

Application of Multi-fidelity Optimization Techniques to Benchmark Aerodynamic Design Problems

Jie Ren¹, Andrew Thelen², Anand Amrit³, Xiaosong Du⁴, and Leifur Leifsson⁵
Iowa State University, Ames, Iowa 50011

Yonatan A. Tesfahunegn⁶, and Slawomir Koziel⁷
Engineering Optimization & Modeling Center, Reykjavik University, Menntavegur 1, 101 Reykjavik, Iceland

Two-dimensional benchmark cases involving lift-constrained drag minimization in inviscid and viscous transonic flows are solved using derivative-free multi-fidelity optimization algorithms (space mapping and manifold mapping) and are compared with direct gradient-based optimization algorithms using adjoint sensitivities and trust regions. With 8 B-spline design variables, the multi-fidelity algorithms yield optimized shapes comparable to the shapes obtained by the direct algorithms but at a fraction of the cost. In particular for the inviscid case, the multi-fidelity algorithms needed less than 150 equivalent high-fidelity model evaluations (only flow solutions) taking approximately 460 minutes on a HPC with 32 processors, whereas the direct algorithm needed 391 high-fidelity model evaluations (flow and adjoint) taking approximately 4,494 minutes on the same HPC. For the viscous case, the multi-fidelity algorithms yield an optimized shape using less than 125 equivalent high-fidelity evaluations taking approximately 17.4 hours on the HPC. The direct algorithm was unsuccessful in optimizing the baseline shape in this case. A simple variation of surrogate-based optimization, the sequential approximate optimization (SAO), is utilized to optimize the twist distribution of a rectangular unswept wing in inviscid flow. Using 3 B-spline design variables, the SAO algorithm is able to obtain an optimized design with a near-elliptic section lift distribution. The total optimization cost is 22 high-fidelity model evaluations or approximately 42.5 hours on a HPC with 32 processors.

I. Introduction

This paper presents the results of solutions to several benchmark aerodynamic design problems, developed by the AIAA Applied Aerodynamics technical committee, using multi-fidelity optimization techniques as well as direct optimization (DO) techniques. The results are presented in a special session at the annual AIAA SciTech 2016 conference. This is the third time the special session is held and numerous research groups, from both academy and industry, present their results using various computational methods and optimization techniques.

In our prior work^{1,2}, we attempted to solve the two-dimensional cases: (i) drag minimization of NACA 0012 in inviscid flow, and (ii) lift-constrained drag minimization of RAE 2822 in viscous flow. In our first paper¹, we used multi-fidelity optimization with the space mapping³ (SM) technique to solve both cases. High-fidelity computational simulations were carried out using FLUENT⁴ with the hyperbolic mesh generator by Kinsey and Barth⁵, whereas the viscous-inviscid analysis method in MSES⁶ was used for the low-fidelity computations. The airfoil shapes were parameterized using PARSEC⁷. In the second paper², we attempted to solve the same two-dimensional cases using the multi-level optimization⁸ (MLO) technique. In our solutions, we used a family of low-fidelity models constructed using the hyperbolic mesh generator and the Stanford University Unstructured⁹ (SU²) solver to optimize a high-fidelity model by the same grid generator and flow solver as the low-fidelity models but with a higher mesh

¹ Graduate Student, Department of Aerospace Engineering, Student Member AIAA.

² Graduate Student, Department of Aerospace Engineering, Student Member AIAA.

³ Graduate Student, Department of Aerospace Engineering, Student Member AIAA.

⁴ Graduate Student, Department of Aerospace Engineering, Student Member AIAA.

⁵ Assistant Professor, Department of Aerospace Engineering, Senior Member AIAA.

⁶ Post-doctoral Fellow, School of Science and Engineering, Member AIAA.

⁷ Professor, School of Science and Engineering, Senior Member AIAA.

resolution (fully grid-independent solutions). Both papers demonstrated that multi-fidelity optimization with physics-based models is capable of solving the two-dimensional cases rapidly when compared to direct optimization (DO) with adjoint sensitivity information (see, e.g., Jameson¹⁰), and surrogate-based optimization¹¹ (SBO) with data-driven surrogate models¹². Although the cost of multi-fidelity optimization is much lower, those techniques have so far not matched the design quality of the designs produced by DO and SBO.

In this paper, we re-attempt to solve the two-dimensional benchmark cases (i) and (ii) using multi-fidelity techniques but using improved computational models as well as a new correction technique. In case (i) we generate an O-mesh in Pointwise¹³ for the half-plane (since the airfoil is symmetric and the angle of attack is zero) and solve the compressible Euler equations using SU². This improves the flow solutions and guarantees that the lift coefficient is equal to zero (which previously was hard to attain due to highly nonlinear flow features). In case (ii) we still use the hyperbolic mesh generator and solve the Reynolds-averaged Navier-Stokes (RANS) equations and the Spalart-Allmaras turbulence model¹⁴ with the SU² solver (which has been significantly improved with a more stable multigrid scheme). Both design cases (i) and (ii) are solved using multi-fidelity optimization with SM³ and with manifold mapping^{15,16} (MM), which has recently been applied to aerodynamic shape optimization¹⁷. We use B-spline curves¹⁸ to parameterize the airfoil shapes (8 design variables in each case). For comparison purposes, we solve cases (i) and (ii) with a gradient-based technique with adjoints and trust regions¹⁹. Additionally, case (i) is solved using pattern search^{20,21} (a derivative-free direct optimization technique). We, furthermore, attempt to solve the third benchmark case: (iii) optimize the twist distribution of a rectangular wing for minimum induced drag at fixed lift in inviscid flow. The goal is to see if the optimized wing shape recovers an elliptical section lift distribution (which is the lift distribution to obtain the theoretical minimum induced drag). In this case, we generate an unstructured mesh using the SUMO tool^{22,23} and solve the compressible Euler equations using SU². The twist distribution is parameterized using a B-spline curve with 3 design variables. This design case is solved using the sequential approximate optimization²⁴⁻²⁷ (SAO) technique where local approximation models are constructed using few samples of the high-fidelity model and sequentially optimized to yield an optimized wing twist distribution. This is a data-driven technique but it is efficient since only local approximations are used. Moreover, the technique lends itself to be used with multi-fidelity models, which will be investigated in future work.

The paper is organized as follows. Section II describes the general aerodynamic shape optimization (ASO) problem formulation, and the optimization techniques used in this work. Sections III, IV, and V give the detailed formulations, computational models, and optimization results of benchmark cases (i), (ii), and (iii), respectively. The paper ends with conclusions.

II. Optimization Techniques

This section describes the aerodynamic shape optimization problem formulation, and several techniques for its solution using direct optimization (derivative-free and gradient-based techniques) and multi-fidelity optimization.

A. Problem Formulation

The aerodynamic design problems considered in this work involve lift-constrained drag minimization of airfoils in two-dimensional transonic flow. The flow simulations are performed using accurate, but computationally expensive, high-fidelity computational fluid dynamics (CFD) models. These models are denoted here by \mathbf{f} . In general, nonlinear constrained optimization problems can be formulated as

$$\mathbf{x}^* = \arg \min_{\mathbf{x}} H(\mathbf{f}(\mathbf{x})) \text{ s.t. } \mathbf{g}(\mathbf{x}) \leq 0, \mathbf{h}(\mathbf{x}) = 0, \mathbf{l} \leq \mathbf{x} \leq \mathbf{u}, \quad (1)$$

where \mathbf{x} is the design variable vector of size $p \times 1$, \mathbf{x}^* is the optimized design, H is a scalar valued objective function, $\mathbf{f}(\mathbf{x})$ is a vector of size $q \times 1$ with the figures of merit, $\mathbf{g}(\mathbf{x})$ is a vector of size $M \times 1$ with the inequality constraints, $\mathbf{h}(\mathbf{x})$ is a vector of size $N \times 1$ with the equality constraints, and \mathbf{l} and \mathbf{u} are the design variable lower and upper bounds, respectively, both vectors of the same size as \mathbf{x} . The vectors $\mathbf{f}(\mathbf{x})$, $\mathbf{g}(\mathbf{x})$, and $\mathbf{h}(\mathbf{x})$ are all obtained, or derived, from computationally expensive models.

In ASO, the high-fidelity model \mathbf{f} calculates the figures of merit, e.g., in the two-dimensional case \mathbf{f} can be composed as follows: $\mathbf{f}(\mathbf{x}) = [C_{l,f}(\mathbf{x}) \ C_{d,f}(\mathbf{x}) \ A(\mathbf{x})]^T$ where $C_{l,f}(\mathbf{x})$ is the nondimensional lift coefficient, $C_{d,f}(\mathbf{x})$ is the nondimensional drag coefficient, and $A(\mathbf{x})$ the airfoil cross-sectional area. The subscript f denotes the high-fidelity model. In the case of a drag minimization problem, the objective function in (1) is set as $H(\mathbf{f}(\mathbf{x})) = C_{d,f}(\mathbf{x})$. The inequality constraints are set as $g_1(\mathbf{x}) = C_{m,min} - C_{m,f}(\mathbf{x}) \leq 0$, and $g_2(\mathbf{x}) = A_{min} - A(\mathbf{x}) \leq 0$, where $C_{m,min}$ is a minimum allowable pitching moment coefficient, $C_{m,f}(\mathbf{x})$ is the pitching moment of the current design \mathbf{x} , A_{min} is a minimum cross-

sectional area, and $A(\mathbf{x})$ the cross-sectional area of the design \mathbf{x} . The equality constraint is $h_1(\mathbf{x}) = C_{L,t} - C_{L,f}(\mathbf{x}) = 0$, where $C_{L,t}$ is the target lift coefficient, and $C_{L,f}(\mathbf{x})$ is the lift coefficient of the current design. Here, the angle of attack, α , is used as a dummy variable to find the target lift coefficient value. All the constraints are handled directly in the optimization process.

B. Direct Optimization

Direct optimization, used here for comparison and validation purposes, can be broadly categorized into derivative-free and gradient-based approaches.¹¹ Methods of the derivative-free type are typically more costly than their gradient-based counterparts, but have the benefit of being more immune to numerical noise. The latter may be present when using CFD simulation models, especially with coarse discretization of the structure at hand. Due to the availability of adjoint technology¹⁰, it is possible to perform direct gradient-based optimization of computationally expensive simulation models. Using this technology, the cost of obtaining the gradients is almost equivalent to one flow solution for any number of design variables.

In this work, multi-fidelity algorithms are compared with direct optimization of the high-fidelity model \mathbf{f} using a gradient-based algorithm exploiting adjoint sensitivities¹⁰ and trust regions¹⁹. More specifically, the problem (1) is solved as an iterative process^{11,20}

$$\mathbf{x}^{(i+1)} = \arg \min_{\mathbf{x}, \|\mathbf{x} - \mathbf{x}^{(i)}\| \leq \delta^{(i)}} H(\mathbf{s}^{(i)}(\mathbf{x})), \quad (2)$$

where $\mathbf{x}^{(i)}$, $i = 0, 1, \dots$, is a sequence of approximate solutions to (1), whereas $\mathbf{s}_k^{(i)}(\mathbf{x})$ is a linear expansion of $\mathbf{f}(\mathbf{x})$ at $\mathbf{x}^{(i)}$ defined as

$$\mathbf{s}^{(i)}(\mathbf{x}) = \mathbf{f}(\mathbf{x}^{(i)}) + \nabla \mathbf{f}(\mathbf{x}^{(i)}) \cdot (\mathbf{x} - \mathbf{x}^{(i)}). \quad (3)$$

Here, the gradient of the model \mathbf{f} (applies separately for the drag and lift coefficient) is obtained by the adjoint equation (see, e.g., Jameson¹⁰). The linear model (3) satisfies the zero- and first-order consistency conditions with the function $\mathbf{s}_k^{(i)}(\mathbf{x})$ at $\mathbf{x}^{(i)}$, i.e., $\mathbf{s}_k^{(i)}(\mathbf{x}^{(i)}) = \mathbf{f}(\mathbf{x}^{(i)})$, and $\nabla \mathbf{s}_k^{(i)}(\mathbf{x}^{(i)}) = \nabla \mathbf{f}(\mathbf{x}^{(i)})$. Optimization of the linear model is constrained to the vicinity of the current design defined as $\|\mathbf{x} - \mathbf{x}^{(i)}\| \leq \delta^{(i)}$, with the trust region radius $\delta^{(i)}$ adjusted adaptively using the standard trust region rules²³. The termination conditions for the algorithm (2) are: (i) $\|\mathbf{x}^{(i)} - \mathbf{x}^{(i-1)}\| < \varepsilon_x$, (ii) $|H^{(i)} - H^{(i-1)}| < \varepsilon_H$, (iii) $\delta^{(i)} < \varepsilon_\delta$, where ε_x , ε_H , and ε_δ are user defined convergence tolerances. For direct optimization of the aerodynamics design benchmark problems, we use: $\varepsilon_x = 10^{-6}$, $\varepsilon_H = 10^{-7}$, and $\varepsilon_\delta = 10^{-6}$. The solution of algorithm (2) with the model (3) is carried out using the pattern search algorithm^{20,21} and MATLAB's fmincon algorithm²⁸.

1. Pattern Search

The pattern search algorithm^{20,21} is a stencil-based local optimization method that explores the neighborhood of the current design point. A rectangular grid (i.e., one point in each direction and in each dimension) is used in our implementation. The search process utilizes grid-constrained line search with the search direction determined using the objective function gradient estimated from perturbed designs. In case of a failure the best perturbation (if better than the current design) is selected. Finally, the grid is refined in case the poll step does not lead to an improved design. The poll stage of the pattern search process is illustrated below in Fig. 1. For more information about the particular pattern search algorithm used for this study, refer to Refs. 20, and 21.

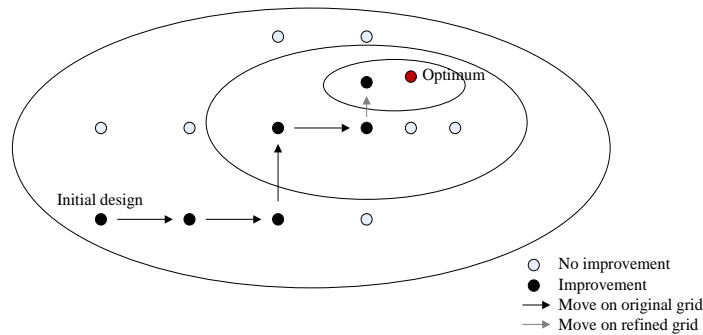


Figure 1. An illustration of the pattern search algorithm^{20,21}.

2. MATLAB's *fmincon*

Matlab's *fmincon*²⁸ utilizes several optimization algorithms (depending on the user preference as well as the scale of the problem at hand). These include trust region reflective algorithm where the objective function is represented using its linear or quadratic expansion models optimized using standard trust region rules¹⁹, a sequential quadratic programming (SQP) algorithm, where the original problem is solved iteratively by replacing the original objective function (and nonlinear constraints) by their respective local quadratic models (linear for constraint functions), as well as an interior point algorithm, where the original constrained problem is replaced by a sequence of approximate minimization problems using a barrier method. In this work, we use the interior point algorithm.

C. Multi-fidelity Optimization with Space Mapping

A generic multi-fidelity optimization algorithm with trust regions produces a sequence $\mathbf{x}^{(i)}$, $i = 0, 1, \dots$, of approximate solutions to (1) using (2) with a surrogate model $\mathbf{s}^{(i)}(\mathbf{x})$ iteration i . The surrogate model \mathbf{s} is a suitably corrected low-fidelity model \mathbf{c} . In this work, two (parametric) correction methods are considered: multi-point space mapping³ and manifold mapping^{15,16,17}.

Multi-point SM constructs an aerodynamic surrogate model as³

$$\mathbf{s}^{(i)}(\mathbf{x}) = \mathbf{A}^{(i)} \circ \mathbf{c}(\mathbf{x}) + \mathbf{D}^{(i)} + \mathbf{q}^{(i)} = [a_l^{(i)} C_{l,c}(\mathbf{x}) + d_l^{(i)} + q_l^{(i)} \quad a_d^{(i)} C_{d,c}(\mathbf{x}) + d_d^{(i)} + q_d^{(i)} \quad A_c(\mathbf{x})]^T, \quad (4)$$

where \circ denotes component-wise multiplication. Note that there is no need to map A_c because $A_c(\mathbf{x}) = A_f(\mathbf{x})$ for all \mathbf{x} . The response correction parameters $\mathbf{A}^{(i)}$ and $\mathbf{D}^{(i)}$ are obtained by solving

$$[\mathbf{A}^{(i)}, \mathbf{D}^{(i)}] = \arg \min_{[\mathbf{A}, \mathbf{D}]} \sum_{k=0}^i \|\mathbf{f}(\mathbf{x}^{(k)}) - \mathbf{A} \circ \mathbf{c}(\mathbf{x}^{(k)}) + \mathbf{D}\|^2, \quad (5)$$

i.e., the response scaling is supposed to (globally) improve the matching for all previous iteration points. The additive response correction term $\mathbf{q}^{(i)}$ is defined as

$$\mathbf{q}^{(i)} = \mathbf{f}(\mathbf{x}^{(i)}) - [\mathbf{A}^{(i)} \circ \mathbf{c}(\mathbf{x}^{(i)}) + \mathbf{D}^{(i)}], \quad (6)$$

i.e., it ensures perfect matching between the surrogate and the high-fidelity model at the current design $\mathbf{x}^{(i)}$, $\mathbf{s}^{(i)}(\mathbf{x}^{(i)}) = \mathbf{f}(\mathbf{x}^{(i)})$ (zero-order consistency).

The term $\mathbf{q}^{(i)}$ can be calculated using (6). $\mathbf{A}^{(i)}$ and $\mathbf{D}^{(i)}$ can be obtained analytically as³

$$\begin{bmatrix} a_l^{(i)} \\ d_l^{(i)} \end{bmatrix} = (\mathbf{C}_l^T \mathbf{C}_l)^{-1} \mathbf{C}_l^T \mathbf{F}_l, \quad \begin{bmatrix} a_d^{(i)} \\ d_d^{(i)} \end{bmatrix} = (\mathbf{C}_d^T \mathbf{C}_d)^{-1} \mathbf{C}_d^T \mathbf{F}_d, \quad (7)$$

where

$$\mathbf{C}_l = \begin{bmatrix} C_{l,c}(\mathbf{x}^{(0)}) & C_{l,c}(\mathbf{x}^{(1)}) & \dots & C_{l,c}(\mathbf{x}^{(i)}) \\ 1 & 1 & \dots & 1 \end{bmatrix}^T, \quad \mathbf{F}_l = [C_{l,f}(\mathbf{x}^{(0)}) \quad C_{l,f}(\mathbf{x}^{(1)}) \quad \dots \quad C_{l,f}(\mathbf{x}^{(i)})]^T, \quad (8)$$

$$\mathbf{C}_d = \begin{bmatrix} C_{d,c}(\mathbf{x}^{(0)}) & C_{d,c}(\mathbf{x}^{(1)}) & \dots & C_{d,c}(\mathbf{x}^{(i)}) \\ 1 & 1 & \dots & 1 \end{bmatrix}^T, \quad \mathbf{F}_d = [C_{d,f}(\mathbf{x}^{(0)}) \quad C_{d,f}(\mathbf{x}^{(1)}) \quad \dots \quad C_{d,f}(\mathbf{x}^{(i)})]^T, \quad (9)$$

which is a least-square optimal solution to the linear regression problem $\mathbf{C}_l a_l^{(i)} + d_l^{(i)} = \mathbf{F}_l$ and $\mathbf{C}_d a_d^{(i)} + d_d^{(i)} = \mathbf{F}_d$, equivalent to (5). Note that the matrices $\mathbf{C}_l^T \mathbf{C}_l$ and $\mathbf{C}_d^T \mathbf{C}_d$ are non-singular for $i > 1$. For $i = 1$ only the multiplicative correction with $\mathbf{A}^{(i)}$ components are used, which can be calculated in a similar way.

The solution of algorithm (2) with the surrogate model (4) is carried out using the pattern search algorithm,^{24,25} and the termination conditions are the same as for the direct optimization algorithm (Sec. II.B) but with $\varepsilon_x = 10^{-3}$, $\varepsilon_H = 10^{-4}$, and $\varepsilon_\delta = 10^{-3}$.

D. Multi-fidelity Optimization with Manifold Mapping

Manifold mapping^{15,16,17} (MM) is a simple yet interesting response correction technique that is capable of comprehensive exploitation of available high-fidelity model data. In its basic version, the MM surrogate model is defined as^{15,16,17}

$$\mathbf{s}^{(i)}(\mathbf{x}) = \mathbf{f}(\mathbf{x}^{(i)}) + \mathbf{S}^{(i)}(\mathbf{c}(\mathbf{x}) - \mathbf{c}(\mathbf{x}^{(i)})) \quad (10)$$

with $\mathbf{S}^{(i)}$ being the 3×3 correction matrix in our case (in general, the size of \mathbf{S} is equal to the number of the components in \mathbf{c} , \mathbf{f} , and \mathbf{s})

$$\mathbf{S}^{(i)} = \Delta \mathbf{F} \cdot \Delta \mathbf{C}^\dagger, \quad (11)$$

where

$$\Delta \mathbf{F} = [\mathbf{f}(\mathbf{x}^{(i)}) - \mathbf{f}(\mathbf{x}^{(i-1)}) \quad \dots \quad \mathbf{f}(\mathbf{x}^{(i)}) - \mathbf{f}(\mathbf{x}^{(\max\{i-n, 0\})})], \quad (12)$$

and

$$\Delta \mathbf{C} = [\mathbf{c}(\mathbf{x}^{(i)}) - \mathbf{c}(\mathbf{x}^{(i-1)}) \quad \dots \quad \mathbf{c}(\mathbf{x}^{(i)}) - \mathbf{c}(\mathbf{x}^{(\max\{i-n, 0\})})]. \quad (13)$$

The pseudoinverse, denoted by † , is defined as^{15,16,17}

$$\Delta \mathbf{C}^\dagger = \mathbf{V}_{\Delta \mathbf{C}} \Sigma_{\Delta \mathbf{C}}^\dagger \mathbf{U}_{\Delta \mathbf{C}}^T, \quad (14)$$

where $\mathbf{U}_{\Delta \mathbf{C}}$, $\Sigma_{\Delta \mathbf{C}}$, and $\mathbf{V}_{\Delta \mathbf{C}}$ are the factors in the singular value decomposition of the matrix $\Delta \mathbf{C}$. The matrix $\Sigma_{\Delta \mathbf{C}}^\dagger$ is the result of inverting the nonzero entries in $\Sigma_{\Delta \mathbf{C}}$, leaving the zeroes invariant. Upon convergence, the linear correction \mathbf{S}^* (being the limit of $\mathbf{S}^{(i)}$ with $i \rightarrow \infty$) maps the point $\mathbf{c}(\mathbf{x}^*)$ to $\mathbf{f}(\mathbf{x}^*)$, and the tangent plane for $\mathbf{c}(\mathbf{x})$ at $\mathbf{c}(\mathbf{x}^*)$ to the tangent plane for $\mathbf{f}(\mathbf{x})$ at $\mathbf{f}(\mathbf{x}^*)$. It should be noted that although MM does not explicitly use sensitivity information, the surrogate and the high-fidelity model Jacobians become more and more similar to each other towards the end of the MM optimization process (i.e., when $\|\mathbf{x}^{(i)} - \mathbf{x}^{(i-1)}\| \rightarrow 0$) so that the surrogate (approximately) satisfies both zero- and first-order consistency conditions with \mathbf{f} . This allows for a more precise identification of the high-fidelity model optimum. On the other hand, the correction matrix $\mathbf{S}^{(i)}$ can be defined using exact Jacobians of the low- and high-fidelity models if available.

The solution of algorithm (2) with the surrogate model (10) is carried out using the pattern search algorithm,^{20,21} and the termination conditions are the same as for the space mapping optimization algorithm (Sec. II.C.1). In the specific case of airfoils, where the model response vectors consist of three components (lift, drag, and cross-section area), the correction matrix \mathbf{S} should be identity with respect to the third components (the area is identical for both the low- and high-fidelity model).

E. Sequential Approximation Optimization

A simple variation of SBO with data-driven surrogates is sequential approximate optimization²⁴⁻²⁷ (SAO). In SAO, the optimization is limited to a certain subregion (typically, rectangular shaped) of the search space, where a local surrogate model, such as a low-order polynomial, is set up based on high-fidelity model data sparsely sampled in the subregion. The new design is found by optimizing the surrogate within the subregion, which is followed by defining a new subregion according to the chosen relocation strategy.

In this paper, we follow the work by Leifsson *et al.*²⁹ and use SAO to search the design space for one of the benchmark cases (see Sec. V). The SAO algorithm works as follows. In each iteration, a local surrogate model is constructed using sparsely sampled \mathbf{f} data and a low-order polynomial approximation. Let $\mathbf{x}^{(j)} = [x_1^{(j)} \ x_2^{(j)} \ \dots \ x_n^{(j)}]^T$ be a design obtained as a result of iteration $j-1$ of the algorithm. Let $\mathbf{d}^{(j)} = [d_1^{(j)} \ d_2^{(j)} \ \dots \ d_n^{(j)}]^T$ be the size parameter that is used to define the vicinity of the vector $\mathbf{x}^{(j)}$. A local response surface approximation (RSA) model is created in the interval $[\mathbf{x}^{(j)} - \mathbf{d}^{(j)}, \mathbf{x}^{(j)} + \mathbf{d}^{(j)}]$. We denote by $\mathbf{X}_T^{(j)} = \{\mathbf{x}_t^{(j,1)}, \dots, \mathbf{x}_t^{(j,N)}\}$ the training set obtained by sampling the aforementioned vicinity. The RSA model is obtained by approximating the data pairs $\{\mathbf{x}_t^{(j,k)}, \mathbf{f}(\mathbf{x}_t^{(j,k)})\}$, $k = 1, \dots, N$. In this work, a second-order polynomial model $\mathbf{q}^{(j)}$ is utilized as follows²⁹

$$\mathbf{q}^{(j)}(\mathbf{x}) = \lambda_0^{(j)} + \sum_{k=1}^n \lambda_k^{(j)} x_k + \sum_{k=1}^n \sum_{s=k}^n \lambda_{k,s}^{(j)} x_k x_s. \quad (18)$$

The unknown coefficients $\boldsymbol{\lambda} = [\lambda_0 \ \lambda_1 \ \dots \ \lambda_n \ \lambda_{11} \ \lambda_{12} \ \dots \ \lambda_{1n} \ \lambda_{22} \ \dots \ \lambda_{nn}]$ are found by solving the following the linear regression problems $q(\mathbf{x}_i^{(j,k)}) = \mathbf{f}(\mathbf{x}_i^{(j,k)})$, $k = 1, \dots, N$. A unique least-square solution to this problem exists and can be found analytically assuming that the number of training points is equal or larger than the number of unknown coefficients.

The optimization algorithm workflow is the following (here, $\mathbf{x}^{(0)}$ is the initial design, and $\mathbf{d}^{(0)}$ is the initial vicinity size, usually, a fraction of the design space size):

1. Set $j = 0$;
2. Sample the interval $[\mathbf{x}^{(j,i)} - \mathbf{d}^{(j)}, \mathbf{x}^{(j,i)} + \mathbf{d}^{(j)}]$ to obtain the training set $\mathbf{X}_T^{(j)}$;
3. Evaluate the function \mathbf{f} at $\mathbf{X}_T^{(j)}$;
4. Identify the RSA model $\mathbf{q}^{(j)}$;
5. Find a candidate design \mathbf{x}^{tmp} as $\mathbf{x}^{tmp} = \arg \min_{\mathbf{x}^{(j,i)} - \mathbf{d}^{(j)} \leq \mathbf{x} \leq \mathbf{x}^{(j,i)} + \mathbf{d}^{(j)}} H(\mathbf{q}^{(j)}(\mathbf{x}))$;
6. Calculate the gain ratio $r = \frac{H(\mathbf{f}(\mathbf{x}^{tmp})) - H(\mathbf{f}(\mathbf{x}^{(j)}))}{H(\mathbf{q}^{(j)}(\mathbf{x}^{tmp})) - H(\mathbf{q}^{(j)}(\mathbf{x}^{(j)}))}$;
7. If $r > r_{incr}$, set $\mathbf{d}^{(j+1)} = \mathbf{d}^{(j)} \cdot m_{incr}$;
8. If $r < r_{decr}$, set $\mathbf{d}^{(j+1)} = \mathbf{d}^{(j)} / m_{decr}$;
9. If $r > 0$, set $\mathbf{x}^{(j+1)} = \mathbf{x}^{tmp}$; otherwise $\mathbf{x}^{(j+1)} = \mathbf{x}^{(j)}$;
10. If the termination condition is satisfied END; else $j = j + 1$ and go to 2;

In the above procedure, the updating parameters for the trust region size, i.e., r_{incr} , m_{incr} , r_{decr} , and m_{decr} are set by the user. In this work, we set $r_{incr} = 0.75$, $m_{incr} = 1.5$, $r_{decr} = 0.25$, and $m_{decr} = 2$. The termination conditions are the same as for the direct optimization algorithm (Sec. II.B), but with $\varepsilon_x = 10^{-3}$, $\varepsilon_H = 10^{-4}$, and $\varepsilon_\delta = 10^{-3}$. Figure 2 illustrates the operation of the optimization procedure for $n = 2$.

III. Benchmark Case I: Drag Minimization of the NACA 0012 in Transonic Inviscid Flow

A. Problem Statement

The objective is to minimize the drag coefficient (C_d) of modified NACA 0012 airfoil section at a free-stream Mach number of $M_\infty = 0.85$ and an angle of attack $\alpha = 0$ deg. subject to a minimum thickness constraint. The optimization problem is stated as

$$\min_{\mathbf{l} \leq \mathbf{x} \leq \mathbf{u}} C_d, \quad (15)$$

where \mathbf{x} is the vector of design variables, and \mathbf{l} and \mathbf{u} are the lower and upper bounds, respectively. The thickness constraint is stated as

$$z(x) \geq z(x)_{baseline}, \quad (16)$$

where $z(x)$ is the airfoil thickness, $x \in [0,1]$ is the chord-wise location, and $z(x)_{baseline}$ is the thickness of the baseline airfoil, which is a modified version of the NACA 0012, defined as

$$z(x)_{baseline} = \pm 0.6(0.2969\sqrt{x} - 0.1260x - 0.3516x^2 + 0.2843x^3 - 0.1036x^4). \quad (17)$$

In our implementation, the objective function is set as $H(\mathbf{f}(\mathbf{x})) = C_d$ and the thickness constraint is handled directly. The design variables and their bounds are described as follows.

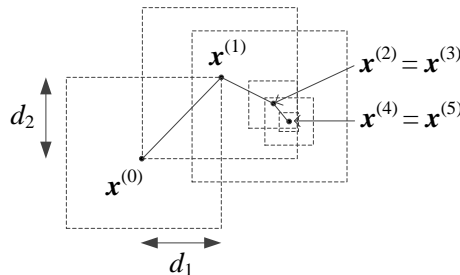


Figure 2. A conceptual illustration of the sequential approximate optimization procedure ($n = 2$).

B. Design Variables

The airfoil shape design variables are defined by the parametrization method. The design variable vector is written as $\mathbf{x} = \mathbf{p}$, where \mathbf{p} is a vector of size $m \times 1$, with m being the total number of control parameters of the shape parameterization technique. In this work, we use B-spline curves (see, e.g., Farin¹⁸) for the shape parametrization. The airfoil surfaces are written in parametric form as¹⁸

$$x(t) = \sum_{i=1}^{n+1} X_i N_{i,k}(t), \quad z(t) = \sum_{i=1}^{n+1} Z_i N_{i,k}(t), \quad (18)$$

where (x, z) are the Cartesian coordinates of the surface, $N_{i,k}$ is the B-spline basis function of order k , (X_i, Z_i) are the coordinates of the B-spline control polygon, and $n+1$ is the total number of control points. Note that the surface description with (18) is continuous. The control points are used as design variables and allowed only to move freely vertically as shown in Fig. 3 (in this figure we only show the upper surface of the airfoil). Each designable control point is free to move in the vertical direction only. Thus, we have $\mathbf{x} = \mathbf{p} = [Z_1 \ Z_2 \ \dots \ Z_{n+1}]^T$ and the corresponding X_i are fixed during the optimization.

In Benchmark Case I, we use 10 control points, as shown in Fig. 3, where two are fixed at the leading- and trailing-edges, and the other ones can move in the vertical direction, yielding 8 design variables. Based on a fit to the modified NACA 0012 of (17), we set the x -locations of the free control points as $\mathbf{X} = [0 \ 0.0536 \ 0.2000 \ 0.4000 \ 0.5854 \ 0.7527 \ 0.8854 \ 0.9706]^T$. The initial design variable vector is $\mathbf{x} = [0.0185 \ 0.0474 \ 0.0654 \ 0.0633 \ 0.0486 \ 0.0323 \ 0.0161 \ 0.0043]^T$. The lower bound of \mathbf{x} is set as zero, i.e., $\mathbf{l} = \mathbf{0}$, and the upper bound is set as one, i.e., $\mathbf{u} = \mathbf{1}$.

C. Inviscid Aerodynamics Model

The SU² computer code⁹ is utilized for the inviscid fluid flow simulations. The steady compressible Euler equations are solved with an implicit density-based formulation. The convective fluxes are calculated using the second order Jameson-Schmidt-Turkel (JST) scheme³⁰. Three multi-grid levels are used for solution acceleration. Asymptotic convergence to a steady state solution is obtained in each case. The flow solver convergence criterion is the one that occurs first of the two: (i) the change in the drag coefficient value over the last 100 iterations is less than 10^{-4} , or (ii) a maximum number of iterations of 1,000 is met.

An O-type computational mesh is generated using Pointwise¹³ (see Fig. 4). Since the airfoil is symmetrical and the angle of attack is fixed at zero, only the half-plane is considered. The farfield boundary is set 55 chord lengths away from the airfoil surface. The mesh density is controlled by the number of cells on the airfoil surface and the number of cells normal to the surface. The results of a grid convergence study, given in Table 1, revealed that a 512×512 mesh (shown number 5 in the table) is required for convergence within 0.1 drag count (1 drag count is $\Delta C_d = 10^{-4}$) when compared with the next mesh. Distance to the first grid point is $0.0015c$. The flow simulation for Mesh 5 takes about 4.2 minutes. An adjoint solution for the drag coefficient take approximately the same amount of time. It should be noted that throughout an optimization run the airfoil shape may change significantly and the flow and adjoint simulation times may vary depending on the particular shape.

For the optimization studies, Mesh 5 will be used as the high-fidelity model \mathbf{f} , and Mesh 3 as the low-fidelity model \mathbf{c} (used only for space mapping and manifold mapping). For the low-fidelity model, the maximum number of solver iterations is set to 300. Figure 5(a) shows the solver convergence of the low-fidelity model, and Fig. 5(b) gives a comparison of the low- and the high-fidelity models. The comparison indicates that the low-fidelity model may be a relatively good representation of the high-fidelity one.

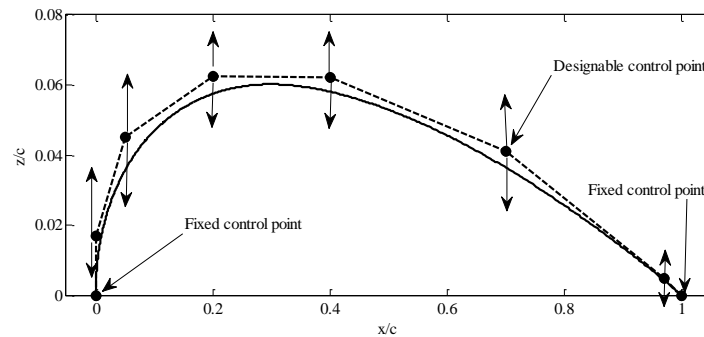


Figure 3. B-spline parameterization for the upper surface of the airfoil.

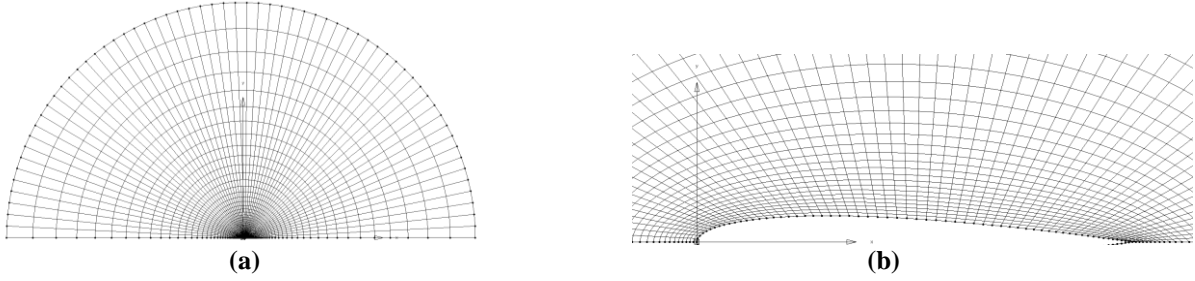


Figure 4. The half-plane O-mesh used in the inviscid model: (a) farfield view, (b) a view close to the surface.

Table 1. Grid convergence study for the baseline shape of BC I at $M = 0.85$ and $\alpha = 0.0^\circ$.

Mesh	Grid Size	C_l (cts)	C_d (cts)	Simulation Time* (min)
1	32×32 (961)	0.0	496.7194	0.29
2	64×64 (3,969)	0.0	480.7083	0.41
3	128×128 (16,129)	0.0	470.2201	0.71
4	256×256 (65,025)	0.0	469.3722	1.72
5	512×512 (261,121)	0.0	468.4470	4.24
6	$1,024 \times 1,024$ (1,046,529)	0.0	468.3714	24.60

* Computed on a high-performance cluster with 32 processors. Flow solution only.

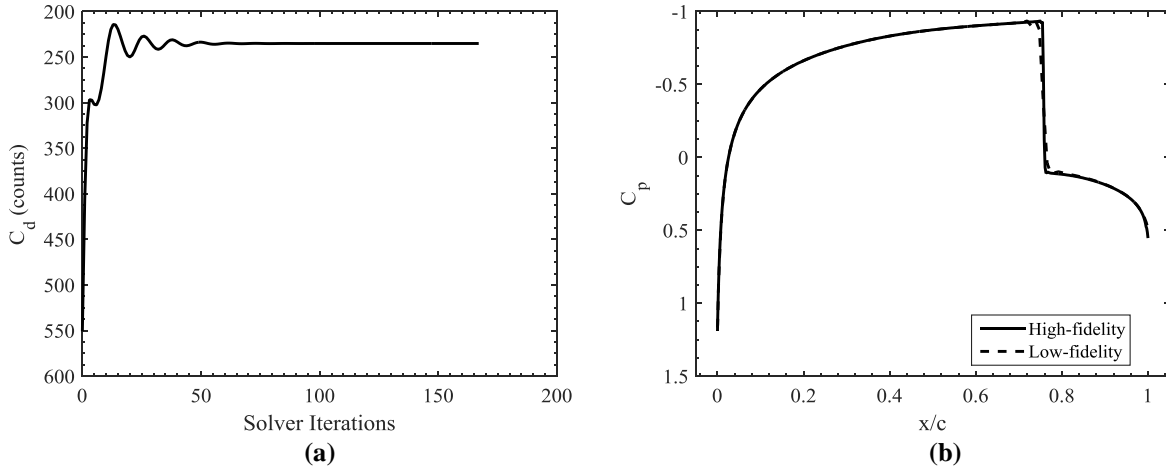


Figure 5. Inviscid flow simulation results for the modified NACA 0012 at $M = 0.85$ and $\alpha = 0.0^\circ$: (a) the evolution of the drag coefficient obtained by the low-fidelity model; (b) a comparison of the pressure distributions obtained by the high- and low-fidelity models.

D. Optimization Results

Benchmark Case I (BC I) is solved using the optimization algorithms listed in Table 2. Figure 6 shows the convergence history of the algorithms. In particular, Fig. 6(a) shows the convergence of the argument \mathbf{x} , and Fig. 6(b) shows the evolution of the objective function H (the drag coefficient in this case). The direct algorithms A and B terminate on the objective function, whereas the space mapping and manifold mapping algorithms terminate on the argument.

In terms of design quality, Direct B obtained the lowest drag coefficient value of 47.7 counts (Table 3). Direct A obtained a drag coefficient value of 55.1 counts, whereas the space mapping and manifold mapping algorithms converge to the same design with a drag coefficient value of 74.5 counts. Although, there is a significant difference in the drag coefficient values the overall shapes look very similar (see Fig. 7(a)). However, there is a slight difference between the shapes obtained by the direct algorithms and the multi-fidelity algorithms. In particular, the

shapes obtained by the direct algorithms have a slightly fuller shape between $x/c = 0.6$ to $x/c = 0.8$ than the shapes obtained by the multi-fidelity algorithms. Other parts of the shapes are comparable. In terms of the pressure coefficient distributions, shown in Figs. 7(b), the shape obtained by Direct B has a distinctly lower peak near the leading-edge of the airfoil compared to the others. All the shapes have reduced the shock strength significantly when compared to the baseline pressure distribution. Figure 8 shows the pressure coefficient contours for all the shapes, including the baseline. The results of a grid convergence study of the optimized shape obtained by manifold mapping, shown in Table 4, indicates that the high-fidelity grid used in the optimization run (the 512×512 grid) is converged within 0.3 drag counts. This indicates that a finer mesh may have to be used for the high-fidelity model in order to ensure a drag coefficient resolution of less than 0.1 drag counts.

In terms of computational cost, the multi-fidelity algorithms need the least amount time. In particular, they need 4 high-fidelity model evaluations (N_f), and 778 low-fidelity model evaluations (N_c); which, in total, is equivalent to less than 150 high-fidelity model evaluations (N_{tot}). The total optimization time is about 459 minutes. Direct B needed 391 high-fidelity model evaluations (includes both flow and adjoint solutions) and the total optimization time was about 4,949 minutes. Note that the multi-fidelity methods used in this study only require the flow solutions but not the adjoint solutions like the gradient-based algorithm. Consequently, the space mapping and manifold mapping algorithms are more efficient than the Direct B algorithm. Direct A needed 1,383 high-fidelity model evaluations (only flow solutions) and the total optimization time was about 6,939 minutes.

Table 2. Details of the optimization algorithms used for BC I.

Algorithm	Driver	Model	Mesh for f	Mesh for c	Adjoint	Trust Region
Direct A	Pattern search	f	5	N/A	N/A	N/A
Direct B	MATLAB's fmincon	(3)	5	N/A	Yes	Yes
Space Mapping	Pattern search	(4)	5	3	No	Yes
Manifold Mapping	Pattern search	(10)	5	3	No	Yes

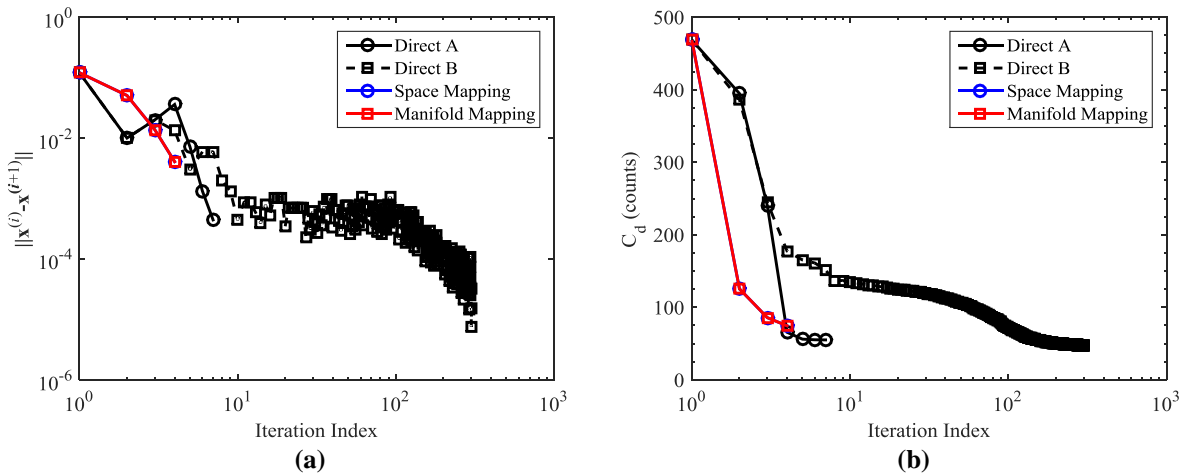


Figure 6. BC I convergence histories of the optimization runs.

Table 3. Optimization results for BC I.

Parameter/Method	Baseline	Direct A	Direct B	SM	MM
C_l (l.c.)	0.0	0.0	0.0	0.0	0.0
C_d (d.c.)	468.45	55.10	47.68	74.53	74.53
N_c	—	—	—	778	778
N_f	—	1,383	391*	4	4
N_{tot}	—	1,383	391*	< 150	< 150
t_c (min)	—	—	—	447.0	445.8
t_f (min)	—	6,939	4,494	12.3	13.1
t_{tot} (min)	—	6,939	4,494	459.3	458.9

* Primary flow solutions as well as adjoint solutions.

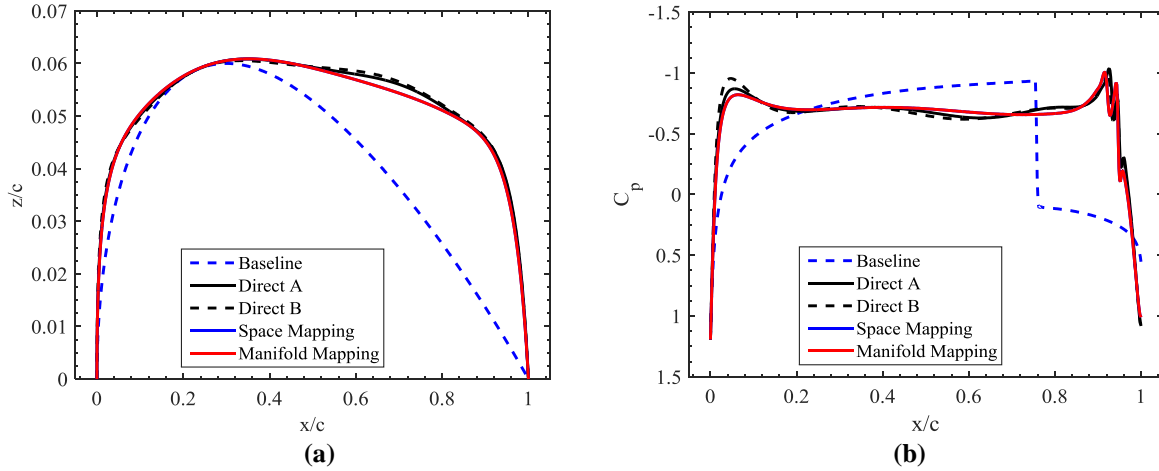


Figure 7. BC I baseline and optimized characteristics: (a) shapes, (b) pressure coefficient distributions.

Table 4. Grid convergence study for the optimized MM shape of BC I at $M = 0.85$ and $\alpha = 0.0^\circ$.

Mesh	Grid Size	C_l (cts)	C_d (cts)	Simulation Time* (min)
1	32×32 (961)	0.0	272.7192	0.25
2	64×64 (3,969)	0.0	129.5981	0.42
3	128×128 (16,129)	0.0	83.0096	0.76
4	256×256 (65,025)	0.0	77.2443	1.59
5	512×512 (261,121)	0.0	74.5209	4.41
6	$1,024 \times 1,024$ (1,046,529)	0.0	74.1940	23.34

* Computed on a high-performance cluster with 32 processors. Flow solution only.

IV. Benchmark Case II: Drag Minimization of the RAE 2822 in Transonic Viscous Flow

A. Problem Statement

The objective is to minimize the drag coefficient (C_d) of the RAE 2822 at a free-stream Mach number of $M_\infty = 0.734$, lift coefficient of 0.824, and Reynolds number of 6.5×10^6 subject to an area and pitching moment constraint. The task is to solve the following constrained optimization problem:

Solve

$$\min_{\mathbf{1} \leq \mathbf{x} \leq \mathbf{u}} C_d, \quad (19)$$

subject to the following constraints

$$C_l = 0.824, \quad (20)$$

$$C_m \geq -0.092, \quad (21)$$

$$A \geq A_{baseline}, \quad (22)$$

where C_m is the moment coefficient and A is the airfoil cross-sectional area nondimensionalized with the chord length squared.

The constant lift coefficient constraint (20) is implicitly satisfied in the flow solver by using the angle of attack as a dummy parameter (see Sec. III.B.3). In the implementation of the multi-fidelity algorithms, the pitching moment and cross-sectional area constraints (21) and (22), respectively, are handled through a penalty function. In the direct optimization algorithms, those constraints are handled directly.

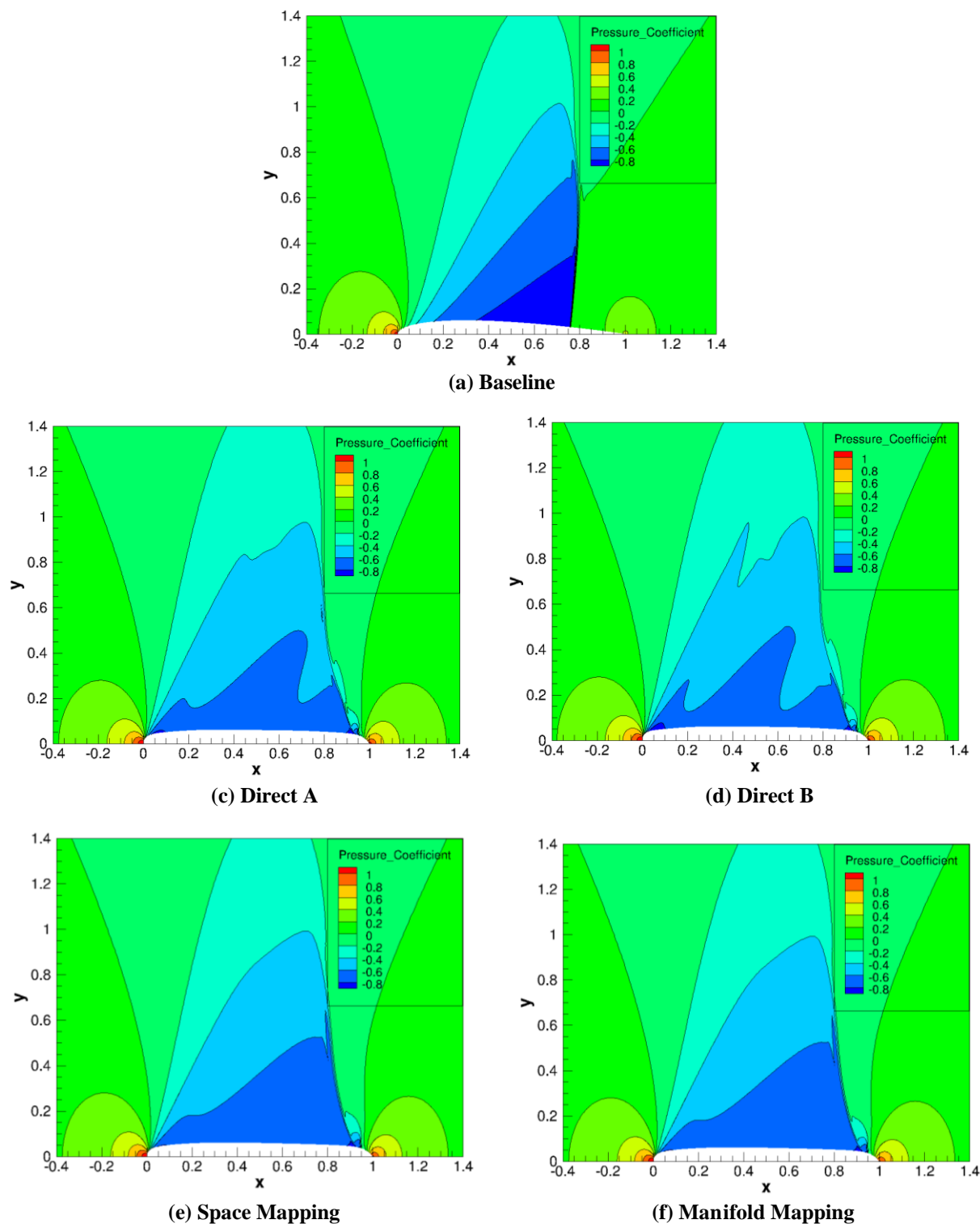


Figure 8. BC I pressure coefficient contours for baseline and optimized shapes.

B. Design Variables

The B-spline parameterization approach, described in Sec. III.A.2, is used in Benchmark Case II for the upper and lower surfaces. We use 10 control points, as shown in Fig. 9, where two are fixed at the leading- and trailing-edges, and the other ones, 4 for each surface, can move in the vertical direction. This yields 8 design variables. Based on a fit to the RAE 2822, we set the x -locations of the free control points as: $\mathbf{X} = [\mathbf{X}_u; \mathbf{X}_l]^T = [0.0 \ 0.15 \ 0.45 \ 0.80; 0.0 \ 0.35 \ 0.60 \ 0.90]^T$. The initial design variable vector is $\mathbf{x} = [\mathbf{x}_u; \mathbf{x}_l]^T = [0.0175 \ 0.0498 \ 0.0688 \ 0.0406; -0.0291 \ -0.0679 \ -0.0384 \ 0.0054]^T$. The lower bound of \mathbf{x} is set as $\mathbf{l} = [0 \ 0 \ 0 \ 0; -1 \ -1 \ -1 \ -1]^T$, and the upper bound is set as $\mathbf{u} = [1 \ 1 \ 1 \ 1; 0 \ 0 \ 0 \ 0]^T$.

C. Viscous Aerodynamics Model

The SU² implicit density-based flow solver²⁸ is used for the viscous case, solving the steady compressible Reynolds-averaged Navier-Stokes (RANS) equations with the Spalart-Allmaras turbulent model³¹. The convective flux will be calculated using the second order JST scheme²⁹. One multi-grid level is used for solution acceleration. The turbulent variables are convected using a first-order scalar upwind method. The flow solver convergence criterion is the one that occurs first of the two: (i) the change in the drag coefficient value over the last 100 iterations is less than 10^{-5} , or (ii) a maximum number of iterations of 5,000 is met.

The grids are generated using the hyperbolic C-mesh of Kinsey and Barth³² (see Fig. 10). The farfield is set 100 chords away from the airfoil surface. The grid points are clustered at the trailing edge and the leading edge of the airfoil to give a minimum streamwise spacing of $0.001c$, and the distance from the airfoil surface to the first node is $10^{-5}c$. The grid density is controlled by the number of points in the streamwise direction (N_s), and the number of points in the direction normal to airfoil surface (N_n). We set the number of points in the wake region equal to the number in the normal direction. The grid sizes are denoted by $N_s \times N_n$. Table 5 gives the results of a grid convergence study using the RAE 2822 airfoil at $M = 0.734$ and $C_l = 0.824$. The constant lift condition is determined by externally changing the angle of attack using a numerical optimization algorithm. Typically, 3 to 4 simulations are needed. The simulation times presented in Table 5 give the overall time to compute the constant lift condition. The times do not include the time it takes to calculate the adjoint solutions.

For the optimization studies, we use Mesh 2 for the high-fidelity model **f** and Mesh 1 for the low-fidelity model **c**. The low-fidelity model convergence criteria is the same as the high-fidelity one, but with the following values: drag coefficient convergence set to 10^{-4} , and the maximum number of iterations is set to 2,000. Fig. 11(a) shows that the low-fidelity solver is converged well within the 2,000 iteration limit, and Fig. 11(b) shows that the low-fidelity model is a good representation of high-fidelity one in terms of the pressure coefficient distributions.

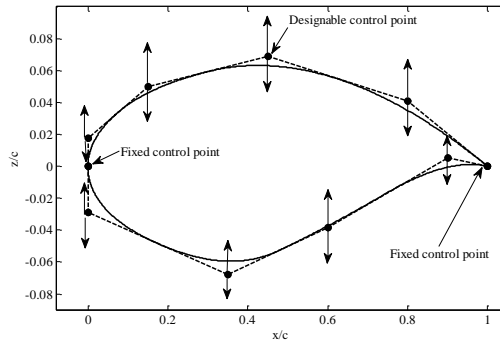


Figure 9. Airfoil shape parameterization using B-spline curves for the upper and lower surfaces.

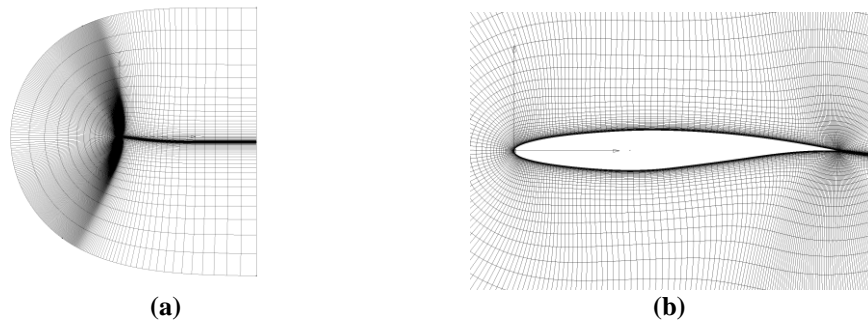


Figure 10. Hyperbolic C-mesh used in the viscous model: (a) farfield view, (b) view close to the surface.

Table 5. Grid convergence study for the baseline shape of Benchmark Case II.

Mesh	Grid Size	C_l (cts)	C_d (cts)	Simulation Time* (min)
1	120 × 40 (8,196)	82.39	257.59	5.5
2	240 × 80 (32,396)	82.35	220.28	19.0
3	480 × 160 (128,796)	82.40	204.67	77.2
4	960 × 320 (513,596)	82.36	198.65	518.8
5	1,920 × 640 (2,051,196)	82.41	203.93	2,269.8

* Computed on a high-performance cluster with 32 processors. Flow solution only.

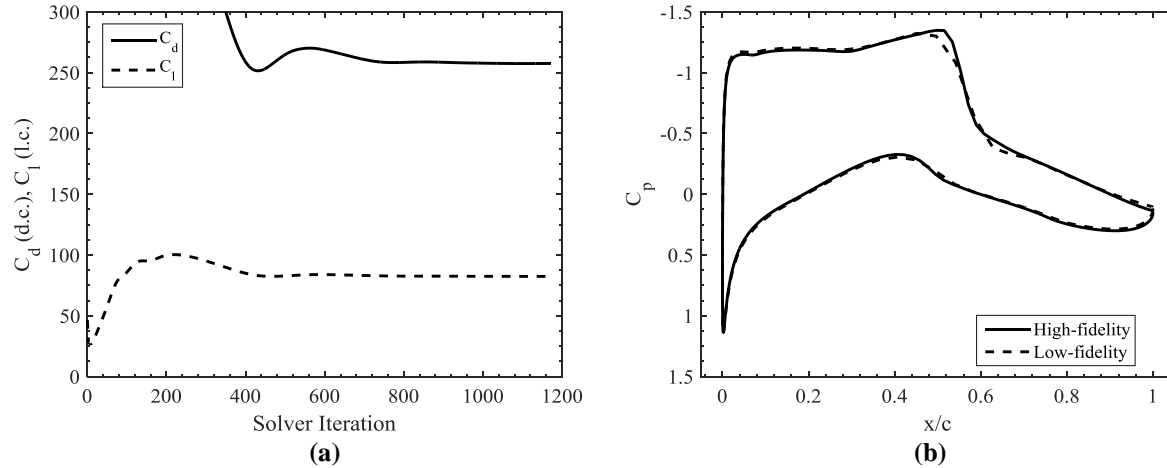


Figure 11. Viscous flow simulation results for RAE 2822 at $M = 0.734$ and $C_l = 0.824$: (a) the evolution of the lift and drag coefficients obtained by the low-fidelity model; (b) a comparison of the pressure distributions obtained by the high- and low-fidelity models.

D. Optimization Results

Benchmark Case II (BC II) is solved using Direct B, space mapping (SM), and manifold mapping (MM) algorithms as described in Table 2, but using Meshes 2 and 1 of Table 5 for the high- and low-fidelity models, respectively. Figure 12 shows the convergence history of the algorithms. The Direct B algorithm was unsuccessful, most likely due to excessive numerical noise of the chosen high-fidelity model, and changes the design only slightly; terminating after 4 design iterations based on the objective function. The multi-fidelity algorithms are, however, able to make significant reduction in the objective function and terminate after 3 design iterations based on the argument.

As can be seen by Table 6, the SM algorithm reduces the drag coefficient value from 220.3 counts to 131.4 counts (or by 88.9 counts), whereas the MM reduces it to 142 counts (or by 78.3 counts). Direct B reduced the drag coefficient value to 218.6 counts (or by 1.7 counts). Figure 13 shows the optimized shapes and pressure coefficient distributions. Figure 14 shows the pressure coefficient contours. Clearly, the SM and MM reduce the shock strength significantly. However, the lift coefficient and cross-sectional area constraints are slightly violated for both SM and MM. In particular, the lift coefficients are less than 0.04 counts below the constraint value of 82.4 counts, and the cross-sectional areas are 1.5% below the baseline value of 0.0779. These violations are minor. The Direct B algorithm violates the pitching moment coefficient constraint is slightly (or by less than 1%).

The optimization cost of the SM and MM algorithms is comparable. SM needs less than 131 equivalent high-fidelity model evaluations (14 high-fidelity and 263 low-fidelity), and MM needs less than 125 high-fidelity equivalent calls (14 high-fidelity and 210 low-fidelity). In terms of time, SM took 17.8 hours and MM 17.4 hours.

Tables 5 and 7 indicate that the meshes used in the studies here are not converged. Future studies will use Mesh 5 (around 2 million cells) as the high-fidelity model to ensure converged flow solutions.

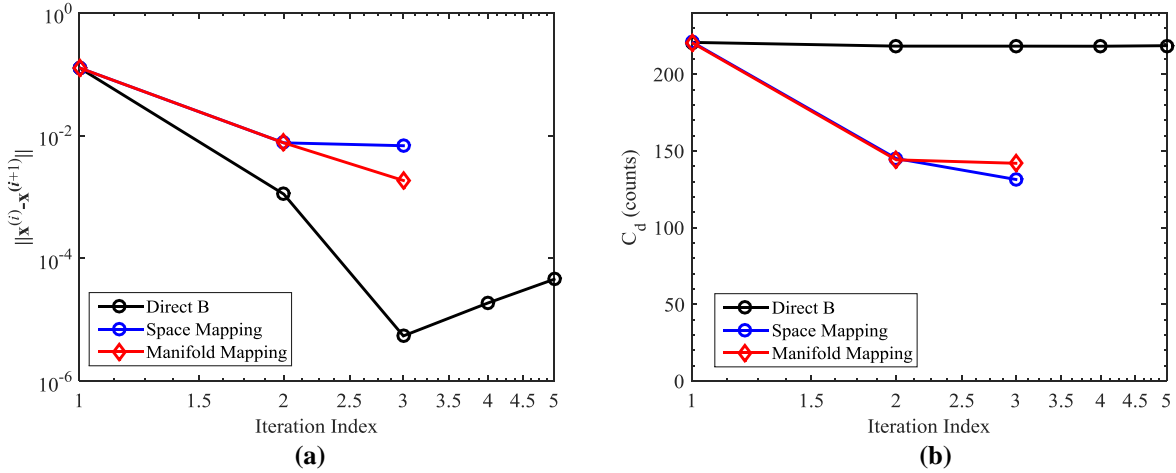


Figure 12. BC II convergence histories of the optimization runs.

Table 6. Optimization results for BC II.

Parameter/Method	Baseline	Direct B	SM	MM
C_l (l.c.)	82.35	82.41	82.39	82.36
C_d (d.c.)	220.28	218.60	131.41	141.96
$C_{m,c/4}$	-0.0905	-0.0925	-0.0917	-0.0904
A	0.0779	0.0782	0.0767	0.0767
N_c	—	—	263	210
N_f	—	54*	14	14
N_{tot}	—	54*	< 131	< 125
t_c (min)	—	—	952.9	924.6
t_f (min)	—	237.2	114.6	116.7
t_{tot} (min)	—	237.2	1,067.5	1,041.3

* Primary flow solutions as well as adjoint solutions.

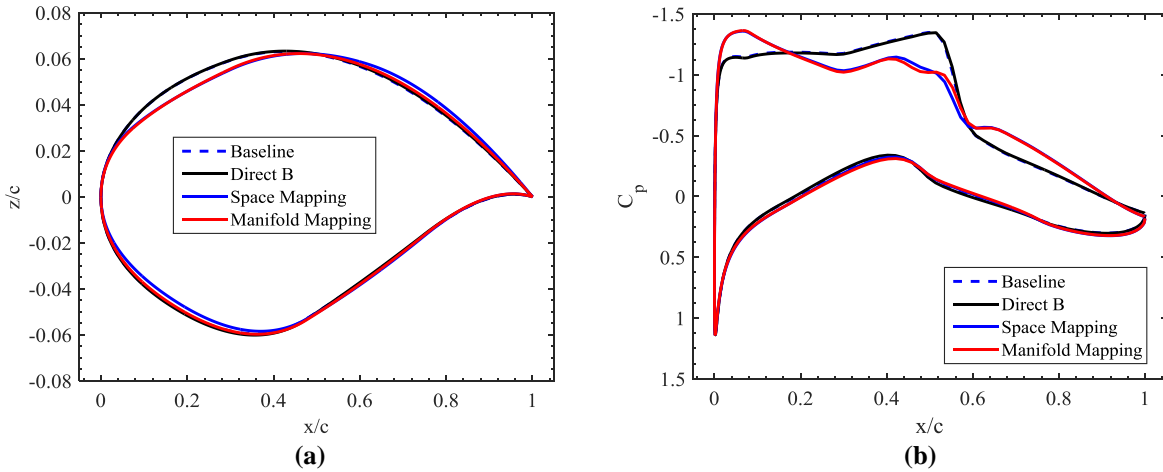


Figure 13. BC II baseline and optimized characteristics: (a) shapes, (b) pressure coefficient distributions.

Table 7. Grid convergence study for the MM optimized shape of BC II.

Mesh	Grid Size	C_l (cts)	C_d (cts)	Simulation Time* (min)
1	120 × 40 (8,196)	0.8239	184.08	6.8
2	240 × 80 (32,396)	0.8236	141.78	21.2
3	480 × 160 (128,796)	0.8235	133.85	55.1
4	960 × 320 (513,596)	0.8236	130.57	532.3
5	1,920 × 640 (2,051,196)	0.8239	134.48	2,880.1

* Computed on a high-performance cluster with 32 processors. Flow solution only.

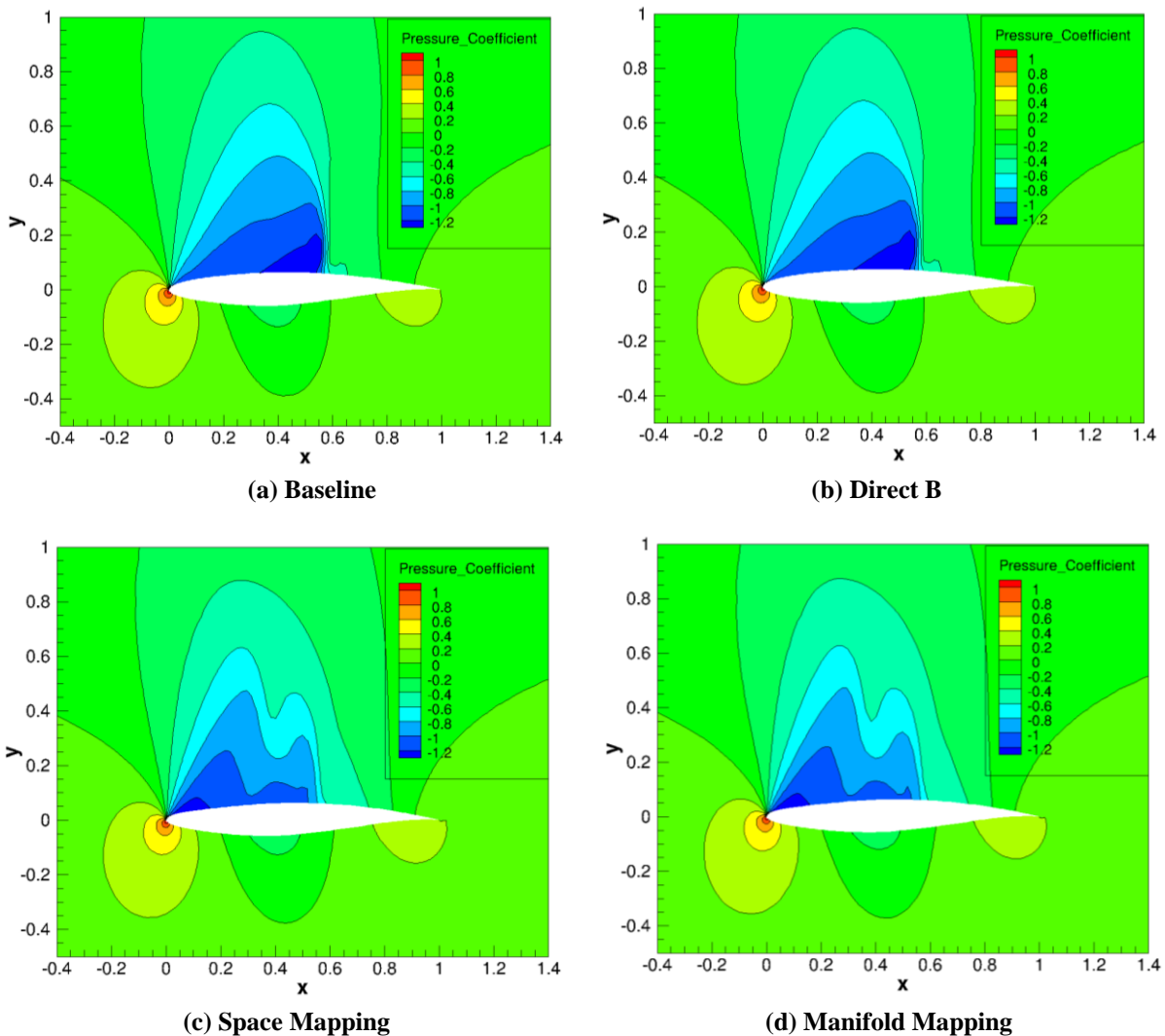


Figure 14. BC II pressure coefficient contours for baseline and optimized shapes.

V. Benchmark Case III: Optimum Twist Distribution of a Rectangular Wing for Minimum Induced Drag at Fixed Lift in Inviscid Flow

A. Problem Statement

The objective is to optimize the twist distribution of a rectangular wing to minimize the induced drag at fixed lift in inviscid flow. The goal is to verify that the optimization methods recover a lift distribution that is close to elliptical and a span efficiency factor approximately equal to one.

The wing planform geometry is given in Fig. 15. The semi-span planform is rectangular over the first 3 chord lengths. The airfoil shape is constant across this part of the wing given by the modified version of the NACA 0012 defined in (17). The last 0.06 chord units consist of a wing-tip cap, which is a surface of revolution based on the modified NACA 0012 airfoil. The initial geometry is untwisted, unswept, and has zero dihedral. The aspect ratio is $AR \equiv b^2/(S) = (6c)^2/(c \cdot 6c) = 6$, where b is the span, c is the chord length, and S is the planform area.

The optimization problem can be written as follows. At a constant Mach number $M = 0.5$, minimize the drag coefficient

$$\min_{1 \leq x \leq u} C_D \quad (23)$$

subject to the constraints

$$C_L = 0.375, \quad (24)$$

$$\left. \frac{d\gamma}{dy} \right|_{y=3c} \geq -10^\circ / c, \quad (25)$$

where \mathbf{x} is the design variable vector (representing the wing twist distribution, γ) with the upper and lower bounds \mathbf{u} and \mathbf{l} , respectively, C_D is the wing drag coefficient, and C_L is the wing lift coefficient.

B. Design Variables

The wing planform and profile shapes are held fixed, i.e., the airfoil profile shape is fixed across the wing, the wing sweep is fixed, the semi-span ($b/2$) and chord (c) lengths are fixed, as well as the dihedral. The design variable is the twist distribution, γ , of the wing. The twist at the root is fixed to zero and the twist of the remaining part of the wing is parameterized using a B-spline curve as shown in Fig. 16. The B-spline curve is defined using three control points. The control point at the root is fixed at the origin (0,0). The control point at the tip has a fixed span-wise location, $\eta \equiv y/(b/2) = 1$, but it can move in the vertical direction. The control point in the middle can move along the η direction and the γ -direction. The design variable vector is written as $\mathbf{x} = [\eta_2 \ \gamma_2 \ \gamma_3]^T$. The lower bound is set $\mathbf{l} = [0.0 \ -6.0^\circ \ -6.0^\circ]^T$, and upper bound is set $\mathbf{u} = [1.0 \ 6.0^\circ \ 6.0^\circ]^T$. The angle of attack, α , is used as a dummy parameter to fulfill the lift constraint (24).

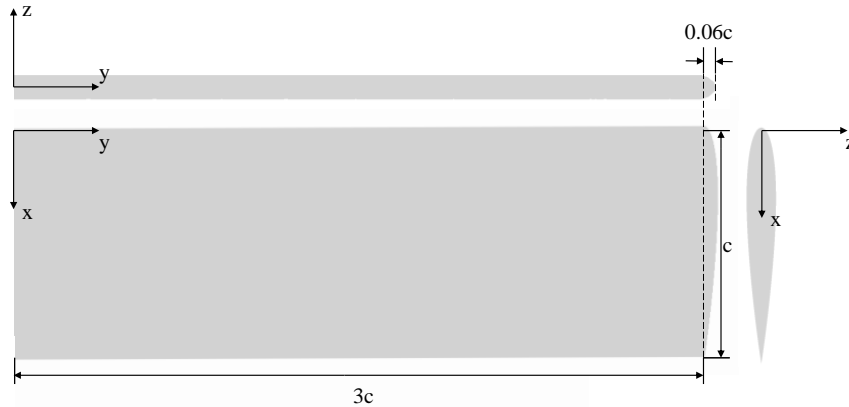


Figure 15. Baseline semi-span geometry for Benchmark Case III.

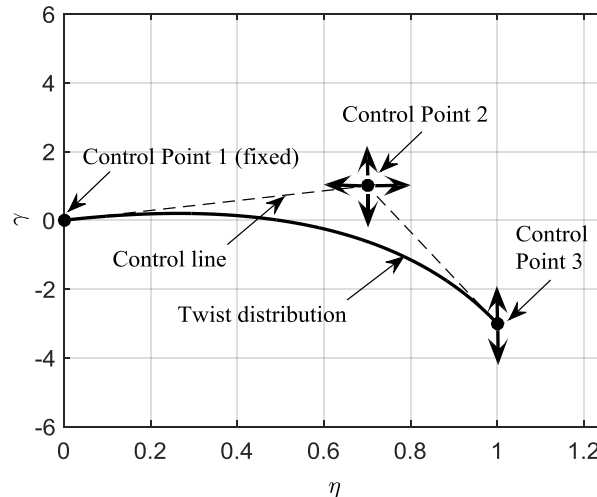


Figure 16. Twist distribution parameterization using a B-spline curve with three control points.

C. Inviscid Aerodynamics Model

The inviscid flow is simulated using the SU^2 solver and the setup is the same as for Benchmark Case I (see Sec. III.C). An unstructured grid is generated for this case using the surface modeler²² (SUMO) which first creates a surface grid and then uses the TetGen volume grid generator²³. The farfield boundary is set 20 chord lengths from the wing surface (Fig. 17(a)). The minimum off-wall spacing is set to $10^{-3}c$ and the grid nodes are clustered near the leading and trailing edges, as well as the wing tip (Fig. 17(b)). Table 8 shows three grids that were generated with number of elements ranging from approximately 240 thousand to 15.6 million with simulation times from 5.4 min to 1 hr and 54 min on a high-performance computing cluster with 32 processors.

D. Optimization Results

The optimization was carried out using the SAO algorithm described in Sec. II.E. The fine grid in Table 8 was used in the optimization studies. The optimization convergence history is given in Fig. 18 showing as a function of the design iterations: the convergence of the design variable argument (left), the patch size (center), and objective function (drag coefficient) evolution (right). The optimizer terminates based on the objective function value convergence criteria.

The baseline and optimized twist distributions are shown in Fig. 19(a). The corresponding section lift distributions are given in Fig. 19(b), as well as the elliptical lift distribution. Table 9 gives the numerical values at selected spanwise locations. The optimized wing has a section lift distribution close to the elliptical one. The drag coefficient is reduced from 87.9 counts to 86.8 counts, or by 1.1 counts (Table 10). The span efficiency factor e grows from 0.930 to 0.955, or by 2.7%. The optimizer needed 22 model evaluations taking around 42.5 hours in total on a high-performance cluster with 32 processors.

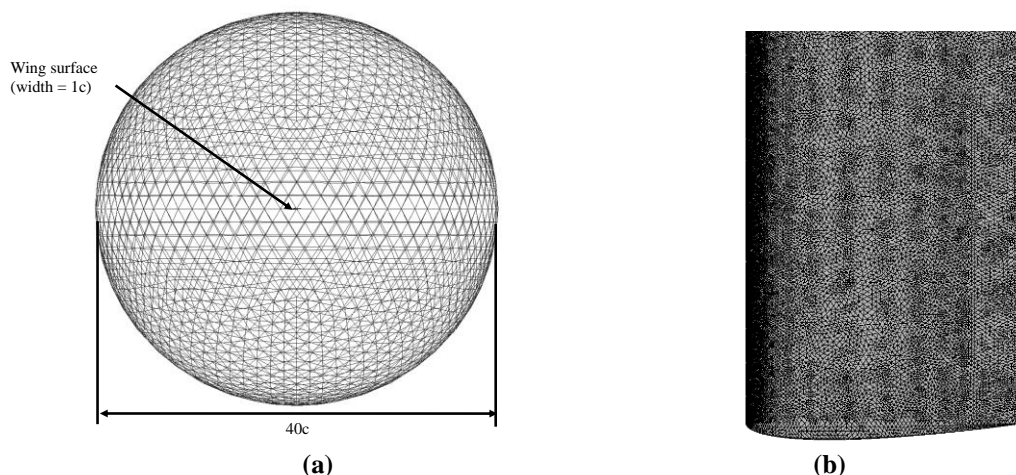


Figure 17. An example unstructured grid generated using SUMO for Benchmark Case III: (a) the farfield, (b) wing surface near the tip.

Table 8. Grid convergence study for the baseline shape of Benchmark Case III at $M = 0.5$.

Grid	Number of Elements	C_L (cts)	C_D (cts)	Simulation Time (hr)
Coarse	~243,000	0.375	150.7	0.0895
Medium	~1,946,000	0.375	95.04	0.3266
Fine	~15,569,000	0.375	87.93	1.8930

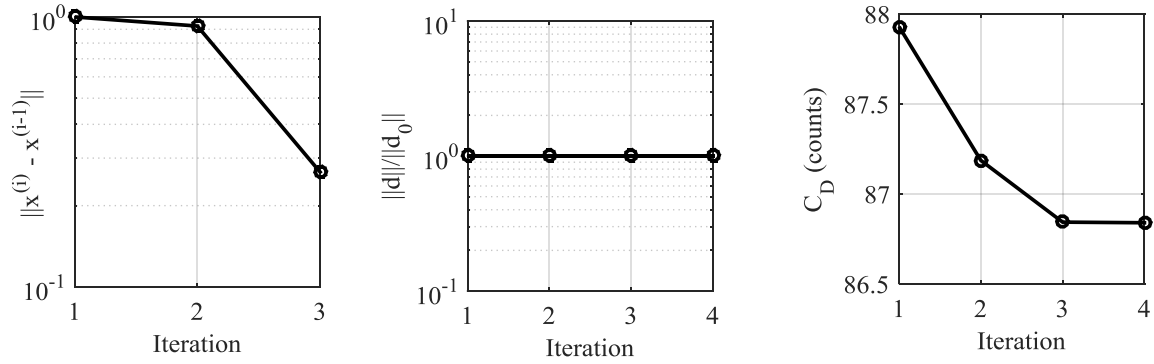


Figure 18. BC III optimization convergence history.

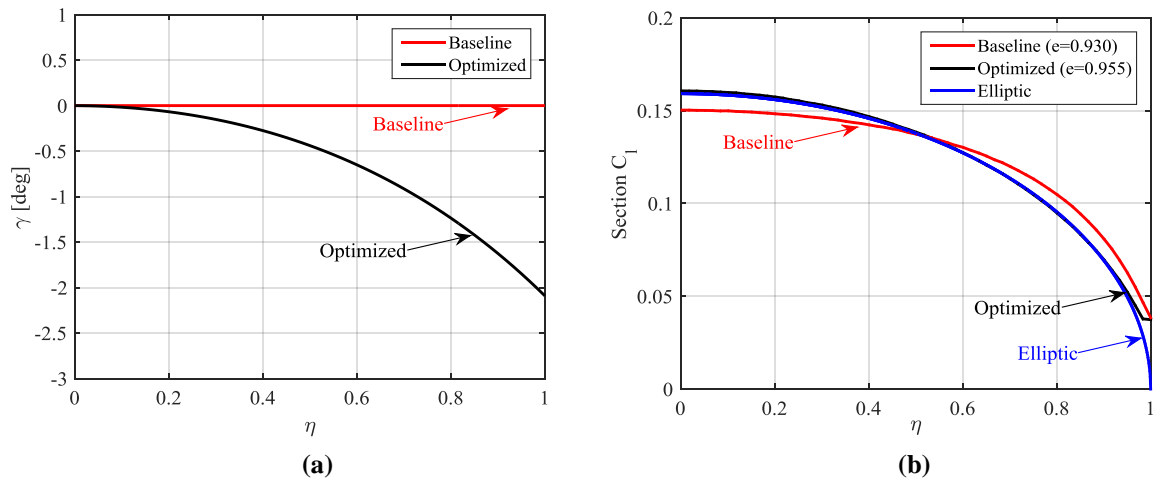


Figure 19. BC III baseline and optimized wing properties: (a) twist distributions, (b) section lift distributions.

Table 9. BC III baseline and optimized twist and lift distributions.

	η	0.0	0.2	0.4	0.6	0.8	0.9	0.95	1.0
Baseline	γ (deg)	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	C_l	0.1503	0.1483	0.1424	0.1303	0.1049	0.0811	0.0629	0.0384
Optimized	γ (deg)	0.0000	-0.0674	-0.2746	-0.6506	-1.2359	-1.6249	-1.8476	-2.0911
	C_l	0.1607	0.1574	0.1468	0.1274	0.0950	0.0699	0.0524	0.0374
Elliptical	C_l	0.1592	0.1559	0.1459	0.1273	0.0955	0.0694	0.0497	0.0000

Table 10. BC III baseline and optimized planform attributes.

Parameter	Baseline	Optimized
α (deg)	4.2988	4.8125
C_L (cts)	37.5004	37.4999
C_D (cts)	87.9308	86.8410
e	0.9302	0.9546
N_f	—	22
t_f (hr)	—	42.5

VI. Conclusion

Benchmark cases in aerodynamic design have been solved using multi-fidelity optimization algorithms and low-dimensional shape parameterizations (up to 8 design variables in the cases considered). Compared with direct gradient-based optimization algorithms equipped with adjoint sensitivities and trust regions, the multi-fidelity algorithms obtain similar optimized shapes but at a significantly lower optimization cost. However, since the multi-fidelity algorithms presented here use pattern search (a derivative-free local search algorithm) to optimize the surrogate model, the cost will grow quickly when increasing the number of design variables. In particular, when considering benchmark cases that require more than 10 design variables, such as the common research model wing, the cost of utilizing the multi-fidelity techniques presented here may be prohibitive. Future work will, therefore, investigate alternatives to efficiently optimize the surrogate model. In particular, we will investigate the extension of the multi-fidelity techniques using adjoint sensitivity information, as well as utilizing techniques to reduce the dimensionality of the design space.

References

- ¹Leifsson, L., Koziel, S., Tesfahunegn, Y.A., Hosder, S., and Gramanzini, J.-R., "Aerodynamic Design Optimization: Physics-based Surrogate Approaches for Airfoil and Wing Design," *AIAA 52nd Aerospace Sciences Meeting*, National Harbor, Maryland, January 13-17, 2014.
- ²Tesfahunegn, Y.A., Koziel, A., Gramanzini, J.-R., Hosder, S., Han, Z.-H., and Leifsson, L., "Direct and surrogate-based optimization of benchmark aerodynamic problems: A Comparative Study," *53rd AIAA Aerospace Sciences Meeting, Science and Technology Forum*, Kissimmee, Florida, Jan 5-9, 2015.
- ³Koziel, S., and Leifsson, L., "Knowledge-based airfoil shape optimization using space mapping," *AIAA Paper 2012-3016, 30th AIAA Applied Aerodynamics Conference*, New Orleans, Louisiana, June 25-28, 2012.
- ⁴FLUENT, ver. 14.5.7, ANSYS Inc., Southpointe, 275 Technology Drive, Canonsburg, PA 15317, 2013.
- ⁵Kinsey, D. W., and Barth, T. J., "Description of a Hyperbolic Grid Generation Procedure for Arbitrary Two-Dimensional Bodies," AFWAL TM 84-191-FIMM, 1984.
- ⁶Drela, M., and Giles, M.B., "Viscous-Inviscid Analysis of Transonic and Low Reynolds Number Airfoils," *AIAA Journal*, vol. 25, no. 10, Oct. 1987, pp. 1347-1355.
- ⁷Sobieczky, H., "Parametric airfoils and wings," Recent developments of aerodynamic design methodologies, Braunschweig/Wiesbaden: Friedr. Vieweg & Sohn Verlagsgesellschaft mbH, 1999, pp. 71-87.
- ⁸Koziel, S., and Leifsson, L., "Multi-Level Surrogate-Based Airfoil Shape Optimization," *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, Grapevine, Texas, January 7-10, 2013.
- ⁹Palacios, F., Colonno, M. R., Aranake, A. C., Campos, A., Copeland, S. R., Economou, T. D., Lonkar, A. K., Lukaczyk, T. W., Taylor, T. W. R., and Alonso, J. J., "Stanford University Unstructured (SU2): An open-source integrated computational environment for multi-physics simulation and design," *AIAA Paper 2013-0287, 51st AIAA Aerospace Sciences Meeting and Exhibit*, Grapevine, Texas, USA, 2013.
- ¹⁰Jameson, A., "Aerodynamic Design via Control Theory," *Journal of Scientific Computing*, Vol. 3, 1988, pp. 233-260.
- ¹¹Koziel, S., Echeverría-Ciaurri, D., and Leifsson, L., "Surrogate-based methods," in S. Koziel and X.S. Yang (Eds.) *Computational Optimization, Methods and Algorithms*, Series: Studies in Computational Intelligence, Springer-Verlag, pp. 33-60, 2011.
- ¹²A.I.J. Forrester and A.J. Keane, "Recent advances in surrogate-based optimization," *Prog. in Aerospace Sciences*, vol. 45, no. 1-3, pp. 50-79, Jan.-April, 2009.
- ¹³Pointwise V17.3R2, Pointwise Inc., 213 South Jennings Avenue, Fort Worth, Texas, 76104-1107, USA.
- ¹⁴Spalart, P. R. and Allmaras, S. R., "A One Equation Turbulence Model for Aerodynamic Flows," *AIAA-Paper-92-0439, 38th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, NV, January 6-9, 1992.
- ¹⁵Echeverría-Ciaurri, D., and Hemker, P.W., "Manifold Mapping: a Two-Level Optimization Technique," *Computing and Visualization in Science*, 11, pp. 193-206, 2008.
- ¹⁶Koziel, S., and Echeverría-Ciaurri, D., "Reliable Simulation-Driven Design Optimization of Microwave Structures Using Manifold Mapping," *Progress in Electromagnetic Research B (PIER B)*, 26, pp. 361-382, 2010.
- ¹⁷Ren, J., Leifsson, L., Koziel, S., and Tesfahunegn, Y.A., "Multi-fidelity Aerodynamic Shape Optimization Using Manifold Mapping," *54th AIAA Aerospace Sciences Meeting, Science and Technology Forum*, San Diego, CA, Jan 4-8, 2016.
- ¹⁸Farin G., *Curves and Surfaces for Computer Aided Geometric Design*, Boston, MA: Academic Press, 1993.
- ¹⁹Conn, A.R., Gould, N.I.M., and Toint, P.L., *Trust Region Methods*, MPS-SIAM Series on Optimization, 2000.
- ²⁰Leifsson, L. and Koziel, S., *Simulation-Driven Aerodynamic Design Using Variable-Fidelity Models*, Imperial College Press, London, UK, 2015.
- ²¹Koziel, S., "Multi-fidelity multi-grid design optimization of planar microwave structures with Sonnet," *International Review of Progress in Applied Computational Electromagnetics*, Tampere, Finland, 2010, pp. 719-724.
- ²²Surface Modeler (SUMO), Larosterna Engineering Dynamics, Vallentuna, Sweden.
- ²³Si, H., "TetGen, a Delaunay-Based Quality Tetrahedral Mesh Generator," *ACM Trans. on Mathematical Software*, 41 (2), Article 11, February 2015.

²⁴Toropov, V., Filatov, A., and Polynkin, A., "Multiparameter structural optimization using FEM and multipoint explicit approximations," *Structural Optimization*, Vol. 6, 1993, pp. 7-14.

²⁵Giunta, A. and Eldred, M., "Implementation of a trust region model management strategy in the DAKOTA optimization toolkit," *Proceedings of the AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Long Beach, CA, September 6-8, 2000.

²⁶Hosder, S., Watson, L., Grossman, B., Mason, W., and Kim, H., "Polynomial response surface approximations for the multidisciplinary design optimization of a high-speed civil transport," *Optimization and Engineering*, Vol. 2, 2001, pp. 431-452.

²⁷Jacobs, J., Etman, L., van Keulen, F., and Rooda, J., "Framework for sequential approximate optimization," *Structural and Multidisciplinary Optimization*, Vol. 27, 2004, pp. 384-400.

²⁸MATLAB V15.0, Mathworks Inc., Corporate Headquarters, 3 Apple Hill Drive, Natick, MA 01760-2098, USA.

²⁹Leifsson, L., Hermannsson, E., and Koziel, S., "Optimal shape design of multi-element trawl-doors," *Journal of Computational Science*, Vol. 10, September, 2015, pp. 55-62.

³⁰Jameson, A., Schmidt, W., and Turkel, E., "Numerical Solution of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes," AIAA 1981-1259, *AIAA 14th Fluid and Plasma Dynamic Conference*, Palo Alto, CA, June 23-25, 1981.