Spaces for Aerodynamic Shape Optimization Under Uncertainty

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Wu Li and Sharon Padula

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER
762-20-61-01

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
NASA Langley Research Center
Hampton, VA 23681-2199

8. PERFORMING ORGANIZATION REPORT NUMBER
L–18355

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
National Aeronautics and Space Administration
Washington, DC 20546-0001

10. SPONSOR/MONITOR'S ACRONYM(S)
NASA

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
NASA/TP–2004–213003

12. DISTRIBUTION/AVAILABILITY STATEMENT
Unclassified-Unlimited
Subject Category 59
Availability: NASA CASI (301) 621-0390 Distribution: Standard

13. SUPPLEMENTARY NOTES
An electronic version can be found at http://techreports.larc.nasa.gov/ltrs/ or http://ntrs.nasa.gov.

14. ABSTRACT
This paper explains why high resolution design spaces encourage traditional airfoil optimization algorithms to generate noisy shape modifications, which lead to inaccurate linear predictions of aerodynamic coefficients and potential failure of descent methods. By using auxiliary drag constraints for a simultaneous drag reduction at all design points and the least shape distortion to achieve the targeted drag reduction, an improved algorithm generates relatively smooth optimal airfoils with no severe off-design performance degradation over a range of flight conditions, in high resolution design spaces parameterized by cubic B-spline functions. Simulation results using FUN2D in Euler flows are included to show the capability of the robust aerodynamic shape optimization method over a range of flight conditions.

15. SUBJECT TERMS
robust optimization, airfoil shape optimization, multiobjective optimization, lift-constrained drag minimization, variational analysis of derivatives, B-spline shape parameterization

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			STI Help Desk (email: help@sti.nasa.gov)
					19b. TELEPHONE NUMBER *(Include area code)*
U	U	U	UU	32	(301) 621-0390