Advances in Engineering Software 41 (2010) 923–930

Contents lists available at ScienceDirect

Advances in Engineering Software

journal homepage: www.elsevier.com/locate/advengsoft

Bezier-PARSEC: An optimized aerofoil parameterization for design

R.W. Derksen [a,*], Tim Rogalsky [b]

[a] Department of Mechanical and Manufacturing Engineering, University of Manitoba, Winnipeg, Canada
[b] Mathematics Department, Canadian Mennonite University, Winnipeg, Canada

ARTICLE INFO

Article history:
Received 1 October 2009
Accepted 2 April 2010
Available online 22 June 2010

Keywords:
Aerodynamic optimization
Airfoil parameterization
Bezier curves
Differential evolution
Airfoil design
Airfoil profiles

ABSTRACT

The numerical search for the optimum shape of an aerofoil is of great interest for aircraft and turbomachine designers. Unfortunately, this process is very computationally intense and can require a large number of individual flow field simulations resulting in very long CPU run times. One of the core issues that the designer must deal with is how to describe the shape of the airfoil. Clearly, we cannot treat the profile on a point by point basis as the problem would have an infinite number of degrees of freedom. Hence the typical practice is to resort to using a series of curves, such as polynomials and Bezier curves, to describe the profile. This typically reduces the number of degrees of freedom to a much smaller, manageable number. The influence of the selection of the parameterization on the optimization has received relatively little consideration to date. We can anticipate that some parameterizations will be less suitable for describing the profile shape and may result in slower convergence times.

Our paper will discuss a new airfoil parameterization, Bezier-PARSEC, that was developed to extend and improve the typical Bezier parameterization found in use. This parameterization was found to fit the known shape of a wide range of existing airfoil profiles as well as resulting in accelerated convergence for aerodynamic optimization using Differential Evolution. Our presentation will present the development and details of the Bezier-PARSEC parameterization and provide evidence that the parameterization is suitable and accelerates convergence.

1. Introduction

The profiles of most airfoils given in the literature are described by a tabulated vector of 50–80 coordinate pairs located on the profile surface as demonstrated by NACA compilation given by Abbot and von Doenhoff [1]. The practical demands of aerodynamic optimization on CPU time require us to search for an effective parameterization that contains fewer degrees of freedom. Four main objectives must be considered in developing an airfoil parameterization: minimizing the number of degrees of freedom as possible, the parameterization should be able to represent a wide range of existing airfoils, the parameters should be simple to formulate and impose, and finally, the parameterization should result in effective and efficient optimization.

The following work describes a new method of representing an airfoil that is an extension of a Bezier parameterization previously used by the authors [2] but uses aerodynamic parameters. These are similar to the PARSEC parameters developed by Sobieczky [3]. In Section 2 the fundamentals of evolutionary optimization are described, as applied to aerodynamic design. In Section 3, the new airfoil parameterization is given. Section 4 compares the abilities of the different parameterizations to represent a range of airfoils. In Section 5 the effect of parameterization on convergence speed is discussed.

2. Evolutionary aerodynamic optimization

Optimization is a required element of aerodynamic design. Genetic Algorithms (GAs) are relatively new algorithms which mimic the theory of natural evolution in their search for an objective function's optimal value. They are robust strategies for aerodynamic design [4], having the ability to find the global optimum even when other algorithms are unable to do so. Differential Evolution (DE) [5] is a recently developed GA which has been found to be particularly effective for aerodynamic optimization [2,6].

DE, like other GAs, starts with a large collection of design vectors, the initial population. It interprets the function value of a vector as a measure of that individual's fitness as an optimum. Then, guided by the principle of survival of the fittest, the initial population of vectors is transformed, generation by generation, into a solution vector.

The overall structure of the DE algorithm resembles that of most other population-based searches. Two arrays are maintained, each of which holds a population of NP, D-dimensional, real-valued vectors. For an effective search, the recommended population size is $NP = 10D$, which is used throughout this paper. The primary array holds the current population while the secondary array

* Corresponding author.
E-mail address: derksen@cc.umanitoba.ca (R.W. Derksen).

0965-9978/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.advengsoft.2010.05.002

accumulates vectors that are selected for the next generation. Selection occurs by competition between the existing vectors and trial vectors. The trial vectors used by DE are formed through mutation and recombination of the vectors in the primary array.

Mutation is an operation that makes small random alterations to one or more parameters of an existing population vector. Mutation is crucial for maintaining diversity in a population, and is typically performed by perturbation. Traditional GAs use predetermined probability distributions to perturb vectors. This leaves them unable to adapt the perturbation magnitude to the topology of the objective function. DE avoids this problem by using the population itself as the source of appropriately scaled perturbations. It uses a weighted "differential" – the difference of two vectors in the primary array, multiplied by weight F – to perturb other vectors. In this way, as convergence approaches, those variables having a narrow and well-defined range around the minimum will have small variation among the population members, resulting in their mutations being relatively small. This automatic adaptation, known as self-organization, significantly improves behaviour of the algorithm as convergence nears the global optimum. Mutation is controlled by the differential weight and the recommended value of $F = 0.85$ is used throughout.

Recombination, or crossover, provides an alternative and complementary means of creating viable vectors. Designed to resemble the natural process by which a child inherits DNA from its parents, new parameter combinations are built from the components of existing vectors. This efficiently shuffles information about successful combinations, enabling the search for an optimum to focus on the most promising areas of the solution space. A primary array vector is first mutated to form a trial vector. That trial vector is then modified using crossover with its parent. Binary crossover is used throughout, which is controlled by the crossover constant CR. For each parameter, CR is compared to random number chosen from a uniform distribution within the interval [0, 1). If the random number is smaller than CR, the trial vector parameter is chosen; otherwise the parameter is kept the same as that of the parent. Note that $CR = 1$ results in no crossover, in which case the trial vector is formed by mutation alone.

While DE is robust for aerodynamic optimization, it is also inefficient, requiring roughly 50,000 function evaluations to converge [2]. Thus the flow-solver used for each trial vector must be computationally inexpensive. Here the aerodynamic calculations are performed using an inviscid potential flow model, Martensen's surface vorticity panel method [7]. While this model simplifies the true fluid dynamics situation, it will accurately predict the flow in most subsonic situations. Ideally, a boundary layer solution should be coupled to the potential flow-solver. However, convergence could then take weeks or even months to occur. Thus some method of accelerating the convergence is required. It will be demonstrated that a different aerofoil parameterization can do just that.

3. Bezier-PARSEC parameterization

A Bezier parameterization is determined by its control points which are physical points in the plane. Four curves form the airfoil, two for the camber line and two for the thickness distribution. The first and last control points of a Bezier curve are the initial and terminal point on the curve itself. However the other control points need not be on the curve even though they determine the shape of the curve. As such a Bezier parameterization of an airfoil has control points that are only indirectly determined by the underlying aerodynamics. It is desirable to have the parameters such as leading edge radius, trailing wedge angle, and so on that do have physical relevance to the flow, such as the PARSEC parameterization, demonstrated by Sobieczky [3,8].

Oyama et al. [9] showed that this type of parameterization improved the robustness and convergence speed for aerodynamic optimization, which makes it more suitable for optimization using genetic algorithms. The superior performance of PARSEC parameterization is likely due to its ability to minimize epistasis – a term borrowed from biology to indicate the nonlinear manner in which the objective function is dependent on the design parameters [10]. In epistatic functions, small changes in several variables can result in large changes in the objective function. They are difficult to minimize because they provide so few clues as to the location of the global minimum. Typically a reduction in nonlinear interaction of the parameters, by having parameters more directly linked to the objective function, will enable the optimizer to converge more quickly.

This motivated the development of a combined Bezier-PARSEC (BP) parameterization to utilize the advantages of both the Bezier and PARSEC parameterizations. Bezier curves are commonly used in industry and in CAD systems generally. The camber-thickness formulation of the Bezier curves is more directly related to the flow than is the upper curve-lower curve formulation for PARSEC, while the PARSEC parameters are more aerodynamically oriented than the Bezier parameters. The BP parameterization uses the PARSEC variables as parameters, which in turn define four separate Bezier curves. These curves describe the leading and trailing portions of the camber line, and the leading and trailing portion of the thickness distributions. While the Bezier parameterization joins the leading and trailing curves with first-order continuity, the BP parameterization uses second-order continuity. Each curve is ultimately a Bezier curve, and the new parameterization will be denoted by BP $ijkl$, where i and j are the order of the leading and trailing thickness curves, and k and l are the order of the leading and trailing camber curves.

The parameters are: leading edge radius – r_{le}, trailing camber line angle – α_{te}, trailing wedge angle – β_{te}, trailing edge vertical displacement – z_{te}, leading edge direction γ_{le}, location of the camber crest – (x_c, y_c), curvature of the camber crest – κ_c, position of the thickness crest – (x_t, y_t), curvature of the thickness crest – κ_t, the half thickness of the trailing edge – dz_{te}, and several Bezier variables, b_0, b_2, b_8, b_{15} and b_{17}. The detailed development of the Bezier-PARSEC curves is given by Rogalsky [11] and only the results will be given here. In the following, the variables x_i and y_i represent the Bezier control points that uniquely define each Bezier curve.

3.1. BP 3333 parameterization

A BP 3333 parameterization uses third degree Bezier curves for all four curves used to define the airfoil. A third degree Bezier curve is given parametrically by

$$x(u) = x_0(1-u)^3 + 3x_1u(1-u)^2 + 3x_2u^2(1-u) + x_3u^3,$$

and

$$y(u) = y_0(1-u)^3 + 3y_1u(1-u)^2 + 3y_2u^2(1-u) + y_3u^3,$$

where u is the parameter that runs from 0 at the beginning to 1 at its terminus. The BP 3333 parameterization relies exclusively on the 12 aerodynamic parameters shown in Fig. 1 – there are no free Bezier points in BP 3333.

3.1.1. Leading edge thickness curve

The control points are given by

$$\begin{aligned} x_0 &= 0 & y_o &= 0 \\ x_1 &= 0 & y_1 &= 3\kappa_t(x_t - r_t)^2/2 + y_t \\ x_2 &= r_t & y_2 &= y_t \\ x_3 &= x_t & y_3 &= y_t \end{aligned}$$

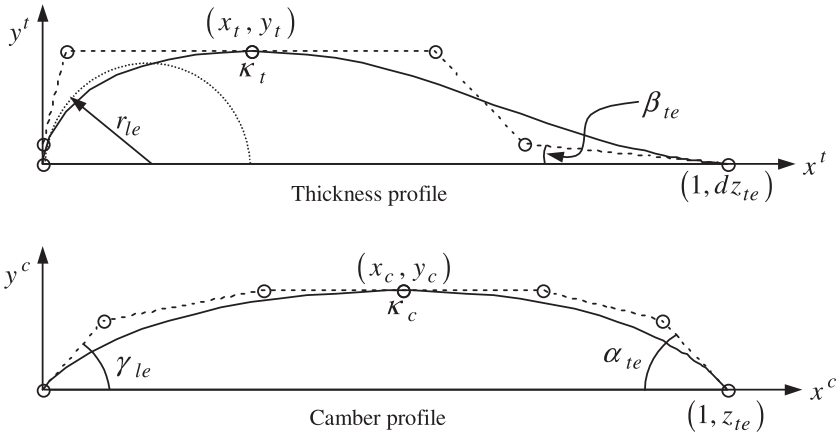

Fig. 1. BP 3333 airfoil geometry and Bezier control points defined by twelve basic aerodynamic parameters.

The parameter r_t is the smallest root of

$$27\kappa_t^2 r_t^4/4 - 27\kappa_t^2 x_t r_t^3 + (9\kappa_t y_t + 81\kappa_t^2 x_t^2/2) r_t^2 + (2r_{le} - 18\kappa_t x_t y_t - 27\kappa_t^2 x_t^3) r_t + (3y_t^2 + 9\kappa_t x_t^2 y_t + 27\kappa_t^2 x_t^4/4) = 0$$

within the bounds given by

$$\max(0, x_t - \sqrt{-2y_t/3\kappa_t}) < r_t < x_t.$$

3.1.2. Trailing edge thickness curve

The control points are given by

$$\begin{aligned} x_0 &= x_t & y_0 &= y_t \\ x_1 &= 2x_t - r_t & y_1 &= y_t \\ x_2 &= 1 + [dz_{te} - (3\kappa_t(x_t - r_t)^2/2 + y_t)]\cot(\beta_{te}) & y_2 &= 3\kappa_t(x_t - r_t)^2/2 + y_t \\ x_3 &= 1 & y_3 &= dz_{te} \end{aligned}$$

3.1.3. Leading edge camber curve

The control points are given by

$$\begin{aligned} x_0 &= 0 & y_0 &= 0 \\ x_1 &= r_c\cot(\gamma_{le}) & y_1 &= r_c \\ x_2 &= x_c - \sqrt{2(r_c - y_c)/3\kappa_c} & y_2 &= y_c \\ x_3 &= x_c & y_3 &= y_c \end{aligned}$$

The parameter r_c is computed from

$$r_c = [16 + 3\kappa_c(\cot\gamma_{le} + \cot\alpha_{te})(1 + z_{te}\cot\alpha_{te})]/[3\kappa_c(\cot\gamma_{le} + \cot\alpha_{te})] \pm 4\sqrt{16 + 6\kappa_c(\cot\gamma_{le} + \cot\alpha_{te})(1 - y_c(\cot\gamma_{le} + \cot\alpha_{te}) + z_{te}\cot\alpha_{te})}$$

and must be within the following bounds

$$0 < r_c < y_c.$$

3.1.4. Trailing edge camber curve

The control points are given by

$$\begin{aligned} x_0 &= x_c & y_0 &= y_c \\ x_1 &= x_c + \sqrt{2(r_c - y_c)/3\kappa_c} & y_1 &= y_c \\ x_2 &= 1 + (z_{te} - r_c)\cot(\alpha_{te}) & y_2 &= r_c \\ x_3 &= 1 & y_3 &= z_{te} \end{aligned}$$

3.2. BP 3434 parameterization

A BP 3434 parameterization uses third degree Bezier curves for all of the leading edge curves and fourth degree Bezier curves for the both trailing curves used to define the airfoil. Five new parameters are introduced, each of them representing a Bezier curve coefficient, and the crest curvature parameters are removed. Thus the BP 3434 parameterization relies on ten aerodynamic parameters, plus five Bezier parameters. These are shown in Fig. 2.

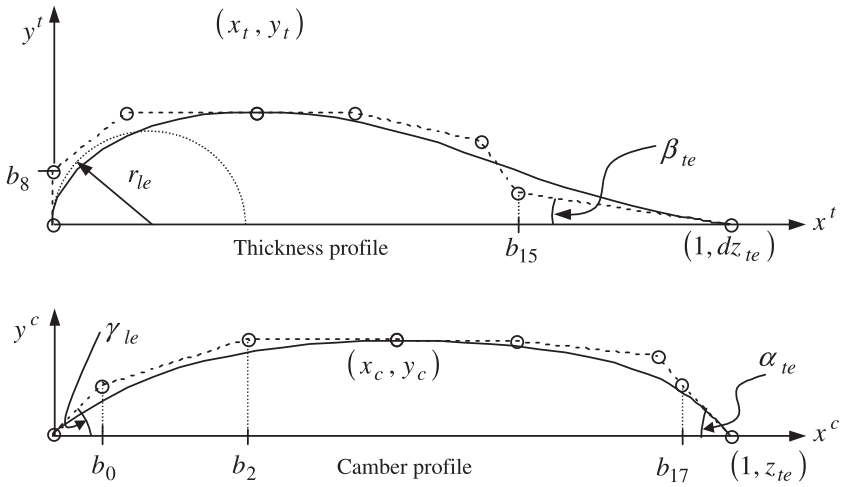

Fig. 2. BP 3434 airfoil geometry and Bezier control points defined by ten aerodynamic and five Bezier parameters.

A fourth degree Bezier curve is given parametrically by

$$x(u) = x_0(1-u)^4 + 4x_1u(1-u)^3 + 6x_2u^2(1-u)^2 + 4x_3u^3(1-u) + x_4u^4,$$

and

$$y(u) = y_0(1-u)^4 + 4y_1u(1-u)^3 + 6y_2u^2(1-u)^2 + 4y_3u^3(1-u) + y_4u^4.$$

3.2.1. Leading edge thickness curve

The control points are given by

$$\begin{aligned} x_0 &= 0 & y_o &= 0 \\ x_1 &= 0 & y_1 &= b_8 \\ x_2 &= -3b_8^2/2r_{le} & y_2 &= y_t \\ x_3 &= x_t & y_3 &= y_t \end{aligned}$$

Here the parameter b_8 is subject to the following restriction

$$0 < b_8 < \min(y_t, \sqrt{-2r_{le}x_t/3}).$$

3.2.2. Trailing edge thickness curve

The control points are given by

$$\begin{aligned} x_0 &= x_t & y_0 &= y_t \\ x_1 &= (7x_t + 9b_8^2/2r_{le})/4 & y_1 &= y_t \\ x_2 &= 3x_t + 15b_8^2/4r_{le} & y_2 &= (y_t + b_8)/2 \\ x_3 &= b_{15} & y_3 &= dz_{te} + (1-b_{15})\tan(\beta_{te}) \\ x_4 &= 1 & y_4 &= dz_{te} \end{aligned}$$

3.2.3. Leading edge camber curve

The control points are given by

$$\begin{aligned} x_0 &= 0 & y_0 &= 0 \\ x_1 &= b_0 & y_1 &= b_0\tan(\gamma_{le}) \\ x_2 &= b_2 & y_2 &= y_c \\ x_3 &= x_c & y_3 &= y_c \end{aligned}$$

3.2.4. Trailing edge camber curve

The control points are given by

$$\begin{aligned} x_0 &= x_c & y_0 &= y_c \\ x_1 &= (3x_c - y_c\cot(\gamma_{le}))/2 & y_1 &= y_c \\ x_2 &= (-8y_c\cot(\gamma_{le}) + 13x_c)/6 & y_2 &= 5y_c/6 \\ x_3 &= b_{17} & y_3 &= z_{te} - (1-b_{17})\tan(\alpha_{te}) \\ x_4 &= 1 & y_4 &= z_{te} \end{aligned}$$

The parameterizations assume that the length of the airfoil has been normalized to unit length.

3.3. Airfoil representation

The parameterization method must be able to represent a wide range of airfoils if it is to contribute to a robust design algorithm. The authors compared the ability of Bezier and BP methods to reproduce 63 known airfoils: 40 NACA symmetric and asymmetric airfoils [1], 15 Eppler airfoils [12], and 8 low-speed airfoils [13]. Data for each of the selected airfoils consisted of an ordered array of coordinates for each profile.

The curves were fit to the coordinates using a modified airfoil design code that used differential evolution to find the parameters that minimized the ℓ_2 – error norm between the representation and the given data points. The DE algorithm was rand-to-best/1/bin with $F = 0.85$, $CR = 1$, and a population of 150 members. The maximum number of generations permitted was 500, however, in a few cases an additional run was used to obtain convergence. The convergence requirement was that the cost limit was less than 0.01. For a unit chord length airfoil this corresponds to an average deviation of approximately 8×10^{-4}.

The following discussion will be limited to a few representative airfoils. For a complete analysis of each of the 63 airfoils, see Rogalsky [11].

3.3.1. NACA symmetric airfoils

The Bezier, BP 3333 and BP 3434 approximations converged for all of the symmetric airfoils. Typically, fewer than 3000 approximations, or 20 generations were required to obtain convergence, with one notable exception, the BP 3333 parameterization of the NACA 67_1-015, which required almost 10,000.

The parameterization of the NACA 0008-34 airfoil will be presented in more detail. The curve fitting required the following: Bezier fit – 6612 function evaluations with an average deviation of 1.06×10^{-4}, BP 3333 fit – 4243 function evaluations with an average deviation of 1.24×10^{-4}, and BP 3434 fit - 6180 function evaluations with an average deviation of 1.26×10^{-4}. Clearly, all of the parameterization methods could successfully approximate the shape of the symmetric profiles. The fit to the NACA 00008-34 is shown in Fig. 3.

3.3.2. NACA asymmetric airfoils

The Bezier and BP 3434 were able to successfully reproduce each of the 20 asymmetric airfoils, with the BP 3333 failing to reproduce one, the NACA 744A315 to the required tolerance. The average number of function evaluations for the successful representations are as follows: Bezier fit – 3302, BP 3333 – 3071, and BP 3434 – 4424. A successful fit for each method is shown in Fig. 4, for the NACA 63_1-212 airfoil.

The failure of the BP 3333 parameterization for the NACA 747A315 was occurred near the trailing edge. This was primarily attributed to a lack of sufficient freedom on the trailing edge camber with minor issues with the trailing edge thickness profile. The NACA 747A315 has a sharp cusp at the trailing edge that is difficult to approximate. This was not an issue with the BP 3434 approximation as it has additional control points at the trailing edge.

3.3.3. Eppler airfoils

Eppler pursued the development of accurate theoretical methods to obtain inverse designs of airfoils with prescribed boundary layer characteristics. This has resulted in a catalogue of well established airfoils [12]. These airfoils represent a greater challenge due to their shape. The attempts to approximate these airfoils were less successful with the following success rates: Bezier – 73%, BP 3333 – 80%, and BP 3434 – 90%. The average number of function evaluations for the successful approximations were: Bezier – 3316, BP 3333 – 3570, and BP 3434 – 4870. The resulting approximations for the E 266 airfoil are shown in Fig. 5.

The reasons for the failure to approximate the Eppler airfoils to the required accuracy are numerous, and include a lack of degrees of freedom, difficulty with airfoils that have a positive leading edge and negative trailing edge directions, and finite trailing edge thickness to name a few issues. For the Bezier parameterization specifically, some airfoils that were represented within the required tolerance did so at the expense of second-order continuity at the crest. For example, the Bezier version of E 337 has camber crest curvature -0.064 from the left, and -1.72 from the right.

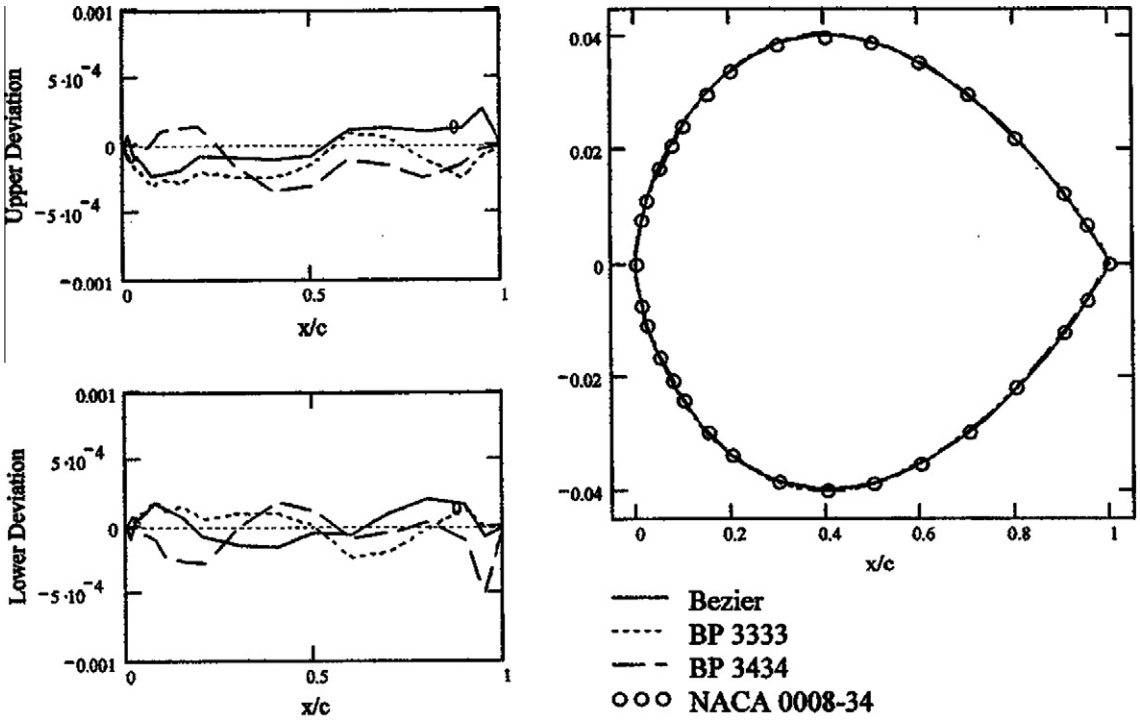

Fig. 3. Representation of the NACA 0008-34 airfoil.

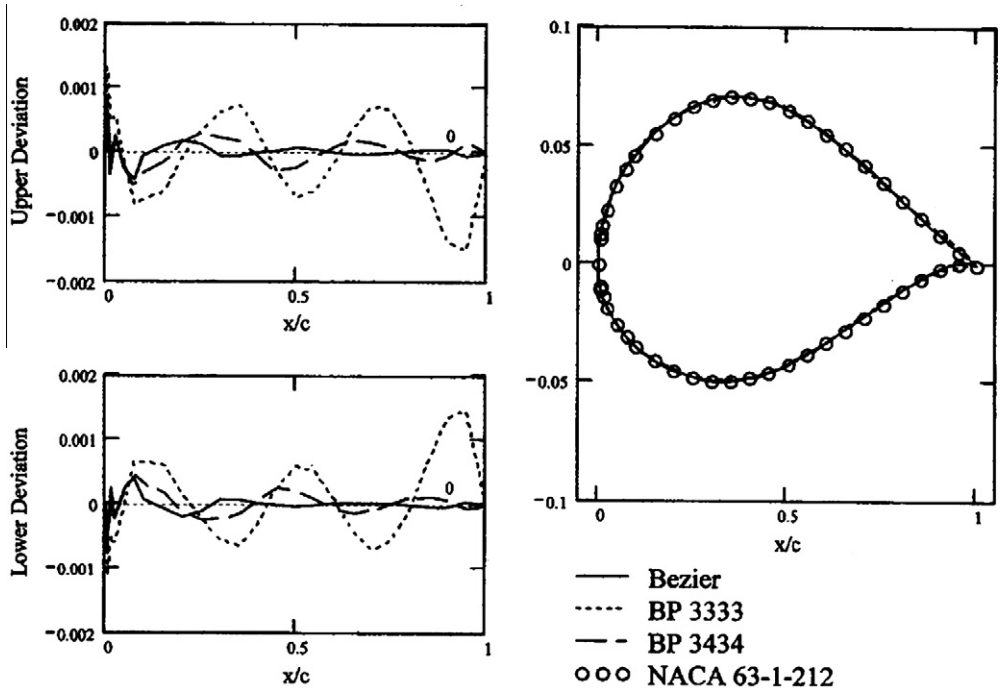

Fig. 4. Representation of the NACA 63-1-212 airfoil.

3.3.4. Low-speed airfoils

All of the methods were able to reproduce the selected set of low-speed airfoils to the desired tolerance. The average number of function evaluations required to fit the airfoils were: Bezier – 5483, BP 3333 – 8834, and BP 3434 – 9994. A representative fit for the low-speed airfoils is given for the FX 74-CL5-140 MOD airfoil shown in Fig. 6.

3.4. Summary

Each of the three parameterization methods was able to represent a high percentage of the sample airfoils. Of the 63 airfoils, the Bezier parameterization reproduced 58, BP 3333 reproduced 59, and BP 3434 reproduced 62. In most cases the approximation required less than 10,000 function evaluations. Typically, the Bezier

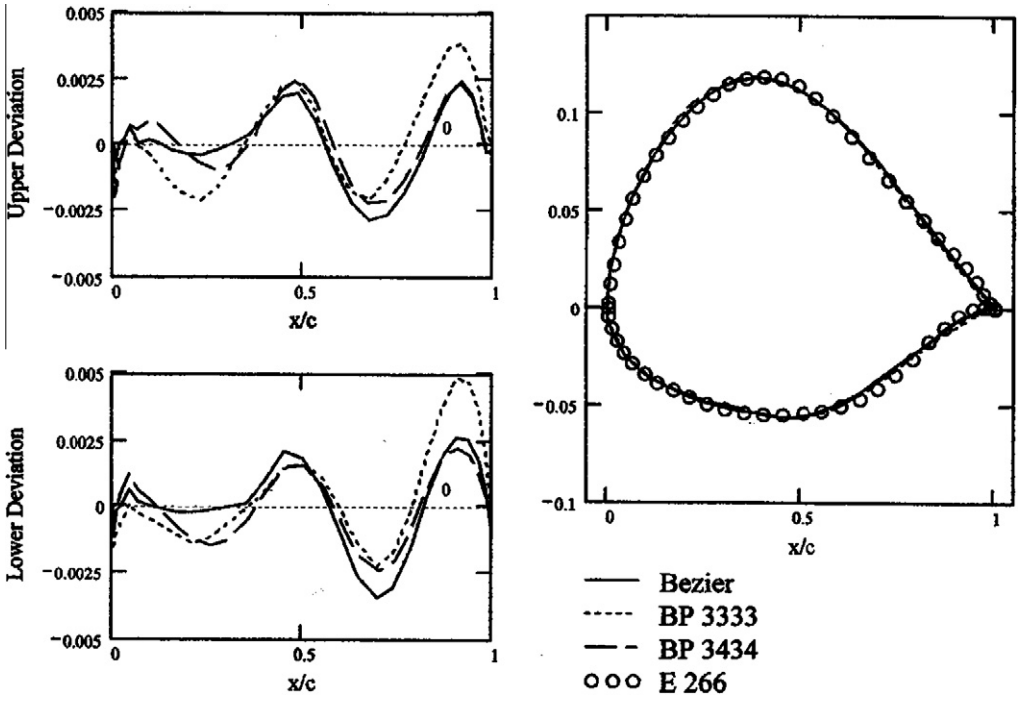

Fig. 5. Representation of the E 266 airfoil.

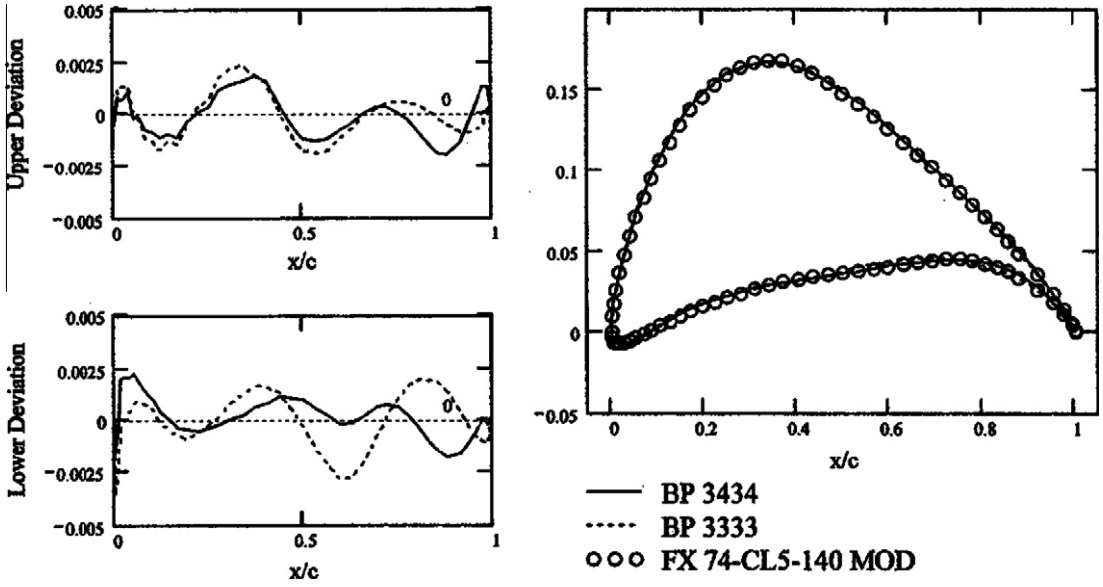

Fig. 6. Representation of the FX 74-CL5-140 MOD airfoil.

approximations required the fewest function evaluations and the BP 3434 approximations required the most.

The biggest limitation of the Bezier approximations is the discontinuous second derivative at the camber and thickness crests which can result in inaccurate flow field simulations. A less significant difficulty is the zero trailing edge thickness which is a problem for some airfoils such as the Eppler E 863.

The BP 3333 approximation has the least control point freedom which can be beneficial in some cases. Because the design space to search is smaller, the optimizer spends less time perusing far from the actual solution. In some cases, e.g. the Eppler E 417, this resulted in BP 3333 finding the best approximation to the target shape. Two classes of airfoil cannot be successfully represented by BP 3333; those with a radical change in the trailing edge camber curve and those with a camber curve that dips below the *x*-axis.

The BP 3434 approximation is the most robust due to the additional control points at the trailing edge. This can be a problem in that it can result in the camber crest being pushed to far back or premature convergence with the shapes with sharp edges of incorrect trailing edge directions.

4. Effect of parameterization on design speed

The previous discussion showed that all three parameterization methods can represent a broad range of airfoils. The next issue of

concern is the effect of the parameterization on the robustness and rate of convergence for an inverse design problem. The focus is narrowed to a comparison between the Bezier and BP 3333 parameterizations. While the solution space of BP 3333 is smaller than that of BP 3434, it also has fewer parameters, and all are aerodynamic. Thus the BP 3333 parameterization should have more potential for accelerated convergence.

The results of four design optimizations will be presented here. In three cases, the optimization was based on finding the geometry that best matches the known pressure distribution for a given airfoil. This allows us to determine the robustness and convergence properties of the optimization process, while ensuring that the optimum has been reached. The final example uses a theoretical pressure distribution, designed to generate a high lift to drag ratio.

4.1. Sample design optimization algorithm

As a sample aerodynamic design problem, an inverse method is used to design fan blades that will exhibit a prescribed pressure distribution. In addition to the blade profile, the stagger angle and blade spacing must also be optimized. See [2] for details. An objective function first calculates the airfoil geometry, then the corresponding pressure distribution, and finally the deviation of this flow from the target distribution, using an L_2-error norm (the least-squares measure of the deviation between the distributions). Optimization is then accomplished by searching among candidate solutions for those that minimize the objective function.

The most effective DE variant for optimization with the Bezier parameterization is $CR = 1$, that is, mutation alone with no crossover. Mutation alone results in trial vector moves that are parallel to the coordinate axes. That is, mutation alone is rotationally invariant. Price et al. [5] have shown that, for epistatic functions, a rotationally invariant algorithm is preferred. Increased crossover rate results in slower convergence. Thus $CR = 1$ is the recommended strategy for problems in which the parameters interact nonlinearly.

The BP parameterization, on the other hand, is designed to reduce epistasis. It should result in a more separable or decomposable objective function, in which each the optimization parameters are more independent. This allows for a different optimization approach. Binary crossover introduces rotation by taking some parameters from the trial vector and some from the parent vector. Values of CR lower than 1 should thus be able to exploit the increased decomposability of the objective function. After some experimentation, it was found that $CR = 0.95$ was an effective parameter. This is used throughout for the BP 3333 solutions.

4.2. Convergence for a 112° cambered turbine blade

The first two test cases have been used by Lewis to validate the numerical flow simulations of the Martensen's surface vorticity panel method [7]. In the first example, the target is a highly cambered impulse cascade profile, based on a 112° circular arc camber. The target pressure distribution was calculated for inlet angle 50°, stagger angle 0°, and pitch/chord ratio 0.5899644. The convergence rate for the Bezier and BP 3333 parameterizations are shown in Fig. 7. Clearly the BP 3333 parameterization is an improvement. It results in a more rapid convergence than the Bezier parameterization, to a solution that more closely approximates the target pressure distribution.

4.3. C4/70/C50 compressor blade

The second example is a blade designed for compressors used in the UK in the 1960s. It has a C4 base profile distributed along a 70° circular arc camber. C-series airfoils are still representative of turbine blading today. The target pressure distribution was calculated

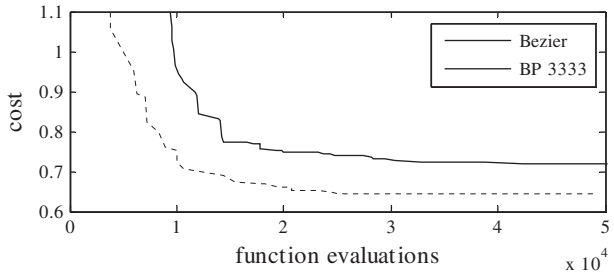

Fig. 7. Effect on convergence to the 112° cambered turbine blade.

with inlet angle −35°, stagger angle 0°, and pitch/chord ratio 0.900364. In this case, the convergence results are essentially comparable (Fig. 8). BP 3333 converges slightly more slowly at the outset, but by 30,000 function evaluations the costs are the same.

4.4. E850 propeller tip

To test the convergence properties further, one airfoil from Section 3 was selected at random for an inverse design. The Eppler E850 airfoil is designed for the tip of a propeller. The target pressure distribution was calculated with inlet angle 5°, stagger angle 0°, and pitch/chord ratio 1. The BP 3333 design converges significantly faster, and by 50,000 function evaluations, the cost is the same for both designs. See Fig. 9.

4.5. Liebeck

For the final example, the solution is not known. The Liebeck [14] pressure distribution family has been demonstrated to generate airfoils with high lift to drag ratios. They avoid separation by a specified, small margin along the pressure recovery region of the suction surface. In principle this recovers the maximum pressure rise over a given distance. The design target has inlet angle 30°, and outlet angle 0°. The BP 3333 design converges sooner and to a lower cost than the Bezier design (Fig. 10).

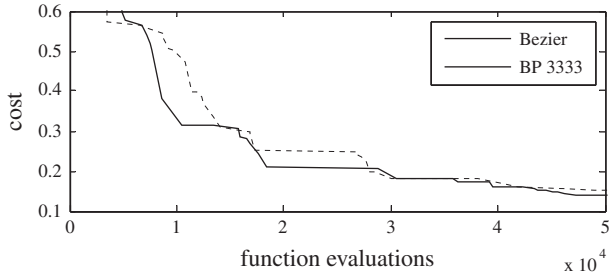

Fig. 8. Effect on convergence to the C4/70/C50 compressor blade.

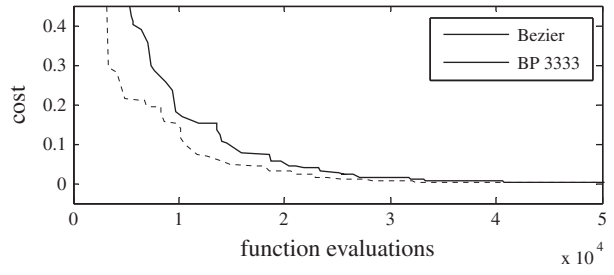

Fig. 9. Effect on convergence to the E850 propeller tip.

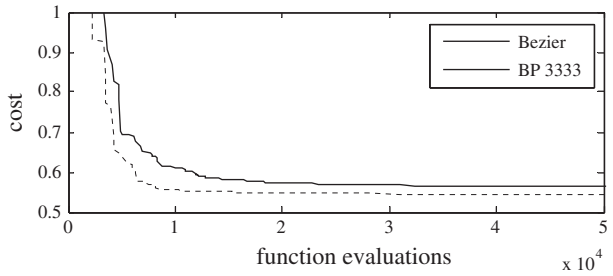

Fig. 10. Effect on convergence to the Liebeck pressure distribution.

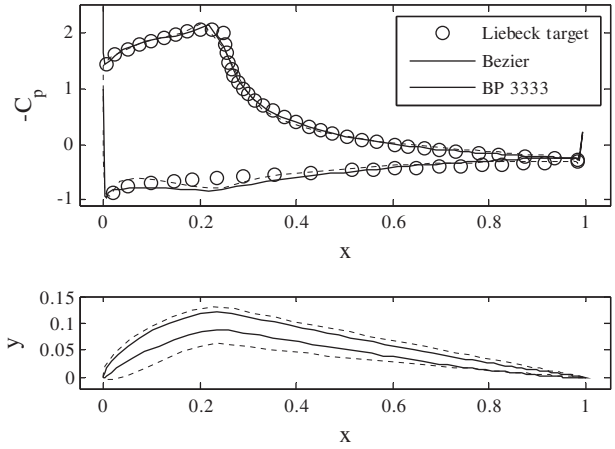

Fig. 11. Comparison of Liebeck designs.

The geometries found are also slightly different. The BP 3333 shape is thinner, with a higher curvature on the lower surface (Fig. 11). In initial trials, the Bezier design converged to a shape with a leading edge cusp. To avoid this, constraints had to be placed on the Bezier control points at the leading edge of the thickness distribution, resulting in the shape shown. This kind of ad hoc constraining of the curve was unnecessary for the BP 3333 design. In addition to converging faster, the BP 3333 parameterization is able to avoid a sharp leading edge simply by virtue of the new leading edge radius parameter.

4.6. Summary

In three of the four inverse design cases, BP 3333 outperforms the Bezier parameterization, with accelerated convergence to a better design. These include the highly cambered turbine blade, the E850 propellor tip, and the inverse Liebeck design. For these three cases, the Bezier designs require between 40,000 and 50,000 function evaluations to converge, which is consistent with the results reported in [2]. The BP 3333 parameterization, on the other hand, requires only 20,000–30,000 function evaluations, providing an acceleration on the order of double speed. In the fourth case, the performance for the C4/70/C50 target is very similar. The two parameterizations find roughly the same shape using roughly the same NFE's, both requiring about 50,000 function evaluations for full convergence.

5. Conclusions

The three parameterization methods are all capable of representing a wide range of airfoil shapes with a slight edge going to BP 3434. However, the BP 3333 parameterization performed better, and certainly no worse than the Bezier parameterization for inverse design.

Overall, then, BP 3333 is a definite improvement over the Bezier parameterization. It is more separable, is more closely linked to the aerodynamics of the shape, and has fewer parameters, all of which increase its potential for further acceleration of the optimization algorithm. Even without this acceleration, convergence is at least comparable, and in three of four cases significantly better. It has improved continuity characteristics over the Bezier parameterization. It has the ability to steer the design away from undesirable features. Also, constraints become much easier to envision and impose due to the aerodynamic nature of the variables. For example, a minimum leading edge radius or minimum crest curvature can be imposed to avoid sharp corners. Structural constraints such as a minimum wedge angle are similarly easy to achieve.

While BP 3434 is more robust than BP 3333, it also uses more parameters, and some of these will contribute nonlinearity to the objective function. BP 3333 thus has more potential for acceleration. Furthermore, by avoiding sharp edges, BP 3333 has the ability to focus the search within regions of acceptable aerodynamic shapes. It is unable to represent a small percentage of airfoils, but this shortcoming is overcome by its other advantages.

An interesting result is the discrepancy in the convergence speed for the two different kinds of optimization tasks. Matching the geometry (Section 3) typically required fewer than 10,000 function evaluations. Finding a shape that matches a pressure distribution (Section 4) required between 20,000 and 50,000 function evaluations. The first task is a more direct optimization problem, and is simpler for DE to perform. The second is indirect, and is more difficult.

References

[1] Abbot IH, von Doenhoff AE. Theory of wing sections. New York: Dover Publications; 1959.
[2] Rogalsky T, Derksen RW, Kocabiyik S. Differential evolution in aerodynamic optimization. Can Aeronaut Space J 2000;46(4):183–90.
[3] Sobieczky H. Parametric airfoils and wings. Recent developments of aerodynamic design methodologies. Braunschweig/Wiesbaden: Friedr. Vieweg & Sohn Verlagsgesellschaft mbH; 1999. p. 71–87.
[4] Chan YY. Applications of genetic algorithms to aerodynamic design. Can Aeronaut Space J 1998;44(3):182–7.
[5] Price K, Storn R, Lampinen J. Differential evolution: a practical approach to global optimization. Berlin: Springer-Verlag; 2005.
[6] Amoiralis EI, Nikolos IK. Freeform deformation versus B-spline representation in inverse airfoil design. J Comput Inform Sci Eng 2008;8(2):024001-1–024001-13.
[7] Lewis RI. Vortex element methods for fluid dynamic analysis of engineering systems. Cambridge: Cambridge University Press; 1991.
[8] Sobieczky H. Manual aerodynamic optimization of an oblique flying wing. In: 36th aerospace sciences meeting & exhibit, Reno, NV. AIAA Paper 98-0598; 1998.
[9] Oyama A, Obayashi S, Nakahashi K. Fractional factorial design of genetic coding for aerodynamic optimization. AIAA Paper 99-3298; 1999.
[10] Bäck T. Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms. Oxford: Oxford University Press; 1996.
[11] Rogalsky T. Acceleration of differential evolution for aerodynamic design. Ph.D. Thesis, University of Manitoba; 2004.
[12] Eppler R. Airfoil design and data. Berlin: Springer-Verlag; 1990.
[13] Selig MS, Guglielmo JJ, Broeren AP, Giguère P. Summary of low-speed airfoil data, vol. 1. Virginia Beach: SoarTech Publications; 1995.
[14] Liebeck RH. A class of airfoils designed for high lift in incompressible flow. J Aircraft 1973;10(10):610–7.