A Locally Adaptive Subdivision Parameterisation Scheme for Aerodynamic Shape Optimisation

D. A. Masters*,
Department of Aerospace Engineering, University of Bristol

N. J. Taylor†,
MBDA UK Ltd, Filton

T. C. S. Rendall‡ and C. B. Allen§
Department of Aerospace Engineering, University of Bristol

This work presents a shape parameterisation method based on multi-resolution subdivision curves and investigates its application to aerodynamic optimisation. Subdivision curves are defined as the limit curve of a recursive application of a subdivision rule, which provides an intrinsically hierarchical set of control polygons that can be used to provide surface control at varying levels of fidelity. In this work they are used to construct a hierarchical set of aerofoil parameterisations that can be changed throughout an optimisation procedure. This enables an optimisation to be initialised with a small number of design variables, and then periodically increased in resolution throughout. This brings the benefits of a low dimensional design space (high convergence rate, increased robustness, low cost finite-difference gradients) while still allowing the final results to be from a high-dimensional design space. In this work two approaches to subdivision aerofoil parameterisation are investigated. A multi-level refinement technique the periodically refines the parameterisation globally and an adaptive refinement scheme that refines (and coarsens) the parameterisation based on adjoint surface sensitivities. Both of these approaches are tested on a variety of optimisation problems and for each problem a range of single-level subdivision schemes (equivalent to cubic B-splines) are also used as a control group. For all the optimisation cases the multi-level and adaptive schemes converge to solutions comparable or better than the single-level methods, generally providing a significant computational advantage, and in many cases allowing a solution to be found when the single-level method would otherwise finish prematurely in a local optimum.

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]. Within an optimisation procedure the choice of shape parameterisation controls the relationship between the optimisation design variables and the aerodynamic surface itself. Consequently the choice of shape parameterisation method can have a significant impact on the effectiveness and efficiency of the overall procedure[3]. Many different methods have been used within an aerodynamic optimisation framework, from standard geometric curve definitions such as B-Splines[4] or NURBS[5] to aerospace-specific methods such as CST[6,7], Hicks-Henne bump functions[8,9] or PARSEC[9,10] to Free-Form Deformation[11–14], proper orthogonal decomposition[2,15,16] or the discrete method[17]. All of these approaches are subject to the 'curse of dimensionality'; in the context of aerodynamic optimisation this refers to the problems associated with increasing the number of design variables used in the optimisation procedure. For many optimisation schemes the number of objective function evaluations is proportional to the number of design variables used, in conjunction with this a large number of design variables can lead to poor convergence rates and poor design space conditioning. Considering that for aerodynamic optimisation each objective function evaluation equates to a single, often expensive, aerodynamic solution the impact of dimensionality can be large. On the other hand, the fidelity of the parameterisation, and therefore the design space of the problem, is directly linked to the number of design variables. This often leads to a compromise between available resources and desired accuracy of the results.

[*] 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

One approach to reducing this effect is to control the shape with a series of nested, hierarchical parameterisation schemes and increase the fidelity at intervals throughout the optimisation process. This approach was first used in an aerodynamic optimisation setting by Beux and Dervieux[18] and has since been applied to a range of aerofoil optimisation problems using a variety of different parameterisation frameworks such as Bèzier curves[19–22], Bèzier surface FFD[23–25], RBFs[26] and B-Splines with a knot insertion algorithm[27,28]. There are two main approaches to applying this kind of parametrisation framework, the first is to uniformly increase the fidelity of the parameterisation periodically throughout the optimisation[18–28] , this is often referred to as a 'multi-level' approach. The second is to locally change the fidelity of the parameterisation based on some objective[20,21,25,26], this is referred to as an 'adaptive' approach. In general, it was shown that implementation of multi-level and adaptive parameterisations can improve the convergence rate, robustness and final result of an optimisation procedure. This paper investigates the application of both of these parameterisation approaches to multi-resolution subdivision curves for aerodynamic optimisation procedures.

Subdivision curves are a shape parameterisation method used predominantly in computer graphics and animation[29]. They describe a smooth curve based on an initial coarse network of points and a simple subdivision rule of refinement. By successively applying the subdivision rule increasingly fine networks are created which at the refinement limit create a curve. In some cases these limit curves are equivalent to B-Splines, for example Chaikin's corner cutting scheme[30] is equivalent to uniform quadratic B-Splines; further extensions to higher order uniform B-Splines can also be derived[31]. B-Splines and subdivisions share many characteristics; the method of implementation is, however, one area of major difference. B-splines utilise continuous parametric representation whereas subdivisions use a hierarchical process of discrete refinement. It is the innately hierarchical nature of subdivisions that make them easily applicable to multi-resolution analysis.

Multi-resolution analysis utilises hierarchical nested data sets to efficiently store data and allow operations to be performed at varying levels of detail. For geometry and shape parameterisation applications this typically means the ability to implement either coarse geometry changes while maintaining the fine detail, or fine geometry changes while maintaining the overall shape. This approach has been implemented comparably from both B-spline[32] and subdivision[33] perspectives.

The aim of this work is to explore the use of multi-resolution subdivision curves for aerodynamic shape optimisation with particular emphasis on how they can be used in multi-level and adaptive schemes to improve both the efficiency and accuracy of current optimisation procedures.

II. Subdivision Curves

Subdivision curves are defined as the limit of a process of repeated subdivision refinement of an initial control polygon. Each subdivision refinement defines a new set of smoother, denser points as a weighted average of the old points. For this reason the refinements can conveniently be expressed as a simple matrix transformation

$$C^{n+1} = P^n C^n \tag{1}$$

from old points C^n to new points C^{n+1}. For simple subdivision schemes on closed polygons these matrices are just the two row offset repetition of a subdivision 'mask'. Two common examples are Chaikin's rule[30] (equations 2 and 3) and the Cubic B-spline rule (equations 4 and 5). Figure 2 shows a simple implementation of Chaikin's rule on a closed polygon. Note how the rows of the transformation matrices all sum to 1 and therefore describe a weighted averaging of the previous points, this is a key feature of all subdivision transformation matrices. Equations 2 and 4 describe the matrix for areas of smooth subdivision whereas equations 3 and 5 represent an area with a corner or endpoint, where the non-averaged matrix row represents the corner or endpoint itself.

Given a set of subdivision matrices P the N^{th} subdivision level C^N can be expressed as

$$C^N = P^{N-1} \dots P^{n+1} P^n C^n \tag{6}$$

for some $n < N$. The limit surface can therefore be described as

$$C^\infty = \lim_{N\to\infty} C^N = \dots P^{n+1} P^n C^n. \tag{7}$$

In practice this calculation must be truncated at some point and the limit surface calculated. A typical solution to this is to calculate an evaluation matrix based on eigenanalysis[34] that pushes the subdivision points to their limit locations. With this method however the user cannot directly parameterise the surface itself and therefore cannot specify the distribution of points around the surface. As the distribution of points around an aerofoil can be very important for aerodynamic applications an alternative method is used in this work. This is done by exploiting the equivalency between B-spline curves and some subdivision formulations, and creating a continuous B-spline transformation between

Chaikin's rule:

$$P^n = \begin{bmatrix} \ddots & \ddots & & & \\ 0.25 & 0.75 & & & \\ & 0.75 & 0.25 & & \\ & 0.25 & 0.75 & & \\ & & 0.75 & 0.25 & \\ & & 0.25 & 0.75 & \\ & & & \ddots & \ddots \end{bmatrix}, \quad (2) \qquad P^n = \begin{bmatrix} \ddots & \ddots & \\ 0.5 & 0.5 & \\ & 1 & \\ & 0.5 & 0.5 \\ & 0.25 & 0.75 \\ & & \ddots \end{bmatrix}. \quad (3)$$

Cubic B-Spline rule:

$$P^n = \begin{bmatrix} \ddots & \ddots & & & \\ 0.125 & 0.75 & 0.125 & & \\ & 0.5 & 0.5 & & \\ & 0.125 & 0.75 & 0.125 & \\ & & 0.5 & 0.5 & \\ & & 0.125 & 0.75 & 0.125 \\ & & & \ddots & \ddots \end{bmatrix}, \quad (4) \qquad P^n = \begin{bmatrix} \ddots & \ddots & & & \\ 0.5 & 0.5 & & & \\ & 1 & & & \\ & 0.5 & 0.5 & & \\ & & 0.75 & 0.25 & \\ & & 0.1875 & 0.6875 & 0.125 \\ & & & \ddots & \ddots \end{bmatrix}. \quad (5)$$

Figure 1. Matrix representations of the Chaikin and Cubic B-Spline subdivision schemes.

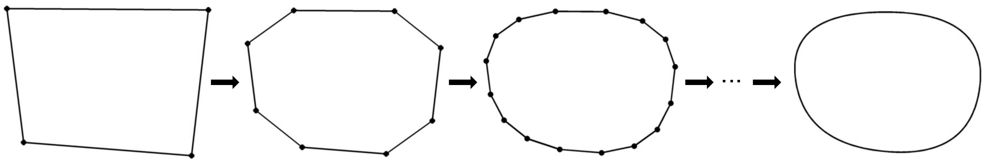


Figure 2. Simple closed subdivision using Chaikin's Scheme

the final subdivision level N and the limit curve. This B-Spline transformation can then be formulated as matrix P^N_{BS} such that a desired point distribution is achieved. This method is however constrained to subdivisions with B-spline equivalents.

It is then convenient to define

$$\phi^n = P^N_{BS} P^{N-1} \dots P^n \tag{8}$$

and therefore the relationship between any level of control polygon and the limit surface can be expressed as

$$C^\infty = \phi^n C^n. \tag{9}$$

The columns of ϕ_i represent the basis functions of the subdivision scheme. Figure 3 shows the basis functions for a Chaikin subdivision with fixed endpoints, these are identical to the basis functions of an equivalent quadratic B-spline. From this it can be seen that the different subdivision levels can be used to control and deform the limit surface at varying levels of fidelity.

III. Reverse Subdivision Curves

Given a set of points it can often be desirable to obtain the subdivision polygon (or closest possible match) that produced them. Consider a fine set of points C^n; the coarser set of points C^{n-1} can simply be calculated as the least squares solution of equation 1,

$$C^{n-1} = (P^{n-1})^+ C^n, \tag{10}$$

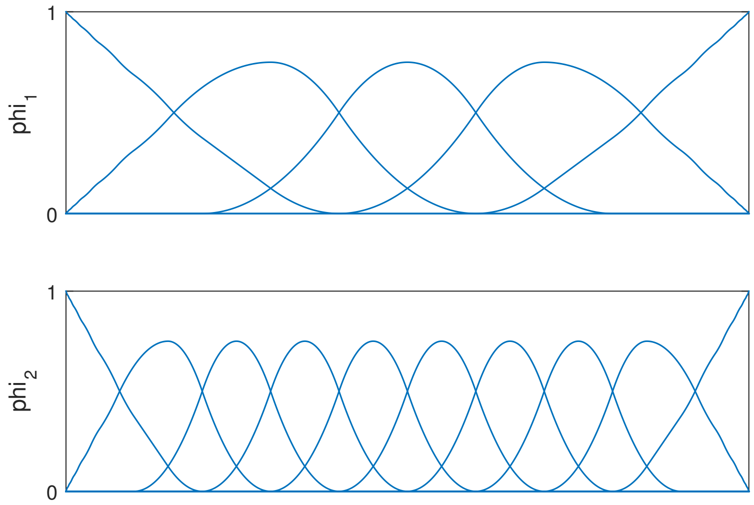


Figure 3. The basis functions for the first two levels of a Chaikin subdivision with fixed endpoint conditions with a 4 point initial control polygon

where + denotes the Moore-Penrose pseudo-inverse. However as P^n is a non-square, non-invertible matrix this leads to some loss of information with the result that, for almost all cases,

$$P^{n-1}C^{n-1} \neq C^n. \tag{11}$$

For this reason it is important to retain any errors created through the least squares process and include them in any subsequent refinement. This can be done very conveniently and efficiently by extending the refinement matrices P^n by any orthogonal compliment $Q^n = null((P^n)^T)$ and extending the subdivision refinement equation such that

$$\begin{aligned} C^{n+1} &= \begin{bmatrix} P^n & Q^n \end{bmatrix} \begin{bmatrix} C^n \\ D^n \end{bmatrix} \\ &= P^n C^n + Q^n D^n, \end{aligned} \tag{12}$$

for some set of error coefficients D^n. Then by letting $\begin{bmatrix} P^n & Q^n \end{bmatrix}^{-1} = \begin{bmatrix} A^n \\ B^n \end{bmatrix}$ equation 10 can be re-expressed as the reverse subdivision equations

$$C^{n-1} = A^n C^n, \tag{13}$$

$$D^{n-1} = B^n C^n. \tag{14}$$

This importantly creates a one-to-one relationship between the subdivision refinement levels and thus allows information to be propagated uniquely and exactly in either the refinement or coarsening direction (figure 4).

$$\begin{tikzcd} C^i \arrow[r] & C^{i+1} \arrow[r] & C^{i+2} \arrow[r] & \cdots \\ D^i \arrow[ur] & D^{i+1} \arrow[ur] & D^{i+2} \arrow[ur] & \end{tikzcd}$$

a)

$$\begin{tikzcd} C^i \arrow[r] \arrow[dr] & C^{i-1} \arrow[r] \arrow[dr] & C^{i-2} \arrow[r] \arrow[dr] & \cdots \\ & D^{i-1} & D^{i-2} & \cdots \end{tikzcd}$$

b)

Figure 4. Process for subdivision refinement (a) and reconstruction (b)

Equation 9 can then be reformed using equation 12 such that

$$C^\infty = \phi^n C^n + \sum_{i=n}^{N} \phi^{i+1} Q^i D^i. \tag{15}$$

By storing and including these error terms it means that any shape can now be represented by any subdivision rule as long as the correct error terms D^n are used.

IV. Subdivision Basis Elevation

In shape optimisation, as well as many other applications, it can be advantageous to have varying degrees of fidelity in different areas of the shape. This can allow high fidelity control in areas that need it and low fidelity control in areas that do not. To achieve this within the subdivision framework a process of basis elevation has been devised that promotes a given subdivision basis functions to the first subdivision level. In this context 'basis elevation' refers to the process of copying a subdivision control point or basis function from a given level up to the first level, here on referred to as the 'active' level. All shape control is then be performed on the active level only with the subdivision configuration itself changed to control fidelity.

Figure 5a shows a graphical representation of an initial subdivision configuration, an open set of five control points with end-points, subdivided using a cubic B-spline subdivision scheme. Just the first three levels are included here but in practice they would be further refined to a much finer limit surface. Each node represents a subdivision control point and the arrows between the nodes represent a weighted association from a 'parent' to a 'child' where the darkness is proportional to its weighting. It should be noted that the sum of any node's parent associations must equal one, this is because a subdivision control point is always a weighted average of its parent points; this is not true of a node's child association.

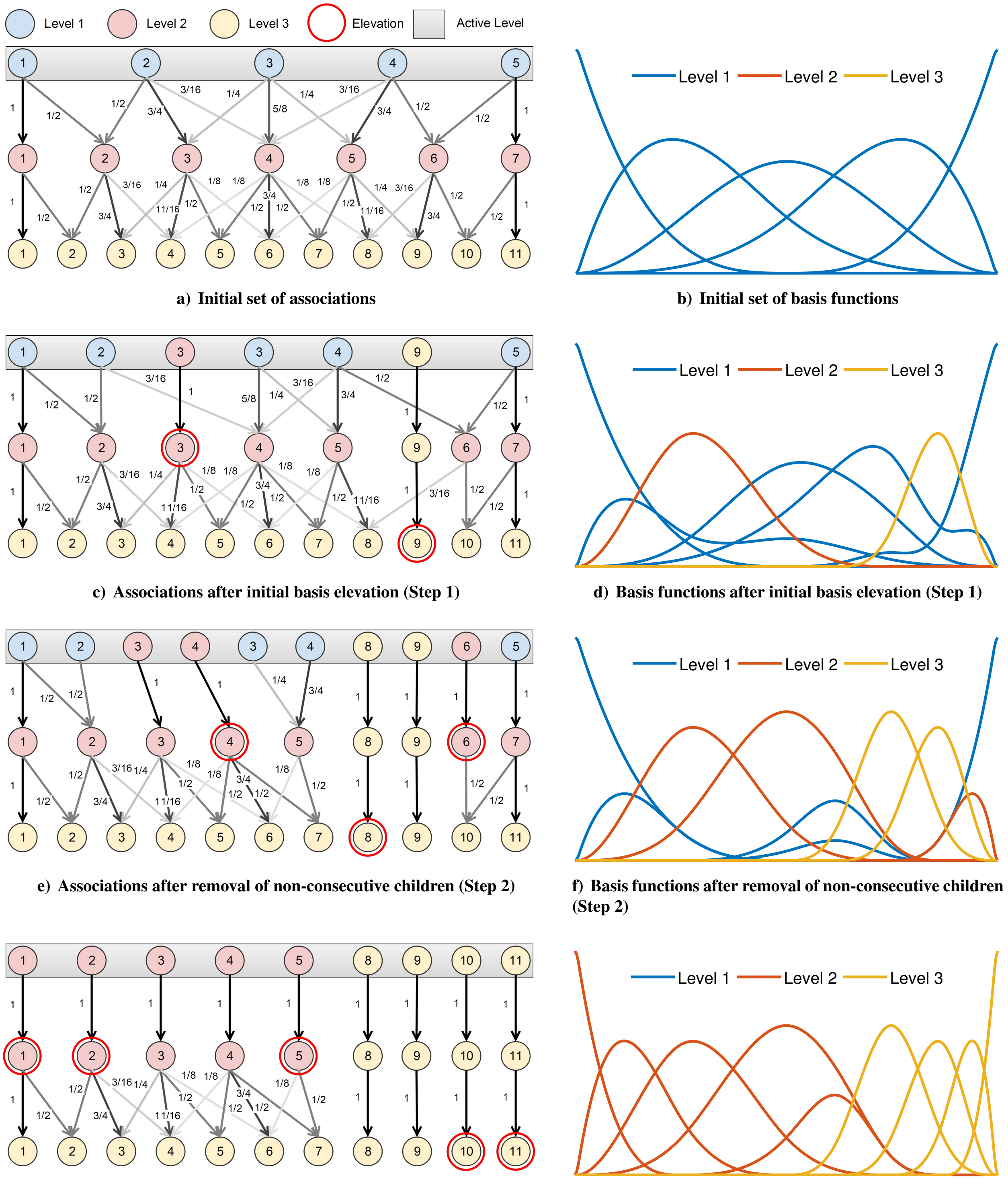
To 'elevate' a given 'target' node to the active, first level a chain of surrogate parent nodes are created in the intermediate levels. Each of these is associated with a weighting of 1 which represent a duplication of the target control point. However as the sum of all parent associations must sum to one, all of the original associations between the target node and its parents must be removed. Figure 5c shows the result of elevating the two highlighted nodes.

This elevation process allows any subdivision control point from any level to be pushed to the first level and used together. It has however been found that if this basis elevation process is applied naively the resulting set of basis functions can be undesirable; containing redundant, highly overlapping or multi-maxima functions, figures 5d and 5f show examples of this. The authors have found that these features can negatively influence an optimisation process. For this reason a three step process has been implemented to ensure that a suitable set of final basis functions are found:

1. Elevate desired control points to active level
2. Identify non-consecutive children and elevate control points to remove them.
3. Remove redundant control points and simplify configuration to achieve minimal overlap of basis functions.

Figure 5 shows an example of the process to elevate the 3rd control point on level 2 and the 9th control point on level 3. From here on the control points will be referred to in the form $[level, id]$, e.g. $[2, 3]$ and $[3, 9]$. Figures 5c and 5d show how elevating these two control points (step 1) affects the subdivision system and basis functions. It can be seen that two of the basis functions have two maxima, this is due to some of the nodes having non-consecutive children. This can be seen in figure 5c where the children of control points $[1, 2]$ and $[1, 4]$ have been split by the elevated nodes and are now not consecutive. To remove this, the isolated control points $[2, 4]$, $[2, 6]$ and $[3, 8]$ are elevated; this is step 2 and the result of this can be seen in figures 5e and 5f. It should be noted that this step does increase the number of design variables but the authors have found this to be a reasonable compromise to remove the multi-maxima basis functions.

A third and final simplification step is then implemented that removes redundancy and minimises the overlap of basis functions. This is done by finding any set of nodes equal or larger in size than their combined children, then elevating those children. This ensures that the design space is maintained and the most efficient set of basis functions is chosen. It can be seen that there are three instances in figure 5e where the above condition is true: control points $[1, 1]$,$[1, 2]$; $[1, 3]$,$[1, 4]$ and $[2, 6]$,$[2, 7]$. Their combined children were then also elevated to give the final configuration shown in figures 5g and 5h.



a) Initial set of associations

b) Initial set of basis functions

c) Associations after initial basis elevation (Step 1)

d) Basis functions after initial basis elevation (Step 1)

e) Associations after removal of non-consecutive children (Step 2)

f) Basis functions after removal of non-consecutive children (Step 2)

g) Associations after final simplification (Step 3)

h) Basis functions after final simplification (Step 3)

Figure 5. Example of basis elevation of 3rd control point on level 2 and 9th control point on level 3.

V. Adaptive Design Variable Prioritisation

The ability to use a variety of design variables from different subdivision levels opens the possibility of adapting the design variable used to suit the problem at hand. For an optimisation this could be done once at the start to allow greater control in an area of particular interest, periodically throughout the optimisation or even after every iteration. However in order to achieve this, a robust strategy for prioritising the design variables must first be implemented. The approach described below aims to prioritise the design variables such that the best approximation of the exact surface gradient is achieved while minimising the number of design variables used.

Given an adjoint approach to aerodynamic simulation a full surface gradient for the objective function J, $\frac{\partial J}{\partial z_j}$, is available at every iteration. From this the gradients for any particular set of design variables, a_i, can be calculated as

$$\underbrace{\begin{bmatrix} \frac{\partial J}{\partial a_1} \\ \frac{\partial J}{\partial a_2} \\ \vdots \\ \frac{\partial J}{\partial a_n} \end{bmatrix}}_{\text{Gradients}} = \underbrace{\begin{bmatrix} \frac{\partial z_1}{\partial a_1} & \cdots & \frac{\partial z_m}{\partial a_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial z_1}{\partial a_n} & \cdots & \frac{\partial z_m}{\partial a_n} \end{bmatrix}}_{\substack{\text{Geometric} \\ \text{Sensitivities}}} \underbrace{\begin{bmatrix} \frac{\partial J}{\partial z_1} \\ \frac{\partial J}{\partial z_2} \\ \vdots \\ \frac{\partial J}{\partial z_m} \end{bmatrix}}_{\substack{\text{Surface} \\ \text{Sensitivities}}} . \tag{16}$$

Within the subdivision framework the geometric sensitivities for a given level L are given by ϕ^L where each column ϕ_i^L represents the basis function for design variable a_i^L. The gradients for each subdivision level are therefore given by

$$\mathbf{g}^L = \frac{\partial J}{\partial a_i^L} = \left(\phi_i^L\right)^T \frac{\partial J}{\partial z_j}. \tag{17}$$

These gradients can then be projected onto the aerofoil surface to obtain the parametrised surface sensitivities, $\phi_i^L \mathbf{g}^L$. For an unconstrained steepest descent optimisation this is equivalent to the surface deformation direction. As the subdivision level, and therefore fidelity, is increased these parameterised surface sensitivities will converge to the exact adjoint surface sensitivity. The importance of each design variable can then be characterised by the change in gradient as the subdivision level is increased. This can be calculated directly by comparing the gradients at a given level and its subdivided previous level, i.e.

$$\Delta \mathbf{g}^L = \hat{\mathbf{g}}^L - P^{L-1} \hat{\mathbf{g}}^{L-1} \tag{18}$$

where the hat represents the normalisation $\hat{\mathbf{g}}^L = \frac{\mathbf{g}^L}{||\phi_i^L \mathbf{g}^L||}$. The rationale behind this metric is that a large change in the gradients between levels suggests that the addition fidelity available at that level has a significant impact. This method then localises this so that the influence of individual design variables can be identified. The values of Δg_i^L can then be compared across the levels with the largest values representing the most important design variables. In this work a tolerance is specified such that if $\Delta g_i^L > l_{tol}$ the design variable is used.

One assumption that this metric makes is that the adjoint surface sensitivity is always the best search direction for an optimisation. Previous work[35,36] has however suggested that using high fidelity gradients, such as exact adjoint surface sensitivities, is not always appropriate and that maintaining a smooth surface throughout the optimisation process is crucial. This was also investigated by the authors[37] who showed that using un-smooth surface deformations, particularly in the early stages, can hinder both the rate of convergence and robustness of an optimisation. For this reason a limitation on the available design variables is applied at the start of the optimisation process. A maximum subdivision level, L_{max}, is set for the first optimisation phase, this is then increased by some integer u (typically $u = 2$) at every refinement stage.

A further limitation is also included for the profile-constrained optimisation presented in section VIII.C. This is applied to reduce unnecessary fidelity in areas where constraints are going to be immediately violated. In this work the profile constraint enforces that the optimised aerofoil surface must always be outside the initial aerofoil surface. It can therefore be deduced that any design variable supporting an area of the surface sufficiently close to the constraint boundary with an inwards pointing gradient will immediately violate the constraint and thus not be used by the optimiser. For this reason any design variable for which this is true $\Delta g_i^L = 0$.

VI. Aerofoil Parameterisation

To parameterise the aerofoils in this work a cubic B-Spline subdivision scheme is used with a single, closed initial polygon with 'corners' at the leading and trailing edges. This ensures that the position of the leading and trailing edges are equal to the 'corner' control points at every subdivision level. This is equivalent to using two distinct subdivision curves for the upper and lower surfaces with shared endpoints at the leading and trailing edges. In physical terms the trailing edge is a corner, however the leading edge is not, for this reason the control points closest to the leading edge are constrained to lie directly above and below it; this enforces the vertical surface required.

In this work the first subdivision level has been defined by six control points with two points defining the leading and trailing edges at $[x/c, z/c] = [0, 0]$ and $[1, 0]$ respectively, and points at $x/c = 0$ and $x/c = 0.5$ on each surface. All points are free to move in the z direction apart from the trailing edge point that is fixed. In this work, unless otherwise specified, the aerofoils have been defined by 601 cosine distributed points and the maximum level is defined by the eighth subdivision polygon containing 260 points. Table 1 shows the total number of control points in each level used.

Subdivision Level	Number of Control Points
1	6
2	8
3	12
4	20
5	36
6	68
7	132
8	260

Table 1. Number of control points at each subdivision level

Given an initial aerofoil $C_{initial}$, the starting initial subdivision positions can then be calculated by the recursive application of equation 13. The resulting control point positions, $C^n_{initial}$, represent the least-squares approximations of the initial aerofoil for the limit surfaces $\phi^n C^n_{initial}$. At each of these subdivision levels a set of fixed error terms D^n can then be calculated by applying equation 14. Then for the set of control points $C^n_{initial}$ calculated, equation 15 implies that

$$C_{initial} \equiv \phi^n C^n_{initial} + \sum_{i=n}^{N} \phi^{i+1} Q^i D^i, \quad \forall n \leq N. \tag{19}$$

The benefit of this is that any subdivision optimisation can be set to start from exactly the prescribed initial aerofoil rather than a best approximation. This approach has been used for all the optimisations in this paper. Figure 6 shows the initial positions for the first four subdivision levels; each of which reproduce a NACA0012 exactly.

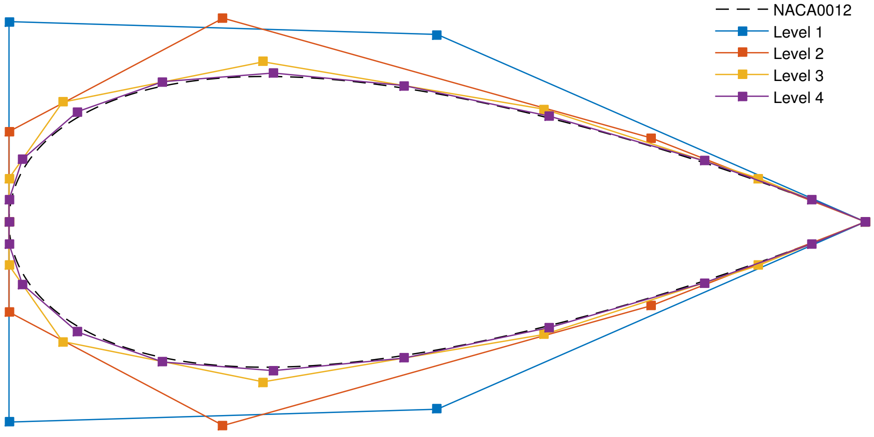


Figure 6. Initial control point positions for the first four subdivision levels for a NACA0012.

Each of these available subdivision levels can then be used to parametrise the aerofoil independently, in what have been described hereafter as ‘single-level’ schemes, in an ascending series, which have been described as ‘multi-level’ or by adaptively modifying the subdivision scheme (sections IV and V), which have been described as ‘adaptive’. For the single-level schemes the aerofoils are defined by the equation

$$C_{aero} = \phi^n C^n + \sum_{i=n}^{N} \phi^{i+1} Q^i D^i \tag{20}$$

for the desired subdivision level n with control points C^n. For the multi-level schemes the same definition applies but when a 'refinement trigger' is activated $n \to n+1$. This increases the number of design variables, and therefore available fidelity, while maintaining the aerofoil shape exactly. For the the adaptive methods a reconfiguration of the full subdivision system is triggered by the 'refinement trigger' with the design variables chosen through prioritisation technique described in section V. This recalculates the parameterisation such that fidelity is placed in areas where it is needed, there is therefore not always a strict increase in the total number of design variables. The aerofoil shape is again maintained exactly.

The aim of these refinement methods is to use fewer design variables in the early stages of optimisation to increase the rate of convergence, robustness and, for finite-difference gradients, reduce gradient calculation times. Then, as the lower dimensional design space looks to be fully exploited, the refinement process is applied to allow the larger design space to be explored.

For each optimisation case investigated a variety of these subdivision schemes have been applied. Each available subdivision level has been applied as a single-level parameterisation, as four multi-level schemes starting from the first, second, third and fourth levels respectively and refining up to the final level, as well as a single adaptive scheme. The single-level schemes are equivalent to using normal cubic B-Splines and act as the control group for which the benefits of the other methods have been compared against.

VII. Optimisation Methodology

In this work subdivion curves have been applied to a range of aerodynamic optimisation problems. For all of these tests the multi-purpose large-scale optimiser SNOPT[38] was used. This is a gradient-based sequential-quadratic programming (SQP) method that employs a reduced-Hessian BFGS search-direction and, in this work, a non-derivative line-search technique. This was coupled with two different aerodynamic solvers; a potential flow panel-code and the SU^2 Euler solver[39,40]. For the potential flow solver function gradients were calculated by finite-difference, importantly this means that the cost of calculating the gradients is proportional to the number of design variables used. For the adaptive method adjoint surface sensitivities are required to prioritise the design varaibles, for the potential flow solver this is calculated by finite-differencing each of the aerofoil points. For SU^2 however, the gradients were calculated using the adjoint equations, and consequently the cost of calculating the gradients is independent of the number of design variables used.

Optimiser convergence was determined based on the activation of one of three criteria:

1. The Karush-Kuhn-Tucker (KKT) first-order optimality condition[38] satisfying a tolerance level
2. The optimiser unable to improve the objective function
3. For the schemes *not* on the final level, satisfying the refinement condition (equation 21)

The refinement condition aims to trigger the refinement of the subdivision scheme when the optimisation has exploited most of the available gains from the current design space and is approaching the local minimum. This moment can be difficult to identify as it is very hard to differentiate between the optimiser converging to a local optimum and the optimiser traversing a difficult area of the design space. If refinement is triggered too early the under-exploitation of design space can result in slower convergence and possibly a poorer final result, and if it is triggered too late, over-exploitation of the design space can waste resources. A method for approximating the optimum refinement time was proposed by Anderson[26] where refinement is triggered when the convergence of the objective function with respect to the iterations drops below some proportion $t < 1$ of the maximum attained. A slightly modified version of this is implemented in this work. This triggers refinement if the rolling average of the log-scaled gradients is less than a proportion t of the max rolling average of the log scaled gradients; i.e.

$$\frac{1}{w}\sum_{j=0}^{w-1} |G_{k-j}| < \max_{m \le l \le k} \frac{t}{m}\sum_{j=0}^{m-1} |G_{l-j}| \tag{21}$$

$$\text{where} \qquad G_i = \log_{10}(J_{i-j}) - \log_{10}(J_{i-j-1}) \tag{22}$$

for objective function J, iteration k (for current optimisation phase), $t < 1$ and integers w and m. The parameter t controls the change in gradient required to trigger the scheme and w and m control the size of the rolling average windows for the maximum and current gradients. If small values are used for w and m this defines a very aggressive triggering system, for well behaved, consistently converging optimisations this ensures that iterations are not wasted converging areas close to a local minimum. For more complex optimisation procedures this can however cause premature triggering when the optimiser only makes a small improvement through a highly non-linear area. For this reason these parameters can be increased to average the gradients over a given window and only trigger refinement when improvements are consistently small.

VIII. Optimisation Results

To test benefits of using the multi-level and adaptive subdivision parameterisations described above, a series of optimisations of varying difficulty have been performed. A set of geometry matching problems present the simplest test, minimising the RMS geometry error between the aerofoil surface and a target aerofoil. A further set of inverse design problems present a more challenging optimisation with a final constraint drag optimisation test representing the most difficult test.

A. Geometry Matching

Three geometry matching problems have been considered in this work. Each one starts from an initial NACA0012 then targets RAE2822, NACA4410 and ONERA M6 aerofoils. For all of these optimisations the objective function is defined as the root-mean-squared difference between the current and target aerofoils, i.e.,

$$J = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(z_i - z_i^{target}\right)^2}, \tag{23}$$

the gradients are then calculated analytically. The refinement parameters used are $w = m = 1$ and $t = 0.3$ and the adaptive parameters are $l_{tol} = 5 \times 10^{-3}$, $L_{max} = 4$ and $u = 2$.

Figures 7, 8, 9 show the results of these optimisations. For each of the three test cases in can be seen, for the single-level parameterisations, that as more design variables are used the results improve. This is what you would expect as the design space is increased, it can however be seen that it can take quite a large number of iterations to reach the final result. For the multi-level cases it can be seen that they always achieve a final result equivalent to the 'single-level 8' case, it appears that this is the global optimum for all three test cases. The multi-level case, however, converges quicker in most instances obtaining the optimum in fewer iterations. The rate of convergence of the multi-level methods also seems to be dependant on the starting level used, with the '4→8' case performing best for all three test cases. The adaptive method then provides further improvement on the '4→8' multi-level method for the RAE2822 test case and shows comparable performance for the other two test cases.

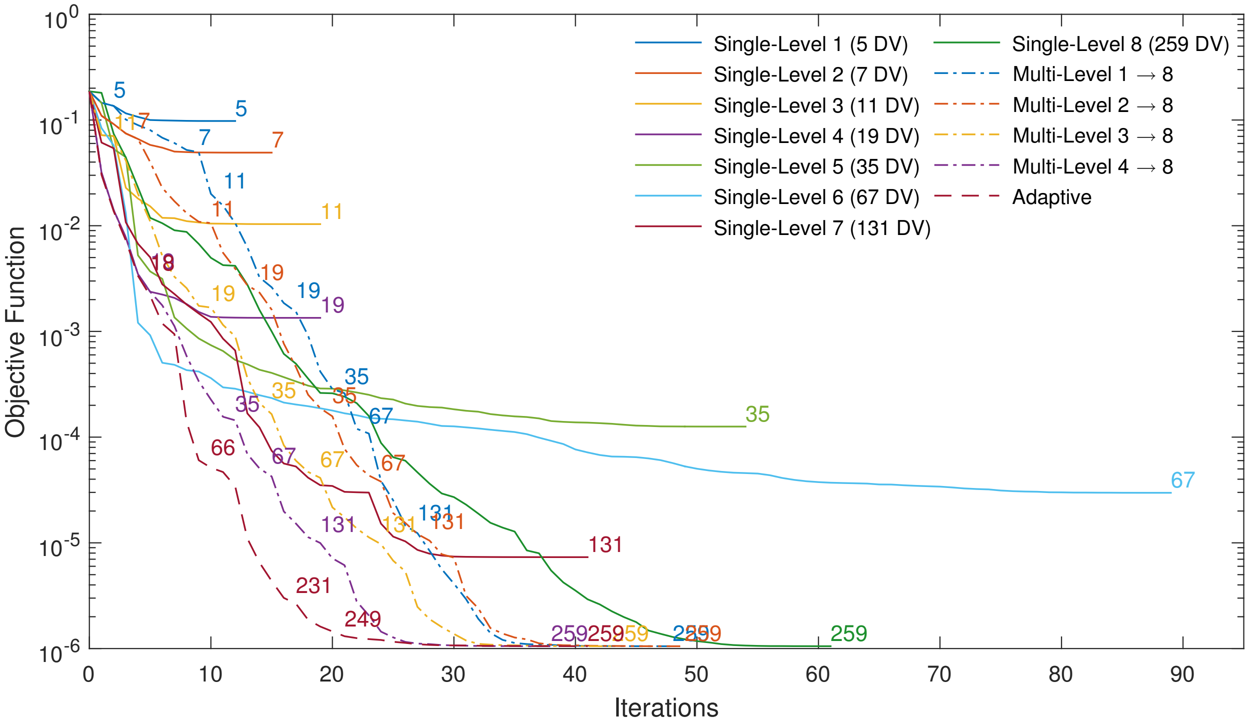


Figure 7. Results for the geometry matching optimisation from a NACA0012 to an RAE2822. Numbers represent the number of design variables for the previous optimisation phase.

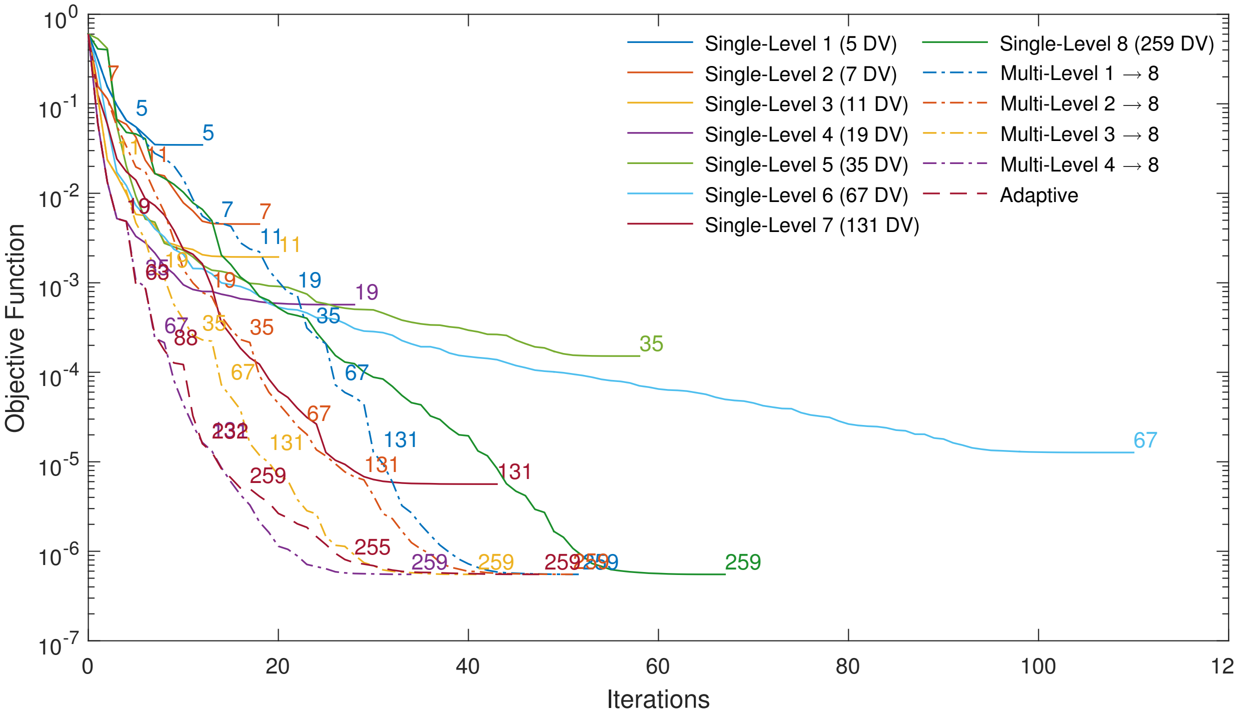


Figure 8. Results for the geometry matching optimisation from a NACA0012 to a NACA4410. Numbers represent the number of design variables at the end of an optimisation phase.

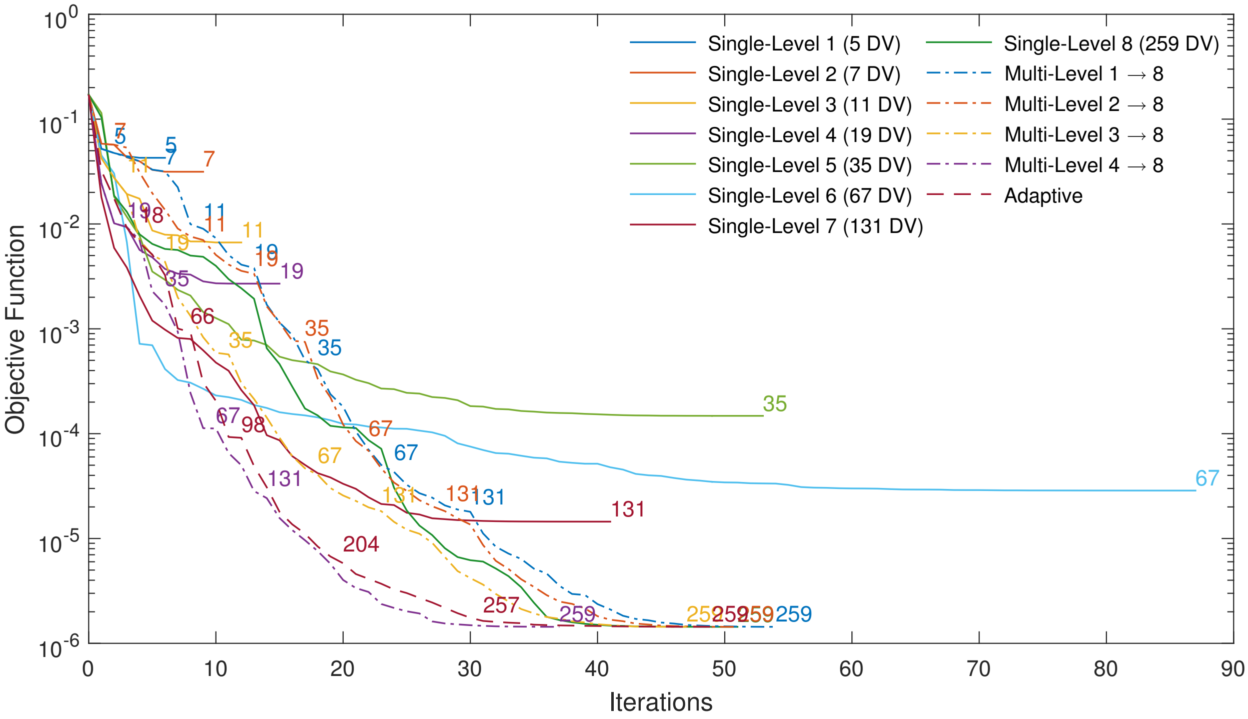


Figure 9. Results for the geometry matching optimisation from a NACA0012 to an ONERA M6. Numbers represent the number of design variables at the end of an optimisation phase.

B. Inverse Design

Three inverse design problems have also been considered to compare the subdivision aerofoil parametrisations. Each one starts from an initial NACA0012 then targets RAE2822, NACA4410 aerofoils with $\alpha = 0$ and an ONERA M6 with $\alpha = 3$. For all of these the objective function is defined as the root-mean-squared difference between the current and target pressure distributions, i.e.,

$$J = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(Cp_i - Cp_i^{target}\right)^2}. \tag{24}$$

The pressure distributions are calculated with a potential flow panel code and the gradients are calculated through finite-forward-differencing with a step size of 10^{-10}. The refinement parameters are $w = m = 1$ and $t = 0.1$ and the adaptive parameters are $l_{tol} = 10^{-2}$, $L_{max} = 4$ and $u = 2$.

Figures 10, 11, 12 show the inverse design test case results. For the single-level methods it can again be seen that, in general, the final results improve with increased fidelity. There are however some instances where this is not true, for example, for the NACA4410 case (figure 11) the 6th, 7th and 8th level methods fail to improve on the final result obtained using just the 5th level. This shows that the optimiser has failed to reach the global optimum. It can also be seen that the rate of convergence tends to increase with fidelity for these cases, in particular it can be seen for the RAE2822 case (figure 10) where the 'single-level 8' case takes a significantly longer to converge than any of the other methods. The multi-level methods converged to the optimum result consistently for all cases investigated and have better convergence rates than the high fidelity single-level cases. For all three test cases the adaptive method then showed further improvements to convergence rates, particularly for the high fidelity regions. Figure 13 plots the objective function obtained at the end of each optimisation phase against the number of design variables used. It shows that the adaptive cases always requires fewer design variables than the multi-level schemes to obtain a given objective function. In particular a result comparable to the optimum is achieved with between 162-175 design variables for all three test cases, compared to 259 required by the multi-level methods. This emphasises the efficiency of the design variables used by the adaptive parameterisation.

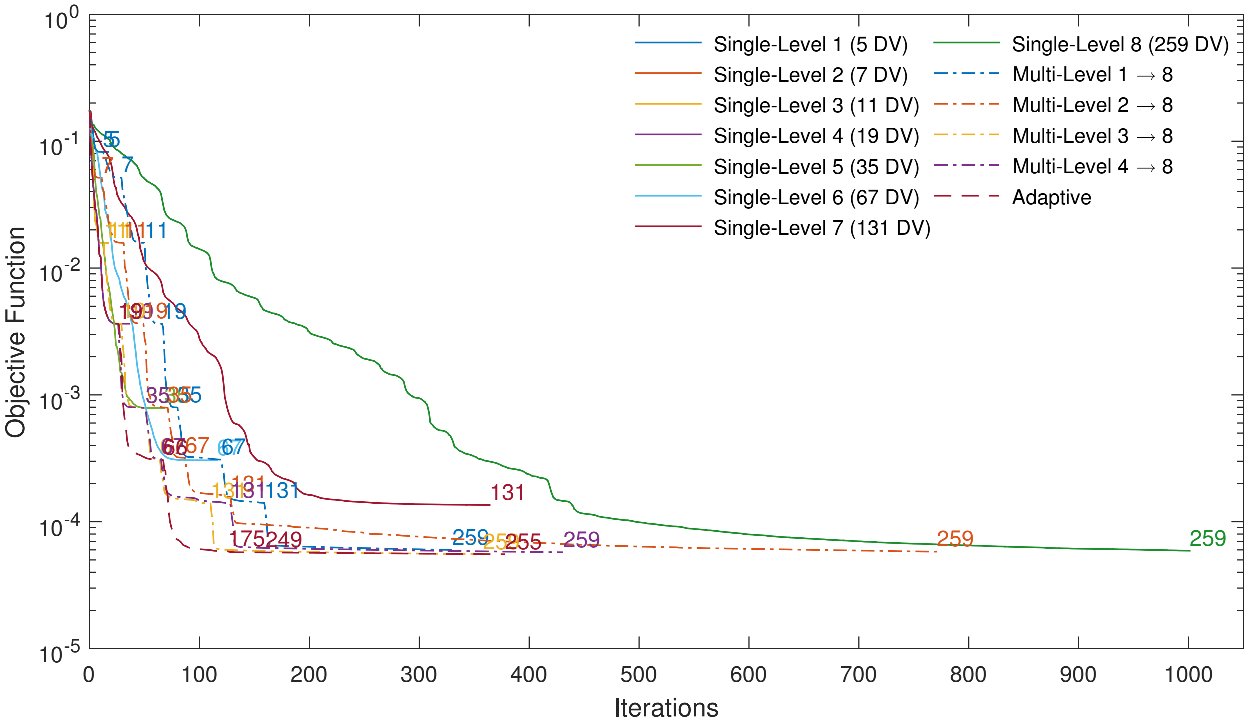


Figure 10. Results for the inverse design from a NACA0012 to an RAE2822 using a potential flow solver at $\alpha = 0$. Numbers represent the number of design variables at the end of an optimisation phase.

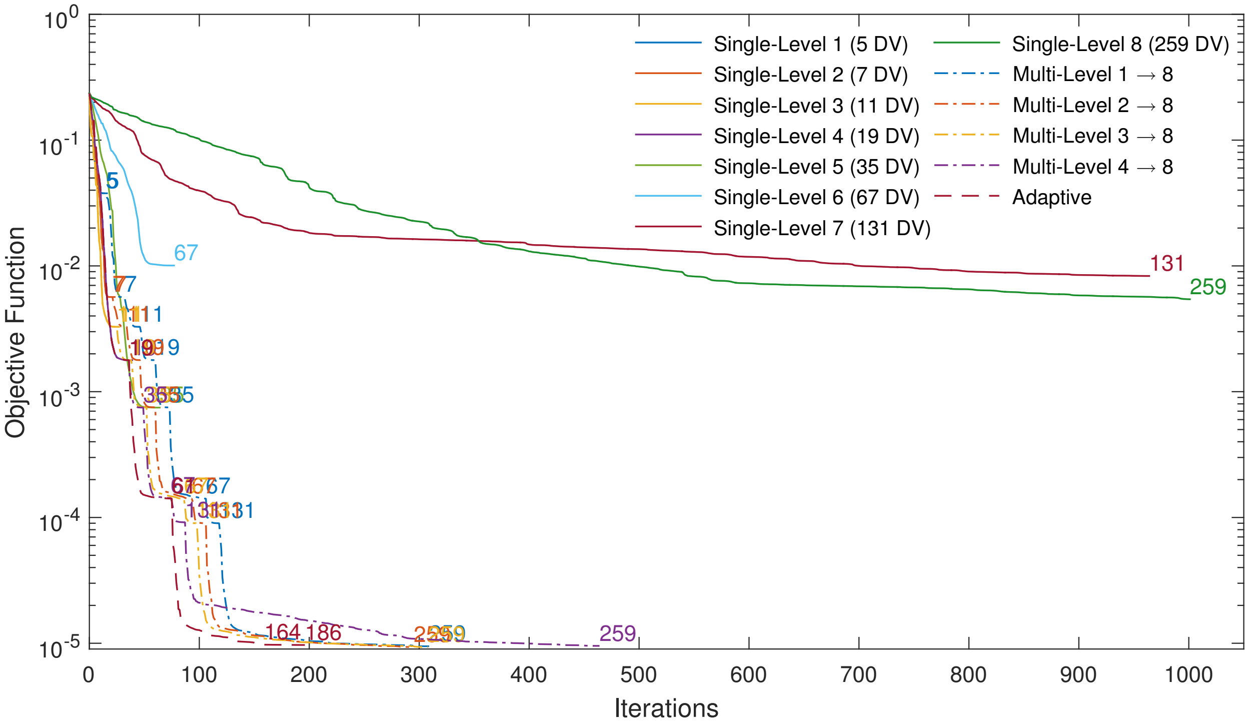


Figure 11. Results for the inverse design from a NACA0012 to a NACA4410 using a potential flow solver at $\alpha = 0$. Numbers represent the number of design variables at the end of an optimisation phase.

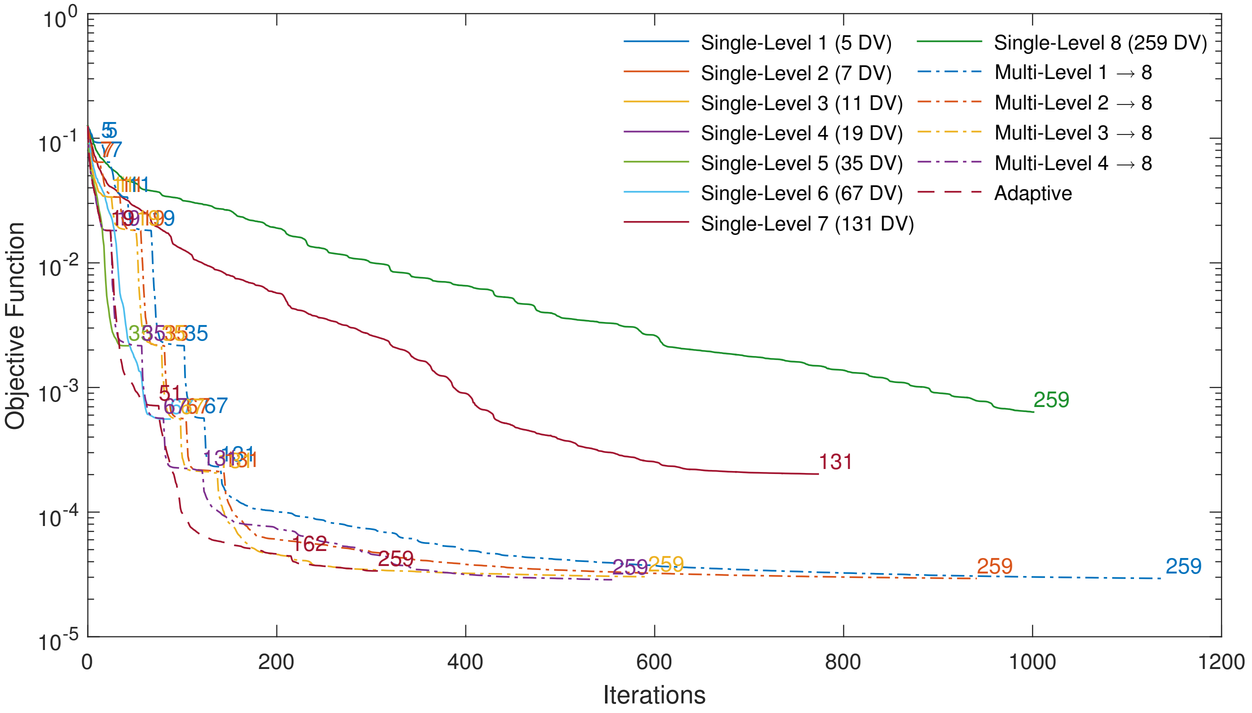


Figure 12. Results for the inverse design from a NACA0012 to an ONERA M6 using a potential flow solver at $\alpha = 3$. Numbers represent the number of design variables at the end of an optimisation phase.

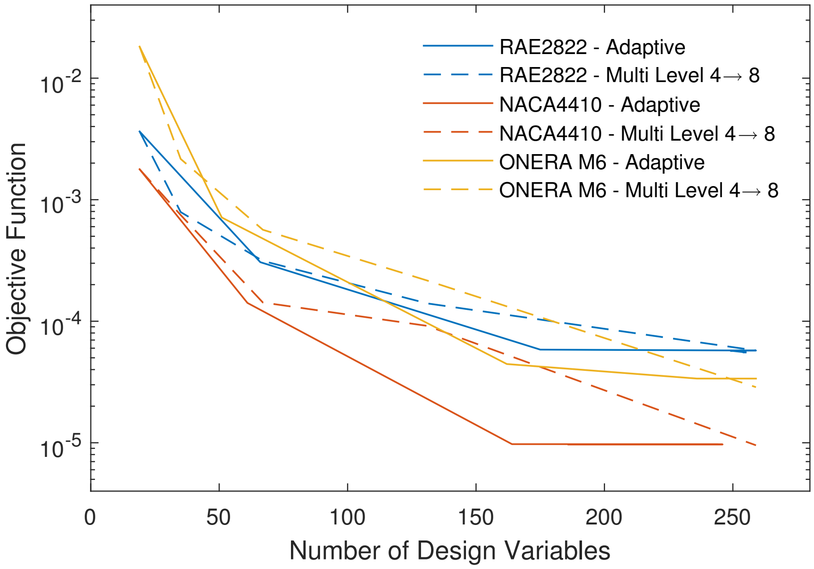


Figure 13. Comparison of objective function vs number of design variables for all three inverse design test cases.

C. Inviscid Drag Reduction

TEST CONFIGURATION

$$\begin{aligned}
\text{Initial Aerofoil:}\quad & NACA0012,\\
\text{Flow Conditions:}\quad & \alpha = 0,\\
& M = 0.85,\\
\text{Minimise:}\quad & C_D,\\
\text{Subject to:}\quad & C_L = 0,\\
& y \geq y_{NACA0012} \quad \forall x \in [0, 1],\\
\text{Refinement Parameters:}\quad & t = 0.1, \quad w = 3, \quad m = 3,\\
\text{Adaptive Parameters:}\quad & l_{tol} = 10^{-2}, \quad L_{max} = 3, \quad u = 1.
\end{aligned}$$

The final optimisation is a test case outline by the Aerodynamic Design Optimisation Discussion Group[41]. This is the inviscid drag reduction of a NACA0012 at $M = 0.85$ and $\alpha = 0$. A considerable number of papers have been published on this case[1,2,37,42–49] and it has been shown to be a particularly difficult problem to optimise due to a range of characteristics such as multiple local minima[37], non-symmetric solutions[48] and hysteresis[49].

These tests have been performed with the unstructured CFD solver SU^2 with the local thickness constraint applied as linearly in SNOPT at every chord-wise percentile. To calculate the design variable gradients the adjoint equations[50] are solved to calculate the surface sensitivities which are then projected, in the z direction, on to the design variables. The benefit of this method is that the calculation of the gradients is independent of the number of design variables used. Due to the symmetry of the problem a symmetry plane was aligned with the aerofoil chord and just half the mesh was solved. This meant that only half of the aerofoil surface, and therefore only half of the subdivision curve, needed to be modelled. For this reason each subdivision level has half the number of control points and design variables used in the other optimisation cases. The full optimisation study was run on a 257 × 257 half-mesh (figure 14), generated using a conformal mapping approach where all surface cells have aspect ratio one with a farfield distance of 50 chord lengths. The mesh was then deformed at every iteration using an RBF method[51] with a support radius of 10 and Wendland's C4 RBF[52]. This ensures the surface deformations are dissipated smoothly across the mesh to maintain mesh quality. The RBF deformations are also mirrored with respect to the x-axis to ensure the symmetry plane stays fixed throughout the optimisation. At each CFD run the solution was started from the previous best result to increase the chances of the lower branch of the hysteresis loop being found. This was found to be a largely successful approach and is believed to be the main contributor to the improvement in results compared to previous work from the authors[37,53].

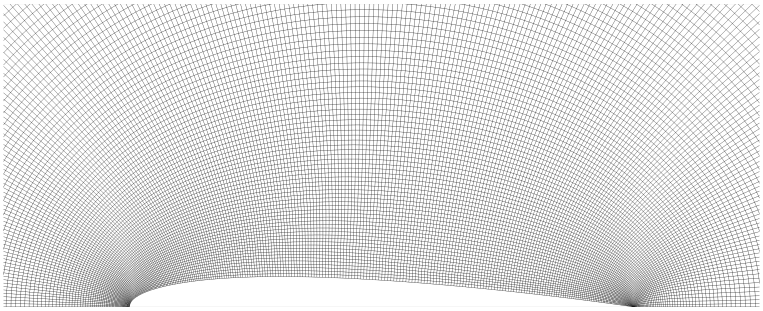


Figure 14. 257 × 257 **mesh used for the inviscid, symmetric NACA0012 optimisation.**

It was found that no improvement could be made at subdivision level 1 (with 2 design variables). For this reason only levels two and above are included in the results for this test case.

The results for this case are displayed in figure 15. The drag convergence results in figure 15a show that the single-level methods improve with fidelity up to level 5 after which they fail to successfully exploit the available design space. This can further be seen in figures 15b and 15c where the single-level results show significantly different aerofoil shapes and C_P distributions to the optimum found. In particular the aerofoil shapes for levels 6, 7 and 8 are not very smooth. The formation of these unsmooth shapes is likely to be the reason for their premature stagnation which is in agreement with the conclusions of Masters *et al.*[37]. It can then be seen that the multi-level methods all converge to good results and all surpass those from the single-level methods. In this case the '3 → 8' method produces the joint best result of 15.7 drag counts. Figures 15b and 15c show that all three of the multi-level methods converge to similar aerofoil shapes though there are some small discrepancies in the C_P distributions. Interestingly, even though the '2 → 8' and '4 → 8' methods converge to very similar drag values they have slightly different C_P distributions at the leading and trailing edges. This indicates that there may be some degree of multi-modality around the optimum solution. For this test case the adaptive method takes a very similar path to that of the '3 → 8' scheme, achieving an equivalent final result though it uses slightly fewer design variables in the latter optimisation stages.

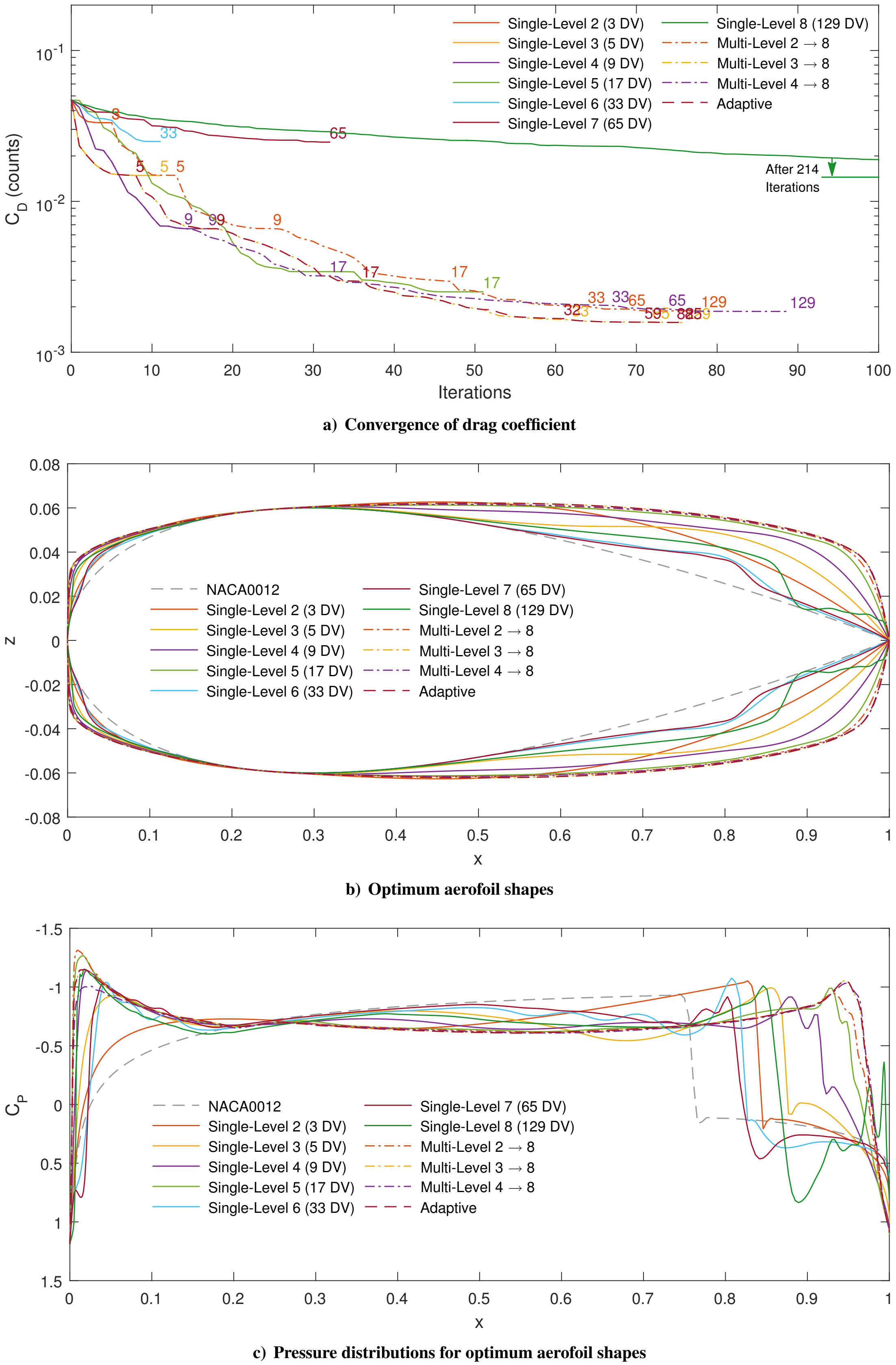
	Iterations	C_D (Counts)
Baseline	~	*469.4*
Single-Level 2 (3 DV)	6	332.2
Single-Level 3 (5 DV)	12	148.5
Single-Level 4 (9 DV)	18	65.9
Single-Level 5 (17 DV)	52	25.1
Single-Level 6 (33 DV)	12	246.5
Single-Level 7 (65 DV)	33	247.0
Single-Level 8 (129 DV)	214	144.7
Multi-Level 2→8	83	18.8
Multi-Level 3→8	81	15.7
Multi-Level 4→8	93	18.6
Adaptive	81	15.7

Table 2. Table of final drag results (in counts) for the inviscid NACA0012 optimisation at $\alpha = 0$ and $M = 0.85$ with 257×257 mesh.

A limited study was also done on the effects of increasing the mesh resolution. For this the '3 → 8' multi-level optimisation was run with increased mesh resolutions of '513×513' and '1025×513'. The result of these optimisations are displayed in figure 16 and show that the all three mesh resolutions produce similar results up to approximately 16 drag counts where the increased mesh resolution allows a further reduction in drag over the initial result. The final drag results are then summarised in table 3 which show that the finest mesh produces the best result of just 4.2 drag counts.

	Mesh Resolution	C_D (Counts)
Multi-Level 3→8	257×257	15.7
Multi-Level 3→8	513×513	8.2
Multi-Level 3→8	1025×513	4.2

Table 3. Table of final drag results (in counts) for the inviscid NACA0012 optimisation at $\alpha = 0$ and $M = 0.85$ for increased mesh resolution.



a) Convergence of drag coefficient

b) Optimum aerofoil shapes

c) Pressure distributions for optimum aerofoil shapes

Figure 15. Results for the symmetric inviscid optimisation of a NACA0012 at $\alpha = 0$ and $M = 0.85$ with 257×257 mesh.

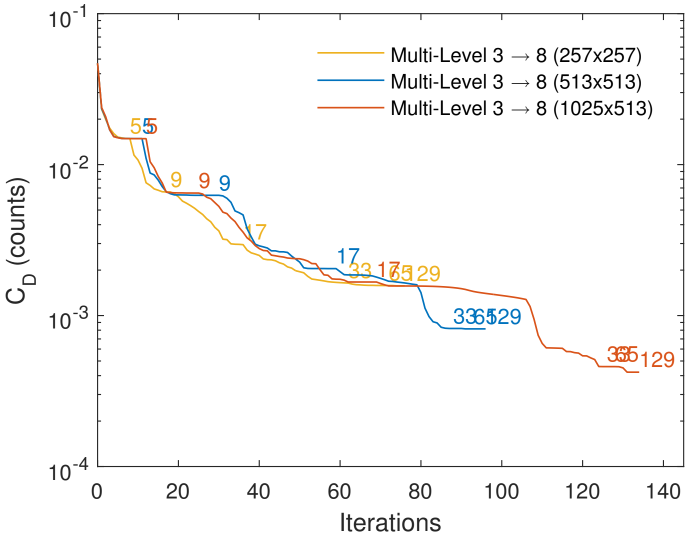


Figure 16. Results for the symmetric inviscid optimisation of a NACA0012 at $\alpha = 0$ and $M = 0.85$ with increased mesh resolution.

IX. Conclusion

In this work the benefits of using hierarchical subdivision parametrisations for aerofoil optimisation has been investigated. Three different parameterisation methods have been included: a 'multi-level' scheme that globally refines the parametrisation on optimiser convergence; an 'adaptive' scheme that locally refines and coarsens the parameterisation based on the adjoint surface sensitivities and a 'single-level' control group that uses no refinement. Each of these methods have been applied with a range a parameters.

Seven test cases have been investigated, three geometric shape matching case, three inverse design cases and the symmetric inviscid drag reduction of a NACA0012. These provide a range of tests that vary in computational requirement as well as complexity. The geometric shape matching cases represent the simplest of the optimisations. For all three of the tests it appears that all of the individual cases reach their global minimum. It was however found that by using the multi-level methods compared to the single-level methods the optima could be reached in fewer iterations. The adaptive method then provided additional improvement for one of the cases and comparable results for the other two.

The three inverse design cases then represent a significant increase in complexity compared to the geometry matching optimisations. For these cases it was found that some of the high-fidelity single-level methods stopped prematurely in local minima and failed to fully exploit the available design space. The multi-level methods, however, did not have this problem and always reached, what appears to be, the global minimum. They also showed better convergence rates than the equivalent single-level methods. The adaptive cases also reached the optimum in all cases, as well as showing further improvement in the convergence rates over the multi-level methods and requiring fewer design variables throughout the optimisation.

The final optimisation case, the inviscid, symmetric drag reduction of a NACA0012, represents the most complex optimisation case investigated. For this, the high fidelity single-level methods again struggled to navigate the design space and often stopped in local minima. The multi-level and adaptive cases however always converged to very good results with the 'multi-level 3 → 8' and adaptive cases achieving the best results of 15.7 drag counts. A limited study on the effects of increasing the mesh resolution was then performed. This showed that increasing the mesh resolution significantly reduced the final drag results with the finest '1025 × 513' mesh producing a result of 4.2 drag counts.

This work has shown that using the hierarchical multi-level and adaptive parameterisation can improve the robustness and rate of convergence of a range of optimisation cases when compared to the single-level results which are equivalent to cubic B-splines. They converged to the vicinity of the optimum result in all instances whereas the single-level methods often failed to reach the global minimum if large numbers of design variables were used. The adaptive method then showed that further improvements to convergence could be made through the efficient selection of design variables such that local fidelity was only increased where necessary.

X. Acknowledgements

This work was carried out using the computational facilities of the Advanced Computing Research Centre, University of Bristol - http://www.bris.ac.uk/acrc/. The authors also 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, doi: 10.2514/6.2014-0568.

[2] Poole, D. J., Allen, C. B., and Rendall, T. C. S., "Application of control point-based aerodynamic shape optimization to two-dimensional drag minimization," *52nd Aerospace Sciences Meeting*, 2014, doi: 10.2514/6.2014-0413.

[3] Masters, D. A., Taylor, N. J., Rendall, T. C. S., Allen, C. B., and Poole, D. J., "Review of Aerofoil Parameterisation Methods for Aerodynamic Shape Optimisation," *53rd AIAA Aerospace Sciences Meeting*, Jan 2015, doi: 10.2514/6.2015-0761.

[4] 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, doi: 10.1016/0045-7825(84)90132-4.

[5] Lépine, J., Trépanier, J.-Y., and Pépin, F., "Wing aerodynamic design using an optimized NURBS geometrical representation," *The 38th AIAA Aerospace Sciences Meeting and Exhibit*, No. AIAA–2000–699, Reno, Nevada, 2000.

[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] Kulfan, B. M., "A universal parametric geometry representation method-CST," *45th AIAA Aerospace Sciences Meeting and Exhibit*, January 2007.

[8] Hicks, R. M. and Henne, P. A., "Wing design by numerical optimization," *Journal of Aircraft*, Vol. 15, No. 7, 1978, pp. 407–412, doi: 10.2514/3.58379.

[9] 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.

[10] Sobieczky, H., "Parametric airfoils and wings," *Recent Development of Aerodynamic Design Methodologies*, Springer, 1999, pp. 71–87.

[11] 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, doi: 10.1002/fld.1769.

[12] 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, doi: 10.1243/09544100JAERO393.

[13] 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, doi: 10.1007/s11081-011-9179-6.

[14] Chauhan, D., Praveen, C., and Duvigneau, R., "Wing shape optimization using FFD and twist parameterization," *12th Aerospace Society of India CFD Symposium*, 2010.

[15] Poole, D. J., Allen, C. B., and Rendall, T. C. S., "Metric-based mathematical derivation of efficient airfoil design variables," *AIAA Journal*, Vol. 53, No. 5, 2015, pp. 1349–1361, doi: 10.2514/1.J053427.

[16] 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, doi: 10.2514/1.41420.

[17] Jameson, A., "Aerodynamic design via control theory," *Journal of Scientifical Computing, also ICASE Report No.88-64*, Vol. 3, 1988, pp. 233–260, doi: 10.1007/BF01061285.

[18] Beux, F. and Dervieux, A., "A hierarchical approach for shape optimization," *Engineering Computations*, Vol. 11, No. 1, 1994, pp. 25–48.

[19] Desideri, J. and Zolesio, J., "Inverse shape optimization problems and application to airfoils," *Control and Cybernetics*, Vol. 34, No. 1, 2005, pp. 165.

[20] Desideri, J. and Dervieux, A., "Hierarchical methods for shape optimization in aerodynamics I: Multilevel algorithms for parametric shape optimization," *LECTURE SERIES-VON KARMAN INSTITUTE FOR FLUID DYNAMICS*, Vol. 3, 2006, pp. 10.

[21] Désidéri, J.-A., El Majd, B. A., and Janka, A., "Nested and self-adaptive Bézier parameterizations for shape optimization," *Journal of Computational Physics*, Vol. 224, No. 1, 2007, pp. 117–131.

[22] Martinelli, M. and Beux, F., "Multi-level gradient-based methods and parametrisation in aerodynamic shape design," *European Journal of Computational Mechanics/Revue Européenne de Mécanique Numérique*, Vol. 17, No. 1-2, 2008, pp. 169–197.

[23] Andreoli, M., Ales, J., and Désidéri, J.-A., "Free-form-deformation parameterization for multilevel 3D shape optimization in aerodynamics," 2003.

[24] Duvigneau, R., Chaigne, B., and Désidéri, J.-A., "Multi-level parameterization for shape optimization in aerodynamics and electromagnetics using a particle swarm optimization algorithm," 2006.

[25] El Majd, B. A., Duvigneau, R., and Désidéri, J., "Aerodynamic shape optimization using a full and adaptive multilevel algorithm," *ERCOFTAC Conference Design Optimization: Methods and Applications, Canary Island, Spain*, 2006.

[26] Anderson, G. R. and Aftosmis, M. J., "Adaptive Shape Control for Aerodynamic Design," *56th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Jan 2015.

[27] Han, X. and Zingg, D. W., "An adaptive geometry parametrization for aerodynamic shape optimization," *Optimization and Engineering*, Vol. 15, No. 1, 2014, pp. 69–91.

[28] Sherar, P., Thompson, C., Xu, B., and Zhong, B., "An Optimization Method Based On B-spline Shape Functions & the Knot Insertion Algorithm." *World congress on engineering*, Citeseer, 2007, pp. 862–866.

[29] DeRose, T., Kass, M., and Truong, T., "Subdivision surfaces in character animation," *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, ACM, 1998, pp. 85–94.

[30] Chaikin, G., "An algorithm for high-speed curve generation," *Graphical Models /graphical Models and Image Processing /computer Vision, Graphics, and Image Processing*, Vol. 3, 1974, pp. 346–349.

[31] Cashman, T. J., Hormann, K., and Reif, U., "Generalized Lane–Riesenfeld algorithms," *Computer Aided Geometric Design*, Vol. 30, No. 4, 2013, pp. 398–409.

[32] Forsey, D. R. and Bartels, R. H., "Hierarchical B-spline refinement," *ACM SIGGRAPH Computer Graphics*, Vol. 22, ACM, 1988, pp. 205–212.

[33] Finkelstein, A. and Salesin, D. H., "Multiresolution curves," *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, ACM, 1994, pp. 261–268.

[34] Sabin, M., "Eigenanalysis and artifacts of subdivision curves and surfaces," *Tutorials on multiresolution in geometric modelling*, Springer, 2002, pp. 69–92.

[35] Jameson, A., "Aerodynamic shape optimization using the adjoint method," *Lectures at the Von Karman Institute, Brussels*, 2003.

[36] Schmidt, S., Ilic, C., Gauger, N., and Schulz, V., "Shape gradients and their smoothness for practical aerodynamic design optimization," *Optim. Eng.(20 08) Preprint-Number SPP1253-10-03*, 2008.

[37] Masters, D. A., Taylor, N. J., Rendall, T. C. S., and Allen, C. B., "Impact of Shape Parameterisation on Aerodynamic Optimisation of Benchmark Problems," *54rd AIAA Aerospace Sciences Meeting*, Jan 2016.

[38] Gill, P. E., Murray, W., and Saunders, M. A., "SNOPT: An SQP algorithm for large-scale constrained optimization," *SIAM journal on optimization*, Vol. 12, No. 4, 2002, pp. 979–1006.

[39] Palacios, F., Alonso, J., Duraisamy, K., Colonno, M., Hicken, J., Aranake, A., Campos, A., Copeland, S., Economon, T., Lonkar, A., Lukaczyk, T., and Taylor, T., "Stanford University Unstructured (SU^2): An open-source integrated computational environment for multi-physics simulation and design," *Aerospace Sciences Meetings*, American Institute of Aeronautics and Astronautics, Jan. 2013.

[40] Palacios, F., Economon, T. D., Aranake, A. C., Copeland, S. R., Lonkar, A. K., Lukaczyk, T. W., Manosalvas, D. E., Naik, K. R., Padrón, A. S., Tracey, B., et al., "Stanford University Unstructured (SU2): Open-source analysis and design technology for turbulent flows," *AIAA paper*, Vol. 243, 2014, pp. 13–17.

[41] Nadarajah, S., "Aerodynamic Design Optimization: Drag Minimization of the NACA 0012 in Transonic Inviscid Flow," Retreived from https://info.aiaa.org/tac/ASG/APATC/AeroDesignOpt-DG/Test%20Cases/ADODG%20Case%201%20and%202%20NACA0012%20and%20RAE%202822.pdf.

[42] Zhang, M., Rizzi, A. W., and Nangia, R. K., "Transonic Airfoils and Wings Design Using Inverse and Direct Methods," *53rd AIAA Aerospace Sciences Meeting*, Jan 2015.

[43] Vassberg, J., Harrison, N., Roman, D., and Jameson, A., "A Systematic Study on the Impact of Dimensionality for a Two-Dimensional Aerodynamic Optimization Model Problem," *29th AIAA Applied Aerodynamics Conference*, Jun 2011.

[44] Bisson, F. and Nadarajah, S., "Adjoint-Based Aerodynamic Optimization of Benchmark Problems," *52nd Aerospace Sciences Meeting*, Jan 2014.

[45] Nadarajah, S., "Adjoint-Based Aerodynamic Optimization of Benchmark Problems," *53rd AIAA Aerospace Sciences Meeting*, Jan 2015.

[46] Poole, D. J., Allen, C. B., and Rendall, T. C. S., "Control Point-Based Aerodynamic Shape Optimization Applied to AIAA ADODG Test Cases," *53rd AIAA Aerospace Sciences Meeting*, Jan 2015.

[47] Anderson, G. R., Nemec, M., and Aftosmis, M. J., "Aerodynamic Shape Optimization Benchmarks with Error Control and Automatic Parameterization," *53rd AIAA Aerospace Sciences Meeting*, Jan 2015.

[48] Lee, C., Koo, D., Telidetzki, K., Buckley, H., Gagnon, H., and Zingg, D. W., "Aerodynamic Shape Optimization of Benchmark Problems Using Jetstream," *53rd AIAA Aerospace Sciences Meeting*, 2015.

[49] Meheut, M., Destarac, D., Carrier, G., Anderson, G., Nadarajah, S., Poole, D., Vassberg, J., and Zingg, D. W., "Gradient-Based Single and Multi-points Aerodynamic Optimizations with the elsA Software," *53rd AIAA Aerospace Sciences Meeting*, Jan 2015.

[50] Economon, T. D., Palacios, F., and Alonso, J. J., "A viscous continuous adjoint approach for the design of rotating engineering applications," *AIAA Paper*, Vol. 2580, 2013, pp. 24–27.

[51] 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, doi: 10.1002/nme.2219.

[52] Wendland, H., *Scattered data approximation*, Cambridge University Press Cambridge, 2005.

[53] Masters, D. A., Taylor, N. J., Rendall, T. C. S., and Allen, C. B., "Progressive Subdivision Curves for Aerodynamic Shape Optimisation," *54rd AIAA Aerospace Sciences Meeting*, Jan 2016.