AIAA 2006-1820

Airfoil Design and Optimization Using Multi-Fidelity Analysis and Embedded Inverse Design

Thomas R. Barrett*, Neil W. Bressloff† and Andy J. Keane‡
Computational Engineering and Design Research Group, School of Engineering Sciences, University of Southampton, SO17 1BJ, U.K.

The increasing popularity of high fidelity computational fluid dynamics simulations in aerodynamic design has stimulated research into more efficient design optimization methods. A common contributor to this efficiency is a reduction in problem dimensions via the use of novel parameterization techniques. The focus of this paper is on the high fidelity aerodynamic design of airfoil shapes. A multi-fidelity design search method is presented which uses a parameterization of the airfoil pressure distribution followed by inverse design, giving a reduction in the number of design variables used in optimization. While an expensive analysis code is used in evaluating airfoil performance, computational cost is reduced by using a low-fidelity code in the inverse design process. This method is run side by side with a method which is considered to be a current benchmark in design optimization. The two methods are described in detail, and their relative performance is compared and discussed. The newly presented method is found to converge towards the optimum design significantly more quickly than the benchmark method, providing designs with greater performance for a given computational expense.

Nomenclature

x	=	airfoil ordinate in the horizontal (chordwise) direction
z	=	airfoil ordinate in the vertical direction
z''	=	airfoil surface second derivative with respect to x
C_p	=	airfoil pressure coefficient
C_p^T	=	target pressure coefficient
C_p^k	=	computed pressure coefficient at design iteration k
C_D	=	force coefficient in the x direction (drag coefficient)
C_L	=	force coefficient in the z direction (lift coefficient)
c	=	airfoil chord
α	=	airfoil angle of attack
t	=	airfoil maximum thickness
z_c	=	airfoil maximum mean thickness (camber)
r_{LE}	=	airfoil leading edge radius
a, b	=	constant coefficients of polynomial expressions
R	=	relaxation factor

I. Introduction

ADVANCES in computational fluid dynamics (CFD) simulations have lead to a significant increase in their use for aerodynamic design. Despite enhancements in computing power, the expense of high fidelity CFD analysis necessitates continued research into more efficient design optimization methods. Traditionally, *direct* CFD based design methods[1] involve the manipulation of a component's geometry based on the results of analysis, in order to improve a design metric. An inherent limitation associated with high-fidelity aerodynamic shape optimization performed in this way is the requirement for a large number of design variables in order to define, in sufficient

* Graduate Research Student, School of Engineering Sciences, University of Southampton. Member AIAA.
† Senior Research Fellow, School of Engineering Sciences, University of Southampton.
‡ Professor of Computational Engineering, School of Engineering Sciences, University of Southampton.

detail, the geometry of the component being studied. The resulting process can be very expensive computationally. As well as designing a component via the relationship between its geometry and its overall performance, it is also possible to design by matching flow features with a set specified by the designer.[2,3] These so-called *inverse* design methods have been used widely, particularly in the context of designing airfoils to prescribed surface pressure distributions. Inverse methods require that the flow features of the intended design are specified *a priori*; traditionally this would be the task of an experienced aerodynamicist, but this specification can also be the result of an optimization on these flow features.[3] Inverse methods have proven valuable because once the target pressure distribution is specified the required airfoil geometry can be obtained with very few CFD evaluations, compared to a direct optimization on the geometry.[2,3] Of course, the inverse process always requires a construction of the geometry in order to perform CFD analysis, but knowledge of the target surface pressure allows rapid convergence towards the final design.

The method introduced here does not approach inverse design from this traditional perspective. Rather, the inverse process is used as a tool within a more complex multi-fidelity search and is called upon whenever an airfoil geometry is required whose flow features match those specified by a parametric model, i.e., the design search method uses a parameterization of the flow features coupled with inverse design. This is compared with a direct method which focuses purely on the geometry of a component and the resulting lift and drag. The motivation for investigating this more involved strategy is the desire to reduce the number of design variables used in the design search process. Thus, development using high-fidelity CFD analysis is accelerated without requiring the specification of the target pressure distribution *ab initio*, which is the hallmark of classical inverse design and one of its principal drawbacks. This alternative approach is based on the observation that the pressure distribution resulting from an airfoil shape can often be simpler to define than its geometry, i.e., the flow features of a component can potentially be described by fewer design variables compared to the geometry, independent of simulation fidelity. This reduction in problem dimensions means that fewer CFD evaluations are required in the design search process, making it less computationally expensive to achieve a given level of performance. The pressure distribution is an intuitive choice of flow feature since small perturbations in pressure can often require large variations in the entire geometry, for a given set of flow conditions.[3] The authors of related work[4-7] use various means of optimizing the target pressure distribution before using a single inverse design step to recover the corresponding geometry.

Here, the aerodynamic component being designed is a two-dimensional airfoil section in subsonic flow, and comparisons are made between two design methods. The first is treated as a benchmark in aerodynamic shape optimization. In this, the geometry model is defined by 13 design variables and the design is evaluated using high fidelity CFD software to give a measure of performance (drag at a fixed value of lift). An efficient design search and optimization (DSO) method based on response surfaces is used to maximize this performance measure by varying the design variables. The alternative method adopted uses a parameterization of the pressure coefficient (C_p) profile for the airfoil section using six parameters, i.e., considerably fewer than can be used for the direct optimization method. For each variation in the design variables of this model, an airfoil geometry which produces this prescribed pressure distribution must be found by inverse design. Crucially, a much less expensive CFD solver is used during the inverse design process. Once the resulting geometry is determined its performance is analyzed using the high fidelity CFD software, adopting the same performance metric as in the benchmark method. The efficient DSO method is again employed to maximize the performance, but now by manipulating the variables of the C_p profile.

Since it uses more design variables, the benchmark design method requires a relatively large number of full CFD evaluations for sufficient coverage of the design space, but acts directly on the geometry. The alternative method calls for fewer full CFD evaluations, but for each variation of the design variables an inverse design step using low fidelity CFD is required to first recover the geometry. This work investigates the relative computational expense of these two methods in reaching a similar level of design performance.

II. CFD Solver Setup

In the current work, the high-fidelity solver used is FLUENT®[8] and the low-fidelity code (used in the newly proposed method) is the full potential flow solver VGK.[9] Calibration of the CFD solvers requires a set of accepted experimental results for a standard airfoil whose geometry can be reproduced. In the current study a subsonic flow regime is used, as this simplifies the pressure distribution to be parameterized in the alternative method. The NASA low-speed (LS) airfoil family[10,11] provides a suitable baseline for drawing comparisons with the results of a subsonic design study. These airfoils are designed to operate at a lift coefficient of 0.4 and typically a flow speed Mach number of 0.15 and so these conditions are used in the present design studies. Ref. 11 gives detailed experimental surface pressure results for the NASA LS(1)-0013 airfoil. This data has been collated such that comparisons can be

made with results from FLUENT and VGK for the same airfoil (Figure 1). Because the flow solvers are validated with the flow conditions specified in Refs. 10 and 11, these conditions are used in calculating the design objective for the two design methods; this also allows the resulting airfoil designs to be compared directly with the NASA low-speed airfoils.

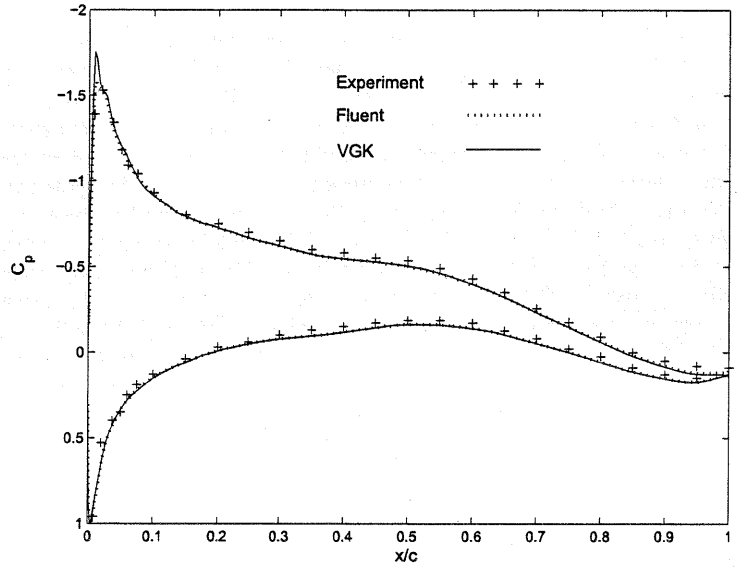

Fig. 1 Comparison of pressure distributions generated using the FLUENT and VGK CFD solvers, for the NASA LS(1)-0013 airfoil. This is compared against experimental data.

A. FLUENT

The FLUENT model is set up with the aim of minimizing the computational effort required for the analysis, giving robust convergence, while providing accurate results in close agreement to the experimental data as shown in Figure 1. The FLUENT geometry generation and meshing tool, GAMBIT®, is used to mesh the flow domain to be solved by FLUENT. The extent of the flow domain is a square with 40 metre edges, with the airfoil normalized to a chord of 1 metre. A structured boundary layer mesh of 20 cells thick is attached to the airfoil surface, with the first cell given a thickness of 0.045% airfoil chord and the cells growing in size towards the free stream. The remaining flow domain consists of an unstructured triangular mesh, with a size function applied to the airfoil to grow the cell size with distance from the surface, thus reducing the computational expense of the analysis. Typically, the full mesh consists of 17,500 mesh elements, including the structured boundary layer. In this configuration an increase in the mesh size gives a negligible variation in the resulting force coefficients, providing good numerical accuracy.

In the CFD model described here, the equations of momentum and viscosity are solved in a coupled manner and the Spalart-Allmaras turbulence model[12] is employed, as this is known to be a relatively accurate method for external flow over an airfoil, giving robust convergence. On studying the convergence history of the solver using this setup, it is observed that 2000 iterations of the RANS calculations are sufficient to provide a converged solution. At this point the variation in the drag coefficient for the airfoil is within ±0.1 counts (±0.00001 C_D) of the fully converged value. For the converged solution the non-dimensional distance, $y+$, over the airfoil surface lies in the range suitable for a log-law wall function representation of the boundary layer, i.e., between 30 and 60.

Figure 1 shows the FLUENT results for the NASA LS(1)-0013 airfoil using the above setup and for the flow conditions specified in Ref. 11, i.e., a flow speed Mach number of 0.15 and Reynolds number of 4×10^6. Figure 1 confirms the strong similarity between the pressure profiles from FLUENT and the experimental data.

For each airfoil design analyzed using FLUENT, the desired performance metric is drag at a fixed value of lift. This is determined by running the analysis at a number of values of angle of attack until the desired lift is achieved. In practice, the analysis is run at two initial angles; following this the correct angle is calculated from the lift curve, providing a converged solution at the desired value of lift to within ±1%. Calculating the drag using the current CFD

set up takes on average 21.4 minutes when running on a Xeon 2.8GHz compute node with 2Gb memory, and this process is used to calculate the airfoil performance metric in both design methods under consideration.

B. VGK

The method newly presented in the current work requires a computationally inexpensive CFD solver to compute the airfoil pressure distributions during the inverse design step. The low-fidelity software used here is VGK, written by DRA Farnborough and distributed by the Engineering Sciences Data Unit (ESDU). VGK is a 2-dimensional viscous coupled finite difference code which solves the full potential equations, written specifically for the analysis of airfoils. The airfoil geometry is input as a matrix of co-ordinates, following this a computational mesh is built in the flow domain using a series of radial and circumferential grid lines. The full potential equations are solved iteratively over the grid using a finite-difference approach. Typically, each airfoil analysis takes around 2 seconds on a Xeon 2.8GHz compute node with 2Gb memory. The VGK CFD model is set up as a viscous solve with the same Reynolds number and flow speed as used when running FLUENT. The boundary layer transition point is specified to be at 3% airfoil chord. As with the FLUENT CFD model, the solver was run for the standard NASA LS(1)-0013 airfoil for comparison with the experimental data; this is also shown in Figure 1. With the exception of the pressure *spike* at the leading edge, the VGK method gives accurate pressure distribution results compared to the experimental data.

III. Design Strategies

A. Direct optimization

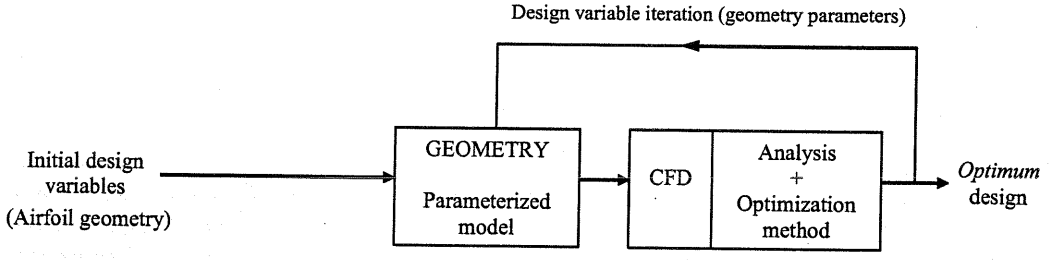

Fig. 2 Benchmark *direct* aerodynamic design optimization method.

A direct design optimization method is used here to act as the benchmark in aerodynamic shape optimization, against which the alternative method is compared. In the benchmark method (Figure 2) the airfoil geometry is parameterized and design iterations, requiring high fidelity CFD analysis, are automated using an optimization algorithm. Ultimately, one hopes to arrive at an *optimum* design, but in reality when there are more than two or three variables this becomes an exercise in design improvement.

The choice of parameterization method is a critical factor in the performance of direct searches such as the benchmark design method used here. This has been the subject of extensive research,[13] seeking for representations which reduce the number of design variables while retaining the ability to capture a global range of designs. Commonly used in aerospace design are computer aided design (CAD) based B-spline or Bézier curves as well as analytical approaches.[14,15] Song and Keane[16] compared an interpolating B-spline method with an analytical based method, and reported that while the spline approach is computationally expensive it is able to capture a larger range of geometries accurately. An interpolating segmented cubic polynomial spline is chosen here to parameterize the airfoil for the direct search method.

The airfoil shape is defined by ten spline segments or eleven control points in (x,z) space, which are interpolated by the parametric curve (Figure 3). The design variables are selected from the possible x and z movements of the control points, with the aim of minimizing the number of design variables while retaining the ability to produce smooth and varied airfoil shapes. The polynomial is cubic in form, except on the leading edge segments where the terms with x of order two and three have been replaced with a $\sqrt{x}$ term, a technique used in the NACA airfoil definition.[17] The airfoil is separated into upper and lower surfaces, while the leading edge point $(0,0)$ is shared by

both surfaces and remains fixed. Of the remaining five points on the upper surface, the near leading edge point (point A in Figure 3) is free to move in both x and z directions and the trailing edge point (B) is fixed, while the other three points are constrained to movement in the z direction only. The same applies to the lower surface, with the exception of the point adjacent to the trailing edge point (C), which is free to move in both directions. The trailing edge thickness is fixed at the value corresponding to the initial design in the search process. Additionally, the gradient of each surface at the trailing edge is added to the list of design variables, since the exit angle is important in the governing aerodynamics. Thus, there are 13 design variables in total defining the geometry of the airfoil, which must be searched by the optimizer and analyzed using CFD software. It is thought that 13 is a reasonable number of variables for an airfoil design problem of this nature; it is not uncommon for such a problem to make use of 22 or more variables.[16, 18, 19]

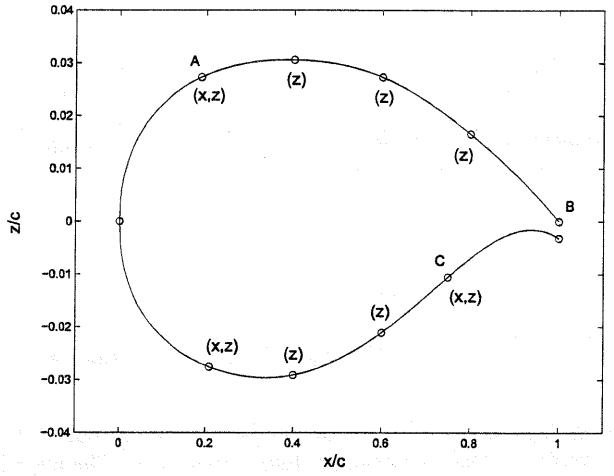

Fig. 3 Airfoil geometry representation for the direct optimization method, showing control point degrees of freedom.

Using the high-fidelity CFD solver (FLUENT), at each airfoil design iteration the lift and drag coefficients (C_L and C_D) can be calculated, which are then used as metrics of performance. As already noted, the design objective of the optimization procedure is to minimize C_D calculated at a constant value of C_L, allowing the angle of attack to float. The optimization strategy in the benchmark design method uses a response surface model (RSM) approach. For optimization using high-fidelity CFD, it is imperative to minimize the number of objective function calls. The RSM method is therefore a popular choice for global optimization using expensive functions[20,21] since it can be used to predict promising areas of the design space with relatively few objective function calls, compared to other design search methods. Further savings in overall time can be made by constructing the RSM with simultaneous calls to the objective function, something that is not always possible in other approaches to design optimization. Initial design iterations are generated as dictated by a formal *design of experiments*[1] (DoE) to populate the designated search space, and the objective function is calculated for each point. A RSM is built based on the initial DoE, and searched in order to predict areas where improved designs may be found. The objective function is evaluated at these points and the surface is updated. The process of building, searching and updating the surface is repeated, providing convergence towards an optimum in the design space. The RSM optimization routine for the current method is implemented using the OPTIONS design exploration system,[22] operating in the MATLAB®[23] environment using the GEODISE toolkit.[24]

The DoE used to seed the initial data-base is a Latin hypercube, which has good coverage of the design space and has the advantage of representing each variable's range equally.[25] Additionally, OPTIONS allows the random number sequence to be changed giving different, but repeatable, initial DoE sets. The response surface model used is an interpolating cubic spline radial basis function (RBF). Once the RSM has been built, it is searched using a genetic algorithm (GA), implemented in OPTIONS. The GA gives a relatively thorough search of the whole design space, which can be tolerated since calls to the response surface are very fast; here 5000 search evaluations are used. Instead of searching for a single optimum point on the surface, five parallel update points are extracted from the search. After evaluating the new design points, the RSM is updated.

The starting airfoil is the first point in the design search, and should be the same for both the benchmark method and the alternative method described below. For the results described in this paper, the starting shape is the NASA LS(1)-0413 from Ref. 10, which is designed to perform well at the selected operating conditions. The capability of the design methods in evolving an existing design is therefore demonstrated.

B. Embedded Multi-Fidelity Inverse Design (EMFID)

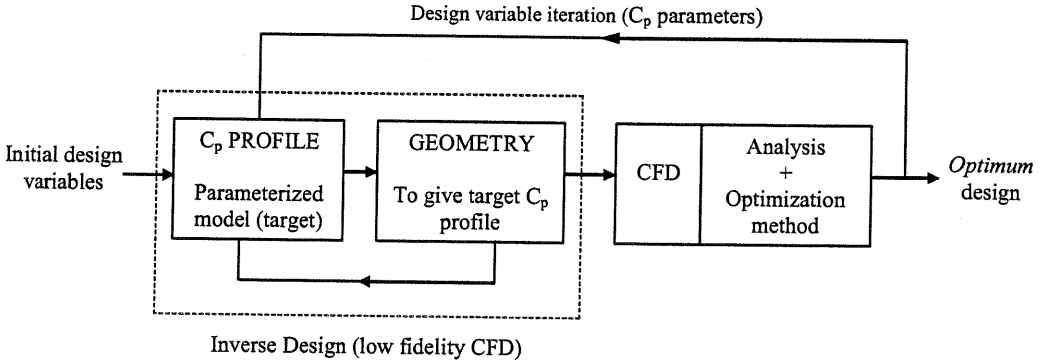

Fig. 4 Alternative *inverse* design method based on parameterizing the C_p profile (EMFID).

Increasingly, multi-fidelity approaches to design optimization are being used both to improve the reliability of the analysis and reduce the computational expense of a design search.[1,26,27] The alternative method described in the current work is illustrated in Figure 4. Instead of directly parameterizing the airfoil geometry to generate design iterations, the pressure distribution over the airfoil surface is parameterized. The geometry which produces this target pressure profile is then recovered using an inverse design method; following this the airfoil performance is calculated via the high fidelity CFD software. The optimization algorithm is used to maximize this performance by manipulating the design variables defining the pressure distribution. While high-fidelity aerodynamic optimization is desired and hence an expensive solver is used to evaluate the airfoil performance, the key to the effectiveness of this method is the use of a lower-fidelity CFD solver for the inverse design steps; thus we have a multi-fidelity search procedure. In using rather few design variables to describe the pressure distribution, the number of calls to the full CFD solver required to populate the design space is reduced considerably over the benchmark optimization method. However, for each call to the objective function an inverse design step must be performed. The effectiveness of this method relies on the saving made in reducing the number of CFD evaluations being greater than the relative cost of the inverse design steps. The configuration of the EMFID method is now described in full, starting with the parameterization of the airfoil pressure distribution.

1. Parameterization technique

The representation of the pressure (or velocity) distribution for an airfoil has been attempted by various authors using different approaches.[3,4,7,28] In most cases these methods use in excess of 12 design variables, partly owing to their ability to represent transonic as well as subsonic flow regimes. Here, the focus is on subsonic flow regimes only, which simplifies the type of pressure distribution to be represented. In order to make comparisons between the EMFID method and the benchmark optimization, the same initial design must be used. This means that the pressure distribution for the NASA LS(1)-0413 airfoil under the chosen design conditions must be capable of being represented reasonably well by the parameterization, with the corresponding parameter values being used to start the DSO process.

With respect to the parameterization method, the objective is to allow the generation of a wide range of realistic subsonic C_p distributions, while also limiting the dimensionality of this model. In this work, the C_p distribution is parameterized using a B-spline curve for each airfoil surface, extending from 1.5% chord to 100% chord. Each B-spline curve contains four knots and four control points. These are constructed on a knot vector of four zeros and four ones, giving a cubic Bézier curve. The control point locations are determined by specifying that the curve must

interpolate four data points. The chordwise positions of these data points are $x/c=\{0.015, 0.6, 0.85, 1\}$. The height of the trailing edge point is fixed while the heights of the three remaining points are the profile design variables, giving a total of six variables for the complete C_p distribution. The first 1.5% of this target C_p profile is used to represent the decay from stagnation pressure. On each surface, a straight line is constructed from C_p=1 to the start of the B-spline curve at 1.5% chord.

Using this parameterization technique, the EMFID design search involves manipulation of the shape of the C_p distribution, and since the integral of surface pressure is lift this results in a different target lift depending on the variable values. In the current work, a direct coupling is maintained between the low-fidelity and high-fidelity analysis of each airfoil design. In other words, the inverse design (using VGK) and the performance calculation (using FLUENT) are performed for identical flow conditions and seeking to achieve the same target lift. To ensure that each target pressure distribution results in the correct target lift, the pressure coefficient values are scaled to give the required enclosed area.

As noted above, the two design search methods evaluated in this work are started from a parameterization of the same initial design, NASA LS(1)-0413. Figures 5 and 6 compare the C_p distribution and geometry for each of these parameterizations with the original NASA shape. The original NASA geometry is shown as a thick solid line. The parameterization used in the benchmark method is shown as a thin dashed line, and this corresponds closely with the shape and C_p profile of the original airfoil (to the extent that its geometry is hidden in Figure 6). The EMFID parameterized representation of the NASA 0413 C_p distribution is shown as a dotted line in Figure 5. This profile was found by running a search on the six pressure profile variables in order to minimize the error with respect to the NASA C_p profile. The resulting target profile is close to that of the NASA airfoil. The airfoil geometry which realizes this parameterized target is found by running the inverse design process (detailed below); this geometry and its C_p distribution are shown as a thin solid line in Figure 5 and 6. It is seen that the disparity between the NASA 0413 pressure profile and the parameterized target has resulted in an airfoil design with a slightly different geometry. This highlights an apparent limitation of the EMFID method using the current parametric model. A model could be configured such that the initial design is replicated more accurately, however such a model would involve more variables, defeating the object of the EMFID process. Moreover, it turns out that the current parameterization is sufficiently flexible for the design of a general subsonic airfoil. The discrepancy between the geometries shown in Figure 6 is thought to be small enough to not disadvantage either method.

Figure 5 shows that the inverse design produces a C_p profile which closely agrees with the parameterized target. Of course, it is not immediately apparent how close one must get to the target profile on each inverse design step, in order to provide a meaningful design search process. However, experience with this process has shown that the target must be sufficiently well reproduced, such that small changes made to the target C_p are represented appropriately in the resulting airfoil shape.

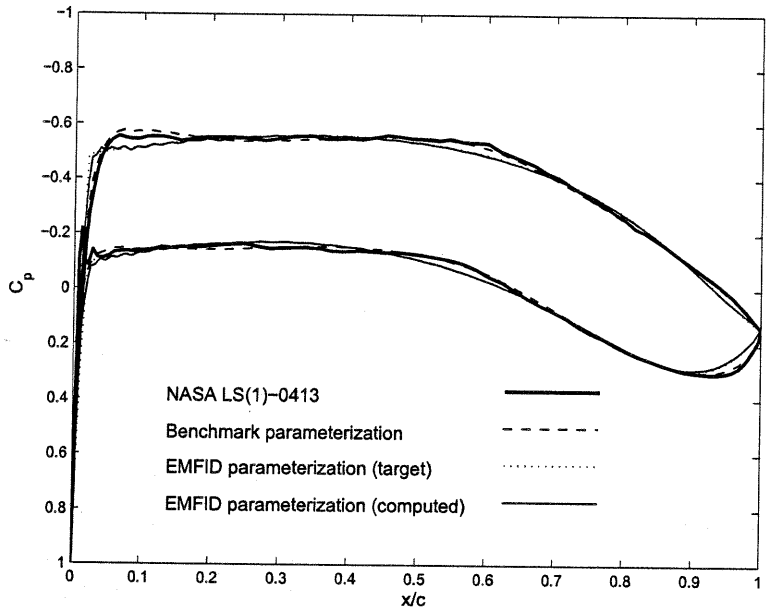

Fig. 5 Pressure distributions for the NASA LS(1)-0413 airfoil and for the representations of this shape using the parameterizations used by the two methods.

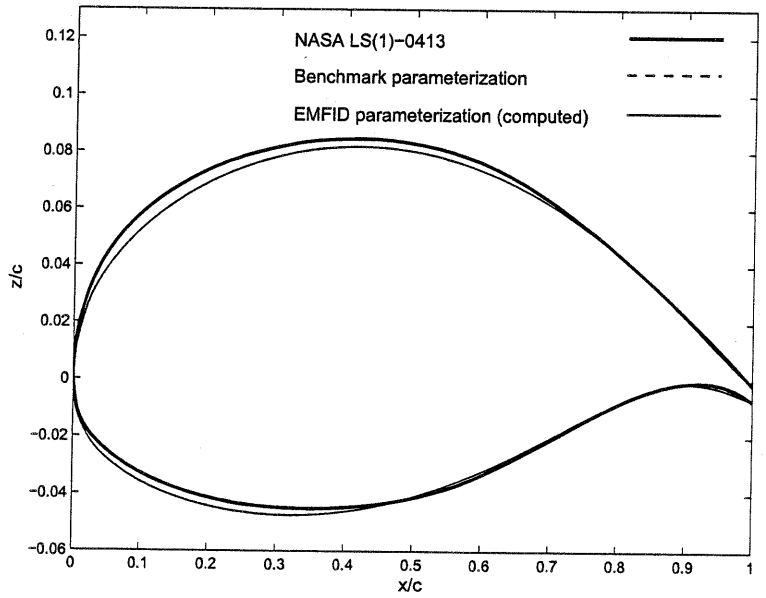

Fig. 6 The NASA LS(1)-0413 airfoil, and geometrical representations of this shape using the parameterizations used by the two methods.

For each iteration of the target pressure distribution generated by the optimization procedure (Figure 4), a geometry which realizes this target at the chosen flow conditions must be found by inverse design; this is the subject of the next section. Following the inverse design step, the airfoil geometry is analyzed using the same high-fidelity CFD as used for the benchmark method. There are therefore three forms taken by the airfoil pressure distribution during an objective calculation in EMFID: the parameterized target, the profile which most closely matched this

target during inverse design (VGK), and the profile generated in the final performance calculation (FLUENT). Figure 7 shows these three forms for the initial NASA 0413 design. Since the flow conditions and target lift are the same in all three cases, the pressure distributions are the same, confirming the direct coupling between the results of the analysis codes. It is not immediately apparent how the EMFID method would perform if the design conditions in each analysis were different; this subject is to be investigated by the authors.

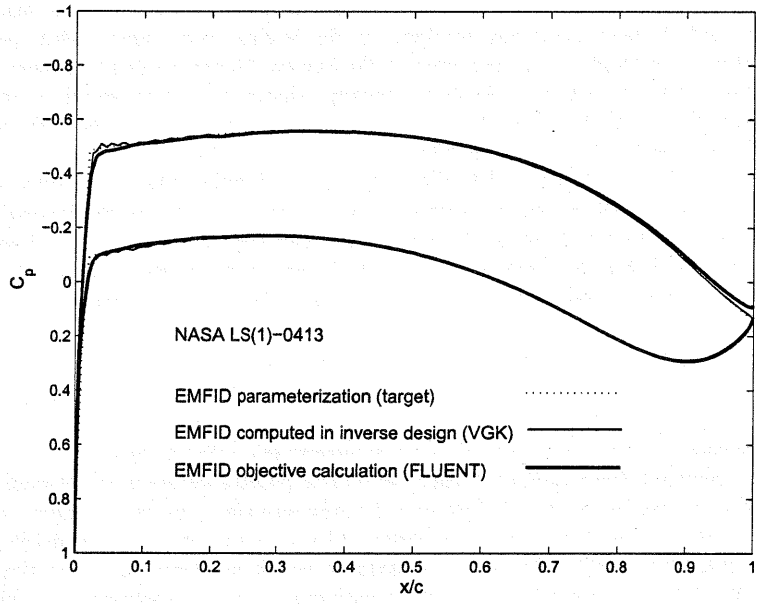

Fig. 7 Pressure distributions generated in an objective calculation in EMFID, for the initial NASA LS(1)-0413 airfoil.

2. Inverse Design

The EMFID method requires an inverse design process to derive an airfoil geometry which produces the specified target pressure distribution, or at least a reasonable approximation to it, at each iteration of the design search. Here, the full potential code VGK is used to calculate the pressure distributions for this inverse process. It must be able to accurately converge onto the target pressure distribution while minimizing the number of CFD calls.

A significant research effort has been put into inverse design methods, much of which focuses on surface flow design, i.e., matching a prescribed surface pressure distribution.[2,29,30] The inverse design method used here is similar to that used by Davis.[31] The method is based on an iterative residual-correction concept. For a given chordwise station, the residual is the difference between the target and computed pressure distributions. At the k^{th} iteration the second derivatives of the airfoil shape are corrected according to the following expression:

$$z''_k = z''_{k-1} + \left(\frac{dz''}{dC_p}\right)_{approx} (C_p^T - C_p^{k-1}) \cdot \qquad (1)$$

For each chordwise ordinate, C_p^T is the target pressure coefficient and C_p^{k-1} is the design pressure coefficient computed by a flow solver at iteration *k-1*. The gradient (dz''/dC_p) is calculated using approximate flow formulae, which can be relatively crude given the iterative nature of the design process. The flow formulation adopted here is the small disturbance theory used by Davis.[31] Thus, the inverse design process proceeds as follows: 1) the pressure distribution is calculated for an initial airfoil using the potential flow solver VGK, and the difference to the target pressure distribution is calculated; 2) an approximation to the pressure distribution is calculated using small disturbance theory, allowing the gradient (dz''/dC_p) to be determined; 3) the surface second derivatives are

corrected using Eq. (1); 4) integration yields the corresponding airfoil surface. The process is repeated until convergence is reached.

The inverse design procedure requires calculation of surface second derivatives, and these must also be integrated twice once they are corrected. The procedure used here is a central differencing scheme for differentiation, and following correction of the surface derivatives using Eq. (1) a simple 1st order interpolation quadrature is employed where the specified boundary conditions are the positions of the leading edge and trailing edge. A number of authors have described problems in the leading edge region when performing the inverse design,[32] largely caused by the high surface curvature in this region. The present authors have also encountered such a problem, the surface correction scheme is found to converge significantly more slowly in the leading edge region compared to the rest of the airfoil. However, the convergence can be greatly improved if the density of the defining co-ordinates is sufficiently high near the leading edge.

The objective of the design strategies described here is to minimize drag for a fixed quantity of lift while allowing the angle of attack, α, to be varied: α therefore becomes a variable in the inverse design process. As a consequence, the process is also given increased capability in matching a given target C_p distribution. Increasing α gives a monotonic increase in lift, and since this increases the area between the C_p curves for the upper and lower surfaces, α can be adjusted at each inverse design iteration using an expression similar to Eq. (1):

$$\alpha_k = \alpha_{k-1} + R\sum_{S}(C_p^T - C_p^{k-1}), \qquad (2)$$

where α_k denotes the angle of attack at iteration k and S represents the airfoil surface.

In Eq. (2), R is a relaxation factor applied at each iteration to prevent excessive corrections to the angle. A value for R of 0.2° is used in the current study. The use of relaxation provides a facility to control the convergence of an iterative procedure, a bigger factor gives faster convergence but increases the risk of instability. The inverse design approach requires a relaxation factor because of its iterative nature and because of the simplicity of the surface pressure approximation. The use of such factors is not uncommon in design optimization,[1] although the appropriate magnitude is likely to depend on the problem set up. A relaxation factor is also applied to the pressure residual term in Eq. (1), since experience with this inverse design method revealed that the surface corrections at each iteration can be overly large. Relaxation is used in Ref. 31, although its magnitude is not specified; a factor of 0.4 is employed here.

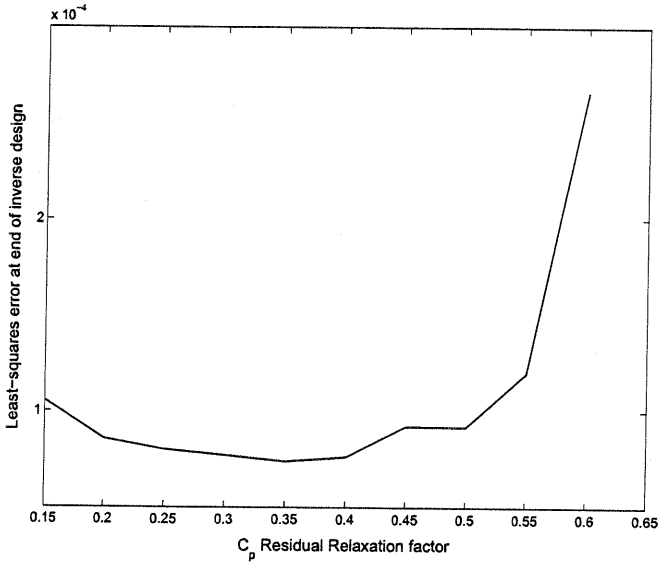

Fig. 8 The error at the end of the inverse design process, plotted against relaxation factor magnitude.

The least-squares error between the target and computed profile is used as a measure of the convergence of the inverse design process. This error diminishes rapidly as the iterative procedure converges onto the desired pressure profile. Eventually, the computed pressure profile is unable to match the target any closer, and the error increases

fractionally; at this point the process is deemed to be converged and is halted. The relaxation factor magnitudes specified here have been found by monitoring the number of iterations required for convergence and the minimum least-square error, as the relaxation factor is varied. Figures 8 and 9 illustrate the process of selecting the pressure residual relaxation factor used in Eq. (1). While Figure 8 shows that a factor of around 0.35 gives a slightly smaller error than 0.4, Figure 9 shows that a factor of 0.4 requires fewer iterations (20 compared to 25). Experience has shown that the form of these graphs is largely independent of the target pressure distribution.

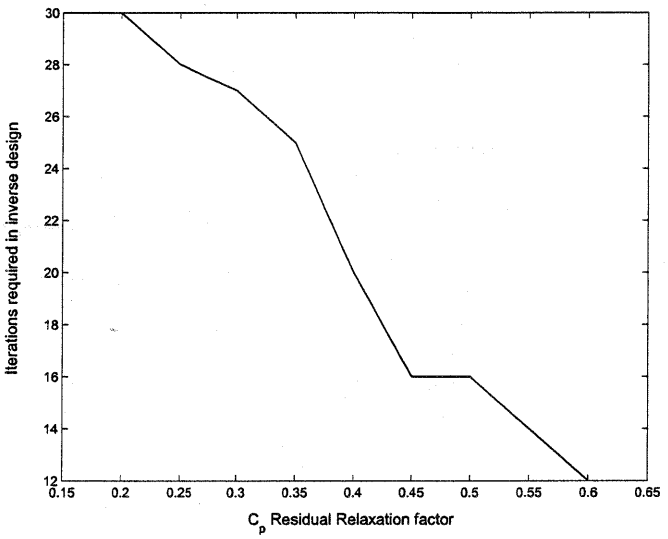

Fig. 9 The number of iterations required in the inverse design process, plotted against relaxation factor magnitude.

Using the method described above, the inverse design step uses approximately 20 VGK calls to capture the target pressure profile. However, it may require as few as five VGK calls or as many as 30 for certain target profiles, particularly when matching features at the leading edge. If the convergence criterion is not met after 40 inverse design iterations the process is stopped. If, during the procedure, a geometry is passed to VGK which causes the calculations to fail (for example, if the solution diverges), the geometry passed to the high fidelity CFD is the last one for which VGK gave a converged solution.

3. *EMFID search setup*

In order to make fair comparisons between the alternative method and the benchmark direct search method described above, the optimization algorithm used and its implementation are set the same for both methods. Therefore, the RSM approach described for the benchmark strategy is also used as the optimization method shown in Figure 4.

Figure 10 shows a more detailed diagrammatic description of the EMFID design search process. The optimization strategy (Figure 10a) is the same for both the benchmark method and EMFID. In summary, the process calculates the objective function as dictated by a random DoE, before building a response surface in order predict promising update points. Update points are evaluated until the computational budget is exhausted. A call to the objective function in EMFID (Figure 10b) takes the design variables, calculates the corresponding pressure distribution, and then scales it to give the required total area and target lift. The inverse design code (low fidelity CFD) is then used to calculate an airfoil geometry which realizes the specified pressure variation. Provided that it satisfies the geometrical constraints (see below), this shape is passed to the high fidelity CFD analysis for calculation of C_D and α at the required value of C_L.

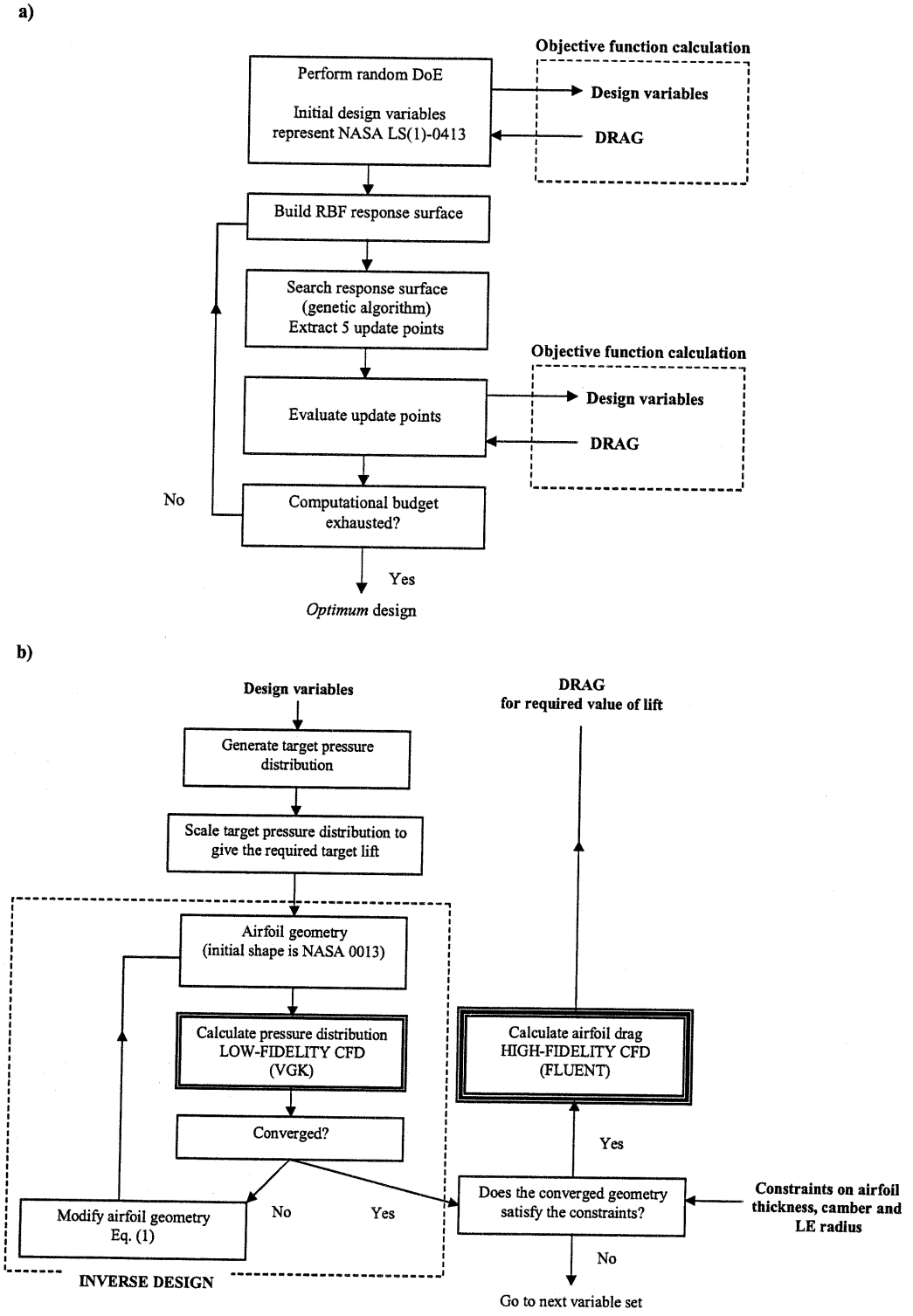

Fig. 10 Flow-chart describing the EMFID design search process: a) optimization strategy and b) an objective function calculation.

C. Comparing the two methods

The two design methods described above are here evaluated side by side to assess their efficiency. The more efficient design method is the one which reaches the highest level of design performance (minimum C_D) given a fixed computational budget. Also of interest is the convergence of the design search process; does the method converge, and if so to what level of performance? Each method has therefore been run five times using different Latin hypercube DoE seeds; their optimization histories, showing the progress of the optimizer in maximizing performance against the number of designs evaluated, are presented.

An important factor when comparing the two methods is the bounds placed on the design variables. These must be equivalent for each method such that one method is not forced to search a much larger or smaller design space than the other. It is also desirable to maintain a conceptual design approach and permit innovative and radical designs. To allow a direct comparison, the EMFID method must have bounds in pressure profile terms which are equivalent to the geometrical bounds of the benchmark method. However, this is difficult to achieve since a modification to the shape of one surface affects the pressure over both surfaces. In order to constrain the two methods fairly, bounds must be placed on design parameters which exist in both methods. In the current study, constraints are therefore placed on the airfoil shape directly, namely on the maximum thickness, maximum camber and leading edge radius.

The variables for the benchmark method control the position of spline points on the airfoil surface, whereas those for the EMFID method control spline points which establish the shape of the target pressure distribution. These variables are therefore each given a relatively larger range, resulting in a large potential design space. For each set of variables selected by the optimization process of either method, if these variables generate an airfoil geometry which violates the constraints on thickness, camber and leading edge radius, the variable set and resulting geometry are rejected, as indicated in Figure 10. The objective function is not calculated for rejected geometries, and they are not included when constructing the RSM. If, however, all five update points requested from the search of the RSM are rejected, one of these points is added to the surface and treated as a failed design point. This action prevents an identical RSM being generated which would stall the optimization process. By equating the search bounds of the two methods a direct comparison can be made between their optimization performances and resulting geometries.

D. Computational Expense

The two design methods described above are here run with equal amounts of computational effort; the one which is able to reach a higher level of design performance is deemed to be the more efficient method. A single objective function evaluation for the benchmark method takes on average 21.4 minutes when running on a Xeon 2.8GHz compute node. An objective function evaluation for the EMFID method uses this effort plus the effort required in the inverse design step. Each VGK evaluation requires approximately 2 seconds; if on average 20 iterations are used in the inverse design step the total objective function call demands 22.1 minutes computational time. Thus, the ratio of computational expense for the two methods is 1:1.031 benchmark to EMFID evaluations.

IV. Case Study

The benchmark design search method (Figure 2) and the EMFID method (Figure 4) have been run side by side to compare their relative performance. The benchmark method is given a budget of 300 calls to the high fidelity CFD code, which equates to 290 objective function evaluations in EMFID after taking into account the effort required for the inverse design. Based on the recommendation of Jones,[20] the number of points in the initial DoE should be ten times the number of design variables. For the benchmark method this gives 130 design points, and for the EMFID method this requires 60 points. The methods use their remaining budget to evaluate designs during the update process, giving 170 update points for the benchmark method and 230 update points in EMFID. The design constraints on airfoil shape maximum thickness, maximum camber, leading edge radius and total lift, respectively, are:

$$12.5\% \leq t/c \leq 15\%,$$

$$0\% \leq z_c/c \leq 2.5\%, \qquad (3)$$

$$r_{LE\ initial} \leq r_{LE}$$

$$C_L = 0.4$$

In these constraints $r_{LE\ initial}$ is the leading edge radius of the initial shape in the design process, i.e., the NASA LS(1)-0413 airfoil. The constraints are arranged with relatively narrow ranges such that the design methods converge more quickly onto promising designs. In addition to the above, a constraint is applied which states that the airfoil thickness at 95% chord shall be no smaller than 0.8% chord. With these constraints applied, the airfoil designs are forced to take a conventional form. As already noted, the target lift and flow conditions are the same as those used in Refs. 10 and 11, allowing comparisons to be made with the NASA low-speed airfoils.

The optimization-iteration histories for the two methods are shown in Figure 11. The progress of the optimizer in each case is plotted against the number of benchmark iterations; the number of iterations of EMFID has been scaled in this plot such that the x-axis can be interpreted as equal computational expense. The design objective is drag, which has been normalized by multiplying by the ratio of lift output from the CFD code to the target lift; this removes the numerical error generated if the airfoil C_L is not exactly 0.4. From Figure 11 it is seen that for a given computational cost, the EMFID method finds a better design performance than the benchmark search method. The rate of convergence towards the optimum is faster for the EMFID method. This is because EMFID is able to find significantly better designs in its initial DoE, and this in turn is due to the reduced problem dimensionality. While neither method has provided convergence onto a single optimum design, the five runs of the EMFID method show greater convergence, indicated by the reduced range of objective values at the end of the search process.

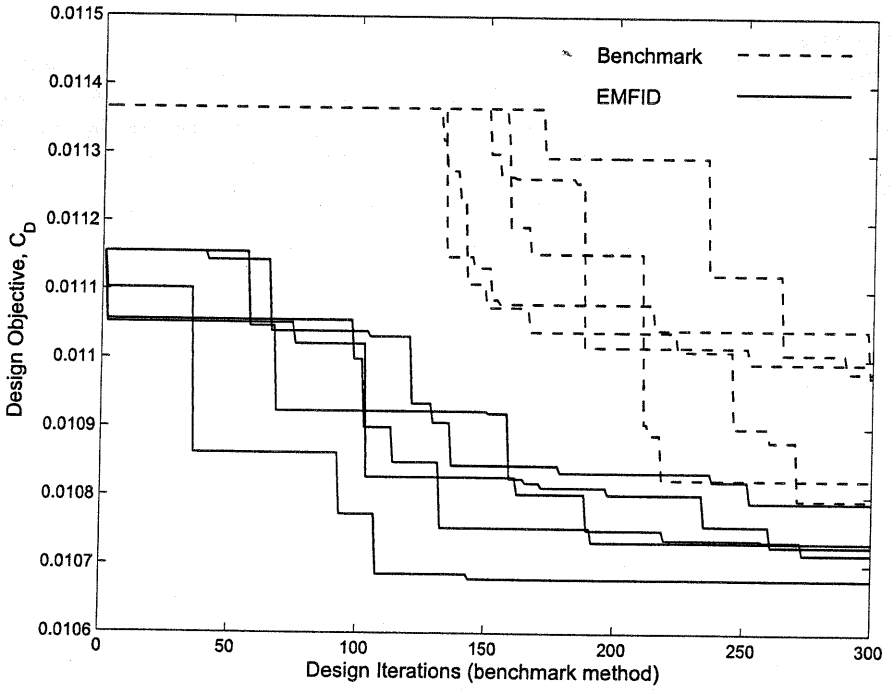

Fig. 11 The five optimization histories for the benchmark and EMFID methods.

The geometries which were found to give the best performance for each of the five runs of the benchmark procedure are shown in Figure 12. It is clear that each initial DoE set has given a different final result for the computational budget used, but all designs appear to be similar in their general form. Table 1 gives data on these geometries. Despite the apparent differences, the maximum camber of the five shapes is in fact very similar, which has also lead to similar angles of attack. However, the airfoil maximum thickness is clearly not converged, possibly a result of the optimization procedure locating many local optima. This is a symptom of the high dimensional design space.

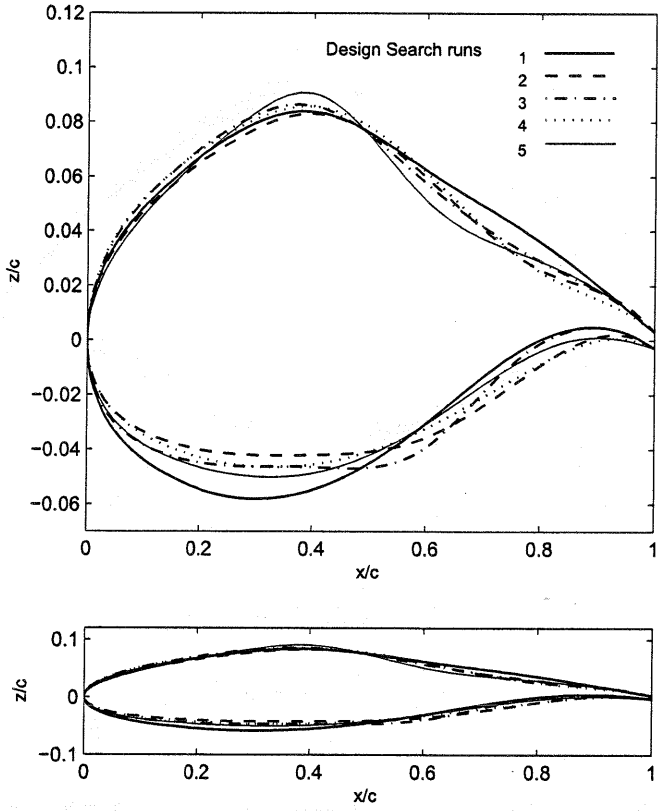

Fig. 12 Final five geometries generated by the benchmark method. The lower figure shows the airfoils on equally scaled axes.

Table 1 Airfoil design data for the five geometries resulting from the benchmark design search: maximum thickness, maximum camber, angle of attack, lift coefficient and drag coefficient.

Run	t	z_c	α	C_L	C_D *(counts)*
1	0.1407	0.0189	0.0308	0.3999	109.8
2	0.1251	0.0205	0.2990	0.3999	108.2
3	0.1327	0.0201	0.1870	0.3999	109.8
4	0.1317	0.0201	0.5557	0.3999	107.9
5	0.1403	0.0209	0.4601	0.3999	109.9

The five best geometries from the computations using the EMFID method are shown in Figure 13. The more advanced state of convergence is made clear by the greater similarity in the five geometries than the benchmark

method achieved, and this is possible because of the reduced dimensionality of the design space in EMFID. It is immediately apparent that these shapes are smoother than the benchmark designs, and this is inherent in the parameterization used. However, although the description of the C_p distribution is simple, the resulting geometry after inverse design can be of high detail and complexity.

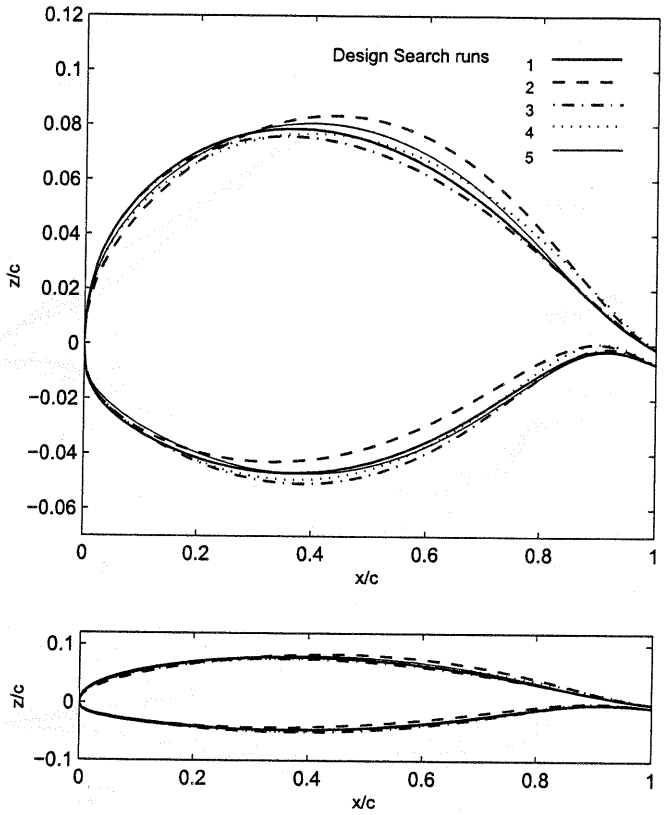

Fig. 13 Final five geometries generated by the EMFID method.

Table 2 Airfoil design data for the five geometries resulting from the EMFID design search: maximum thickness, maximum camber, angle of attack, lift coefficient and drag coefficient.

Run	t	z_c	α	C_L	C_D *(counts)*
1	0.1256	0.0158	0.5941	0.3999	107.3
2	0.1252	0.0229	-0.1831	0.3999	107.9
3	0.1265	0.0132	0.6873	0.3999	106.8
4	0.1262	0.0155	0.1171	0.3999	107.3
5	0.1279	0.0167	0.5634	0.3999	107.2

Table 2 gives the airfoil design data for the five geometries resulting from the EMFID method. Crucially, and rather predictably from looking at the airfoil shapes, the drag for the EMFID designed shapes is lower than those from the benchmark search method. In all five cases the maximum thickness is made to be almost as small as the constraints allow (12.5%), and this is to be expected since the objective is to minimize drag for a single angle of attack. The lower problem dimensionality in EMFID has allowed the optimization procedure to explore promising

areas of the design space more thoroughly than is possible in the benchmark method for the same computational effort.

Figure 14 compares the best two geometries from the five benchmark and EMFID searches. Two NASA low-speed airfoils from Ref. 10 are also shown: the initial design in the search process, NASA LS(1)-0413, which is 12.95% thick, and NASA LS(1)-0013, with a thickness of 12.84%. The 0413 airfoil has 2.2% camber and is designed to give a C_L of 0.4 at 0° angle of attack; the 0013 airfoil has zero camber and gives a C_L of 0.4 at approximately 3.5° angle of attack. The two best airfoils resulting from the benchmark and EMFID methods are 13.17% and 12.65% thick with maximum cambers of 2.0% and 1.3%, respectively. Both designed airfoils feature a cambered adverse pressure region at the lower surface trailing edge, similar to that on the 0413 airfoil. The lower surface of the benchmark airfoil closely resembles that of the 0413 shape; this area has undergone little evolution from the initial NASA 0413 design, whereas the upper surface features *rippling*, associated with the incomplete convergence of the high dimensional search space. The EMFID airfoil has undergone significant development from the initial design. It is interesting that this shape appears to incorporate features from each of the NASA examples: the adverse pressure region on the lower surface, as seen on the 0413 airfoil, and an inflection point at the upper surface trailing edge, a feature of the 0013 shape which gives reduced drag.[11]

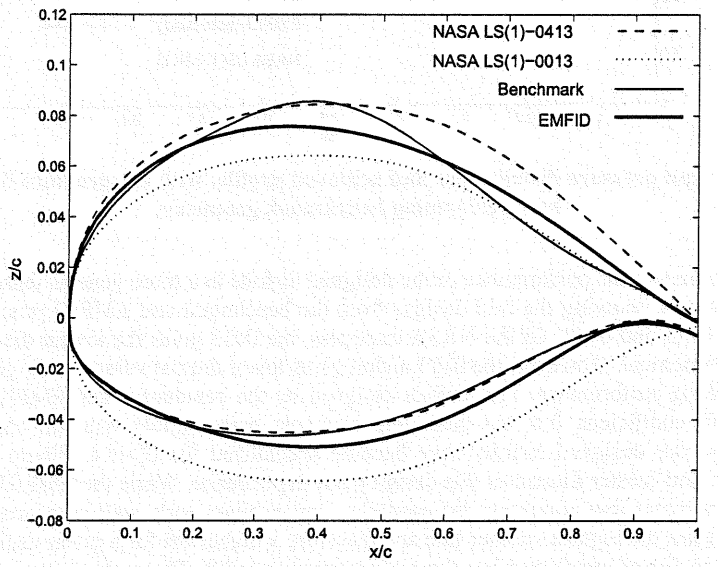

Fig. 14 Comparison of the best performing geometry from each of the two methods, shown with two NASA low-speed airfoils of 13% thickness.

Figure 15 compares the C_p distributions from the best benchmark and EMFID airfoils (i.e., those in Figure 14) with the NASA 0413 C_p distribution, computed using VGK. In the case of the EMFID method two profiles are shown: the best target profile and the achieved profile which most closely matched this during the inverse design process. Note the erratic profile of the benchmark airfoil, an effect of the aforementioned *rippling*.

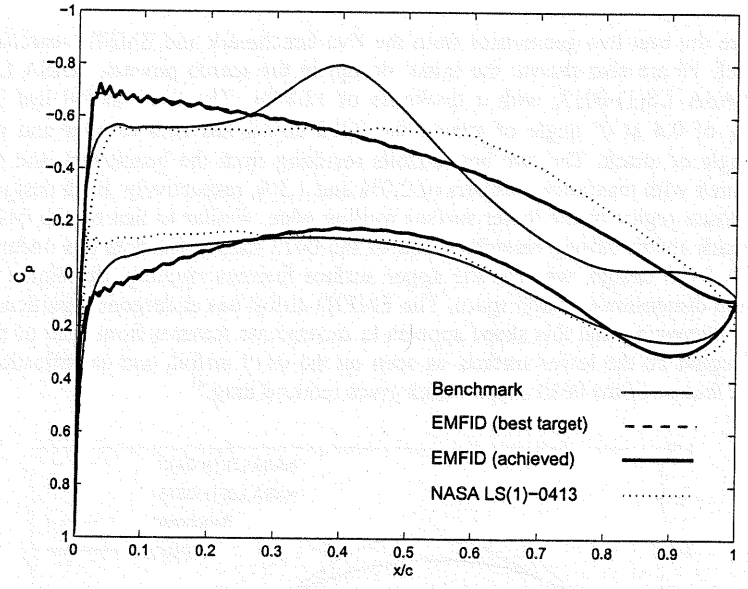

Fig. 15 The final target pressure distribution and achieved profile, with the pressure distribution due to the best performing benchmark geometry.

It is also useful to compare the performance of the designed airfoils in a more general sense. Figure 16 illustrates this as a lift-drag polar plot, showing the best designs from the benchmark and EMFID processes along with plots for the NASA airfoils 0013 and 0413. Of the NASA examples, the 0013 gives the lowest drag; this is at zero angle of attack since it is symmetrical. However, the 0413 airfoil gives lower drag at values of C_L greater than around 0.3, due to its improved lifting performance. The airfoils designed by the benchmark and EMFID methods give lower drag at their design lift coefficient, 0.4, but their drag increases more rapidly with lift compared to the NASA airfoils. Unsurprisingly, the designed airfoils have become specialized for C_L=0.4. Figure 17 shows lift plotted against angle of attack, and further illustrates this design point dependence. While the EMFID airfoil has the lowest drag at C_L=0.4, it has a rather low maximum lift capability and exhibits early stall compared to the other airfoils. The NASA 0413 shape has the highest camber line and therefore a higher lift for a given angle of attack, but it also performs well at high angles of attack and has the highest maximum lift. These observations are an artifact of the design objective used: the NASA airfoils are designed to perform well over a range of angles, whereas the geometries generated here are designed to minimize drag at a single angle of attack (or a single C_L). It is not the purpose of this work to improve on the NASA airfoils; rather it is to demonstrate the use of the EMFID method in design.

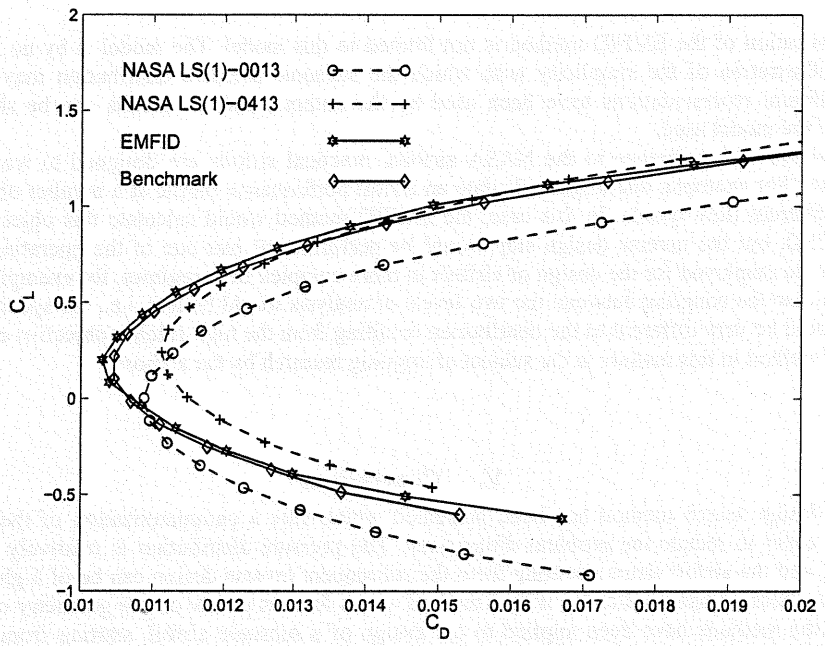

Fig. 16 Lift-drag polar plot for the best designs from the benchmark and EMFID methods, shown with two NASA airfoils.

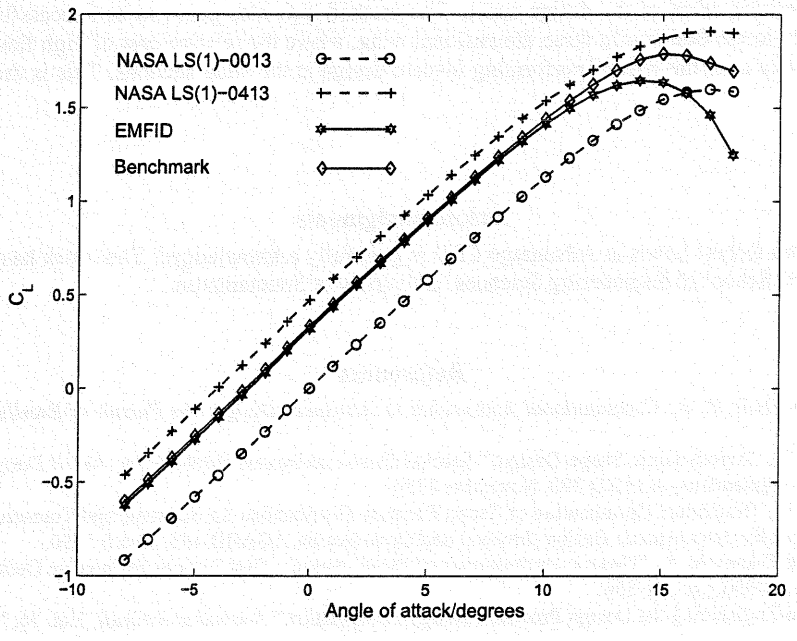

Fig. 17 Lift vs. Angle of attack for the best designs from the benchmark and EMFID methods, shown with two NASA airfoils.

As a final consideration, it is worth discussing the general applicability of the EMFID approach. The B-spline parameterization of the airfoil C_p distribution used in the current work has been shown to perform well, but the

successful application of the EMFID method is not limited to this model. The model is by no means optimal, but serves as an illustration of the simplicity with which the subsonic pressure distribution may be represented. A number of different representations have been used by the authors, and the results can be shown to be largely independent of the model used.

As noted above with reference to the NASA airfoils, practical airfoils are designed to work well at multiple operating points. For example, one might calculate an airfoil performance metric at a number of different angles of attack, or at different flow speeds. In this case, the EMFID method would calculate this objective function using high-fidelity CFD, but the inverse design step would be performed at just one of the operating points. A similar approach could be employed for the design of airfoils in more complex flow regimes, for example, at transonic flow speeds. In this case the coupling between the two levels of analysis would be lost, i.e., the specified target pressure distribution would be very different to the distribution resulting from the high-fidelity objective calculation. The use of the EMFID method in this manner is the subject of ongoing research by the authors.

V. Conclusions

An airfoil design search method has been presented which uses a parameterization of the subsonic pressure distribution in order to reduce the problem dimensions. The pressure distribution is relatively simple to describe parametrically, and the airfoil shape resulting from the subsequent inverse design can be of high quality and detail. This method has been compared directly with a method which focuses purely on the geometry of the airfoil. In the current paper, the methods have been applied to the design of a subsonic airfoil, starting from a design which is known to perform well at the design conditions. For a given computational cost, the proposed embedded multi-fidelity inverse design (EMFID) method is able to return airfoil geometries which perform noticeably better than those of the benchmark method, in terms of aerodynamic efficiency. The use of just six design variables compared to 13 in the benchmark process allows the EMFID method to converge significantly more quickly than the benchmark method. The reduction in the number of design variables has been shown to allow more thorough exploration of promising areas of the design space. The method may also prove to be successful when applied to transonic designs or to the design of a three dimensional wing, where the relative cost of high fidelity CFD analysis increases significantly over the cost of performing inverse design at the wing sections. This is the subject of future work in this area.

Acknowledgments

The support from Robert Lewis at Advantage CFD is gratefully acknowledged. This work has been funded by a scholarship from the School of Engineering Sciences, University of Southampton.

References

[1]Keane, A. J., and Nair, P. B., *Computational Approaches to Aerospace Design: the Pursuit of Excellence*, 1st ed., Wiley, U.K., 2005.

[2]Dulikravich, G. S., "Aerodynamic Shape Design," *Special Course on Inverse Methods for Airfoil Design for Aeronautical and Turbomachinery Applications*, AGARD 780, November 1990.

[3]van Egmond, J. A., "Numerical Optimization of Target Pressure Distributions for Subsonic and Transonic Airfoils Design," *Computational methods for Aerodynamic Design (Inverse) and Optimization*, AGARD 463, March 1990.

[4]Obayashi, S., and Takanashi, S., "Genetic Optimization of Target Pressure Distributions for Inverse Design Methods," *AIAA Journal*, Vol. 34, No. 5, 1996, pp. 881-886.

[5]Jameson, A., "Re-Engineering the Design Process Through Computation," *Journal of Aircraft*, Vol. 36, No. 1, 1999, pp. 36-50.

[6]Kim, H. J., and Rho, O. H., "Aerodynamic Design of Transonic Wings Using the Target Pressure Optimization Approach," *Journal of Aircraft*, Vol. 35, No. 5, 1998, pp. 671-677.

[7]Ahn, T., Kim, H. J., and Rho, O. H., "Inverse Design of Transonic Wings Using Wing Planform and Target Pressure Optimization," *Journal of Aircraft*, Vol. 38, No. 4, 2001, pp. 644-652.

[8]Fluent, "FLUENT 5 User's Guide," *Fluent Inc.*, Lebanon, 2003.

[9]Engineering Sciences Data Unit, "The VGK Method for Two-Dimensional Aerofoil Sections," ESDU 96028, 1996.

[10]McGhee, R. J., Beasley, W. D. and Whitcomb, R. T., "NASA Low- and Medium-Speed Airfoil Development," NASA Technical Memorandum 78709, 1979.
[11]Ferris, J. C., McGhee, R. J. and Barnwell, R. W., "Low-Speed Wind-Tunnel Results for Symmetrical NASA LS(1)-0013 Airfoil," NASA Technical Memorandum 4003, 1987.
[12]Sai, V. A., and Lutfy, F, M., "Analysis of the Baldwin-Barth and Spalart-Allmaras One-Equation Turbulence Models," AIAA Journal, Vol. 33, No. 10, 1995, pp. 1971-1974.
[13]Samareh, J. A., "A Survey of Shape Parameterization Techniques," *CEAS/AIAA/CASE/NASA Langley International Forum on Aeroelasticity and Structural Dynamics*, CP-209136, June 1999, pp. 333-343.
[14]Robinson, G. M., and Keane, A. J., "Concise Orthogonal Representation of Supercritical Airfoils," *Journal of Aircraft*, Vol. 38, No. 3, 2001, pp. 580-583.
[15]Hicks, R. M., and Henne, P. A., "Wing Design by Numerical Optimization," *Journal of Aircraft*, Vol. 15, No. 7, July 1978, pp. 407-414.
[16]Song, W., and Keane, A. J., "A Study of Shape Parameterisation Methods for Airfoil Optimisation," *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, AIAA 2004-4482, August 2004.
[17]Jacobs, E. N., Ward, K. E., and Pinkerton, R. M., "The Characteristics of 78 Related Airfoil Sections from Tests in the Variable-Density Wind Tunnel", N.A.C.A. T. R. No. 460, 1933.
[18]Lépine, J., Guilbault, F. and Trépanier, J. Y., "Optimized Nonuniform Rational B-Spline Geometrical Representation for Aerodynamic Design of Wings," *AIAA Journal*, Vol. 39, No. 11, November 2001, pp. 2033-2041.
[19]Li, W., Krist, S. and Campbell, R., "Transonic Airfoil Shape Optimization in Preliminary Design Environment," *Collection of Technical Papers - 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Comference*, 2004, pp. 3650-3671.
[20]Jones, D. R., "A Taxonomy of Global Optimization Methods Based on Response Surfaces," *Journal of Global Optimization* **21**, 2001, pp. 345-383.
[21]Jones, D. R., Schonlau, M., and Welch, W. J., "Efficient Global Optimization of Expensive Black-Box Functions," *Journal of Global Optimization* **13**, 1998, pp. 455-492.
[22]Keane, A. J., "OPTIONS Design Exploration System," http://www.soton.ac.uk/~ajk [cited April 2005].
[23]The MathWorks, Inc., "MATLAB® – The Language of Technical Computing," http://www.mathworks.com/ [cited April 2005].
[24]The Geodise Project, "Grid Enabled Optimisation and Design Search for Engineering," http://www.geodise.org [cited April 2005].
[25]Sóbester, A., Leary, S. J., and Keane, A. J., "A Parallel Updating Scheme for Approximating and Optimizing High Fidelity Computer Simulations," *Structural and Multidisciplinary Optimization* **27**, 2004, pp. 371-383.
[26]Keane, A. J., "Wing Optimization Using Design of Experiment, Response Surface, and Data Fusion Methods," *Journal of Aircraft*, Vol. 40, No. 4, July 2003, pp. 742-750.
[27]Alexandrov, N. M., Lewis, R. M., Gumbert, C. R., Green, L. L., and Newman, P. A., "Optimization with Variable-Fidelity Models Applied to Wing Design," *38th Aerospace Sciences Meeting and Exhibit*, AIAA 2000-0841, January 2000.
[28]Gopalarathnam, A., and Selig, M. S., "Hybrid Inverse Airfoil Design Method for Complex Three-dimensional Lifting Surfaces," *Journal of Aircraft*, Vol. 39, No. 3, May 2002, pp. 409-417.
[29]LeGresley, P. A., Alonso, J. J., "Airfoil Design Optimization Using Reduced Order Models Based on Proper Orthogonal Decomposition" *Proceedings of Fluids 2000 Conference and Exhibit*, AIAA 2000-2545, June 2000.
[30]Takanashi, S., "Iterative Three-Dimensional Transonic Wing Design Using Integral Equations," *Journal of Aircraft*, Vol. 22, No. 8, August 1985, pp. 655-660.
[31]Davis, W. H. Jr., "Technique for Developing Design Tools from the Analysis Methods of Computational Aerodynamics," *AIAA Journal*, Vol. 18, No. 9, September 1980, pp. 1080-1087.
[32]Milholen II, W. E., "Efficient Inverse Aerodynamic Design Method for Subsonic Flows," *Journal of Aircraft*, Vol. 38, No. 5, September 2001, pp. 918-923.