

Review of Aerofoil Parameterisation Methods for Aerodynamic Shape Optimisation

D. A. Masters*

Department of Aerospace Engineering, University of Bristol

N. J. Taylor†

MBDA, Filton

T. C. S. Rendall ‡, C. B. Allen § and D. J. Poole ¶

Department of Aerospace Engineering, University of Bristol

This paper presents a review of aerofoil shape parameterisation methods that can be used for aerodynamic shape optimisation. Six parameterisation methods are considered for a range in design variables: Class function/Shape function Transformations (CST); B-splines; Hicks-Henne bump functions; a domain element approach using Radial Basis functions (RBF); Bézier surfaces; and a singular value decomposition modal extraction method (SVD); plus the PARSEC method. The performance of each method is analysed by considering geometric shape recovery on over 1000 aerofoils using a range of design variables, testing the efficiency of design space coverage. A more in-depth analysis is then presented for three aerofoils, NACA4412, RAE2822 and ONERA M6 (D section), with geometric error and convergence of the resulting aerodynamic properties presented. In the large scale test it is shown that, for all the methods, a large number of design variables are needed to achieve significant design space coverage. For example at least 25 design variables are needed to cover 80% of the design space regardless of the method used; this is often higher than is desired for two-dimensional studies, suggesting that further work may be required to reduce the number of design variables needed.

I. Introduction and Background

With optimisation becoming more common in aerodynamic design, a significant effort is being made to improve both its effectiveness and its efficiency[1, 2]. A typical aerodynamic optimisation procedure would consist of four stages: shape parameterisation/control; mesh creation/deformation; flow solution; and optimisation. This paper focuses on the efficiency and effectiveness of the shape parameterisation stage, in particular for two-dimensional aerofoil design. Shape parameterisation concerns the way the geometry is handled and deformed by the optimisation algorithm, determining both the fidelity and range of control available. A successful parameterisation method is characterised by the ability to cover a large design space with a limited set of design variables. This paper presents a review of widely-used shape parameterisation methods, by considering geometric shape recovery and analysing how accurately each can recover a large range of different aerofoils.

Shape parameterisation methods can be categorised as either constructive or deformative. Constructive methods represent an aerofoil shape based purely on a series of parameters specified, examples of this

*Graduate Student, AIAA Student Member, dominic.masters@bristol.ac.uk, Bristol, BS8 1TR, UK

†Capability Leader, Aerodynamic Tools & Methods, AIAA Senior Member, nigel.j.taylor@mbda-systems.com, WG3, PO Box 5, Filton, Bristol, BS34 7QW, UK

‡Lecturer, AIAA Member, thomas.rendall@bristol.ac.uk, Bristol, BS8 1TR, UK

§Professor of Computational Aerodynamics, AIAA Senior Member, c.b.allen@bristol.ac.uk, Bristol, BS8 1TR, UK

¶Graduate Student, AIAA Student Member, d.j.poole@bristol.ac.uk, Bristol, BS8 1TR, UK

include polynomials and splines[3, 4], partial differential equation methods (PDE)[5] and CSTs[6]. Deformative methods take an existing aerofoil then deform it to create the new shape; these include discrete,[7] analytical[8], basis vector[9] and free-form deformation (FFD)[10, 11] methods.

Many aerofoil optimisation methods have used the discrete method, which uses the surface points of a discretely defined aerofoil as the design variables[7]. This allows extremely fine control over the shape with no restriction on the design space. However, as point displacements and sensitivities are considered independently of each other deformations are often not smooth, which can present difficulties for flow solvers. A large number of design variables can also lead to slow convergence rates or ill-conditioned optimisation problems so more robust and efficient parameterisation methods are usually favoured over the discrete method.

Hicks and Henne's[8] early analytical approach based on "bump" functions takes a base aerofoil and then adds a linear combination of single-signed sine functions to deform its upper and lower surfaces to create a new aerofoil shape. This concept of adding a linear combination of simple basis functions to a base shape has also been used in a constructive manner, for instance, Kulfan's CST[6] method which adds a combination of Bernstein polynomials to a simple, analytical "aerofoil class" shape. Both of these methods have seen frequent use within the framework of aerodynamic optimisation.

Other more classical methods such as polynomial fitting or B-splines have also been used extensively. Sobieczky's [12] PARSEC (Parameterised Sections) method is popular, approximating each surface by a 6th order polynomial. One positive feature of this method is that it uses real geometric properties such as the aerofoil's crest location and curvature as the design variables, allowing more intuitive control of the shape. However, as the method is limited to only 12 design variables it does not provide the range of fidelity made available by many of the alternative methods.

Attempts have also been made to derive a set of unique modes that can represent an aerofoil in either a constructive or deformative manner. This is typically done through proper orthogonal decomposition (POD) of a set of training aerofoils which will create a set of orthogonal shape modes equal to the number of aerofoils in the collection. This set can then be reduced by only using the most dominant modes. Studies of this nature have been produced by Toal *et al.*[13], Ghoman *et al.*[14] and then by Poole *et al.*[15, 2] who used a large, varied collection of aerofoils and singular value decompositions to produce a universal set of modes representing the deformation of aerofoil shapes.

Another approach to shape parameterisation is to use free-form deformation (FFD) which is typically a method used in soft object animation (SOA). This creates a smooth continuous volume transformation based on the change in position of a series of control points. This volume transformation can also be used to deform computational volume meshes seamlessly with the aerofoil. This can have significant cost benefits particularly in three dimensions. The two principle FFD techniques in use are radial basis functions (RBFs) applied on an arbitrary domain element (a series of user positioned initial control points), and Bézier surfaces (often referred to as just "FFD"), using a structured lattice of initial control points. These control points are commonly grouped together to create global transformations such as thickness and camber, or twist and sweep in three dimensions, to reduce the total number of design variables. The RBF domain element method also allows the local fidelity of movement to be controlled through the proximity and density of the point distribution. Both of these methods have shown promising optimisation results[16, 17, 18, 11].

Previous parameterisation method comparisons include that by Castonguay and Nadarajah[19] who considered the discrete method, Hicks-Henne, B-splines and PARSEC for inverse design and drag minimisation on an ONERA M6 (D section) aerofoil. They found, for the inverse design test case, that the B-spline needed 32 control points (64 design variables) to obtain a satisfactory fit for the geometry but only 16 control points (32 design variables) to get a satisfactory fit on the pressure distribution. They used a fixed bump position, fixed bump width configuration of Hicks-Henne and found they needed 32 bumps to get a satisfactory fit for both the geometry and the pressure. B-splines, for equal design variables, always out-performed Hicks-Henne, and PARSEC failed to approximate the ONERA M6 to a useful degree of accuracy. For the drag minimisation test case both the B-splines and Hicks-Henne bumps gave comparable results.

This study was then followed up by Mousavi, Castonguay and Nadarajah[20] who also performed an inverse design case on the ONERA M6 aerofoil and a drag minimisation (though on a NACA0012) but instead used the mesh point, B-spline and CST parameterisation methods. They found very similar results for B-splines as Castonguay[19] and found that the CST method gave good results with 22 design variables (comparable to the best B-spline cases) but then decreased in accuracy with additional design variables due to the influence of high frequency oscillations. The drag optimisation showed comparable results across both

methods and all orders of accuracy with the CST method using only 10 design variables attaining a 40% drag reduction.

Sripawuadkul *et al.*[21] compared the Hicks-Henne, B-spline, PARSEC and CST methods, though instead of looking at individual test cases rated each method out of four across five different criteria: parsimony; completeness; orthogonality; flawlessness; and intuitiveness. Over these criteria, they rated the CST method highest, followed by PARSEC, B-splines and Hicks-Henne bumps.

The study presented here aims to add to these previous studies by including a larger range of methods and testing them over an extensive database of aerofoils. The motivation behind using a wide range of aerofoils is firstly to create a significantly broad test for the parameterisation methods, but also to ensure that results can be generalised with confidence to other aerofoil shapes. Selected aerofoil case studies have also been included to highlight specific traits of the methods found for particular shape transformations.

II. Parameterisation Methods

Aerofoil parameterisation methods fall loosely into two categories; deformative and constructive methods. Deformative methods take an existing aerofoil shape and deform it to create the new shape whereas constructive methods define the new aerofoil shape purely from the design variables. The methods considered here are presented below, and to unify the analysis, all methods are formulated such that $z = f(x)$ only.

A. Constructive Methods

1. Class Function/ Shape Function Transformations (CST)

CSTs were developed by Kulfan[22, 6] primarily as a method of defining a wide range of aerofoils with relatively few design variables; however, the method can also be extended to other shapes such as square-like and circle-like objects and three-dimensional surfaces. It is defined as

$$z_{upper} = C_{N2}^{N1}(x) \cdot S_{upper}(x) + x \cdot \Delta z_{upper}, \quad (1)$$

$$z_{lower} = C_{N2}^{N1}(x) \cdot S_{lower}(x) + x \cdot \Delta z_{lower} \quad (2)$$

where the class function

$$C_{N2}^{N1}(x) = x^{N1} \cdot (1 - x)^{N2}, \quad (3)$$

Δz defines the trailing edge thickness and $x \in [0, 1]$.

The configuration $N1 = 0.5$ and $N2 = 1$ creates the aerofoil class but, for example, $N1 = 0.01$, $N2 = 0.01$ will create a rectangle or $N1 = 0.5$, $N2 = 0.5$ will create an ellipsoid. The surface function $S(x) \in [0, 1]$ is then defined to create the desired shape. In particular, Kulfan[6] highlighted, for the aerofoil class, that the leading edge radius (r_{le}), the boat-tail angle (β) and the trailing edge thickness (Δz) are directly related to the bounding values of $S(x)$ by the relations

$$S(0) = \sqrt{2r_{le}} \quad \text{and} \quad S(1) = \tan \beta + \Delta z. \quad (4)$$

Kulfan[22] suggested defining $S(x)$ as the linear combination of Bernstein polynomials i.e.

$$S(x) = \sum_{i=0}^n a_i b_{i,n}(x) \quad (5)$$

where

$$b_{i,n}(x) = \binom{n}{i} x^i (1 - x)^{n-i}, \quad (6)$$

a_i is the Bernstein coefficient and n is the degree of the polynomials.

Bernstein polynomials are mathematically equivalent to the set of standard polynomials of the form $a_i x^i$. One advantage of this is that it implies that the Stone-Weierstrass Theorem[23] holds. This implies the existence of an approximation to any aerofoil topologically similar to the “base” aerofoil to any tolerance; it, however, makes no guarantee on the number of design variables required.

Kulfan[24] later presented a leading edge modification (LEM) to the CST method, including an extra polynomial and coefficient, to improve the approximation of aerofoils with large leading edge camber. This proposed adding an additional shape term such that

$$S(x) = \sum_{i=0}^n a_i b_{i,n}(x) + a_{n+1} x^{0.5} (1-x)^{n-0.5}. \quad (7)$$

2. SVD Method

The Singular Value Decomposition (SVD) method uses POD to derive a set of ordered, orthogonal basis modes from a set of pre-determined training aerofoils. New aerofoil shapes are then constructed as a linear combination of these modes where the fidelity in construction is determined by the number of modes used. This technique was first employed by Toal *et al.*[13] then by Ghoman *et al.*[14] and Poole *et al.*[15]. The design space and fidelity of this method is wholly defined by the training aerofoils used. Ghoman *et al.*[14] used a series of supercritical aerofoils to derive the modes and showed that other supercritical aerofoils could efficiently be reconstructed. Poole *et al.*[15] then extended this to show that a broad range of aerofoils could be represented given a wide choice of training aerofoils.

It should be noted that, in the previous work of Poole *et al.*[15], the SVD method was used to create a set of deformation modes to deform an initial aerofoil shape. Alternatively, the modes shapes can be constructed from the absolute aerofoil shape, and this is the approach used here.

In order to construct the aerofoil shape modes it is crucial that the training library is normalised such that the aerofoil shapes are defined in the same way and discretized by the same number of points. In this work they are transformed to have a sharp trailing edge and discretized such that all the aerofoils have an equal distribution of points along the x -axis. This second condition means that the modes only need to be constructed in the z direction. The z coordinates of the training aerofoils, translated around the mean \bar{z} such that $\hat{z} = z - \bar{z}$, then form the rows of the matrix

$$M = \begin{bmatrix} \hat{z}_1^1 & \hat{z}_2^1 & \cdots & \hat{z}_n^1 \\ \hat{z}_1^2 & \hat{z}_2^2 & \cdots & \hat{z}_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ \hat{z}_1^m & \hat{z}_2^m & \cdots & \hat{z}_n^m \end{bmatrix} \quad (8)$$

for m training aerofoil each of length n .

This matrix can then be decomposed into the singular value decomposition

$$M = U \cdot S \cdot V \quad (9)$$

Where the columns of $V = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_{\min(n,m)}]$ represent the ordered, orthogonal aerofoil modes.

New aerofoils are then constructed as linear combination of these modes such that

$$\mathbf{z} = \bar{\mathbf{z}} + z_{TE} \mathbf{x} + \sum_i a_i \mathbf{v}_i \quad (10)$$

where a_i represents the design variables and z_{TE} represents the trailing edge thickness. It should also be noted that in the work of Poole *et al.*[15], all aerofoils in the library were parameterised in terms of surface arc length, not the streamwise coordinate used here.

3. B-Splines

B-splines are a widely used method for producing piecewise polynomial curves. Much like many of the other parametrisation methods, B-splines rely on a set of basis functions, however in this case, the coefficients are defined spatially by a set of discrete control points $\mathbf{P}_i \in \mathbb{R}^3$. The curve, parametrised by the scalar $u \in [u_0, u_m]$, is given by

$$C(u) = \sum_{i=0}^{n-1} N_{i,p}(u) \mathbf{P}_i \quad (11)$$

where the $n = m - p$ basis functions of order p (Note: p must be less than or equal to the number of control points) are given by

$$N_{i,0}(u) = \begin{cases} 1 & u_i \leq u < u_{i+1}, \\ 0 & \text{otherwise} \end{cases}, \quad (12)$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u), \quad (13)$$

with the increasing knot vector

$$U = [u_0, \dots, u_m], \quad u_i \leq u_{i+1}. \quad (14)$$

The knot vector then takes the form

$$U = [\underbrace{u_0, \dots, u_0}_{p+1}, \underbrace{u_{p+1}, \dots, u_{n-1}}_{p+1}, \underbrace{u_m, \dots, u_m}_{p+1}] \quad (15)$$

where the values u_{p+1}, \dots, u_{n-1} indicate the points where the basis functions meet; if u_p, \dots, u_n are equally distributed and the B-spline is said to be “uniform”. In the special case where $p = n$ the B-splines are called “Bézier Curves”, furthermore if $u \in [0, 1]$ the basis functions are the Bernstein polynomials of order $p - 1$.

A useful property of B-splines is that the basis order p has a large impact on the locality of the influence of the control points meaning that for a low order curve the influence of any change in control point position will be very localised compared to if a high order curve was used. However, using low order curves can make maintaining smoothness difficult.

B-splines can be used to represent aerofoils in a variety of different ways but in order to satisfy the constraints imposed by the other parameterisation methods the following configuration has been used. Each aerofoil is represented by two B-splines, for each B-spline \mathbf{P}_0 is fixed at the leading edge $(0, 0)$, \mathbf{P}_{n+1} is at the trailing edge and \mathbf{P}_1 is aligned vertically with the leading edge. The other points \mathbf{P}_i are distributed on a half cosine scale between $(0, 1)$ in the chord-wise direction and only allowed to vary in the vertical direction, i.e.

$$\mathbf{P}_0 = (0, 0), \quad \mathbf{P}_i = \left(\frac{1}{2} \left[1 - \cos \left(\frac{\pi(i-1)}{n+1} \right) \right], a_i \right), \quad \mathbf{P}_{n+1} = (1, z_{te}), \quad (16)$$

where a_i denotes a design variable and z_{te} the z -wise component of the trailing edge position. Figure 1 shows an example of this configuration.

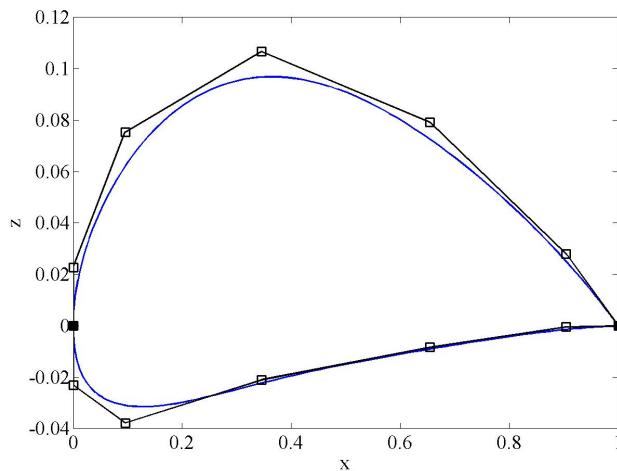


Figure 1. A cubic B-spline approximation of a NACA4412 with 10 design variables. The filled squares represent fixed control points.

4. PARSEC

The PARSEC method was developed by Sobieczky[12] as a system to create analytically defined aerofoils based on meaningful properties such as the upper/lower crest position, max thickness, leading edge radius and boat-tail angle. This was done by proposing that the upper and lower surfaces should be defined by the following 6th order polynomials:

$$z_{up}(x) = \sum_{i=1}^6 a_i x^{i-0.5}, \quad (17)$$

$$z_{lo}(x) = \sum_{i=1}^6 b_i x^{i-0.5}. \quad (18)$$

Twelve equations, subject to twelve free parameters that define the characteristics of an aerofoil, were then defined with the resulting system solved for a_i, b_i for $i = 1 : 6$. These parameters are shown in figure 2 and their definitions are given in table 1. This method is therefore dependant on twelve design variables though this is reduced to ten if the trailing edge position is defined.

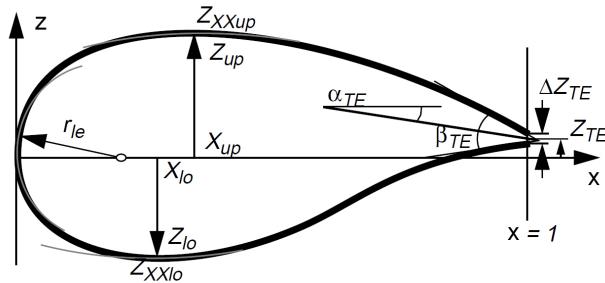


Figure 2. A geometric representation of the parameters used for PARSEC method[4].

Parameter	Definition
Upper leading edge radius (r_{leu})	$r_{leu} = a_1$
Lower leading edge radius (r_{lel})	$r_{lel} = b_1$
Upper crest point (Z_{up})	$Z_{up} = z_{up}(X_{up})$
Lower crest point (Z_{lo})	$Z_{lo} = z_{lo}(X_{lo})$
Position of upper crest (X_{up})	$z'_{up}(X_{up}) = 0$
Position of lower crest (X_{lo})	$z'_{lo}(X_{lo}) = 0$
Upper crest curvature (Z_{XXup})	$z''_{up}(X_{up}) = Z_{XXup}$
Lower crest curvature (Z_{XXlo})	$z''_{lo}(X_{lo}) = Z_{XXlo}$
Trailing edge offset (Z_{TE})	$z_{lo}(1) = Z_{TE}$
Trailing edge thickness (ΔZ_{TE})	$z_{up}(1) = Z_{TE} + \Delta Z_{TE}$
Trailing edge angle (α_{TE})	$z'_{up}(1) = -\tan(\alpha_{TE} + \frac{\beta_{TE}}{2})$
Boat-tail angle (β_{TE})	$z'_{lo}(1) = -\tan(\alpha_{TE} - \frac{\beta_{TE}}{2})$

Table 1. The parameters used for the PARSEC method and their definitions.

B. Deformative Methods

1. Hicks-Henne Bump Functions

Hicks-Henne bump functions use a base aerofoil definition plus a linear combination of a set of n basis functions defined between 0 and 1 to determine the final aerofoil shape. Each surface is defined by

$$z = z^{base} + \sum_{i=0}^n a_i \phi_i(x) \quad (19)$$

for basis functions $\phi_i(x)$ and coefficients a_i for $i = 1, \dots, n$.

One set of possible basis functions proposed by Hicks and Henne[8] were the sine functions defined by

$$\phi_i(x) = \sin^{t_i}(\pi x^{m_i}), \quad (20)$$

$$m_i = \ln(0.5)/\ln(x_{M_i}) \quad (21)$$

where x_{M_i} is the location of the maxima of the basis function and t_i controls the width of the functions. Each bump function is therefore defined by three variables, each of which can be optimised or fixed. Various combinations of fixing and optimising these variables have been performed, for example Wu[25] opted to fix $t_i = 4$ and let

$$x_{M_i} = \frac{1}{2} \left[1 - \cos \left(\frac{i\pi}{n+1} \right) \right], \quad i = 1, \dots, n. \quad (22)$$

whereas Khurana[26] varied all three variables.

For this study a_i will be varied and M_i will take the values from equation 22 and $t_i = \text{constant}$ will be determined later.

2. Radial Basis Function Domain Element

The parameterisation labelled RBF here is a domain element approach, using radial basis function interpolation to transfer the deformations of a set of domain element control points to smooth deformations of the aerofoil. The general theory of RBFs is outlined by Wendland[27] and Buhmann[28] and its use as a parametrisation technique is presented extensively in Rendall and Allen[10] and Morris[16].

The general solution for the interpolated control point deformation vector, $\Delta\mathbf{x}$, is given by

$$\Delta\mathbf{x} = \sum_{i=1}^N \beta_i \phi(\|\mathbf{x} - \mathbf{x}_i\|) + p(\mathbf{x}) \quad (23)$$

where i indicates the i th control point, \mathbf{x}_i its centre and β_i its coefficient. $p(\mathbf{x})$ is a linear polynomial used to ensure that translation and rotation are captured without added shape deformation.

The coefficients β_i are found by requiring the exact recovery of the original function when the control points are in their original positions. The system is then completed by the additional requirement

$$\sum_{i=1}^N \beta_i q(\mathbf{x}) = 0 \quad (24)$$

where $q(\mathbf{x})$ is a polynomial with order less than or equal to $p(\mathbf{x})$.

When a discrete set of points in the original domain is to be transformed the problem can be formulated with matrix multiplication. Exact recovery of original points implies that

$$\Delta X_{DE} = C\Lambda \quad (25)$$

where

$$\Delta X_{DE} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \vdots & \vdots \\ \Delta x_{DE_1} & \Delta z_{DE_1} \\ \vdots & \vdots \\ \Delta x_{DE_N} & \Delta z_{DE_N} \end{pmatrix}, \quad \Lambda = \begin{pmatrix} \gamma_0^x & \gamma_0^z \\ \gamma_x^x & \gamma_x^z \\ \gamma_z^x & \gamma_z^z \\ \beta_{DE_1}^x & \beta_{DE_1}^z \\ \vdots & \vdots \\ \beta_{DE_N}^x & \beta_{DE_N}^z \end{pmatrix} \quad (26)$$

and

$$C = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & \cdots & 1 \\ 0 & 0 & 0 & x_{DE_1} & x_{DE_2} & \cdots & x_{DE_N} \\ 0 & 0 & 0 & z_{DE_1} & z_{DE_2} & \cdots & z_{DE_N} \\ 1 & x_{DE_1} & z_{DE_1} & \phi_{DE_1 DE_1} & \phi_{DE_1 DE_2} & \cdots & \phi_{DE_1 DE_N} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{DE_N} & z_{DE_N} & \phi_{DE_N DE_1} & \phi_{DE_N DE_2} & \cdots & \phi_{DE_N DE_N} \end{pmatrix} \quad (27)$$

with

$$\phi_{DE_1 DE_2} = \phi(\|\mathbf{x}_{DE_1} - \mathbf{x}_{DE_2}\|) \quad (28)$$

indicating the basis function on the distance between DE_1 and DE_2 and the subscript DE representing a domain element control point.

To locate the deformed points the following matrix must be formed where the subscript a indicates the original position of an aerodynamic point:

$$A = \begin{pmatrix} 1 & x_{a_1} & z_{a_1} & \phi_{a_1 DE_1} & \phi_{a_1 DE_2} & \cdots & \phi_{a_1 DE_N} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{a_N} & z_{a_N} & \phi_{a_N DE_1} & \phi_{a_N DE_2} & \cdots & \phi_{a_N DE_N} \end{pmatrix}. \quad (29)$$

The displacement of aerodynamic points, given by the rows of matrix X_d , are then calculated as

$$\Delta X_d = A\Lambda \quad (30)$$

$$= AC^{-1}\Delta X_{DE} \quad (31)$$

$$= H\Delta X_{DE}. \quad (32)$$

Note that as H is invariant of the current control point positions it only needs to be calculated once.

The design space of this method is dependent on the configuration and initial position of the domain element control points. For this study, unless specified otherwise, each control point will act independently of one another and, to reduce the total number of design variables and ensure solutions satisfy the constraints outlined in section III, only variations in the z direction will be allowed. This means that the number of control points will equal the number of design variables.

The initial control point positions have been specified symmetrically about the x -axis on an ellipse around the initial aerofoil with the first, 6 point, configuration having points located at $x = -0.1, 0.5$ and 1.1 . Additional points are then placed with x locations bisecting two points from the previous configuration, with priority given at the leading edge and then the trailing edge. Figures 3-6 show examples of these configurations.

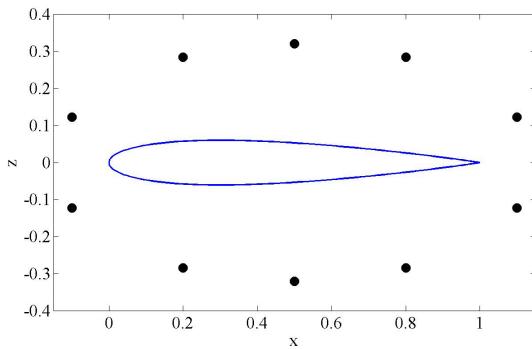


Figure 3. RBF domain element initial configuration for 10 control points.

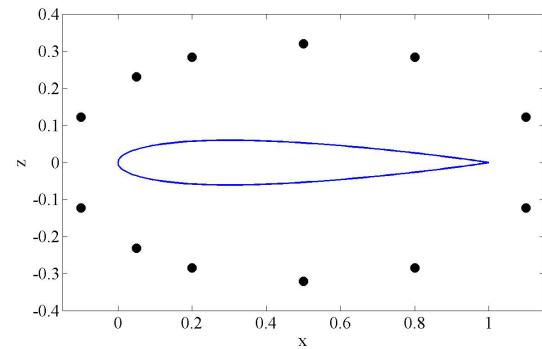


Figure 4. RBF domain element initial configuration for 12 control points.

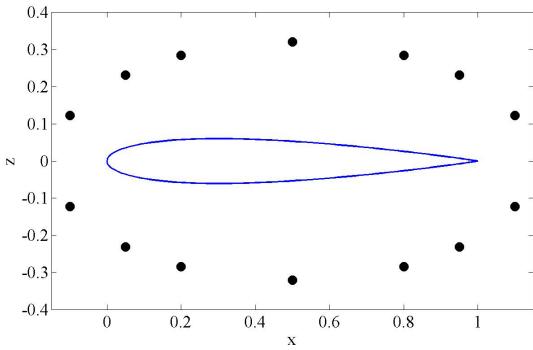


Figure 5. RBF domain element initial configuration for 14 control points.

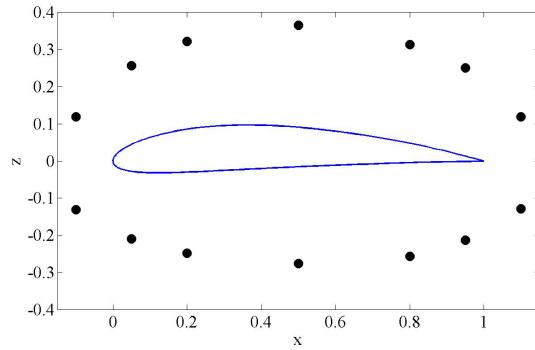


Figure 6. 14 RBF domain element control points deformed to approximate a NACA4412.

3. Bézier Surfaces

A Bézier surface is a B-spline surface of Bézier curves. Although these are usually used to create surfaces in three-dimensional space, they can also be used as a deformation tool in two-dimensional space by constraining the control points to a plane. To create a deformable domain from this surface a rectangular set of $(m + 1) \times (n + 1)$ uniformly spaced control points, \mathbf{P}_{ij} , is placed around an initial aerofoil. It is key that these are uniformly spaced otherwise the initial aerofoil will not be recovered with the original positions of the control points. Given an undeformed domain $\mathcal{A}(x, z) \in [x_{min}, x_{max}] \times [z_{min}, z_{max}]$ we therefore get

$$\mathbf{P}_{ij} = \left(x_{min} + \frac{i}{m} (x_{max} - x_{min}), z_{min} + \frac{j}{n} (z_{max} - z_{min}), 0 \right) \quad (33)$$

for $i = 0, \dots, m$, $j = 0, \dots, n$.

The two-dimensional Bézier surface, $S(u, v)$, spanning the deformed domain $\mathcal{D}(x, z)$, is then given by

$$S(u, v) = \sum_{j=0}^n \sum_{i=0}^m B_{i,m}(u) B_{j,n}(v) \mathbf{P}_{ij} \quad (34)$$

where $u, v \in [0, 1]$ and $B_{i,m}$ are Bernstein polynomials.

Now, to create the one-to-one deformation mapping required the undeformed domain \mathcal{A} must be normalised to the unit domain $\mathcal{N} \in [0, 1] \times [0, 1]$ by the transformation

$$s(x) = \frac{x - z_{min}}{z_{max} - z_{min}}, \quad t(z) = \frac{z - x_{min}}{x_{max} - x_{min}}. \quad (35)$$

This implies that the required deformation transformation, from $\mathcal{A}(x, z) \rightarrow \mathcal{D}(x, z)$, is given by

$$\mathbf{X}(x, z) = \sum_{j=0}^n \sum_{i=0}^m B_{i,m}(s(x)) B_{j,n}(t(z)) \mathbf{P}_{ij}. \quad (36)$$

Similarly to the treatment of the RBF design variables, in this case, only variations in the z direction are allowed to reduce the number of design variables and to ensure the constraints outlined in section III are satisfied.

III. Large Aerofoil Database and Geometry Handling

An extensive aerofoil database^a collated by University of Illinois has been used for the large scale testing in this paper. It contains a wide range of aerofoils used for both fixed wing and rotorcraft applications. The motivation behind using a wide range of aerofoils is firstly to create a sufficiently broad test for the parameterisation methods but also to ensure that results can be generalised with confidence to other aerofoil shapes.

^ahttp://aerospace.illinois.edu/m-selig/ads/coord_database.html accessed 5th June 2014.

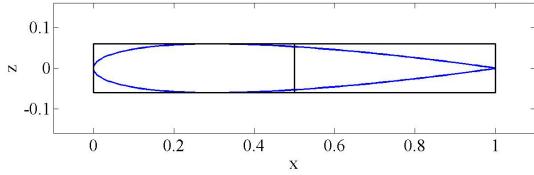


Figure 7. Bèziersurface initial lattice for 2×3 configuration.

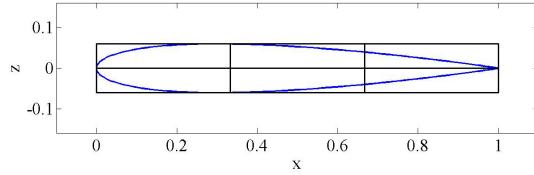


Figure 8. Bèziersurface initial lattice for 3×4 configuration.

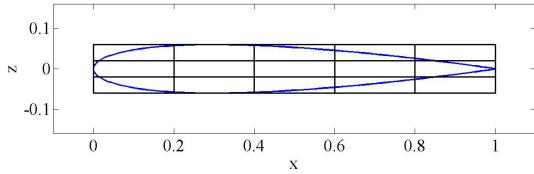


Figure 9. Bèziersurface initial lattice for 4×6 configuration.

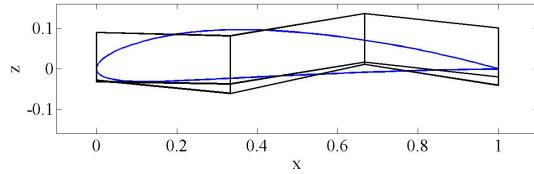


Figure 10. 3×4 Bèziersurface deformed to approximate a NACA4412.

The original database contained 1533 aerofoils defined by between 24 and 205 points; however, some aerofoils were removed as they represented shapes not easily attained by the parameterisation methods being investigated. The results presented here are based on a database of 1064 aerofoils. The aerofoils were then splined and normalised such that all the aerofoils were defined “like-for-like” to remove any definition based bias and satisfy the inherent constraints of the parameterisation methods. To do this each aerofoil was normalised such that

$$\mathbf{x}_i = (x_i, z_i), \quad \text{for } i = 1 : 301 \quad (37)$$

$$\mathbf{x}_{le} = \mathbf{x}_{151} = (0, 0), \quad (38)$$

$$\mathbf{x}_{te}^{upper} = \mathbf{x}_{301} = (1, 0), \quad (39)$$

$$\mathbf{x}_{te}^{lower} = \mathbf{x}_1 = (1, z_1), \quad (40)$$

$$z_1 \leq 0, \quad (41)$$

$$x_i = \left(1 - \cos \left(\pi \left(\frac{i-1}{150} - 1 \right) \right) \right) / 2, \quad (42)$$

$$\|\mathbf{x}_{le} - \mathbf{x}_{te}^{upper}\| \geq \|\mathbf{x}_i - \mathbf{x}_{te}^{upper}\| \quad \forall i, \quad (43)$$

where \mathbf{x}_i denotes the i th point of the aerofoil and the subscripts le and te denote the leading and trailing edges, respectively.

A selection of three aerofoils has also been chosen for a set of case studies; NACA4412, RAE2822[29] and ONERA M6 (D section)[30]; these have also been normalised in this fashion.

IV. Testing Methodology

The objective of this study is to analyse the ability of each parameterisation method to approximate a variety of different aerofoils for varying degrees of fidelity. For each individual test the best approximation has been acquired through a least squares solution, from which errors in the geometry and curvature can then calculated.

The least squares solution $\mathbf{x}_i^{approx} = (x_i^{approx}, z_i^{approx})$ is the solution that minimises the expression

$$\frac{1}{N} \sum_{i=1}^N w_i (z_i^{target} - z_i^{approx})^2 \quad (44)$$

for some set of weights w_i given that $x_i^{target} = x_i^{approx} = x_i$ for $i = 1 : N$.

For each of the parameterisation methods z_i^{approx} can be expressed in the form of a linear combination

$$F\mathbf{a} = \mathbf{z}^{approx} \quad (45)$$

where F is a matrix defined by the method and \mathbf{a} is a column vector of the design variables. The weighted least squares solution can then be calculated as

$$\mathbf{a} = (WF)^+ W \mathbf{z}^{target} \quad (46)$$

where the superscript $+$ denotes the Moore-Penrose pseudoinverse and W is a diagonal matrix with $W_{ii} = w_i$.

To obtain measurable, comparable results it is important that results are processed into well defined error metrics and it is imperative that these metrics reflect the aims of the experiment; in this case the ability of shape parameterisations methods to replicate different aerofoil shapes. In a study by Kulfan[6] confined to the CST method the geometric errors in the solution aerofoils were frequently compared to the wind tunnel tolerance

$$z^{error} < \begin{cases} 4 \times 10^{-4} & \text{if } x/c < 0.2 \\ 8 \times 10^{-4} & \text{if } x/c > 0.2 \end{cases} \quad (47)$$

with solutions deemed acceptable if they were within this range. Kulfan then showed for two cases, the RAE2822 and NASA NSC 2-0714, that L/D converged to within $\approx 2\%$ for errors within this bound. This suggests that a tighter tolerance may been needed if full L/D convergence is required. Throughout the testing in this paper this tolerance (equation 47) is used as a benchmark for a successful approximation. This tolerance also motivates using the least squares weighting

$$w_i = \begin{cases} 2 & \text{if } x_i < 0.2 \\ 1 & \text{if } x_i > 0.2 \end{cases} \quad (48)$$

to provide a tighter fit in the leading edge area.

The six parameterisation methods have been tested in two stages. First a large scale test was performed on the UIUC aerofoil database to test the performance of each method across a broad spectrum of aerofoils. Then, three specific aerofoil case studies were performed.

For the case studies, errors in the geometry are again considered, and lift and drag, calculated using XFOIL[31], analysed. XFOIL was used for convenience due to the large number of computations performed and was run under the conditions $Re = 10^7$, $M = 0.2$ and $\alpha = 0$.

The geometry error is presented in terms of the RMS error in the z direction:

$$RMS_Z = \left(\frac{1}{N} \sum_{i=1}^N (z_i^{target} - z_i^{approx})^2 \right)^{1/2}. \quad (49)$$

For the large database tests, the weighted maximum z error has been calculated, where this is defined as:

$$Error_z = \max_i |w_i(z_i^{target} - z_i^{approx})| \quad (50)$$

The advantage of using the weighted maximum z error is that it encompasses the importance of better approximation at the leading edge but also a value of $Error_z = 8 \times 10^{-4}$ corresponds directly to the Kulfan wind tunnel tolerance.

These errors are then presented in terms of the percentage of the aerofoil database that can be approximated to a prescribed tolerance. By presenting the results in this manner, for a given design variable interval, the chance that an unknown aerofoil can be approximated to a specific tolerance can be estimated.

For all the deformative methods a NACA0012 aerofoil has been used as the initial aerofoil shape. However, due to the inclusion of blunt trailing edge aerofoils in this study a modification has been made such that

$$z_{lo}^{N0012*} = z_{lo}^{N0012} + x z_{te}^{target}. \quad (51)$$

V. Method Implementation Analysis

For each method there are choices that must be made regarding the set up of the design variables. In a this section a variety of different implementations have been considered for each method in an attempt to

obtain the best possible results, where the possible implementations are compared through their performance in the large database tests.

For the RBF method three different implementations have been compared. The first, labelled “RBF (simple)” takes the initial control points defined on an ellipse as in section 2. The second, labelled “RBF (w/ x-variation)” uses the same initial configuration, but has additional fixed points at the leading and trailing edges and allows some x variation to the control points. The third method, labelled “RBF (optimised)” uses a set of optimised initial control points. These have been obtained using a gravitational search algorithm[32] that maximises the number of aerofoils in the large database that can be approximated to within the typical wind tunnel tolerances. It should be noted that due to the computational expense of these optimisations this method has not been tested across as large a range of design variables as the other methods.

Figure 11 compares the percentage of the large aerofoil database that can be approximated to within the wind tunnel tolerance for each of the three variations. For this criteria the optimised implementation performs best for all but one point at 14 design variables, the RBF method will therefore be implemented in this way in the results section.

The implementation choice that must be made for the Bézier surface method is the size of the control point lattices to use. All $n \times m$ lattices for $n = 2 : 7$ and $m = 2 : 15$ have been tested with figure 12 showing the results (where m is on the horizontal axis). The starred points represent the configurations that have been used for further testing.

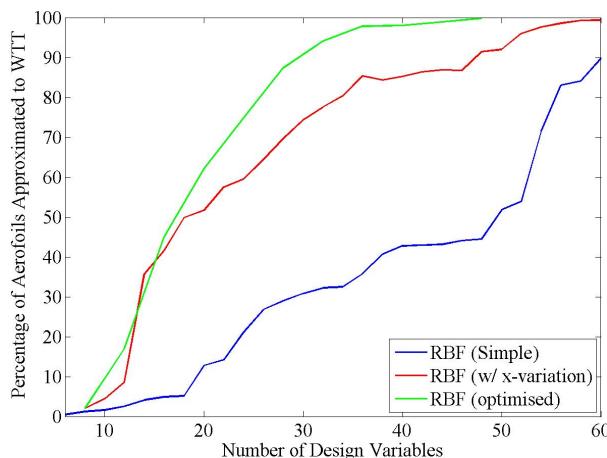


Figure 11. Percentage of aerofoil database approximated to within the typical wind tunnel tolerance for three possible RBF method implementations.

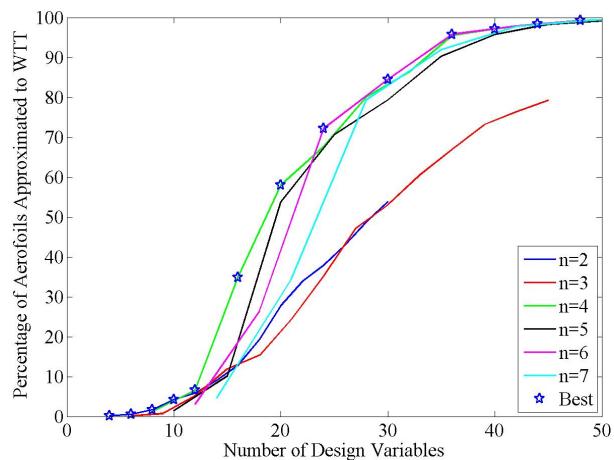


Figure 12. Percentage of aerofoil database approximated to within the typical wind tunnel tolerance for various Bézier Surface implementations.

For the CST method two options were considered, either the original method specification, or with the leading edge modification (LEM). Figure 13 shows the comparison between these two methods with the LEM showing a significant boost in accuracy.

For the Hicks-Henne method, the variable t_i controls the thickness of each bump. Though they can be varied as design variables to keep the problem linear they have been predetermined and set such that $t_i = t \forall i$. Figure 14 shows the area spanned for the results as t is varied from 0.1 to 2. $t = 1$ has been chosen as the best result.

For the B-spline method various orders of spline have been tested with the results shown in figure 15. It shows that for third, fourth and fifth-order B-splines the accuracy is very similar for over ≈ 25 design variables, however for less than that the fifth-order curve is slightly better. For this reason the fifth-order curve has been used. It should be noted that this choice limits the minimum number of design variables possible to eight.

For the SVD method two variations have been considered. The inclusive method and the exclusive method. The inclusive method used the full aerofoil library for both the training of the modes and also the testing. The exclusive method used a random set of 150 aerofoils from the aerofoil database to train the modes and then excludes this set from the testing aerofoils. This removes any bias included as a result of using the same aerofoils for both the training and testing of the method. Figure 16 shows the comparison

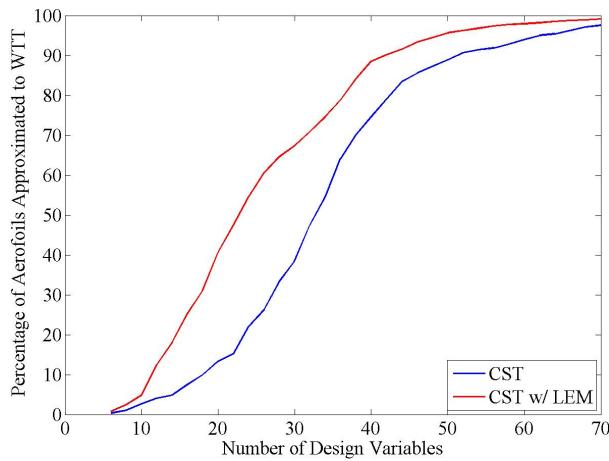


Figure 13. Percentage of aerofoil database approximated to within the typical wind tunnel tolerance for two possible CST method implementations.

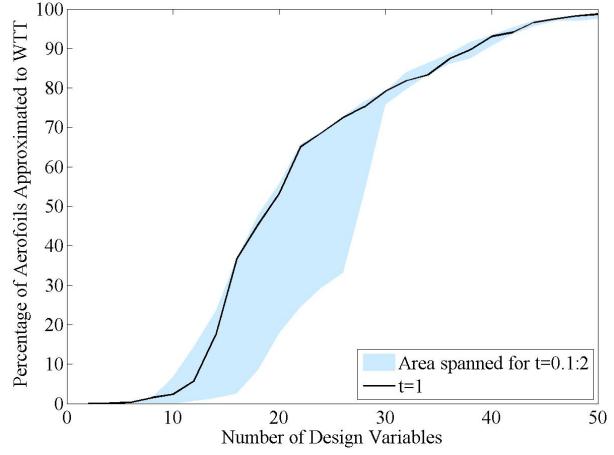


Figure 14. Percentage of aerofoil database approximated to within the typical wind tunnel tolerance for Hicks-Henne method with varying thickness variable t .

of these two methods. Unsurprisingly the inclusive method performs better, though the proximity of the result curves is clear. This indicates that the modes obtained with a particular library can be successfully generalised to aerofoils outside the training library.

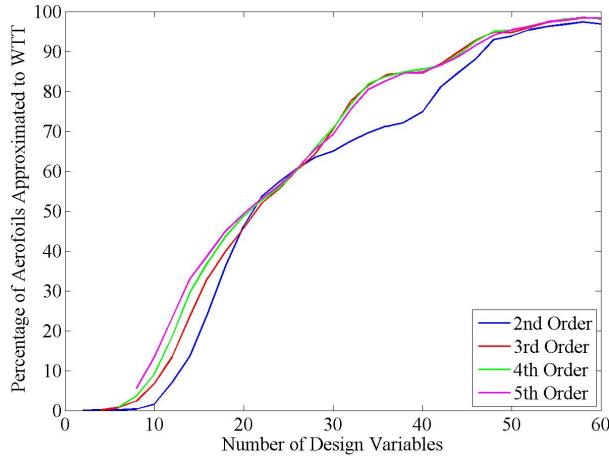


Figure 15. Percentage of aerofoil database approximated to within the typical wind tunnel tolerance for varying orders of B-spline method.

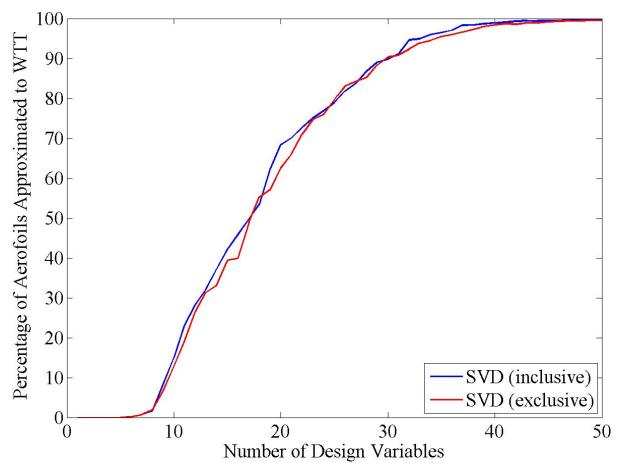


Figure 16. Percentage of aerofoil database approximated to within the typical wind tunnel tolerance for two possible SVD method implementations.

VI. Results

A. Large Database Test

Each of the methods was tested over the large database discussed in section (III). Here the performance of each method is characterised by the percentage of the database that was successfully approximated to a prescribed tolerance across the design variable range. Figure 17 shows the results when the wind tunnel tolerance used by Kulfan[6] was used. Results for the PARSEC method are also presented here as a benchmark.

For this criteria all the methods (bar PARSEC) show broadly comparable performance in that, if more than 50%, say, of the aerofoil library is to be approximated, all the methods require more than 17 design variables to do so. For a set approximation recovery across the library, the number of design variables needed

varies by about 5-10 depending on the method chosen. It should be noted that these results are sensitive to the technique used for re-sampling the aerofoil database as well as the method used to calculate the best fit.

Figures 18-23 then show the sensitivity of these results to $Error_z$ (equation 50): the thick line shows the results presented in figure 17 ($Error_z = 8 \times 10^4$). This presents a more detailed picture of the accuracy that can be expected by each of the methods across the range of design variables. The results indicate that as the tolerance is refined the SVD method continues to show better design coverage across the range of design variables than the other methods considered, albeit at the cost of additional design variables. For example, considering the tolerance $Error_z = 2 \times 10^{-4}$ at 60 design variables, the SVD method approximates 95% of the database to this tolerance whereas the Bezier surface, B-spline, CST and Hicks-Henne methods satisfy 81%, 47%, 52% and 73% respectively.

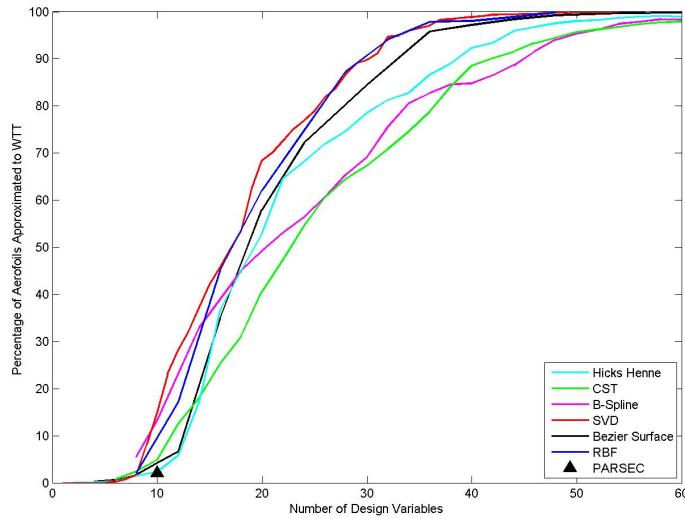


Figure 17. Percentage of aerofoil database approximated to within the typical wind tunnel tolerance (4×10^{-4} for $x/c < 0.2$, 8×10^{-4} for $x/c > 0.2$) for varying numbers of design variables.

B. Case Studies

The first case study considered was the NACA4412 aerofoil. For all the case studies, if an initial aerofoil was required a NACA0012 was used. Figure 24 shows the actual geometry approximation for each method for 12 design variables. Figure 25 demonstrates how the RMS of z error decreased for increasing complexity in the parameterisation methods. The CST and B-spline methods seemed to perform slightly better than the other results given this criteria.

The resulting surfaces were used for XFOIL[31] simulations, and the L/D results are presented in figure 26. The line labelled “Target” denotes the L/D calculated for the original aerofoil. Here the starred points, which indicate the instance where each method first satisfies the wind tunnel tolerance, all fall within 4% of the target L/D . This implies that a tighter tolerance may be required to establish full L/D convergence. Furthermore, no assessment has been made here on the influence of disparity between the aerofoil definition and the discretized surface boundary mesh.

Figures 27-29 show the results for the RAE2822. Figure 28 shows the RMS of the z/c errors for this case, demonstrating the results are similar for all the methods given this criteria. Figure 29 again shows the L/D convergence. For this case the starred points lie close to the target solution, all within 2% of the target L/D . Though this still suggests further refinement of the tolerance may be needed to establish full L/D convergence.

Figures 30 to 32 show similar plots for the ONERA M6 aerofoil which can be more challenging to fit due to local discontinuities in curvature at $x/c \approx 0.05$ and potentially elsewhere. Figure 31 shows the RMS of the z error, with similar traits to the previous two case, now with the RBF method looking slightly better than the alternative methods. In contrast, the CST method is less competitive which is attributable to the local discontinuities in curvature present. As the ONERA M6 is a symmetric aerofoil, figure 32 shows the

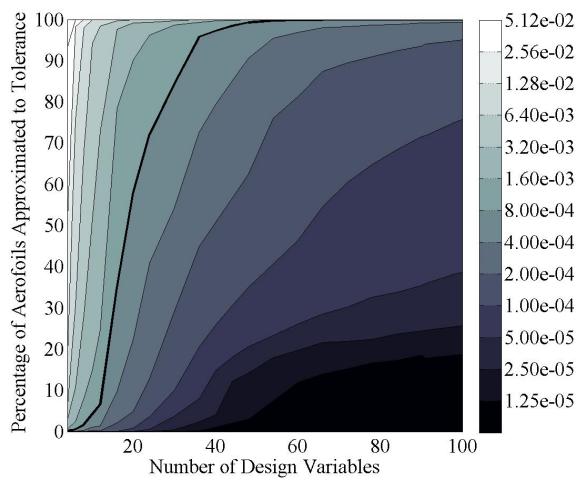


Figure 18. Percentage of aerofoil database approximated to varying $Error_z$ tolerance (right) for the Bézier Surface method.

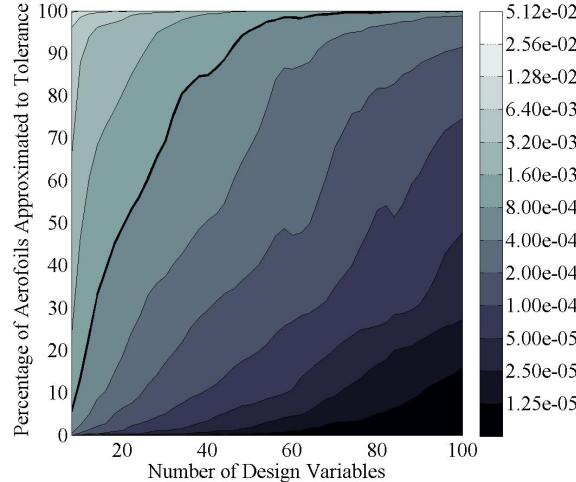


Figure 19. Percentage of aerofoil database approximated to varying $Error_z$ tolerance (right) for the B-spline method.

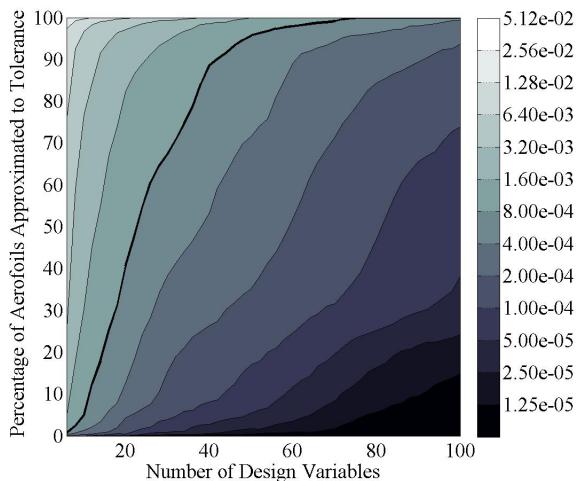


Figure 20. Percentage of aerofoil database approximated to varying $Error_z$ tolerance (right) for the CST method.

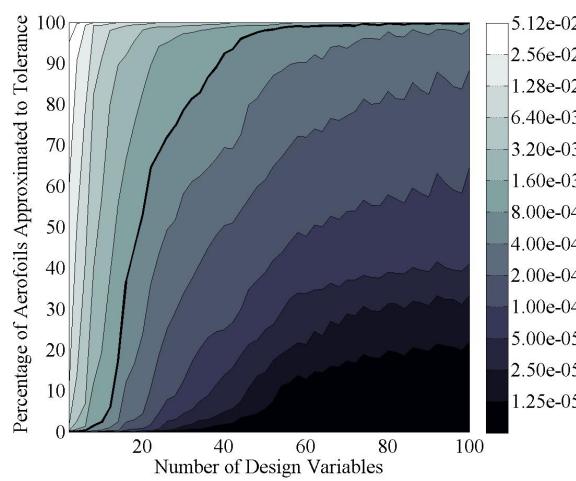


Figure 21. Percentage of aerofoil database approximated to varying $Error_z$ tolerance (right) for the Hicks-Henne method.

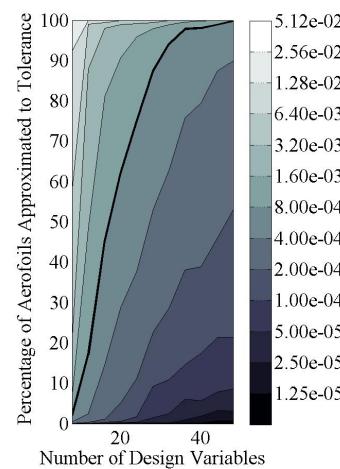


Figure 22. Percentage of aerofoil database approximated to varying $Error_z$ tolerance (right) for the RBF method.

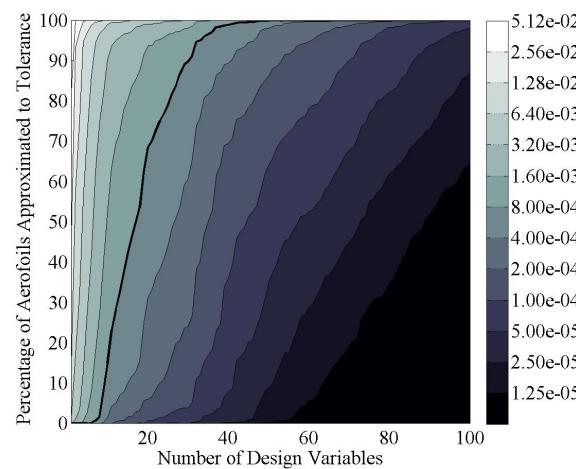


Figure 23. Percentage of aerofoil database approximated to varying $Error_z$ tolerance (right) for the SVD method.

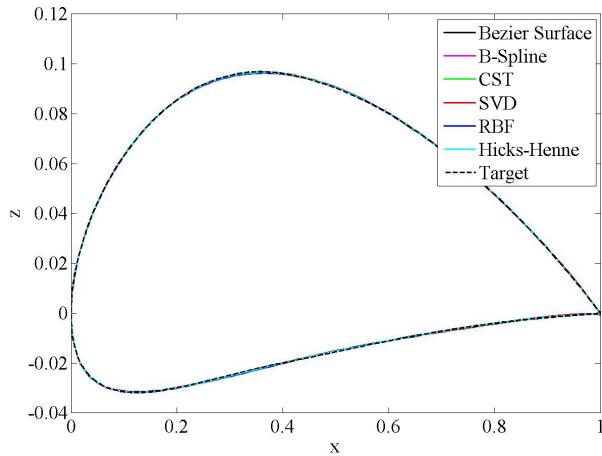


Figure 24. Optimum approximations of a NACA4412 for all six parameterisation methods using 12 design variables.

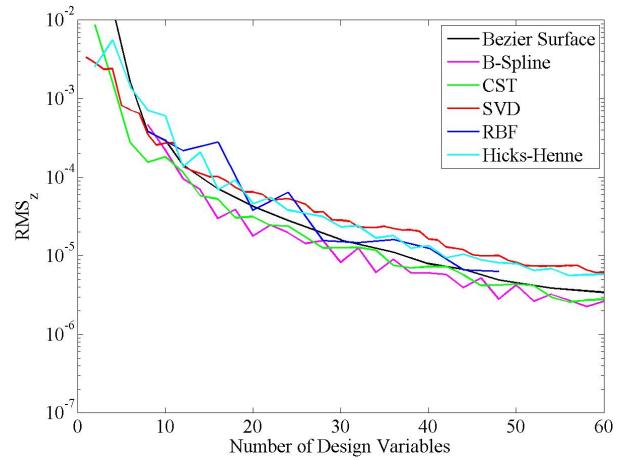


Figure 25. RMS of the z errors for the best approximations of a NACA4412 for all six parameterisation methods.

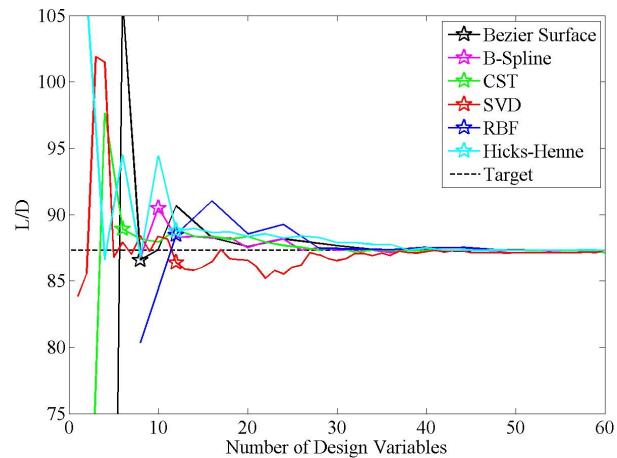


Figure 26. Convergence of L/D for the best approximations of a NACA4412. Starred points represent the first instance the wind tunnel tolerance is satisfied.

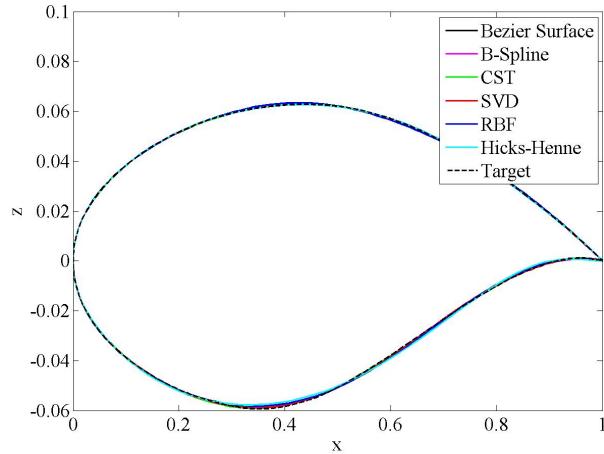


Figure 27. Optimum approximations of an RAE2822 for all six parameterisation methods using 12 design variables.

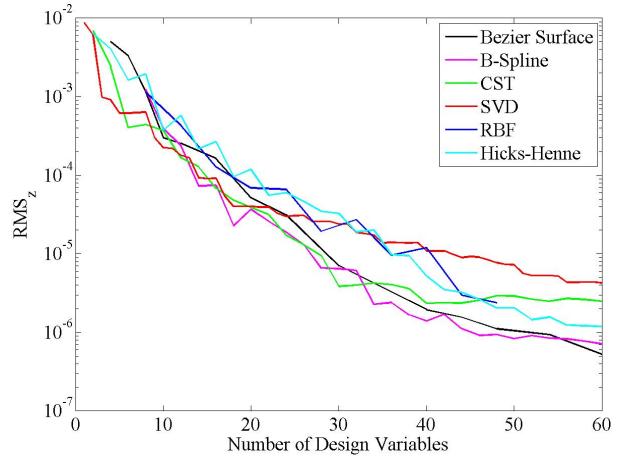


Figure 28. RMS of the z errors for the best approximations of an RAE2822 for all six parameterisation methods.

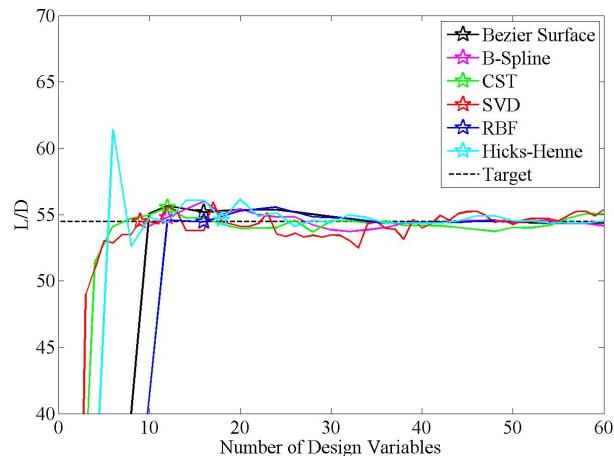


Figure 29. Convergence of L/D for the best approximations of an RAE2822. Starred points represent the first instance the wind tunnel tolerance is satisfied.

convergence of the drag coefficient only. Again, all the methods seem to show similar results with the starred points gathered close to the target drag coefficient. As for the RAE2822, it remains to be seen how far these results reflect those obtained under the transonic conditions for which the aerofoils were designed.

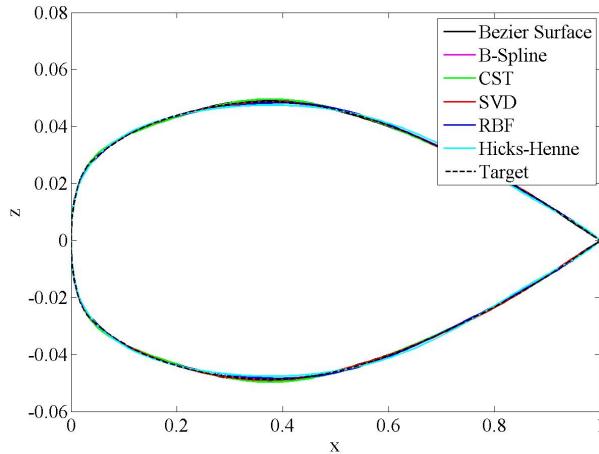


Figure 30. Optimum approximations of an ONERA M6 for all six parameterisation methods using 12 design variables.

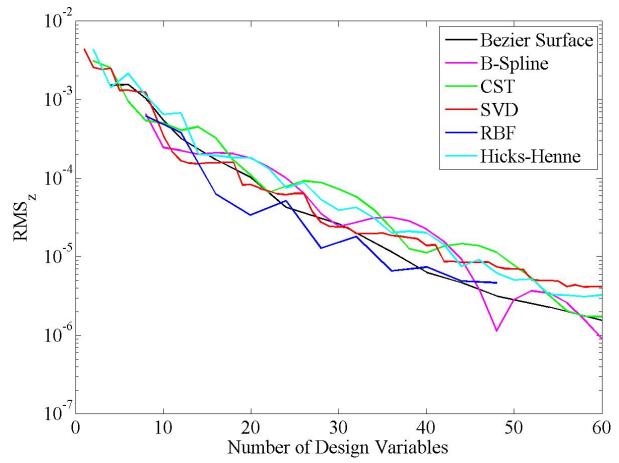


Figure 31. RMS of the z errors for the best approximations of an ONERA M6 for all six parameterisation methods.

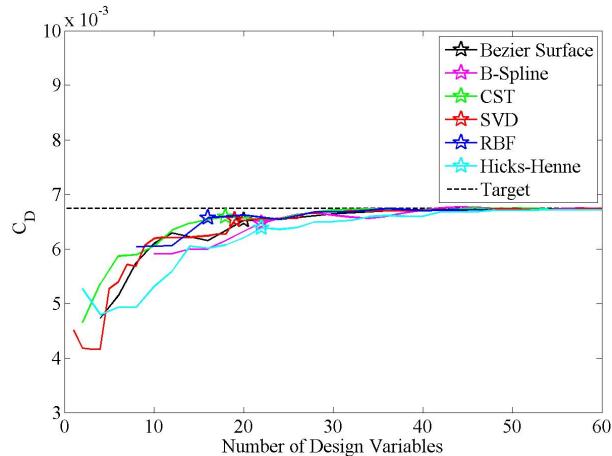


Figure 32. Convergence of drag coefficient for the best approximations of an ONERA M6. Starred points represent the first instance the wind tunnel tolerance is satisfied.

VII. Conclusions

In this work the accuracy and efficiency of six aerofoil parameterisation methods has been investigated, by considering a range of geometric shape recovery problems. The performance of each method has been analysed by considering geometric shape recovery of 1064 aerofoils using a range of design variables, testing the efficiency of design space coverage. Using Kulfan's wind tunnel geometric tolerance[6], the percentage of the aerofoil database that can be recovered to within this tolerance has been determined for varying numbers of design variables.

This has provided general insight into design space coverage; for example, it suggests that to obtain coverage of approximately 80% of the design space of all aerofoils, at least 25 design variables need to be used. This, however, is higher than is often desired for two-dimensional design studies which suggests that further work may be required to establish whether it is possible to devise a rationale for reducing the number of design variables needed for particular design exercises. It is also not clear that the z/c tolerance, alone,

is sufficient to guarantee the desired aerodynamic characteristics e.g L/D convergence; for example it says nothing about local surface curvature. Furthermore, an important issue, not considered here, is the impact of error associated with the discretization of the surface boundary conditions in the aerodynamic prediction methods.

VIII. Acknowledgements

The authors wish to acknowledge the financial support provided by Innovate UK: the work reported herein has been undertaken in GHandI (TSB 101372), a UK Centre for Aerodynamics project.

References

- ¹ Carrier, G., Destarac, D., Dumont, A., Meheut, M., Salah El Din, I., Peter, J., Ben Khelil, S., Brezillon, J., and Pestana, M., "Gradient-based aerodynamic optimization with the elsA software," *52nd Aerospace Sciences Meeting*, Jan 2014.
- ² Poole, D. J., Allen, C. B., and Rendall, T., "Application of control point-based aerodynamic shape optimization to two-dimensional drag minimization," *52nd Aerospace Sciences Meeting*, 2014.
- ³ Braibant, V. and Fleury, C., "Shape optimal design using B-splines," *Computer Methods in Applied Mechanics and Engineering*, Vol. 44(3), 1984, pp. 247–267.
- ⁴ Sobieczky, H., "Geometry generator for CFD and applied aerodynamics," *Courses and Lectures-International Centre for Mechanical Sciences*, 1997, pp. 137–158.
- ⁵ Anderson, W. K., Karman, S. L., and Burdshaw, C., "Geometry parameterization method for multidisciplinary applications," *AIAA Journal*, Vol. 47, 2009, pp. 168–1578.
- ⁶ Kulfan, B. M. and Bussoletti, J. E., "Fundamental parametric geometry representations for aircraft component shapes," *11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, September 2006.
- ⁷ Jameson, A., "Aerodynamic design via control theory," *Journal of Scientifical Computing, also ICASE Report No.88-64*, Vol. 3, 1988, pp. 233–260.
- ⁸ Hicks, R. M. and Henne, P. A., "Wing design by numerical optimization," *Journal of Aircraft*, Vol. 15, No. 7, 1978, pp. 407–412.
- ⁹ Pickett, J. R., Rubinstein, M., and Nelson, R., "Automated structural synthesis using a reduced number of design coordinates," *14th Structures, Structural Dynamics, and Materials Conference*, 1973.
- ¹⁰ Rendall, T. C. S. and Allen, C. B., "Unified fluid–structure interpolation and mesh motion using radial basis functions," *International Journal for Numerical Methods in Engineering*, Vol. 74, No. 10, 2008, pp. 1519–1559.
- ¹¹ Chauhan, D., Praveen, C., and Duvigneau, R., "Wing shape optimization using FFD and twist parameterization," *12th Aerospace Society of India CFD Symposium*, 2010.
- ¹² Sobieczky, H., "Parametric airfoils and wings," *Recent Development of Aerodynamic Design Methodologies*, Springer, 1999, pp. 71–87.
- ¹³ Toal, D. J. J., Bressloff, N. W., Keane, A. J., and Holden, C. M. E., "Geometric filtration using proper orthogonal decomposition for aerodynamic design optimization," *AIAA Journal*, Vol. 48, 2010, pp. 916–928.
- ¹⁴ Ghoman, S. S., Wan, Z., Chen, P. C., and Kapadia, R. K., "A POD-based reduced order design scheme for shape optimization of air vehicles," *53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, 2012.
- ¹⁵ Poole, D. J., Allen, C. B., and Rendall, T. C. S., "Metric-based mathematical derivation of efficient aerofoil design variables," *AIAA Journal*, 2014, Published online.

- ¹⁶ Morris, A. M., Allen, C. B., and Rendall, T. C. S., "CFD-based optimization of aerofoils using radial basis functions for domain element parameterization and mesh deformation," *International Journal for Numerical Methods in Fluids*, Vol. 58, No. 8, Nov 2008, pp. 827–860.
- ¹⁷ Morris, A. M., Allen, C. B., and Rendall, T. C. S., "Aerodynamic shape optimization of a modern transport wing using only planform variations," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, Vol. 223, No. 6, 2009, pp. 843–851.
- ¹⁸ Allen, C. B. and Rendall, T. C. S., "CFD-based optimization of hovering rotors using radial basis functions for shape parameterization and mesh deformation," *Optimization and Engineering*, Vol. 14, No. 1, Mar 2013, pp. 97–118.
- ¹⁹ Castonguay, P. and Nadarajah, S. K., "Effect of shape parameterization on aerodynamic shape optimization," *45th AIAA Aerospace Sciences Meeting and Exhibit*, 2007.
- ²⁰ Nadarajah, S., Castonguay, P., and Mousavi, A., "Survey of shape parameterization techniques and its effect on three-dimensional aerodynamic shape optimization," *18th AIAA Computational Fluid Dynamics Conference*, 2007.
- ²¹ Sripawadkul, V., Padulo, M., and Guenov, M., "A comparison of airfoil shape parameterization techniques for early design optimization," *13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference*, 2010.
- ²² Kulfan, B. M., "A universal parametric geometry representation method-CST," *45th AIAA Aerospace Sciences Meeting and Exhibit*, January 2007.
- ²³ Stone, M. H., "Applications of the theory of Boolean rings to general topology," *Trans. Amer. Math. Soc.*, Vol. 41, No. 3, Mar 1937, pp. 375–375.
- ²⁴ Kulfan, B. M., "Modification of CST airfoil representation methodology," Retrieved from <http://www.brendakulfan.com/docs/CST8.pdf>.
- ²⁵ Wu, H.-Y., Yang, S., Liu, F., and Tsai, H.-M., "Comparison of three geometric representations of airfoils for aerodynamic optimization," *16th AIAA Computational Fluid Dynamics Conference, Orlando, Florida*, 2003.
- ²⁶ Khurana, M., Winarto, H., and Sinha, A., "Airfoil geometry parameterization through shape optimizer and computational fluid dynamics," *46th AIAA Aerospace Sciences Meeting and Exhibit*, 2008.
- ²⁷ Wendland, H., *Scattered data approximation*, Cambridge University Press Cambridge, 2005.
- ²⁸ Buhmann, M. D., "Radial basis functions," *Acta numerica*, Vol. 9, 2000.
- ²⁹ Cook, P., McDonald, M., and Firmin, M., "Aerofoil RAE 2822 - Pressure Distributions, and Boundary Layer and Wake Measurements," *Experimental Data Base for Computer Program Assessment, AGARD Report AR 138*, 1979.
- ³⁰ Schmitt, V. and Charpin, F., "Pressure Distributions on the ONERA-M6-Wing at Transonic Mach Numbers," *Experimental Data Base for Computer Program Assessment, Report of the Fluid Dynamics Panel Working Group 04, AGARD AR 138*, 1979.
- ³¹ Drela, M., "Xfoil Subsonic airfoil development system," <http://web.mit.edu/drela/Public/web/xfoil/>, 2000–2013.
- ³² Rashedi, E., Nezamabadi-pour, H., and Saryazdi, S., "GSA: A gravitational search algorithm," *Information Sciences*, Vol. 179, No. 13, 2009, pp. 2232 – 2248, Special Section on High Order Fuzzy Sets.