UNIVERSITY OF SOUTHAMPTON

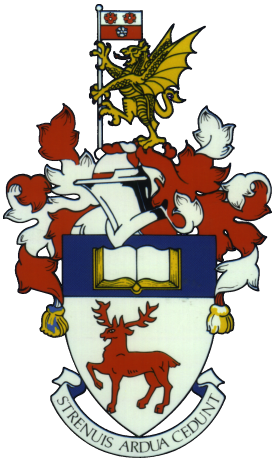
FACULTY OF ENGINEERING AND PHYSICAL SCIENCES

Aerodynamics and Flight Mechanics

On the Sensitivity and Efficiency of Aerodynamic Shape Optimisation

by

Guangda Yang



Thesis for the degree of Doctor of Philosophy

October 2019

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING AND PHYSICAL SCIENCES
Aerodynamics and Flight Mechanics

Doctor of Philosophy

ON THE SENSITIVITY AND EFFICIENCY OF AERODYNAMIC SHAPE OPTIMISATION

by Guangda Yang

Computational fluid dynamics (CFD) has become the method of choice for aerodynamic shape optimisation of complex engineering problems. To date, however, the sensitivity of the optimal solution to numerical parameters has been largely underestimated. Meanwhile, aerodynamic shape optimisation based on high-fidelity CFD remains a computationally expensive task. The thesis consists of two research streams aimed at addressing each of the challenges identified, namely revisiting the optimal solution and developing an efficient optimisation framework. This work primarily focuses on the assessment of optimal design sensitivity and computational efficiency in gradient-based optimisation of aeronautical applications.

Two benchmark cases for NACA0012 and RAE2822 aerofoil optimisation are investigated using the open-source SU2 code. Hicks–Henne bump functions and free-form deformation are employed as geometry parameterisation methods. Gradients are computed by the continuous adjoint approach. The optimisation results of NACA0012 aerofoil exhibit strong dependence on virtually all numerical parameters investigated, whereas the optimal design of RAE2822 aerofoil is insensitive to those parameter settings. The degree of sensitivity reflects the difference in the design space, particularly of the local curvature on the optimised shape. The closure coefficients of Spalart–Allmaras model affect the final optimisation performance, raising the importance of quantifying uncertainty in turbulence modelling calibration. Non-unique flow solutions are found to exist for both cases, and hysteresis occurs in a narrow region near the design point.

Wing twist optimisations are conducted using two aerodynamic solvers of different levels of fidelity. A multi-fidelity aerodynamic approach is proposed, which contains three components: a linear vortex lattice method solver, an infinite swept wing solver, and a coupling algorithm. For reference, three-dimensional data are obtained using SU2. Two optimisation cases are considered, featuring inviscid flow around an unswept wing and viscous flow around a swept wing. A good agreement in terms of lift distribution and aerodynamic shape between the multi-fidelity solver and high-fidelity CFD is obtained. The numerical optimisation using the multi-fidelity approach is performed at a negligible computational cost compared to the full three-dimensional CFD solver, demonstrating the potential for use in early phases of aircraft design.

Contents

List of Figures

List of Tables

Declaration of Authorship

I, Guangda Yang , declare that the thesis entitled *On the Sensitivity and Efficiency of Aerodynamic Shape Optimisation* and the work presented in the thesis are both my own, and have been generated by me as the result of my own original research. I confirm that:

- this work was done wholly or mainly while in candidature for a research degree at this University;
- where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
- where I have consulted the published work of others, this is always clearly attributed;
- where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
- I have acknowledged all main sources of help;
- where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
- parts of this work have been published, as listed in Section 1.3

Signed:.....

Date:.....

Acknowledgements

First and foremost I would like to thank my supervisors Dr Andrea Da Ronch and Dr Zheng-Tong Xie. Thank Dr Andrea Da Ronch for his invaluable support and continuous guidance throughout the course of my Ph.D study. His diligence and insight into research have a profound impact on me in pursuing the future career. Thank Dr Zheng-Tong Xie for providing timely advices and having valuable discussions. His integrity and professionalism have also influenced me a lot over the last four years.

I would like to acknowledge the financial support from the University of Southampton and China Scholarship Council (CSC). I also gratefully acknowledge the IRIDIS high performance computing facility and the associated services at the University of Southampton, in the completion of this work.

I would like to thank my colleagues in our research group for their useful discussions and the memorable time we spent together. Thanks go to Dr Jernej Drofelnik, Muhammad Anas Abd Bari, Daniel Kharlamov, Benjamin Moss, Riccardo Rubini, Marco Saporito.

Special thanks go to two of my best friends, Meng Wang and Chao You, who shared so much with me about research and life, enriching my experience in Southampton.

Last but not least, I would like to thank my parents and family for everything else.

Nomenclature

Alphanumeric

a	Bump amplitude in HHBF
b	Wing span
C_D, C_L	Wing drag and lift coefficients
C_{L_α}	Wing lift curve slope
C_d, C_l, C_m	Aerofoil drag, lift and pitching moment coefficients
C_{d_α}, C_{l_α}, C_{m_α}	Aerofoil drag, lift and pitching moment curve slopes
C_p	Pressure coefficient
c	Chord length
c_p	Specific heat at constant pressure
D	Drag force
E	Total energy density or Young's modulus
e	Span efficiency of wing
h	Bump peak position in HHBF
L	Lift force
M	Mach number
N_{dv}	Number of design variables
N_x	Number of VLM panels in chordwise direction
N_y	Number of VLM panels in spanwise direction
Pr	Prandtl number
p	Pressure
Re	Reynolds number
S	Surface area
T	Temperature
Tr	Trace of a matrix
t	Time or bump width control parameter in HHBF
u, v, w	Velocity components in x, y, and z directions
u', v', w'	Velocity components in x', y', and z' directions
V	Velocity magnitude
w	Downwash velocity magnitude
x, y, z	Cartesian coordinates in global FoR

x', y', z'	Cartesian coordinates in local FoR
y^+	Non-dimensional off-wall distance

Greek

α	Angle of attack
ϵ	Convergence tolerance
γ	Twist design variable or ratio of specific heats
κ	Surface curvature or SA turbulence model closure coefficient
Λ	Sweep angle
μ	Turbulence viscosity
ν	Pseudo eddy viscosity or Poisson's ratio or relaxation factor
ω	Vorticity magnitude
ρ	Density
θ	Wing twist angle
ξ, η, ζ	Parametric coordinates in FFD volume

Symbols

$\cdot$	Vector inner product
$\times$	Vector cross product
$\bigotimes$	Vector outer product
$\nabla(\cdot)$	Gradient operator
$\nabla \cdot (\cdot)$	Divergence operator
$\nabla \times (\cdot)$	Curl operator

Abbreviations

2D	Two-Dimensional
3D	Three-Dimensional
AD	Algorithmic Differentiation or Automatic Differentiation
ADODG	Aerodynamic Design Optimisation Discussion Group
AIC	Aerodynamic Influence Coefficient
AR	Aspect Ratio
ASO	Aerodynamic Shape Optimisation
BFGS	Broyden–Fletcher–Goldfarb–Shanno
BWB	Blended Wing Body
CAD	Computer Aided Design
CFD	Computational Fluid Dynamics
CFL	Courant–Friedrichs–Lewy
CRM	Common Research Model

CST	Class Shape Transformation
DGM	Delaunay Graph Mapping
DLM	Doublet Lattice Method
DoE	Design of Experiment
FALCon	Fast Aircraft Load CalculatiONs
FDM	Finite Difference Method
FFD	Free-Form Deformation
FoR	Frame of Reference
FSQP	Feasible Sequential Quadratic Programming
FVM	Finite Volume Method
GA	Genetic Algorithm
HHBF	Hicks–Henne Bump Function
IATA	International Air Transport Association
ICAO	International Civil Aviation Organisation
ISW	Infinite Swept Wing
JST	Jameson–Schmidt–Turkel
KKT	Karush–Kuhn–Tucker
LE	Leading Edge
LLT	Lifting-Line Theory
MDO	Multidisciplinary Design Optimisation
NURBS	Non-Uniform Rational B-Splines
PARSEC	PARameterised SECtions
PDE	Partial Differential Equation
POD	Proper Orthogonal Decomposition
PSO	Particle Swarm Optimisation
Q3D	Quasi-Three-Dimensional
RANS	Reynolds-Averaged Navier–Stokes
RBF	Radial Basis Function
RHS	Right Hand Side
RSM	Response Surface Model
SA	Spalart–Allmaras
SLSQP	Sequential Least SQuares Programming
SNOPT	Sparse Nonlinear OPTimiser
SQP	Sequential Quadratic Programming
SVD	Singular Value Decomposition
TE	Trailing Edge
TFI	TransFinite Interpolation
UQ	Uncertainty Quantification
VLM	Vortex Lattice Method

Chapter 1

Introduction

1.1 Background and motivation

In recent years there has been a rapid and steady growth in the volume of air traffic, especially for commercial air transportation. This trend is anticipated to continue in the foreseeable future. According to the latest Airbus' Global Market Forecast[1], the air traffic is predicted to grow at an annual rate of 4.4% and more than double in the next 20 years. Meanwhile, there will be a demand for around 37,400 new passenger and freight aircraft as shown in Figure 1.1.

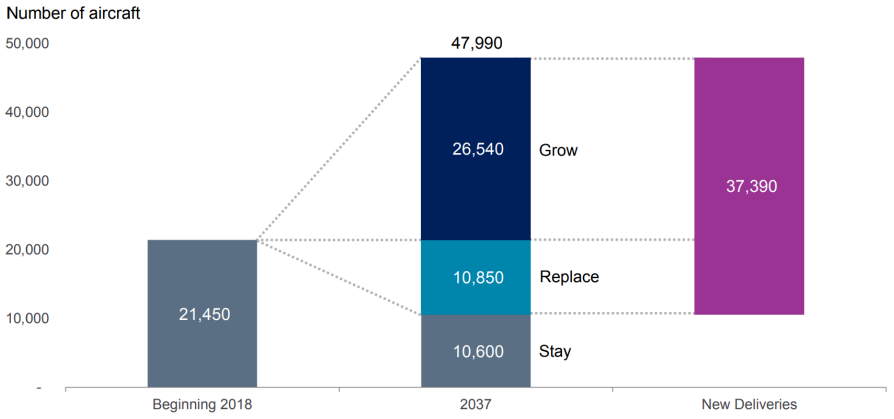


Figure 1.1: Forecast of deliveries of new aircraft within 2018-2037 (Source: Airbus Global Market Forecast 2018).

Continuing growth in the aviation industry has led to carbon dioxide (CO_2) emissions from aircraft becoming a significant contributor to the climate change, raising public concern. In order to address the impact of aviation greenhouse gas emissions on the global climate, the International Civil Aviation Organisation (ICAO) has adopted a new standard for aircraft CO_2 emissions. The International Air Transport Association

[1] https://www.airbus.com/aircraft/market/global-market-forecast.html [retrieved 2019]

(IATA)[2] has also adopted a set of targets to mitigate the carbon emissions from air transport, including setting a limit on net aviation CO_2 emissions from 2020 (i.e. carbon-neutral growth) and achieving a reduction in carbon emissions of 50% by 2050 (relative to 2005 levels). As such, many aircraft and jet engine companies are making efforts to deliver innovative fuel reduction technologies for next generation aircraft. Moreover, as fuel costs account for around 20% of flight operating expenses, oil price variability reduces profit margins of airlines. The uncertainty in fuel price therefore forces airlines to purchase the most efficient aircraft to remain competitive. It is the presence of these two factors, environmental goals and economic considerations, that aircraft manufacturers must continue to push technological boundaries to reduce fuel burn.

To address the challenge of reducing fuel consumption, designers are devoted to developing new techniques and methodologies in diverse fields, such as employing more efficient jet engines and applying composite materials more widely. Of the various considerations, one key aspect is to improve the aerodynamic efficiency by redesigning the aerodynamic shape of aircraft.

The endeavours towards aerodynamic design generally fall into two categories: exploitation and exploration. Exploitation represents the common practices existing in the iterative design process of an aircraft, and it seeks to make improvement and refinement for the conventional tube-and-wing aircraft configurations. Two typical examples of this type are shown in Figure 1.2. The Airbus A350 XWB integrates state-of-the-art aerodynamics, including a unique morphing technology that continuously tailors the wing profile during different phases of flight to reduce the drag. This aircraft also adopts a blended winglet instead of traditional wingtip fence, which reduces interference drag at the seams between the wing and the winglet. An outstanding feature of the Boeing 777X is the design of folding wingtips. When the wingtips are fully extended in flight, the increased span reduces induced drag and maximises fuel efficiency. On the ground, the aircraft can maintain gate compatibility by retracting the wingtips.

(a) Airbus A350 XWB (b) Boeing B777X

Figure 1.2: Examples of state-of-the-art aircraft representing exploitation.

[2] https://www.iata.org/policy/environment/Pages/climate-change.aspx [retrieved 2019]

On top of exploiting the full potential of tube-and-wing configuration, researchers are also exploring new regions that go beyond the limits of conventional designs. Preliminary studies and wind tunnel tests have shown that some proposed designs are promising for achieving fuel burn reduction. Figure 1.3 shows the view of a blended wing body (BWB) concept, which merges efficient high-lift wings with a wide aerofoil-shaped body, allowing the entire aircraft to generate lift and minimise drag. The BWB airframe can produce a significantly high lift-to-drag ratio and thus improve the fuel economy.

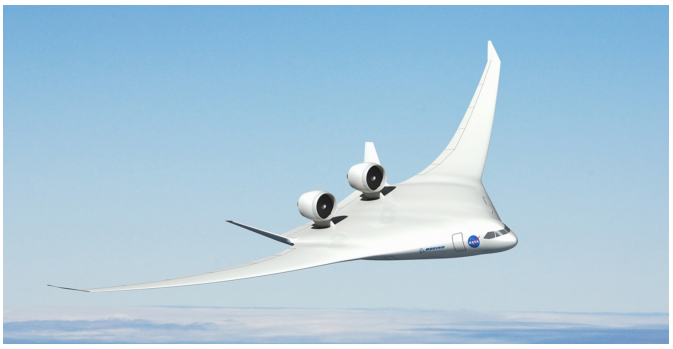


Figure 1.3: BWB concept proposed by Boeing/NASA.

With the rapid development of computer technology and computational fluid dynamics (CFD), numerical optimisation has shown the potential to be a powerful tool in fulfilling the task of fuel burn reduction. By integrating a CFD solver with an optimisation algorithm, geometry parameterisation and grid deformation techniques, designers are enabled to perform aerodynamic shape optimisation (ASO) with ease. A robust ASO framework allows both exploitation and exploration: exploitation of the design space for traditional aircraft configurations as well as exploration of unconventional designs with the potential for bigger improvements in aerodynamic performance.

A number of ASO frameworks and packages have been developed in the research community over the past years. Various approaches and tools have been used for each component of an ASO framework. For benchmarking purposes, the AIAA Aerodynamic Design Optimisation Discussion Group (ADODG)[3] has established a set of aerodynamic optimisation problems with increasing complexity, ranging from single-point aerofoil optimisation to multi-point wing-body-tail optimisation. Many research groups have exercised the benchmark problems, presented the optimisation results, and highlighted their own contributions.

As with any optimisation problem, following CFD-based aerodynamic optimisation, two questions would naturally arise: a) Is the optimal solution obtained from an ASO framework truly optimal? b) Can the aerodynamic optimisation be performed more efficiently, especially for aeronautical applications with relatively more complexity? These two questions reflect the common interest of how to obtain the best aerodynamic design within

[3] https://info.aiaa.org/tac/ASG/APATC/AeroDesignOpt-DG/default.aspx [retrieved 2016]

a restricted optimisation budget. Despite a great deal of research into these two areas, some problems still remain unsolved and warrant further study.

With respect to the first question of optimality, much effort has been put into finding the global optimum so as not to get trapped in the local optima. However, for a given optimisation framework, the numerical parameters existing in each constitutive part may potentially affect the final results. Thus, there is the need to revisit the optimal solution by tuning the numerical parameters involved in optimisation.

Regarding the second question of optimisation efficiency, the computational efficiency of the numerical optimisation can be improved in several aspects. For instance, the grid regeneration can be a time-consuming process, and therefore it is usually replaced by robust grid deformation techniques. For gradient-based optimisation cases with large numbers of design variables, the gradient evaluation using finite difference method (FDM) is computationally expensive. The recent applications of adjoint approach have significantly eased the burden of gradient computation as it overcomes the "curse of dimensionality". Nonetheless, the computational cost of CFD simulations for complex aerodynamic geometries remains expensive even with high-performance computing capability. This prohibits the routine use of high-fidelity CFD solvers in industry, as a large amount of repetitive aerodynamic evaluations are required in the early phases of aircraft design. Although a surrogate model can provide a good approximation, there is still the need to develop physics-based aerodynamic approaches with high efficiency. Recently the multi-fidelity aerodynamic models, particularly the quasi-three-dimensional (Q3D) approach, gain popularity in aerodynamic analysis and design optimisation due to their ability to capture the flow physics and to perform rapidly the aerodynamic analysis.

1.2 Research objectives

This research focuses on the assessment of sensitivity and efficiency in gradient-based ASO for aeronautical applications, addressing the two issues existing in ASO, namely revisiting the optimal solution and improving optimisation efficiency. The primary aim of this research is two-fold: to establish the sensitivity of the optimal solution to numerical parameters, particularly those in geometry parameterisation, and to demonstrate the efficiency of a multi-fidelity aerodynamic solver in wing twist optimisation. The key objectives of this work are summarised as follows.

1. **Drag minimisation of two-dimensional aerofoils**
   Two-dimensional (2D) aerofoils represent the cross-sections of a wing, and the design of the aerofoil is crucial for the aerodynamic performance of an aircraft. The two benchmark cases of aerofoil optimisation defined by the AIAA ADODG are studied using the SU2 solver. The first case is the drag minimisation of the NACA0012 aerofoil

in transonic, inviscid flow, with a minimum thickness constraint; the second case is the drag minimisation of the RAE2822 aerofoil in transonic, viscous flow, subject to lift, pitching moment and area constraints. The Hicks–Henne bump function (HHBF) and free-form deformation (FFD) are employed as geometry parameterisation methods. This part of the work aims to investigate:

(a) The influence of the geometry parameterisation method on the optimisation performance.

(b) The sensitivity of the optimal solution to numerical settings in geometry parameterisation.

(c) The impact of closure coefficients of Spalart–Allmaras (SA) turbulence model on the optimisation results.

(d) The impact of design variable dimensionality on the optimal solution.

(e) The robustness of the optimal solution at conditions around the design point.

2. **Twist optimisation of three-dimensional wings**
Three-dimensional (3D) wings are the primary lifting surfaces of an aircraft. The finiteness of the span causes lift-induced drag due to the pressure difference between the lower and upper surface. Minimisation of the induced drag can be achieved by tailoring the wing twist distribution in the spanwise direction. This research firstly studies the AIAA ADODG benchmark case of twist optimisation of an unswept wing in inviscid flow, and then extends to twist optimisation of a swept wing in viscous flow. Two aerodynamic solvers are employed in this work. The first solver is the high-fidelity SU2 code, and the second is the multi-fidelity FALCon solver. In this part of research, the main objectives are to:

(a) Develop an optimisation framework based on the multi-fidelity FALCon solver.

(b) Verify the computational accuracy of FALCon and its effectiveness in wing twist optimisation.

(c) Demonstrate the high computational efficiency of FALCon relative to that of SU2.

(d) Investigate the impact of design variable dimensionality on the optimal solution.

(e) Investigate the influence of twist interpolation and geometry control methods on the optimisation results.

1.3 Original contributions

The contributions of this thesis are centred around two research streams. The first part investigates the sensitivity of optimal solution to a series of numerical parameters for both 2D aerofoil and 3D wing aerodynamic optimisations. It employs a high-fidelity

computational aerodynamic approach based on full 3D Euler/Navier–Stokes equations utilising high-performance computing facility. The second part focuses on the development, implementation, and application of an efficient multi-fidelity aerodynamic solver to wing twist optimisation problems. Some parts of the work in this thesis have been published as conference proceedings and journal articles, and some are also intended for publications. The contributions of this research are summarised as follows:

1. **Drag minimisation of two-dimensional aerofoils**
   The influence that numerical parameters have on the optimisation results for two aerofoil problems (NACA0012 and RAE2822) in transonic flow is investigated. Gradient-based numerical optimisations are performed using the open-source tool SU2, and gradients are computed using the continuous adjoint method. The optimisation result of NACA0012 aerofoil exhibits strong dependence on all numerical parameters investigated, whereas the optimal design of RAE2822 aerofoil in insensitive to those numerical settings. The degree of sensitivity reflects the difference in the design space, particularly of the local curvature on the optimised shape. The closure coefficients of the SA turbulence model affect the final optimisation performance, raising the need for a good calibration of the turbulence model. Non-uniqueness of flow solutions for optimised aerofoil geometries is found, and a hysteresis loop of the drag coefficient exists around the design point, indicating the need for a robust aerofoil design procedure.

   Guangda Yang, Andrea Da Ronch, Jernej Drofelnik, and Zheng-Tong Xie, "Sensitivity Assessment of Optimal Solution in Aerodynamic Design Optimisation using SU2", *Aerospace Science and Technology*, vol. 81, pp. 362-374, 2018, DOI: 10.1016/j.ast.2018.08.012.

   Guangda Yang, Andrea Da Ronch, "Aerodynamic Shape Optimisation of Benchmark Problems Using SU2", in *AIAA SciTech*, Kissimmee, United States, AIAA 2018-0412, 2018.

2. **Twist optimisation of three-dimensional wings**
   An efficient optimisation framework based on a multi-fidelity aerodynamic solver, FALCon (**F**ast **A**ircraft **L**oad **C**alculati**on**s) [1], is developed and applied to wing twist optimisation. The multi-fidelity aerodynamic approach consists of three main components: a linear vortex lattice method (VLM) solver, an infinite swept wing (ISW) solver of steady/unsteady Reynolds-averaged Navier–Stokes (RANS) equations, and a coupling algorithm. For reference, 3D data are obtained from SU2. Two test cases are considered featuring inviscid flow around an unswept wing and viscous flow around a swept wing. An overall good agreement in terms of lift distribution and aerodynamic shape between the multi-fidelity solver and 3D CFD is obtained. The optimisation using the multi-fidelity aerodynamic approach is performed at a negligible computational cost compared to that using the full 3D CFD solver. The

optimal twist distribution shows sensitivity to the number of design variables, twist interpolation methods as well as the geometry control approach. As demonstrated in this study, the multi-fidelity solver finds easy application to other areas by a straightforward extension and coupling of the three constitutive blocks.

Guangda Yang, Daniel Kharlamov, Andrea Da Ronch, Jernej Drofelnik, and Zheng-Tong Xie, "An Efficient Multi-fidelity Aerodynamic Approach for Wing Twist Optimisation", *AIAA Journal* (under revision).

Guangda Yang, Andrea Da Ronch, Daniel Kharlamov, and Jernej Drofelnik, "Wing Twist Optimisation Using Aerodynamic Solvers of Different Fidelity", in *31st Congress of the International Council of the Aeronautical Sciences*, Belo Horizonte, Brazil, ICAS 2018-0440, 2018.

Data supporting this thesis are openly available from the University of Southampton repository at https://doi.org/10.5258/SOTON/D1113.

1.4 Thesis outline

The remainder of the thesis is organised in the following manner. Chapter 2 gives an extensive review of the previous work related to ASO. Chapter 3 details the computational methodology involved in the numerical optimisations. Chapter 4 and Chapter 5 present the optimisation results for NACA0012 aerofoil and RAE2822 aerofoil, respectively. Chapter 6 investigates wing twist optimisation using aerodynamic solvers of different fidelity. The conclusions and findings in this research are provided in Chapter 7 with recommendations for future work.

Chapter 2

Literature Review

2.1 Overview of design optimisation

2.1.1 Introduction

Many engineering design problems seek to maximise some measure of performance, such as minimising the aerodynamic drag or structural weight of an aircraft. With the advance of computer technology and optimisation techniques, it is of significant benefit to use numerical optimisation in design. Mathematically, numerical optimisation consists in the use of algorithms to minimise or maximise a given function by varying a number of variables. The problem may or may not be subject to constraints. Generally, an optimisation problem can be formulated as

$$\begin{aligned} \text{Minimise:} \quad & f(\boldsymbol{x}) \\ \text{w.r.t:} \quad & \boldsymbol{x} \\ \text{Subject to:} \quad & g(\boldsymbol{x}) \leq 0, \\ & h(\boldsymbol{x}) = 0 \end{aligned}$$

where $\boldsymbol{x}$ is the vector of design variables, $f(\boldsymbol{x})$ is the objective function to be minimised, $g(\boldsymbol{x})$ and $h(\boldsymbol{x})$ represent the inequality and equality constraint functions, respectively. In the case of any maximisation problem, a minus sign can be added to the objective function so as to transform the maximisation problem into a minimisation one.

A design optimisation process follows a similar iterative procedure to that of the conventional design process, with a few key differences. The comparison is illustrated in Figure 2.1. For both approaches, the baseline design could be the result of a previous design process or an initial idea derived from an engineer's experience and knowledge. In the conventional design process, the design may be analysed by numerical modelling or experimental tests, and then evaluated based on the results. The designer decides

whether or not the design performance is satisfactory. If not, likely in early iterations, the designer needs to change the design based on intuition and experience. The final design is reached when the design is satisfactory. For design optimisation, the analysis step is performed numerically without input from the designer. The evaluation of the design is strictly based on numerical values for the objective to be minimised and the constraints that need to be satisfied. An optimisation algorithm is used to automatically make changes to the design variables, avoiding the intervention of the designer. The decision to terminate the iterative process is made by the optimisation algorithm when the current design satisfies the necessary optimality conditions, ensuring the current design is superior to others in the vicinity of the design space.

In the design optimisation process, the designer needs to specify in advance the parameters that can be changed and their lower and upper bounds. The constraints and their limits also need to be defined prior to optimisation. It is crucial that the optimisation problem is well-formulated, otherwise the mathematical optimum obtained from numerical optimisation may not be the engineering optimum. This problem becomes more prominent for engineering systems with increasing complexity, where there are multiple levels and thus more cycles in the design process.

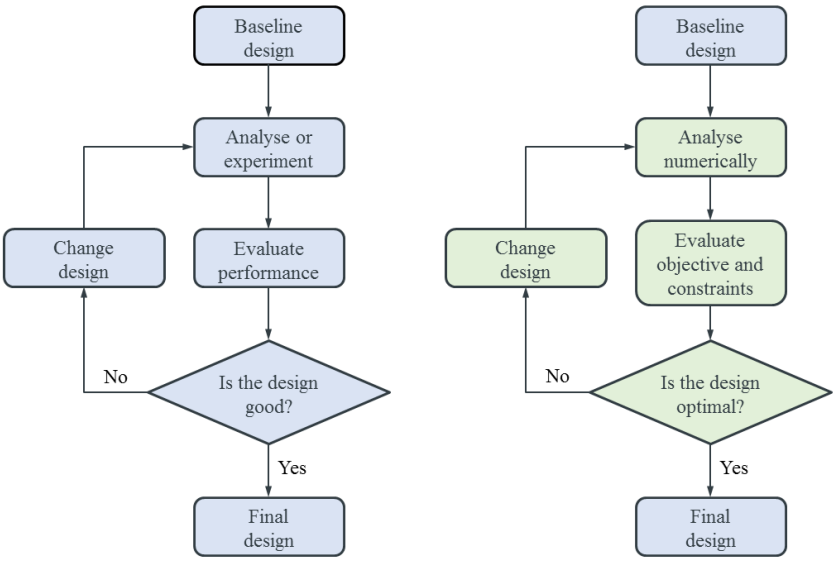


Figure 2.1: Conventional (left) versus optimal (right) design process [2].

Optimisation problems are classified based on various characteristics. A typical classification of optimisation problems is shown in Figure 2.2. As algorithms for solving optimisation problems are tailored to a particular type of problem, it is worth noting that restricting the optimisation to certain types and using appropriate optimisation algorithms can result in a dramatic improvement in the capacity to solve a specific problem.

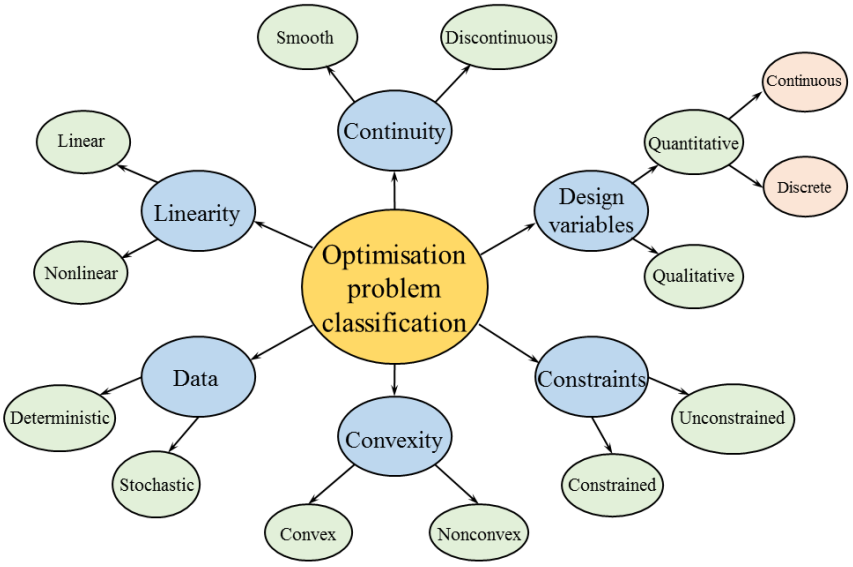


Figure 2.2: Classification of optimisation problems [2].

2.1.2 Multidisciplinary design optimisation

Multidisciplinary design optimisation (MDO) is the application of numerical optimisation techniques to the design of engineering systems that involve multiple disciplines. Aircraft design was one of the first applications of MDO because there is much to be gained by simultaneous consideration of the various disciplines (aerodynamics, structures, propulsion, stability and controls, etc.) that are involved and tightly coupled. For example, the reduction in the structure weight will eventually result in much lower overall weight of the aircraft due to the inner-dependencies among multiple disciplines, leading to the well-known "snowball effect" of weight reduction. On the contrary, design optimisation of the aircraft with one single discipline involved will likely deteriorate the performance in other disciplines. Figure 2.3 gives an illustration of how the aircraft looks like if each of these disciplines works strictly to attain their own requirements. Since none would fulfil all requirements at the same time, there is necessity to tune the needs of various disciplines together and make trade-off amongst them.

In 1977, Haftka [3] published one of the first MDO papers in aircraft wing design, where the disciplines of aerodynamics, structures and controls are coupled. Following the success of its initial application, MDO emerged in the 1980s within the aerospace community and has prospered since. Meanwhile, MDO has been extended to complete aircraft [4, 5, 6] and a wide range of other engineering systems, such as automobiles [7], rotorcraft [8], and wind turbines [9, 10, 11].

There have been a number of surveys of MDO over the past two decades. In 1997, Sobieski and Haftka [12] carried out a review of MDO developments up to that time. Two

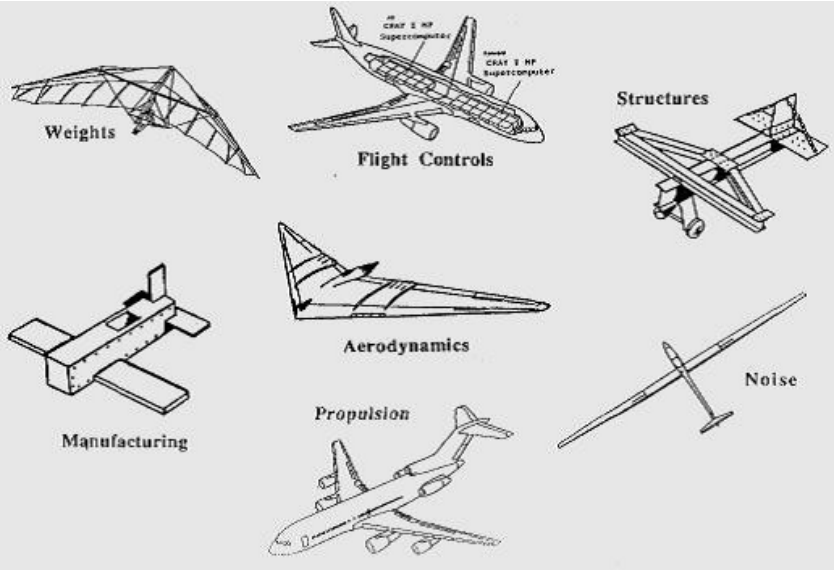


Figure 2.3: Ideal aircraft from the perspective of different disciplines [2].

main challenges addressed in that review were computational expense and organisational complexity. Previous efforts put emphasis primarily on approximation and decomposition strategies to deal with these challenges. Since this exhaustive review, MDO has continued to be an active area of research. More recently, the rapid development of computational capacity has eased the high computational demands to some extent, and some new MDO methodologies have been proposed and developed. In 2013, Martins and Lambe [13] published a comprehensive survey of existing MDO architectures in the literature. A summary and classification of MDO architectures were presented in a unified notation, facilitating the understanding of the various architectures and enabling the comparisons between them. This survey also pointed out the need to benchmark the existing MDO architectures and to develop new architectures.

Nowadays, MDO technologies have reached a maturity level that enables their widespread implementation and use in an industrial environment. Various successful applications of MDO have taken place in both research institutes and industry [14, 15, 16, 17, 18]. Brezillon *et al.* [17] described the ongoing development conducted at DLR, German Aerospace Centre, toward MDO capabilities based on high-fidelity methods. The MDO framework involved several disciplines, including aerodynamics, structures, aeroelasticity, and acoustics. Three representative optimisation scenarios were defined to tackle specific challenges: tightly or loosely coupled multidisciplinary problems, single or multi-objective problems, problems with reduced or large shape variations, etc. All three MDO problems were solved to demonstrate the benefits and limitations of the capabilities developed, providing indications for future work, such as robust and efficient grid deformation techniques in aerodynamics, a dynamic model in structure for fast identification of critical load cases, and suitable MDO architecture, etc. As an example from industry, Piperni *et al.* [18] presented the deployment of MDO technologies within the engineering

organisation at Bombardier Aerospace. As shown in Table 2.1, a multi-level framework in line with the aircraft design stages was developed. At every design stage, the appropriate problem formulation, level of fidelity, analysis tools, and optimisation strategy were implemented to meet the design objectives within the design-cycle time frame. The MDO technologies were deployed incrementally as an evolution of existing engineering methods for the three MDO levels, which correspond to conceptual, preliminary, and detail design stages.

Table 2.1: MDO levels and tool sets in Bombardier Aerospace (Source: [18]).

MDO Level	Fidelity	Aerodynamics	Structures	Propulsion
CMDO	L0	Knowledge-based aerodynamics	Knowledge-based weight prediction	Fixed architecture, scaled engine model
	L1	Quasi-3D methods (3D VLM / Panel method + 2D High-Fidelity CFD)	Beam or thin-shell models	Variable architecture, generic rubber engine
	L1.5	Disciplinary L2 Surrogate Models		Surrogate model(s) from Engine supplier(s)
PMDO	L2	Mid-to-High Fidelity CFD (3D TSD to RANS)	Global FEM	Real engine model (fixed)
	L2.5	Disciplinary L3 Surrogate Models		
DMDO	L3	RANS	Detail FEM	

Recently, an open-source MDO framework, OpenMDAO [19], was developed and released by the NASA Glenn Research Centre. OpenMDAO is a high-performance computing platform for system analysis and multidisciplinary optimisation. It is designed to take advantage of state-of-the-art algorithms to solve coupled models efficiently. OpenMDAO allows the users to combine the analysis tools from multiple disciplines, at multiple levels of fidelity, and to manage the interaction between them, which enables diverse applications in engineering analysis and design.

2.1.3 Aerostructural design optimisation

One of the most common applications of MDO techniques is aerostructural design optimisation, because of the strong interactions between aerodynamics and structures. Much effort has been put into aerostructural optimisation, aiming to optimise the aerodynamic shape and structural sizing simultaneously. Compared with single-disciplinary aerodynamic optimisation, aerostructural optimisation needs to consider the data transfer (i.e. aerodynamic loads and structural displacements) between the two disciplines. The treatment of geometry is also different: a rigid geometric model is considered in aerodynamic optimisation, whereas a flexible model is used for aerostructural optimisation.

One of the earliest studies of aerostructural optimisation was undertaken by Haftka [3], who combined a lifting-line aerodynamic model with structural finite-element analysis

to minimise the wing structural weight subject to drag and stress constraints. Grossman *et al.* then performed the aerostructural optimisation of a sailplane wing [20] and a transport wing [21]. They found that the designs generated from integrated optimisation were superior in terms of either aerodynamic performance or structural weight to those obtained from sequential optimisation (i.e. aerodynamic optimisation followed by structural minimisation). This fact was also confirmed and further explained by Chittick and Martins [22]. The sequential optimisation fails to achieve the true aerostructural optimum because of its inherent flaw that the sequential formulation is an incomplete consideration of the disciplinary interactions. The aerodynamic optimisation does not account for the structural benefit of shifting the lift distribution inboard, and the structural optimisation does not tailor the sizing to produce a deflected wing that is aerodynamically favourable. As a result, the sequential method fails to see the broader picture afforded by the fully-coupled design space.

With the advent of higher fidelity modelling in both structures and aerodynamics, numerical optimisation has been extensively applied to each of the disciplines separately. Since Schmit [23] pioneered structural optimisation in 1960, increasingly detailed finite-element models have been used in wing structural sizing optimisation [24, 25]. The development of CFD as well as adjoint method has made it possible to optimise an aerodynamic shape with respect to hundreds of design variables using both Euler [26, 27, 28] and Navier–Stokes models [29, 30, 31]. Aerodynamic and structural solvers with different levels of fidelity have been used for wing aerostructural optimisation. Elham and van Tooren [32] provided a review of the aerodynamic methods used for aerostructural optimisation. Given the importance of coupling the structures and the aerodynamics in wing design, various techniques have been proposed over the years for coupling CFD to computational structural mechanics solvers, with contributions in load and displacement transfer schemes [33, 34, 35, 36] and solution techniques for solving the coupled system of equations [37, 38].

Maute *et al.* [39] pioneered high-fidelity aerostructural optimisation by coupling an Euler flow solver with a linear finite-element structural model. However, this application was limited to only a few design variables because the cost of gradient computation using the direct analytical approach (as shown later in Section 3.2.5) is proportional to the number of design variables. To take advantage of high-fidelity models, it is desirable to optimise with respect to large numbers of design variables. In order to handle the large numbers of design variables, some researchers employed gradient-based optimisation algorithms together with adjoint methods to compute the required gradients efficiently. Martins *et al.* [40] proposed the use of a coupled adjoint method for aerostructural optimisation using Euler CFD and linear finite-element analysis. They then applied this method to the aerostructural design of a supersonic business jet [41]. To make this approach truly scalable and practical, Kenway *et al.* [42] made some developments and then demonstrated high-fidelity aerostructural optimisation of the NASA common research

model (CRM) using Euler [43] and RANS [44] aerodynamic analysis. Figure 2.4 shows the geometric and structural design variables used for the aerostructural optimisation of the NASA CRM. Some other researchers also implemented coupled adjoint method for the aerostructural equations. Zhang *et al.* [45] demonstrated high-fidelity aerostructural optimisation with an integrated geometry parameterisation and grid movement strategy. Ghazlane *et al.* [46] presented the latest developments at ONERA, French Aerospace Lab, on flexible wing optimisation via the aerostructural adjoint method. The aerodynamic shape and structural geometry were optimised simultaneously for the Airbus XRF1 wing-body configuration, where a weighted sum of aerodynamic drag and structural weight was used as the objective function in both single and multi-point optimisation processes.

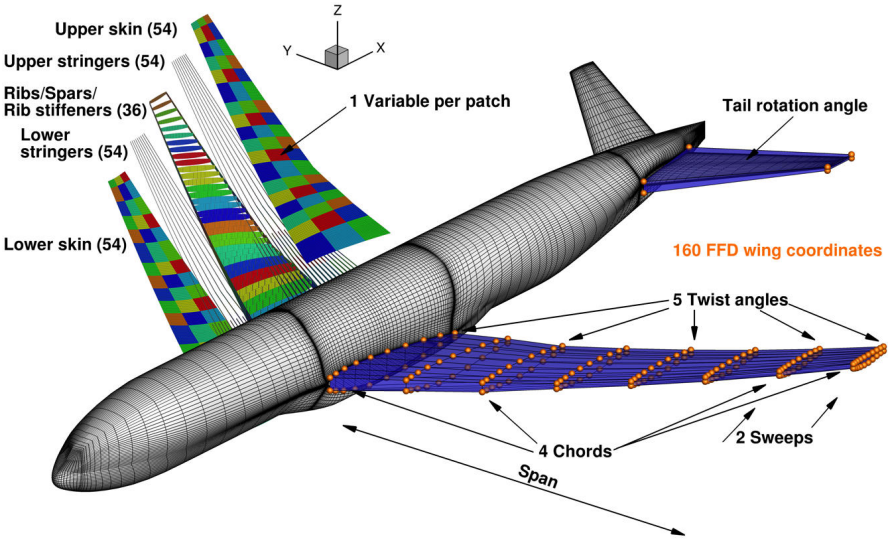


Figure 2.4: Aerostructural design variables of the NASA CRM. Structural design variables grouping (left) and geometric design variables (right). [43]

Aerostructural optimisation using higher-fidelity solvers, especially for aerodynamics, enables better representations of the wing model and thus provides more accurate results. However, high-fidelity wing aerostructural optimisation requires high-performance, parallel computing systems and leads to high computational cost. To address this challenge, some researchers employed a quasi-three-dimensional (Q3D) approach for wing aerodynamic analysis and optimisation. The Q3D approach, which combines 2D viscous aerofoil data with 3D inviscid wing data, computes the wing aerodynamic forces with sufficient level of accuracy and low computational cost. Examples of aerostructural optimisation using Q3D aerodynamic approach can be found in the works of Willcox and Wakayama [14], Jansen *et al.* [47], Mariens *et al.* [48], and Elham and van Tooren [32, 49]. In particular, Elham and van Tooren [32] performed gradient-based optimisation of a transport aircraft wing using the coupled adjoint method to compute the gradients. Compared with fully 3D CFD methods, Q3D approaches can rapidly compute the aerodynamic loads and are suitable for use in early phases of MDO. Nonetheless, Q3D approaches are limited to applications of wings in clean configuration and can not model

the effects of a nacelle or an engine pylon. For a complete aircraft, Q3D approaches are not able to model the effects of the fuselage and hence may have discrepancies (compared to 3D CFD data or experimental measurements) in the surface-pressure distributions near the fuselage [50].

In recent years, the trend of aircraft wing design has been towards enlarging the aspect ratio and using light-weight structures, which increases the wing flexibility. It is therefore essential to include aeroelastic considerations into the design and optimisation process. Kenway *et al.* [44] conducted gust simulations in the post-optimisation analyses, indicating further study to properly constrain the aerostructural design space. Elham and van Tooren [32] considered the index of aileron effectiveness as a constraint in order to satisfy the requirements of manoeuvrability. Variyar *et al.* [51] included the constraint of flutter speed into the early design loop of a strut-braced wing configuration, ensuring the aeroelastic stability. Kontogiannis *et al.* [52] used first bending mode frequency as an additional constraint to make the aerostructural optimisation of the NASA CRM wing more industrially relevant.

More recently, an open-source low-fidelity aerostructural analysis and optimisation tool, OpenAeroStruct [53], was developed within NASA's OpenMDAO framework. OpenAeroStruct combines VLM and finite-element analysis using a 3D beam model to simulate aerodynamic and structural analyses for lifting surfaces, and it uses the coupled adjoint method to compute the aerostructural derivatives. OpenAeroStruct serves as a hands-on learning tool for educators as well as a benchmarking platform for researchers.

2.1.4 Aerodynamic shape optimisation

Aerodynamic shape optimisation, or aerodynamic design optimisation, consists in maximising the performance of a given body (such as an aerofoil or a wing) by changing its shape. The aerodynamic performance is usually evaluated using CFD and the optimisation can be done using a number of algorithms. As described earlier in Section 2.1.1, the aerodynamic optimisation process is iterative: it starts with a given shape and then changes that shape to improve the performance while satisfying the specified constraints.

Over the past years, extensive research has been undertaken in the field of ASO, which has become an indispensable component and a powerful tool for effective and robust aerodynamic design in many applications. In this context, we focus on the application of ASO to aeronautical design. As various ASO frameworks have been developed in the research community, the AIAA ADODG was formed to address the need for ASO benchmarks that can be used to compare the methods between different researchers. The series of benchmark cases defined by ADODG are suitable for exercising ASO in a constrained design space and are diverse enough in searches for both global and local optima for both single- and multi-point optimisations.

Given that ASO is widely investigated across the community, it is impractical to thoroughly review the progress and current status. Therefore, some representative research groups are selected, and their activities and contributions toward ASO are briefly reviewed and introduced as follows.

1. **University of Toronto**
   A high-fidelity gradient-based ASO framework, Jetstream, was developed by Hicken and Zingg [54]. Jetstream features integrated geometry parameterisation and grid movement using the control points of B-spline surface. Gagnon and Zingg [55] addressed the shortcomings of direct surface-based geometry control, and they then implemented into Jetstream the new feature of volume-based geometry control using FFD technique. Lee *et al.* [56] applied both B-spline surface and FFD geometry control methods to several ASO problems and compared the optimal shape performance. Lee *et al.* [57] and Koo and Zingg [58] presented the results from the application of Jetstream to the ADODG benchmark cases. Chernukhin and Zingg [59] investigated the issue of multi-modality (i.e., multiple local optima) in high-fidelity ASO by applying both gradient-free and gradient-based optimisation algorithms to a number of 2D aerofoil and 3D wing problems. Koo and Zingg [60] performed ASO of both planar and nonplanar wings, demonstrating the robustness and ability of the methodology used by Jetstream to handle optimisation problems involving substantial geometric freedom.

2. **University of Michigan**
   The MDO lab focuses on high-fidelity gradient-based optimisation using the adjoint method for computing derivatives. Mader *et al.* [61] presented an approach to develop discrete adjoint solvers using automatic differentiation selectively. Lyu *et al.* [62] developed a discrete adjoint solver for the RANS equations and SA turbulence model. They then solved a series of ASO problems [30] based on the CRM wing benchmark case defined by ADODG. Figure 2.5 shows a detailed comparison of the baseline wing and the optimised wing. Kenway and Martins [63] created multi-point optimisation cases of the CRM wing for ADODG. Chen *et al.* [64] investigated the full aircraft (i.e. wing–body–tail) configuration case of the CRM with the consideration of trim effect on the wing design. In addition, the approach to ASO has also been employed to study other configurations, such as a BWB aircraft [65] and a morphing wing [66]. More recently, Kenway and Martins [67] enforced buffet margin constraints into ASO at transonic conditions and investigated its impact on the optimal designs.

3. **University of Bristol**
   Efforts have been made toward developing new geometry parameterisation approaches and grid deformation methods. Morris *et al.* [68] developed a novel domain element shape parameterisation method using radial basis functions (RBFs) and applied this approach to 2D aerofoil [68] and 3D wing [69] optimisation problems. Poole *et al.* [70]

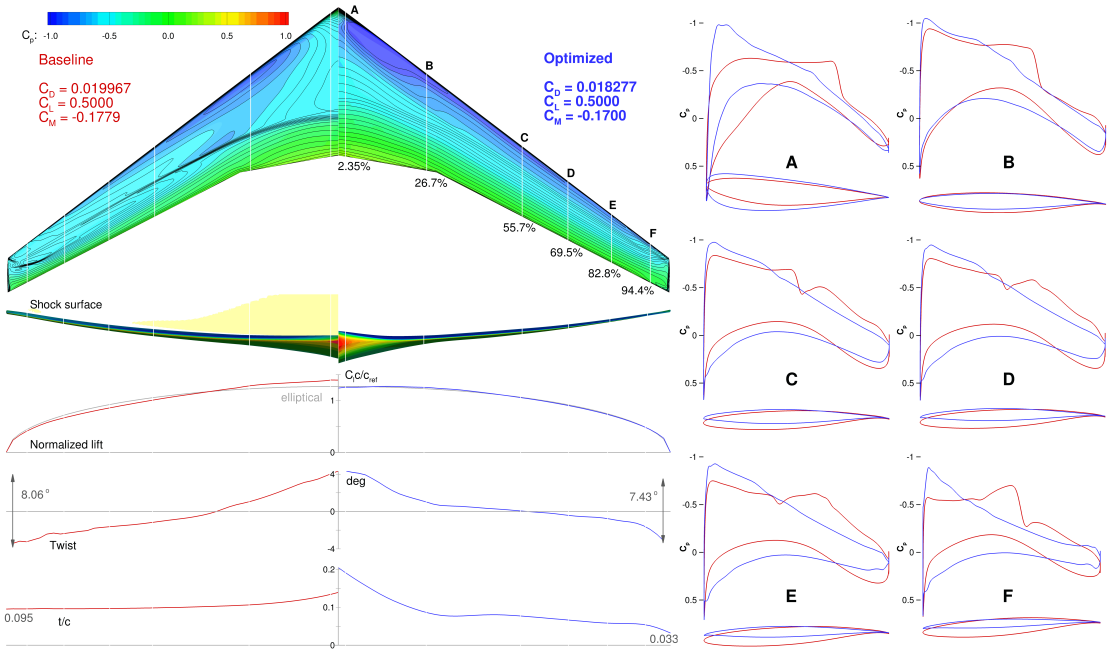


Figure 2.5: Comparison of the baseline wing and optimised wing of the NASA CRM. The optimised wing is shock-free and has 8.5% lower drag. [30]

presented a novel proper orthogonal decomposition (POD) approach using singular value decomposition (SVD) to derive orthogonal aerofoil design variables from a library of training data. The effectiveness of the developed method was then demonstrated through aerodynamic optimisation of 2D aerofoils [71] and 3D wing [72] in transonic flow. Masters *et al.* [73] proposed a multi-level aerofoil parameterisation scheme that uses subdivision curves. More recently, Payot *et al.* [74] presented a novel parameterisation approach, named the restricted snakes volume of solid, for topology optimisation. Kedward *et al.* [75] presented an efficient grid deformation method using multi-scale RBF interpolation. In addition, Poole *et al.* [76] investigated the issue of multi-modality existing in Case 6 of the ADODG benchmarks.

4. **University of Sheffield**

   The research activities cover several aspects of ASO. Le Moigne and Qin [77] presented a variable-fidelity ASO methodology based on a discrete adjoint solver for turbulent flows. The optimisation method was then employed for a systematic aerodynamic study of a BWB aircraft [78], including inverse design of the spanwise lift distribution [79], aerofoil and sweep optimisation [80], and deployment of shock control bumps [81]. Liu *et al.* [82] proposed an efficient dynamic grid deformation technique based on Delaunay graph mapping (DGM) method. This approach was further improved by Wang *et al.* [83, 84], who employed either RBF method [83] or inverse distance weighting method [84] to provide better control of the near-surface grid quality while maintaining the advantage of the efficiency of DGM grid deformation. Vavalle and Qin [85] presented a response surface based optimisation scheme for aerofoil design at transonic speed. Zhu and Qin [86] proposed an intuitive class shape

transformation (CST) method for aerofoil parameterisation. More recently, some work [87, 88, 89] was undertaken to study the grid sensitivity in a discrete adjoint optimisation framework.

5. **Queen Mary University of London**
   The research work primarily focuses on the development and application of adjoint CFD solvers and shape parameterisation methods, particularly computer aided design (CAD)-based parameterisation methods, into engineering design problems. Two flow solvers, namely incompressible GPDE [90] and compressible STAMPS [91], were developed to perform flow analysis and provide sensitivities using the algorithmic differentiation (AD) tool Tapenade. An in-house CAD kernel, termed non-uniform rational B-spline (NURBS)-based parameterisation with continuity constraint (NSPCC) [92, 93], was developed to parameterise the geometry. Yu *et al.* [94] demonstrated the developed method for a 2D aerofoil optimisation. Xu *et al.* [95] applied the CAD-based parameterisation approach to ASO of a wing–body configuration of the DLR-F6 model. Zhang *et al.* [96] performed adjoint-based ASO for the ONERA M6 wing using both B-spline and NURBS parameterisations.

6. **University of Southampton**
   Considerable research has been carried out in the field of surrogate modelling [97] and other model order reduction techniques in the design and optimisation of engineering systems. With the aim of reducing the optimisation cost, Toal *et al.* [98, 99, 100] conducted extensive investigation into kriging, including investigating hyperparameter tuning strategies [98], applying an adjoint of the likelihood function [99], and presenting a variable reduction technique using POD [100]. Forrester *et al.* [101] presented a multi-fidelity optimisation methodology using co-kriging. Toal and Keane [102] then applied a multilevel co-kriging model for multi-point design optimisation. Toal [103] also defined a set of guidelines regarding the use of multi-fidelity kriging. Viswanath *et al.* [104, 105] proposed a dimension reduction method called generative topographic mapping to facilitate the optimisation process. Parr *et al.* [106, 107] presented a new infill sampling criterion for surrogate-based constrained optimisation. Sóbester [108, 109] investigated concise parametric aerofoil formulations using NURBS and Kulfan's CST parameterisation respectively. Sóbester *et al.* [110] also demonstrated the surrogate-based optimisation techniques to engineering design applications, such as a regional airliner wing. More recently, Kontogiannis *et al.* [111] investigated the efficient handling and parameterisation of waverider geometries.

While some literature [112, 113, 114] summarised the state-of-the-art in ASO, the majority of review articles target at one specific area within ASO. Forrester and Keane [115] reviewed the advancement of constructing surrogate models and their use in optimisation strategies, and they also provided guidance regarding the strengths and weaknesses of each method. Lian *et al.* [116] reviewed the progress in aerodynamic design optimisation

using evolutionary algorithms. Han [117] conducted a review of the theory, algorithms and recent progress with respect to the kriging surrogate model and its application to ASO and MDO. Masters *et al.* [118] presented a comprehensive review of aerofoil shape parameterisation methods used for ASO, and also compared their efficiency with respect to the coverage of aerofoil design space. Zhao *et al.* [119] provided a review of robust ASO methodologies, highlighting the key techniques and primary challenges.

Recently, the open-source SU2 code [120, 121] has gained popularity in the aerospace community due to its capability of solving engineering problems through adjoint-based optimisation. The SU2 suite employs the state-of-the-art numerical methods to perform CFD analysis for a range of flow problems. With the aid of an adjoint method, the gradient information can be computed efficiently, enabling powerful analysis and design optimisation for complex engineering systems. In this thesis, SU2 is used to perform adjoint-based ASO of both 2D aerofoil and 3D wing geometries [122, 123, 124].

2.1.5 Section summary

In this section, an overview is provided regarding the recent development, current status, and future trends of design optimisation within the aeronautical community. The mathematical formulation, design process and taxonomy of numerical optimisation was firstly introduced. The literature review was then carried out, starting from MDO, moving on to aerostructural optimisation, and finally narrowing down to ASO. In particular, instead of conducting an exhaustive review of ASO, some representative research groups were chosen and their research activities were reviewed, highlighting their contributions to the research community. A brief summary is given in Table 2.2. In addition, some open-source optimisation frameworks were introduced, which are of interests and use for many researchers.

Table 2.2: Summary of ASO studies from a selection of research groups.

Institution	Principal Investigator	Research Areas
University of Toronto	Prof. David Zingg	High-fidelity gradient-based ASO [54]; Integrated geometry parameterisation and grid movement [55]; B-spline and FFD [56]; ADODG benchmarks [57, 58]; Multi-modality [59]
University of Michigan	Prof. Joaquim Martins	High-fidelity gradient-based ASO [62]; Discrete adjoint [61, 62]; ADODG CRM benchmark [30, 64]; Multi-point CRM ASO [63]; Novel configurations [65, 66]
University of Bristol	Prof. Christian Allen	Novel geometry parameterisation methods [68, 70, 72, 74]; Aerofoil parameterisation [73, 118]; Grid deformation [75]; ADODG benchmarks [76, 125]; Multi-modality [76]
University of Sheffield	Prof. Ning Qin	Variable-fidelity ASO [77]; BWB studies [78, 79, 80, 81]; Grid deformation [82, 83, 84]; Intuitive CST parameterisation; Discrete adjoint [86]; Grid sensitivity [87, 88, 89]
Queen Mary University of London	Dr. Jens Müller	CAD-based parameterisation using NURBS [92, 93]; Discrete adjoint solvers [90, 91]; Automatic differentiation [92, 93]; Aeronautical applications [94, 95, 96]
University of Southampton	Prof. Andy Keane	Surrogate modelling [97]; Kriging [98, 99, 100]; Infill sampling criteria [106, 107]; Multi-fidelity ASO [101, 102, 103]; Dimension reduction [104, 105]; Geometry parameterisation [108, 109]

2.2 Computational techniques in ASO

ASO can be dated back to the 16th century, when Newton used calculus of variations to minimise the fluid drag of a body of revolution with respect to the body's shape. Although there were many significant developments in optimisation theory after that, it was only in the 1960s that numerical optimisation emerged to be a feasible tool for routine applications due to both the theoretical advancements and exponential developments in computer hardware. In this section, the state-of-the-art and commonly-used computational techniques are summarised and reviewed for each constitutive part of ASO framework.

2.2.1 Aerodynamic solver

In an ASO framework, the prediction of aerodynamic properties of a given body requires the numerical analysis of an aerodynamic solver. Over the years, various aerodynamic methods of different levels of fidelity have been developed, suitable to different flow regimes and different phases of the aircraft design process [18].

Early studies rely on empirical methods, such as Engineering Science Data Unit (ESDU) and Data Compendium (DATCOM), but the quest for more efficient aircraft make these methods too inaccurate. Panel methods are slightly more costly approaches, which solve incompressible potential flow over 2D and 3D geometries. XFOIL [1] is an open-source software developed by Drela at MIT. It can model the flow around any 2D aerofoil using panel methods, and it can also perform corrections for viscosity and compressibility. A solution for 3D wings of any general form can be obtained by using a vortex lattice method (VLM) [126]. The wing is modelled as a set of lifting panels, and each panel contains a single horse-shoe vortex. Extensions have been made for unsteady flow. For the case of harmonically oscillating surfaces, Albano and Rodden [127] developed the doublet lattice method (DLM) by augmenting the vortex with an oscillating doublet. DLM is formulated in the frequency domain and it reduces to VLM for steady cases. Both VLM and DLM are widely used in the early stages of aircraft design, however, they are limited to the linear flow regime. Kennedy and Martins [128] combined the use of panel method for induced drag estimation and semi-empirical methods for viscous and wave drag estimation. Piperni *et al.* [15] used a 3D transonic small disturbance code coupled to boundary layer calculations for aerodynamic analysis in the transonic regime. However, transonic small disturbance codes are suitable for drag estimation at transonic conditions with relatively weak shock and attached flow. Drela [129] developed a 3D integral boundary layer formulation to enable rapid viscous analysis of 3D aerodynamic flows via strongly-coupled viscous/inviscid solution methods. The correctness

[1] http://web.mit.edu/drela/Public/web/xfoil/ [retrieved 2019]

and accuracy of the approach were demonstrated on a number of test cases of general configurations.

In later stages of aircraft design, the importance of the accuracy of drag prediction becomes prominent. According to industry criteria [130], the accuracy of drag prediction by numerical methods should be within one drag count (i.e. one ten thousandth of the drag coefficient). Therefore, high-fidelity solvers are required for aerodynamic analysis and optimisation in order to correctly predict the drag, particularly for the design of transonic aerofoils and wings where strong nonlinearity exists due to viscous and compressibility effects. A number of high-fidelity aerodynamic codes that solve Euler and RANS equations have been developed and used for ASO, such as Jetstream [57], SUmb [30], STAMPS [91], etc. Some software packages were also developed within the research institutes, such as ONERA *elsA* flow solver [131] and DLR-TAU code [17]. In addition, the open-source CFD software, such as SU2 [120] and OpenFOAM [2], are also extensively used to perform high-fidelity analysis for complex aeronautical configurations.

Although it produces numerical results with sufficient accuracy, the use of a high-fidelity aerodynamic model in aircraft design also poses a challenge of computational cost even with high-performance computing facilities. Various approaches have been employed to address this issue. For example, Kenway and Martins [43] used the Euler equations to model the flow over the aircraft, and the viscous drag was estimated by empirical equations. However, the uncertainty of total drag calculation due to the skin friction estimation outweighs the cost saving brought by the Euler solver instead of a Navier–Stokes solver. Recently, the development of reduced-order modelling techniques, such as Q3D aerodynamic solvers, has eased this problem to some extent. More details about Q3D solvers and their application in ASO are discussed in Section 2.4.

2.2.2 Geometry parameterisation

Geometry parameterisation, also known as shape or surface parameterisation, concerns how the geometry is represented and perturbed during optimisation process, and it defines a design space by a number of design variables. For any ASO framework, geometry parameterisation is a critical aspect as the ability for the optimiser to fully explore the design space is driven by the ability for the degrees of freedom adopted to represent any shape within the design space. Therefore, the choice of parameterisation scheme has significant impact on the design space and thus the final optimisation result. Typically, an effective parameterisation method is characterised as being: 1) flexible enough to allow sufficient design space investigation; 2) robust enough to be applicable to any geometry or design surface; and 3) efficient enough to cover the design space with a small number of design parameters [70].

[2] `www.openfoam.org` [retrieved 2019]

Over the past several decades, numerous geometry parameterisation and surface control techniques have been developed and implemented for shape optimisation of aerospace applications. Based on whether the CAD model is involved or not, shape parameterisation methods can be divided into two kinds, namely CAD-based and CAD-free methods. Similarly, the parameterisation methods can also be categorised as either constructive or deformative depending on whether the geometry creation and perturbation are separated or not. Following the latter taxonomy approach, examples of constructive methods are CST [132] and parametric section (PARSEC) [133]; and deformative methods include discrete [26], analytic [134] and FFD [135], etc. In terms of 2D aerofoil parameterisation, constructive methods represent an aerofoil shape purely from a series of parameters specified, whereas deformative methods take an existing aerofoil and then deform it to create a new shape.

With the continuous development and application of geometry parameterisation and shape control methods, some studies have also been undertaken to provide a survey of available techniques and to compare their effectiveness in ASO. Reviews of a range of parameterisation methods were presented in the work of Samareh [136], Castonguay and Nadarajah [137], Mousavi *et al.* [138] and Masters *et al.* [118]. In particular, Samareh [136] and Poole *et al.* [70] provided a table summary, detailing the relative merits and features of each approach. In this context, a selection of parameterisation methods are introduced and discussed as follows.

Parametric section

The PARSEC method, originally developed by Sobieczky [133], is a specific 2D aerofoil parameterisation method. It targets at representing subsonic and transonic aerofoils based on meaningful properties, such as leading edge radius, maximum thickness, and trailing edge angles, etc. The upper and lower surfaces are firstly described by six-order polynomials; and the resulting 12 coefficients are then obtained by solving a system of linear equations, subject to 12 intuitive parameters (as shown in Figure 2.6) that define the geometric characteristics of the aerofoil. An advantage of the PARSEC method is that it uses geometric parameters as design variables, allowing direct and intuitive control over the shape of the aerofoil. However, as this method is limited to only 12 design variables, it does not provide the range of fidelity made available by many of the alternative methods. Moreover, the PARSEC method can allow large-scale surface changes, and therefore alter the global family of the aerofoil; however, it is unable to provide high flexibility, especially in inverse design problems, as it is difficult to make detailed minor local surface changes.

Hicks–Henne bump function

The HHBF approach was initially used by Hicks and Henne [134] to represent the aerofoil shape at different spanwise stations of a wing. The HHBFs were applied to modify

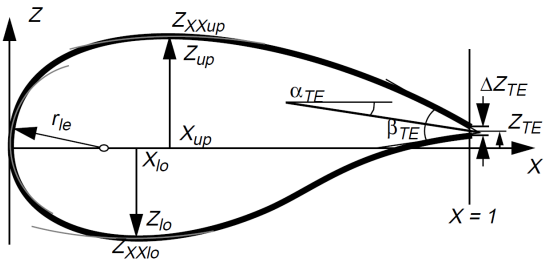


Figure 2.6: A geometric representation of the parameters used for PARSEC method [133].

the baseline aerofoil by adding linearly a series of shape functions. HHBFs recently became popular for modelling small or moderate perturbations of baseline aerofoil shapes for solving various optimisation problems. For instance, Toal *et al.* [98] investigated an inverse design problem (i.e. recreating the surface pressure distribution of the RAE2822 aerofoil via the modification of the baseline NACA0012 aerofoil) by employing multiple HHBFs to produce geometry parameterisations of varying complexity while maintaining a level of continuity. The HHBF approach is employed as one of the two shape parameterisation methods in this study. More technical details are provided in Section 3.2.6.1.

Class shape transformation

The CST method was first proposed by Kulfan [139] to express a wide range of aerofoils with relatively few numbers of variables, and it was then extended to more general 3D applications such as aircraft wing, body, and nacelles [132]. Mathematically, the CST method describes geometries as a product of a class function defining the fundamental baseline geometry profile and a shape function defining the variation or deviation from the baseline geometry. The shape function is formulated with a linear combination of Bernstein polynomials, of which the weighting coefficients are employed as the design variables.

In recent years, the CST method has been extensively used as shape parameterisation scheme in ASO owing to its capability to meet the desirable criteria such as smoothness, mathematical efficiency, size of design space, and ability to handle constraints. Some researchers have made further improvement and enhancement to the CST method. Straathof and van Tooren [140] proposed the class shape refinement transformation method by using a combination of Bernstein polynomials and B-splines to allow for both global and local control of a shape. In addition, the use of B-splines enables the capability to handle volume constraints efficiently. However, by adding the refinement function, the class shape refinement transformation method requires a higher number of design variables compared to the original CST method, which contributes to the complexity of the design problem. Zhu [141] presented a rational CST method by employing rational Bernstein polynomials instead of the standard ones. The rational CST method was shown to increase the fitting accuracy of the original CST method with fewer design

variables. Moreover, Zhu and Qin [86] proposed the intuitive CST method for aerofoil parameterisation, which combines the flexibility and accuracy of Kulfan's CST method and the intuitiveness of Sobieczky's PARSEC method.

Polynomial and spline

The polynomial and spline representations, such as Bézier, B-spline, and NURBS, are powerful tools for geometry parameterisation and are well-suited for shape optimisation. The Bézier form, which is based on Bernstein polynomials, is an effective and accurate representation for simple curves and surfaces. However, it does not have the property of local control, i.e. the manipulation of one control point will influence the entire geometry. This flaw can be overcome by the B-spline form, which relies on a set of basis functions and can be viewed as a composite of low-degree Bézier segments. The B-spline form can represent complex geometries efficiently and accurately. However, it is unable to accurately represent implicit conic sections. The more general NURBS form can represent most parametric and implicit curves and surfaces without loss of accuracy. Generally, the polynomial and spline representations can describe a geometry in a very compact form with a small set of design variables, showing a significant advantage over the discrete approach, in which the grid nodes directly define the geometry and are the design variables. In addition, the polynomial and spline methods have the flexibility to cover the potential optimal solution in the design space and can guarantee the smoothness of a shape.

The polynomial and spline techniques have been widely used within the research community. Nemec and Zingg [142] and Hicken and Zingg [54] applied B-spline surface control in conjunction with an efficient grid movement technique to 2D and 3D aerodynamic optimisation problems, respectively. Lee *et al.* [57] and Bisson *et al.* [143] employed B-spline parameterisation for the optimisation of the ADODG benchmark cases.

CAD-based approach

The CAD-based parameterisation method allows the CAD description of the shape to be maintained inside the automated design loop. This approach avoids the need to develop a separate parameterisation method to describe the degrees of freedom of the design. This approach can import the boundary representation from CAD and produce the optimal shape in the CAD format. Two European Commission projects, About Flow[3] and IODA[4], aim to advance with the systematic integration of adjoint-based design optimisation with CFD into the regular design chains. One key aspect of the projects focuses on the development of CAD-based parameterisation to link the optimisation directly with CAD geometries. Robinson *et al.* [144] and Agarwal *et al.* [145] presented an efficient optimisation process, where the parameters defining the features in a CAD model, such as sketch-based and dress-up features in CATIA V5, are used as design

[3] http://aboutflow.sems.qmul.ac.uk/ [retrieved 2019]
[4] https://ioda.sems.qmul.ac.uk/ [retrieved 2019]

variables. In the research group led by Dr. Jens Müller, Xu *et al.* [95] used B-spline surfaces to model the wing–fuselage configuration of the DLR-F6 aircraft, and Zhang *et al.* [96] performed aerodynamic optimisation of the ONERA M6 wing using both B-spline and NURBS parameterisation approaches.

Free-form deformation

The FFD algorithm, first formally described by Sederberg and Parry [146], is a subset of the soft object animation algorithms used in computer graphics for morphing images and deforming models. The FFD technique is based on the idea of enclosing an object of interest within a flexible volume, and transforming the object by deforming the lattice of the volume. It enables a smooth and continuous deformation through perturbation of the control points of the control volume.

The FFD approach has been extensively employed as geometry parameterisation method to a range of shape optimisation problems [30, 55, 56, 65, 147], including morphing aircraft, NASA CRM wing, BWB, and winglets, etc. The FFD approach parameterises the geometry changes rather than the geometry itself, resulting in a more efficient and compact set of design variables, and thus making it easier to handle complex geometries. The FFD algorithm can also relate the deformation to aerodynamic design variables, such as thickness, camber, twist, and planform, by manipulating groups of control points. Moreover, since the FFD volumes are trivariate Bézier, B-spline or NURBS volumes, the sensitivity derivatives of any point inside the volume can be easily computed. The FFD technique is employed as one of the two shape parameterisation approaches in this study. More technical details are provided in Section 3.2.6.2.

Proper orthogonal decomposition

POD is a method to obtain a low-dimensional approximation to a high-dimensional space by the derivation of dominant components, or modes, and can be done by principal component analysis, Karhunen–Loeve decomposition, or SVD [148]. This mathematical decomposition approach can be used to derive design variables in aerodynamic optimisation, particularly for the case of aerofoils. The POD method takes a large library of aerofoils, termed the training data, and extracts a set of orthogonal shape modes. New aerofoil shapes can then be constructed as a linear combination of these modes. The fidelity of the construction is determined by the number of modes used, though a small set of the most dominant modes are sufficient to have a good design space coverage.

With respect to ASO, the POD technique was initially employed by Toal *et al.* [100], who showed that the design space of the 2D aerofoil could be reduced to a few principal modes. Ghoman *et al.* [149] then showed that POD could be used to derive aerofoil design variables that represent a specific aerofoil family. Poole *et al.* [70] further showed that the extracted geometric modes from POD could represent a wide variety of aerofoil shapes given a broad range of training data. Poole *et al.* [71] and Allen *et al.* [72]

applied this mathematical approach based on SVD to 2D aerofoil and 3D wing optimisation problems, respectively. Furthermore, Masters *et al.* [118] showed that the SVD technique outperforms other commonly-used parameterisation schemes when considering an inverse design problem of a 2D aerofoil.

2.2.3 Grid deformation

In the process of ASO, the boundary surface is perturbed to reflect the changes in the shape design variables. As a result, a new grid has to be generated at each design iteration to fit the deformed surface, or the existing grid has to be allowed to move with the computational domain. Allowing the existing grid to evolve with the computational domain is generally more efficient than generating a new grid. In shape optimisation, the boundary surface undergoes many small changes; it would be too time-consuming to regenerate the grid in response to these deformations. Regenerating grids usually requires manual adjustments for complex geometries and the projection of the solution from the old grid to the new one, because the new grid may not have the same number of nodes and connectivity. Grid deformation, on the other hand, will inherently preserve the original grid connectivity, hence ensuring the consistency of any grid-induced errors in the flow solution (i.e., due to discretisation error) between the initial and deformed grid, provided that a consistent grid quality is maintained. It also ensures continuity in the sensitivity derivatives. The robustness and efficiency of the grid deformation tool is particularly important in gradient-based optimisation because any changes in the grid quality can have significant effects on these derivatives. Most importantly, grid movement algorithms have the potential to significantly reduce engineering cost by allowing the design process to be automated.

Grid deformation techniques can be generally classified into two main categories based on either physical analogy or interpolation. A physical analogy approach, normally requiring connectivity information of the grid, uses certain physics processes to propagate the grid perturbation from the boundary to the entire computational domain, whereas an interpolation approach directly obtains the displacement or the new coordinates of each grid node by applying some interpolation schemes. Some representative methods of both types are introduced and discussed herein.

Transfinite interpolation

Transfinite interpolation (TFI) was first described by Gordon and Hall [150] in 1973, and it has the advantage of providing complete conformity to boundaries in the physical domain. In the early 1980s, Eriksson [151] applied TFI to grid generation for CFD. TFI is essentially a multivariate interpolation procedure, and it can efficiently generate the grid from the boundaries to the interior region of the domain. The choice of blending functions has a considerable influence on the quality and robustness of the grid.

Soni [152] has proposed a set of blending functions based on arc length that is very effective and robust. By the mid-1990s, the algebraic grid generation technique based on TFI had been widely used for grid regeneration and deformation in ASO [153] and MDO [154]. A number of variations of TFI have also been proposed to deal with complicated geometric perturbations. However, the efficiency and robustness of this method are limited to applications with structured grids.

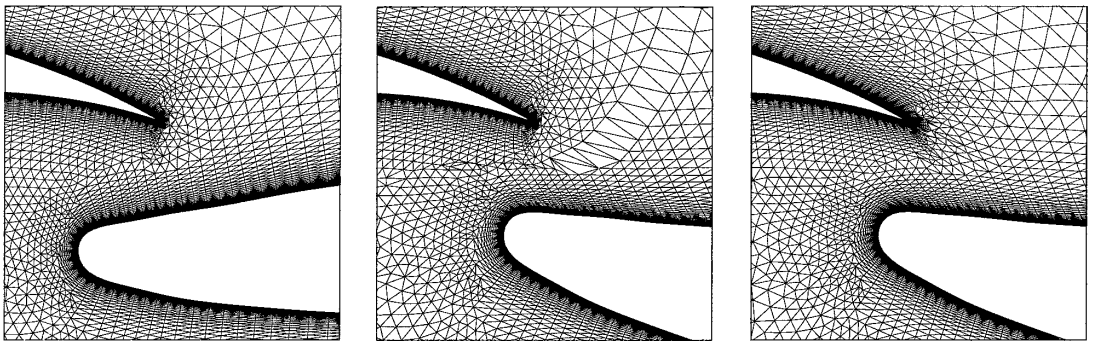
Spring analogy method

The spring analogy method, firstly presented by Batina [155], models the grid as a network of linear springs and solves the system of static equilibrium equations to determine the nodal displacements due to a given boundary movement. The spring stiffness is assumed to be inversely proportional to the edge length. This approach has been employed for aerodynamic optimisation [156]. Although it is found to be fairly efficient and applicable to unstructured or structured grids, the spring analogy method can produce negative volumes for grids with large deformations [157]. Farhat *et al.* [158] improved the robustness of the method by incorporating nonlinear torsional springs. This was later extended to 3D applications [159].

Another approach is to model the grid as a continuum of elastic solid whose properties are defined by the modulus of elasticity and the Poisson's ratio. Nodal movements are governed by the equations of linear elasticity, and grid distortion can be controlled through the elements in the elastic matrix. Johnson and Tezduyar [160] demonstrated that this approach can be used to achieve robust grid movement, even for large shape changes. Nielsen and Anderson [157] extended the linear elasticity theory by assuming an isotropic material and using a spatially varying value of Poisson's ratio. The approach was also found to be more robust than the spring analogy method, although relatively inefficient due to that the equations of linear elasticity are typically less diagonally dominant than the spring analogy equations. Figure 2.7 shows the grids resulting from the two deformation methods, and illustrates the advantage of elasticity method over the spring analogy method. The linear elasticity approach has been used successfully for aerodynamic optimisation [157, 161]. Dwight [162] further augmented the robustness of this method by modifying the equations of linear elasticity. This approach was implemented in the SU2 code and was thus employed in this work. The governing equations and more details of this approach can be referred to Section 3.2.7.

It should be noted that all the physical analogy based methods for grid deformation require iterative solution of the governing equations to determine the grid displacements for every design step. Hence, these methods tend to be more computationally expensive than the interpolation methods.

Delaunay graph mapping



(a) (b) (c)

Figure 2.7: Near-field view of grids with flap deflection on a multi-element aerofoil [157]. (a) Baseline grid; (b) Grid after applying distance function/tension-spring analogy method; (c) Grid after applying modified linear elasticity method.

In 2006, Liu *et al.* [82] introduced an algebraic grid deformation method based on DGM. The proposed DGM method can be viewed as a fast interpolation scheme, and it maps the grid of any topology to a Delaunay graph, which can be moved easily with the geometric shape perturbation. The DGM method is comparable to the spring analogy method in robustness but surpasses it in efficiency, as it is non-iterative and uses significantly less memory. On the other hand, compared to the TFI technique that is limited to structured grids, the DGM method is independent of the grid topology and can be applied to arbitrary types of grids. Despite the efficiency and flexibility of the DGM approach, the grid quality near the boundaries may deteriorate after large deformation. In particular, if the deformation involves large rotation, the Delaunay graph may lose its topology and thus become invalid. To address this problem, Wang *et al.* [83] developed a novel technique based on a combination of the DGM method with the RBF method. The new algorithm maintains the high efficiency of the original DGM method and also improves the grid quality near the boundaries, which is crucial for applications with high Reynolds number. By separating interpolations for translation and rotation motion, the proposed method significantly improves the robustness of the original DGM method, especially for cases with large rotational deformation. More recently, Wang *et al.* [84] presented a new grid deformation method based on the combination of the DGM method and the inverse distance weighting interpolation. Similar as the approach in Reference [83], this approach can also preserve the near-wall grid quality after large rotation deformation while maintaining the efficiency of the DGM method.

Radial basis function

RBF interpolation is a popular tool for general multivariate interpolation and is able to operate on scattered data sets in any multi-dimensional space. RBF methods have been used across a wide variety of disciplines because of its flexibility and generality. Recently there has been interest in using RBF interpolation for grid deformation [163, 164] owing

to its robustness and ability to preserve grid quality. Moreover, RBF methods do not require grid connectivity information and hence are independent of grid types. However, the RBF method in its full form is prohibitively expensive for grids with a large size, as the computational cost scales with the number of control points (i.e., the size of the surface grid). Thus, the full RBF method is impractical to be implemented and used, particularly for 3D applications such as wing.

To address this problem, Wang *et al.* [83] developed a hybrid method combining the advantages of the efficiency of DGM scheme and the grid quality control from RBF interpolation. Alternatively, the efficiency of RBF interpolation can be improved by using approximation techniques with data reduction algorithms. Jakobsson and Amoignon [165] tackled the size of the problem by applying a simple grid coarsening method, in which the control points were reduced to a small, evenly distributed subset of the surface points. Rendall and Allen [166] proposed an approximate RBF method in which only a subset of the surface grid points were used as control points and greedy algorithms were employed to minimise the interpolation error for non-control points. This method was extended [167] by adding a correction step to recover exact surface displacements for all other points once the selected surface points have been moved. Moreover, a number of further improvements were proposed and a series of different data reduction schemes were developed [168, 169, 170, 171], primarily focusing on efficient point selection algorithms and alternative error correction methods. More recently, Kedward *et al.* [75] presented a multi-scale RBF interpolation method, in which all surface points are used to ensure an exact surface representation, but only a single interpolation is required to be built. The multi-scale method outperforms the previous RBF method in Reference [166] in terms of both grid quality and efficiency; and the multi-scale point selection is based purely on geometry, and hence is decoupled from deformation.

The RBF interpolation method for grid deformation has been used in ASO applications. Jakobsson and Amoignon [165] performed gradient-based ASO of the ONERA M6 wing and analysed the grid quality and interpolation error with respect to the parameters of the RBF method. Zhu [141] studied a series of 2D and 3D optimisation problems using RBF grid deformation method that is embedded in DLR-TAU solver. Poirier and Nadarajah [171] demonstrated the proposed RBF-based grid deformation scheme through several 3D cases within an adjoint-based ASO framework, in which the grid sensitivity is based on a combination of both the primary and secondary grid movement algorithms. In addition, Morris *et al.* [68, 69] and Allen and Rendall [172] presented an integrated approach for geometry parameterisation, surface control, and grid deformation; and they also applied this approach to shape optimisation of aerofoils [68], aircraft wings [69], and rotor blades [172]. Poole *et al.* [125] further demonstrated this unified approach through investigating ADODG benchmark problems.

2.2.4 Optimisation algorithm

The optimisation of any engineering design problem usually begins with an initial design point and is then driven by an optimisation algorithm to find the optimum point. Generally, the numerical optimisation algorithms can be classified into two categories: gradient-based and gradient-free, alternatively deterministic and stochastic. Both kinds of optimisation algorithms have been extensively used within the field of ASO; however, gradient-based approaches are more popular compared to gradient-free methods, primarily due to that gradient-based algorithms show fast convergence and require a reasonably low number of function evaluations (i.e., CFD simulations).

Recently, an open-source software package, pyOpt [173], was developed to formulate and solve nonlinear constrained optimisation problems in an efficient, reusable, and portable manner. A range of optimisation algorithms, including both gradient-based and gradient-free methods, are integrated in pyOpt and can be accessed through a common interface. This flexible framework serves as a platform for practitioners and developers to solve their optimisation problems or develop and benchmark their own optimisation algorithms.

In this section, the commonly-used optimisation algorithms of both types are presented and discussed, although the gradient-based algorithms are used in this thesis.

2.2.4.1 Gradient-based algorithms

For large numbers of variables, gradient-based methods are usually the most efficient optimisation algorithms. This class of methods uses the gradient (also known as sensitivity) of the objective function to determine the most promising search direction. Most algorithms for unconstrained gradient-based optimisation can be described as shown in Algorithm 1. The outer loop represents the major iterations (i.e., design iterations). The design variables are updated at each major iteration k using

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \underbrace{\alpha_k \boldsymbol{p}_k}_{\Delta \boldsymbol{x}_k} \tag{2.1}$$

where $\boldsymbol{p}_k$ is the search direction for the k-th iteration, and α_k is the accepted step length from the line search. It is worth noting that the line search usually involves multiple iterations that do not count towards the major iterations. The two iterative loops represent the two subproblems in this type of algorithm: 1) the computation of a search direction $\boldsymbol{p}_k$; and 2) the search for an acceptable step size α_k. The various types of gradient-based algorithms are classified based on the method that is used for computing the search direction (the first subproblem).

Algorithm 1 General gradient-based optimisation algorithm for smooth functions.

1: **Input:** Initial guess, $\boldsymbol{x}_0$
2: $k \Leftarrow 0$
3: **while** Not converged **do**
4: Compute a search direction $\boldsymbol{p}_k$
5: Find a step length α_k, such that $f(\boldsymbol{x}_k + \alpha_k \boldsymbol{p}_k) < f(\boldsymbol{x}_k)$
6: Update the design variables: $\boldsymbol{x}_{k+1} \Leftarrow \boldsymbol{x}_k + \alpha_k \boldsymbol{p}_k$
7: $k \Leftarrow k + 1$
8: **end while**
9: **Output:** Optimum, $\boldsymbol{x}^*$

With respect to unconstrained optimisation, one of the most basic algorithms is the steepest descent method, which uses the gradient vector at $\boldsymbol{x}_k$ as the search direction for the major iteration k. Although a substantial decrease may be observed in the first few iterations, the method "zigzags" in the design space and is usually very slow to converge to a local minimum. A small modification to the steepest descent method is the conjugate gradient method, which computes the descent direction by adding a contribution from the previous direction. Compared to steepest descent method, conjugate gradient method takes into account the history of the gradients and performs much better in terms of convergence. Unlike these methods that use first-order information to obtain a local model of the function, Newton's method uses a second-order Taylor series expansion of the function about the current design point (i.e., a quadratic model). For a general nonlinear function, Newton's method converges quadratically if $\boldsymbol{x}_0$ is sufficiently close to $\boldsymbol{x}^*$ and the Hessian matrix (containing the second-order partial derivatives) is positive definite at $\boldsymbol{x}^*$. As an alternative to Newton's method, quasi-Newton methods use first-order information only, but build second-order information, i.e., an approximate Hessian, based on the sequence of function values and gradients from previous iterations. The various quasi-Newton methods differ in how they update the approximate Hessian, and the Broyden–Fletcher–Goldfarb–Shanno (BFGS) method has shown to be the most effective and popular quasi-Newton method. One of the primary advantages of quasi-Newton methods over Newton's method is that the Hessian matrix does not need to be inverted, hence significantly reducing the computational cost.

Trust region methods, on the other hand, can be viewed as a reversal of the rules in the line search approaches. A maximum step size (i.e., the size of trust region) is firstly determined, and an approximate minimisation is then performed within that region. This minimisation ultimately allows us to choose the direction and step size simultaneously. Trust region methods generally require fewer iterations than quasi-Newton methods but each iteration is more computationally expensive because of the need for at least one matrix factorisation. Compared to line search methods, trust region methods are more strongly dependent on accurate Hessians.

In most real world, engineering design optimisation problems are very rarely unconstrained. Moreover, the constraints that appear in these problems are typically nonlinear. The simplest method is to try and eliminate constraints by reconstructing the optimisation problem, i.e., transforming design variables using constraints. A more direct approach to dealing with constraints is by adding a penalty function to the objective function that depends on the value of the constraints. There are two main types of penalisation methods: exterior penalty functions, which impose a penalty for violation of constraints; and interior penalty functions, which impose a penalty for approaching the boundary of an inequality constraint. However, more efficient approaches involve solving the Karush–Kuhn–Tucker (KKT) conditions [174, 175], which are the necessary conditions for the optimum of a constrained problem. Allowing for inequality constraints, the KKT approach to nonlinear programming [5] generalises the method of Lagrange multipliers, which allows only equality constraints. Sequential quadratic programming (SQP) is one of the most effective methods for local minimisation of nonlinearly-constrained smooth problems. The SQP methods solve a sequence of subproblems, each of which optimises a quadratic model of the objective function subject to a linearisation of the constraints. To some extent, SQP represents the application of Newton's method to the KKT optimality conditions.

SQP methods have been implemented in numerous software packages (SNOPT, SLSQP, NLPQL, FSQP, etc.) and are widely adopted within the ASO community. SNOPT [176] is a sparse nonlinear optimiser that is particularly useful for solving large-scale constrained problems with smooth objective functions and constraints. The algorithm consists of a SQP algorithm that uses a smooth augmented Lagrangian merit function and approximates the Hessian using a limited-memory quasi-Newton method. SNOPT has been applied successfully to a variety of optimisations, including aerofoils [57, 143], wings [30, 54], and full aircraft [55, 65]. SLSQP [177] optimiser is a sequential least squares programming algorithm which uses the Han–Powell quasi-Newton method with a BFGS update of the Hessian matrix. SLSQP is available in the open-source pyOpt and SciPy [6] package, and is also the default optimiser in the SU2 framework. In this work, SLSQP is used as the gradient-based optimiser to produce optimisation results for 2D [123] and 3D [124] aerodynamic shapes.

For gradient-based optimisation, efficient and accurate gradient evaluation is essential to ensure robust and efficient convergence, especially for problems with large numbers of constraints. Martins and Hwang [178] presented an overview of all the existing methods for computing the derivatives of computational models within a unified mathematical framework. Some representative and commonly-used approaches are briefly introduced herein.

[5] In mathematics, nonlinear programming is the process of solving an optimisation problem where some of the constraints or the objective function are nonlinear.

[6] `https://www.scipy.org/` [retrieved 2019]

The most straightforward method of computing gradients is by means of finite differences. Although the FDM has the advantage of simple implementation, it has limited accuracy due to truncation error for large step sizes and subtractive cancellation error for small step sizes. Additionally, the time required is long when compared with the flow solution, and the total time scales linearly with the number of design variables. The subtractive cancellation errors can be eliminated by using the complex step method, which computes derivatives of real functions using complex variables. The theory for this method was laid out by Squire and Trapp [179], and its implementation was discussed by Martins *et al.* [180]. It allows very small step sizes to be used, thereby almost eliminating truncation error. However, the gradient calculation time is similar to that of finite differences.

Algorithmic differentiation (AD), also known as automatic differentiation or computational differentiation, is a well-known method based on the systematic application of the differentiation chain rule to computer programs [181]. This method can easily produce differentiated codes since the implementation can be done automatically in either the forward or reverse modes. In the forward mode, the differentiated code must be run once for each design variable; but in the reverse mode, the code must be run once for each objective, and so it has a run time that is independent of the number of design variables. The main advantage of AD is that it can provide derivatives with accuracy of machine precision. However, the speed and accuracy advantages of AD, especially in the reverse mode, are offset by a very large memory requirement: the intermediate values of each variable must be stored.

There are two main approaches to implementing AD. The source code transformation approach intersperses lines of code that compute the derivatives of the original code, whereas in the operator overloading approach, the original code does not change, but the variable types and the operations are redefined. In addition, there are a variety of AD tools available for most programming languages. They have been extensively used to obtain the gradients in numerical optimisation. For instance, Jones *et al.* [90] and Müller *et al.* [91] developed discrete adjoint CFD solvers using the AD tool Tapenade [7], and these solvers were then applied to aerodynamic optimisation of multiple geometric shapes [94, 95, 96].

Alternatively, the gradient can be computed analytically. Analytic methods are the most accurate and efficient methods available for computing derivatives; however, they are much more involved than the other methods, as they require a detailed knowledge of the computational model and a long implementation time. There are two forms of the analytic method: the direct and the adjoint forms. As will be shown later in Section 3.2.5, the computational cost when using the adjoint method is independent of the number of design variables, and instead proportional to the number of quantities of interest. Therefore, for practical engineering problems with large numbers of design

[7] https://www-sop.inria.fr/tropics/tapenade.html [retrieved 2019]

variables, the adjoint method is much more efficient than the direct method and hence is preferable to be used. Moreover, the adjoint method is able to compute gradients with a cost that is comparable to the cost of solving the corresponding model.

Adjoint methods have been known and used for over three decades. They were originally introduced to ASO by Pironneau [182]. The method was then extended by Jameson to perform aerofoil shape optimisation [26], and since then, it has been extended to 3D problems and has been widely adopted within the community, leading to applications such as ASO of complete aircraft configurations [64, 65] and aerostructural optimisation [41, 43]. The adjoint method has also been generalised for multidisciplinary systems [40]. Conventionally, adjoint methods are divided into continuous and discrete. In the continuous adjoint approach, the adjoint equations are derived from the governing PDEs and then subsequently discretised, whereas in the discrete approach, the adjoint equations are directly derived from the discretised governing equations. Both continuous and discrete adjoint methods have their own merits. For instance, the continuous approach minimises the memory requirements and the CPU cost per iteration, whereas the discrete approach enables a more straightforward process for code development. Regarding the discrete approach, Giles *et al.* [183] presented a number of algorithm developments concerned with the formulation and solution of adjoint Euler and Navier–Stokes equations. In the SU2 framework, the adjoint method is used to provide gradient information, which can be used for optimal shape design [184], uncertainty quantification (UQ) [185], and goal-oriented adaptive grid refinement [121]. In this work, gradient-based aerodynamic optimisations are performed for both 2D aerofoil [123] and 3D wing [124] geometries with the aid of the adjoint capability within the SU2 suite.

In general, gradient-based algorithms are efficient at finding local minima for high-dimensional, nonlinearly-constrained convex problems; however, most gradient-based algorithms have problems dealing with noisy and discontinuous functions, and they are unlikely to locate the global optimum for multi-modal problems unless the optimisation is started in the region of that optimum. Some approaches can be used to find the global optimal solution in the complete design space. One option is to choose multiple starting points for the gradient-based optimiser. Alternatively, the gradient-free optimisers can be used, which are discussed in the following section.

2.2.4.2 Gradient-free algorithms

Generally, gradient-free algorithms rely purely on the evaluation of the objective function and do not require the computation and use of gradient information. Therefore, the key strength of gradient-free methods is their ability to solve problems that are difficult to solve using gradient-based methods. Many gradient-free algorithms mimic mechanisms or behaviour observed in nature or use heuristics, such as evolution in genetic algorithm (GA) [186], cooling in simulated annealing [187], ant colony food searching in ant colony

optimisation [188, 189], swarm behaviour in particle swarm optimisation (PSO) [190], etc. Unlike gradient-based algorithms in a convex search space, most gradient-free algorithms are designed as global optimisers, and they are able to find multiple local optima while searching for the global optimum. Of the various gradient-free algorithms that have been developed, some representative and commonly-used algorithms are briefly introduced and discussed as follows.

Nelder–Mead simplex

The simplex method of Nelder and Mead [191], also known as nonlinear simplex, performs a search in n-dimensional space using heuristic ideas. A simplex is a structure in n-dimensional space formed by $n+1$ points that are not on the same plane. For example, a triangle is a 2D simplex, and a tetrahedron forms a simplex in 3D space. The Nelder–Mead algorithm starts with a simplex ($n+1$ sets of design variables $\boldsymbol{x}$) and then modifies the simplex at each iteration using four simple operations (as illustrated in Figure 2.8): reflection, expansion, outside/inside contraction, and shrinking. Each of these operations generates a new point (or points in the case of shrinking). The sequence of operations performed in one iteration depends on the value of the objective at the new point relative to the other key points. The algorithm moves the simplex in the design space and automatically shrinks the size of simplex, which can be used as the convergence criterion. Note that the simplex method is a local search algorithm.

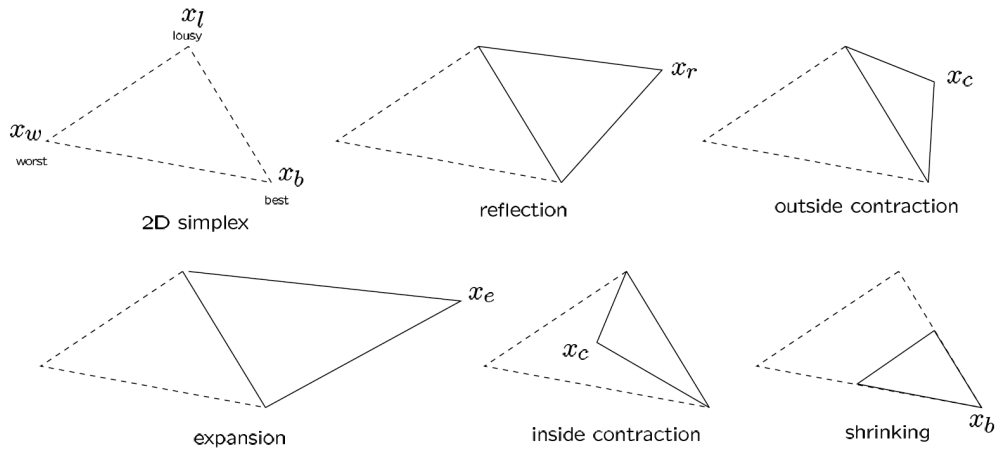


Figure 2.8: Operations performed on the simplex in Nelder–Mead's algorithm for $n = 2$. (Source: [2])

The main strengths of the simplex method are that it requires no derivatives to be computed and it does not require the objective function to be smooth. It has been successfully used in ASO, such as the work done by Widhalm *et al.* [192]. However, the weakness is that it is only efficient for problems with a small number of design variables; otherwise the computational cost for constructing the initial simplex model is expensive, and the convergence becomes increasingly difficult.

Genetic algorithms

The GA is one of the most popular gradient-free optimisation methodologies. Initially developed by Holland [186], GAs are inspired by the process of natural selection, whereby desirable or advantageous traits become more common as a population reproduces and undesirable or disadvantageous traits die out. Like other stochastic optimisation techniques, GAs employ a population which gradually evolves over the course of a number of generations. In the search for a global optimum, GAs are based on three essential components:

1. Selection: survival of the fittest;
2. Crossover: reproduction processes where genetic traits are propagated;
3. Mutation: random operation performed to change the genetic information.

For every component, many different methods and strategies exist. For instance, the popular selection approaches include elitism, tournament selection, and roulette wheel selection, and they have the same idea that individuals with higher fitness have higher chances of being selected into the mating pool. Furthermore, although crossover and reproduction effectively recombine the existing information, some useful genetic information might occasionally be lost. Therefore, the mutation operation is needed to protect against such irrecoverable loss, and it also introduces additional diversity into the population. Generally, the performance of GAs is governed by four important factors: the size of the population, the probability of crossover, the probability of mutation, and the number of generations that the algorithm is run for. The selection of appropriate values of these parameters greatly affects the convergence of an optimisation. In addition, as GAs are probabilistic methods (due to the initial population and mutation), it is crucial to run the optimisation problem multiple times when studying its characteristics.

GAs are radically different from the gradient-based methods. Instead of starting from one point at a time and stepping to a new point for each iteration, GAs attempt to move a population of members away from the undesirable regions of the design space and towards the optimum. Compared to the gradient-based algorithms, the main advantages of GAs are:

- The basic algorithm works on a coding of the design variables (e.g. binary bit string representation) instead of the design variables themselves; consequently, the algorithm can handle mixed continuous, integer, and discrete design variables.
- The population can cover a large range of design space and thus explore multiple local minima simultaneously; as a result, the algorithm increases the likelihood of finding the global optimum by avoiding getting stuck in local minima.
- The implementation is straightforward and easily parallelised.

- GAs can easily be used for multi-objective optimisation. As opposed to gradient-based algorithms that perform multi-objective optimisation through a composite weighted function or by making of the objectives a constraint, GAs determine the fittest population members using the concept of dominance and rank. A Pareto set is generated by collecting the solutions of rank one. The graphical representation of a Pareto set is called a Pareto front, an example of which is shown in Figure 2.9.

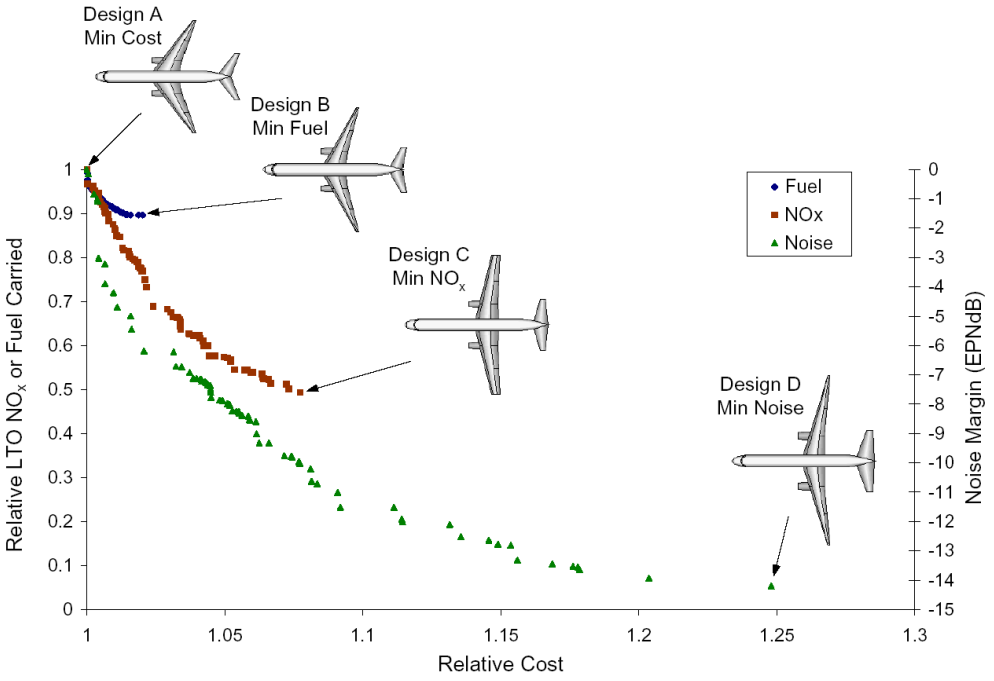


Figure 2.9: An example of Pareto front in aircraft design. (Source: [2])

Despite the merits mentioned above, a typical GA has some drawbacks as well. The main issue is that the GAs are computationally expensive when compared to gradient-based algorithms, especially for problems with large numbers of design variables. The GAs may also suffer from poor exploitation capabilities and premature convergence. Poor exploitation is primarily due to the nature of the encoding for population members, which may prohibit the necessary changes to the variables. Premature convergence can occur due to a lack of population diversity, but can be overcome by adjusting the degree and probability of mutation.

GAs have been successfully applied to ASO problems because of its ease of use, broad applicability, and global perspective. The applications cover a range of topics, including aerofoils [193, 194], wings [194, 195], rotor blades [196], and rocket turbo pumps [197], etc. There are also a number of GA variants such as NSGA-II [198], and the reader is referred to Reference [114] for more discussions.

As a global search method, GAs usually locate the general region of the global optimum but not the precise answer. GAs also exhibit a slow convergence due to the lack of

gradient information. In order to increase the accuracy of the optimum as well as improve the efficiency of optimisation, one common approach is to hybridise a stochastic GA with a deterministic gradient-based method or other local search techniques. Such examples of the hybrid strategy exist in the literature with conjugate gradient [199], quasi-Newton [200], and Nelder–Mead simplex [201] algorithms all employed. All of these hybrid strategies work on the same principle: the GA is used to locate regions of optimal design and the local search method is used to further exploit these regions. The only differences between these strategies are the local search method used and when it is involved during the course of an optimisation. Some hybrid strategies use a local search method to improve on the best member of population at each generation [200]; while others start from the best point of the final generation [201].

Particle swarm optimisation

PSO is a stochastic, population-based optimisation method. It was originally developed by Eberhart and Kennedy [190] after they adopted simulations of simple social behaviour for use in optimisation. Like the GA, the PSO is also inspired by nature; though instead of modelling the evolutionary processes, the PSO models the collective behaviour of a population of animals, such as birds, fish, and insects. The basic PSO formulation contains the following two core points:

- A swarm of agents (or particles) represents a group of potential solutions and moves in n-dimensional design space looking for the best solution.
- Each agent adjusts its movement according to the effects of cognitivism (self experience) and sociocognition (social interaction). In other words, each agent remembers the location where it found its best result so far, and it exchanges information with the swarm about the location where the swarm has found the best result so far.

Mathematically, PSO can be viewed as an iterative process. As shown in Equation (2.2), at each time step a particle adjusts its velocity with an inertial component (a continuation of its previous direction), a weighted random component toward its personal best location, and a weighted random component toward the swarm's best location.

$$v_{k+1}^i = wv_k^i + c_1 r_1 (p_k^i - x_k^i) + c_2 r_2 (p_k^g - x_k^i) \tag{2.2}$$

With the new velocity, particle i simply updates its position as

$$x_{k+1}^i = x_k^i + v_{k+1}^i \tag{2.3}$$

Note that the artificial time dependence Δt is eliminated for convenience. The parameters in the equations are explained as follows:

- p_k^i is particle i's best position so far, and p_k^g is the swarm's best particle position so far.
- c_1 and c_2 are the cognitive parameter (confidence in itself) and social parameter (confidence in the swarm), respectively. Both parameters are in the interval $[0, 2]$ and typically close to 2.
- w is the inertia parameter in the interval $[0, 1.2]$; typical inertial coefficients are between 0.8 and 1.2. A lower value of w dampens the particle's inertia and tends toward faster convergence to a minimum. A higher value of w accelerates the particle's inertia and tends toward increased exploration to potentially help discover multiple minima.
- r_1 and r_2 are random numbers in the interval $[0, 1]$.

Compared to other global optimisation approaches, PSO has several outstanding characteristics. Firstly, PSO is a simple algorithm and is easy to implement. Secondly, PSO is still a population-based algorithm; however, it works well with few particles (usually 10 to 40) and directly updates the design variables, unlike using the generations in the evolutionary approaches. Thirdly, PSO has both global and local search behaviour that can be directly adjusted as desired using the cognitive c_1 and social c_2 parameters. Moreover, PSO allows for convergence balance thorough the inertial weight factor w. In addition, the basic PSO algorithm is inherently an unconstrained optimiser and can include constraints through the use of penalty methods or augmented Lagrangian function.

PSO has grown in popularity since its inception by Eberhart and Kennedy. Based on the original formulation, Shi and Eberhart [202, 203] further improved the performance of the algorithm by adjusting parameters in Equation (2.2). PSO is one of the most widely-used agent-based optimisation algorithms, and has been applied in MDO of an aircraft wing [204] and the conceptual design of aircraft [205]. In the case of aerofoil optimisation, PSO outperforms evolutionary algorithms in terms of efficiency [206] and exploration ability [207], indicating that the PSO algorithm designed for continuous optimisation is more effective.

Although gradient-free optimisation methods can be relied upon to reach a global optimum, they typically require a very large number of evaluations of the objective function to do so. In the scenario of ASO, the objective function evaluation usually involves a high-fidelity CFD simulation, and the computational time required by a gradient-free method can be much higher than that of a gradient-based approach. Even with the parallel computing capability it may be infeasible to use gradient-free method of optimisation for some complex problems.

Surrogate modelling [97], also known as response surface modelling or meta-modelling, aims to facilitate optimisation but with a reduced number of objective function evaluations. It can be viewed as a nonlinear inverse problem with the purpose of determining a continuous function that relates the design variables to output responses from limited data. The surrogate-assisted aerodynamic optimisation alleviates the computational burden by defining a simplified mathematical relationship allowing for fewer numerical simulations to be performed. In recent years, numerous surrogate models have been developed, including polynomial response surface model (RSM) [208], kriging [209, 210], RBFs [211], neural network [212], support vector regression [213], and polynomial chaos expansion [214], etc. Queipo *et al.* [215] and Forrester and Keane [115] provided comprehensive overview and discussions of the surrogate modelling methods and their use in optimisation strategies. In terms of generating initial sample points in the design space, Giunta *et al.* [216] provided an overview of modern design of experiments (DoE) techniques (such as quasi-Monte Carlo sampling, Latin hypercube sampling, and orthogonal array sampling) that can be applied in computational engineering design studies.

Kriging is a statistical interpolation method proposed by Krige [209] in 1951, and was first used by geologists to estimate mineral concentrations within a particular region. It has since been adapted by Sacks *et al.* [210] for use in the creation of surrogate models of deterministic computational experiments. Among the various types of surrogate modelling techniques, the kriging method has gained increasing popularity due to its unique ability to model complicated responses based on sampled data whilst also providing an error estimation (indicating the uncertainty of prediction). Since its initial application to surrogate modelling, kriging has been successfully applied to a variety of engineering problems. Meanwhile, there have been further developments with the aim of improving the efficiency and accuracy of the kriging method. Representative examples of such advancements include gradient-enhanced kriging [217, 218, 219], cokriging [101, 102, 220], and hierarchical kriging [221]. The reader is referred to Reference [117, 221] for more discussions of the kriging method.

The use of surrogate models has helped to reduce the computational cost of optimisation. However, they only work well with a small number of variables. This is because the sample size required to accurately construct the surrogate model is often exponential to the dimensionality of the design space. For 3D wing or more complex problems with large numbers of design variables, a significant amount of sample points are therefore needed to represent the high-dimensional design space, and the high-fidelity simulations at these sample points would pose a computational challenge. As the dimensionality of a design space increases so does the complexity of the optimisation. Eventually the surrogate itself becomes intractable and the surrogate-based optimisation falls into the so-called "curse of dimensionality". Moreover, in a real world engineering environment, there are limited computational resources and time constraints due to the integration of optimisation within a much larger design process. The time limit may have a serious

impact on the performance of the resulting design. Given a restricted simulation budget, the computational expense of generating the surrogate model is increased as the dimensionality is enlarged; as a result, the number of available evaluations is reduced and may become insufficient to optimise the problem effectively. The insufficiency to fully explore the expanded design space would counteract the increased flexibility of shape parameterisation. Consequently, the optimisation with an increased dimensionality may produce poorer designs than at lower dimensions even though the increased flexibility can theoretically produce a better design. The issue of dimensionality remains to be addressed to improve the efficiency of surrogate-based optimisation.

2.2.5 Section summary

In this section, a review is provided regarding the constitutive computational techniques within an ASO framework. An aerodynamic solver is able to model the flow physics around an aerodynamic body. Over the past decades, a number of aerodynamic codes and software with increasing level of fidelity have been developed in the research community. Recently the efficient Q3D aerodynamic approach has also gained popularity to solve 3D wing problems. Geometry parameterisation has a significant impact on the design space and hence the optimisation result. The need for flexible, robust, and efficient parameterisation methods has motivated researchers to develop various approaches originating from different concepts. Once a geometry has been perturbed, the surface and volume grids must also be moved or updated. An efficient and robust grid deformation scheme is essential for aerodynamic optimisation. Various grid deformation techniques have been proposed, based on either interpolation or physical analogy. An optimiser updates the design variables and drives the design loop toward the optimum. For gradient-based optimisation algorithms, adjoint method is able to compute the gradients accurately and efficiently, particularly for problems with large numbers of design variables; for gradient-free optimisation methods, the use of surrogate modelling such as kriging can significantly reduce the computational cost.

2.3 Sensitivity assessment

Sensitivity analysis is the study of how the outputs of a model change in response to changes in its inputs. It plays a key role in gradient-based optimisation, UQ, error analysis, and computational model-assisted decision making. In the engineering literature, the term "sensitivity analysis" is often used to refer to the computation of derivatives [178]. In this context, we use "sensitivity assessment" to distinguish from sensitivity analysis, and to refer to the influence of a specific parameter (or approach) on the optimal solution obtained from ASO.

As discussed in the previous section, there are a variety of methods and techniques for all of the constituent parts of an ASO framework. Each of these methods has its own merits and drawbacks, and may be suitable for a particular type of problems. Due to this fact, the quality and applicability of results gained by numerical optimisation are inherently dependent on a large range of factors, such as the fidelity of the analysis tool used, the flexibility of the shape parameterisation, and the complexity of the optimisation scheme. It is therefore necessary to isolate the contribution of a single component and investigate the influence of different approaches on the optimisation result. Moreover, it is also crucial to assess the sensitivity of the optimal solution with respect to key parameters of the optimisation framework. Much effort has been made in the previous studies toward the sensitivity assessment, and a brief review is provided in this section.

Geometry parameterisation is undoubtedly one of the most critical factors that have a large impact on the optimal solution. Song and Keane [222] studied the effect of two parameterisation approaches, B-spline and orthogonal basis function, on inverse fitting of different aerofoils. It was shown that the B-spline approach produced results with higher accuracy while the basic function approach was more efficient. Castonguay and Nadarajah [137] compared using grid points, B-spline curves, HHBFs, and the PARSEC method for aerofoil design. For a viscous transonic inverse design case, grid points and B-spline curves provided higher level of accuracy. B-spline curves and HHBFs approach showed comparable performance for drag minimisation of the same case. A similar study was presented by Mousavi *et al.* [138] for a wing comparing grid points, B-spline surfaces, and CST. B-spline surfaces and grid points outperformed the CST method for both the inviscid transonic inverse design and drag minimisation studies. Amoiralis and Nikolos [223] conducted a comparison between FFD and B-spline surface control for a series of inverse design optimisations. FFD was found to outperform B-spline approach for a comparable number of design variables. However, the optimisations conducted were 2D aerofoil inverse problems and used the low-fidelity XFOIL solver. Sripawadkul *et al.* [224] employed several desirable metrics to compare five aerofoil parameterisation techniques: Ferguson's curves, HHBFs, B-splines, PARSEC, and CST. The results suggested that the metrics can provide a basis for objective comparison, allowing the designers to select the proper parameterisation according to the problem at hand. Zhu *et al.* [225] investigated aerofoil inverse fitting problems using PARSEC, CST, and MACROS dimension-reduction method. It was shown that MACROS dimension-reduction method produced results with relatively higher accuracy while the CST and PARSEC method provided a design space with higher flexibility.

More recently, Masters *et al.* [226] investigated the influence of shape parameterisation and dimensionality on the optimisation of a benchmark problem described by ADODG (Case 1: drag minimisation of the NACA0012 aerofoil in inviscid transonic flow). The six parameterisation methods used were: Bézier surface FFD, B-splines, CST, HHBFs, a RBF domain-element method, and SVD. It was found that the ability to ensure smooth

geometries is a key requirement for the robustness and fast convergence of the optimisations, and the B-spline approach was shown to achieve the lowest drag. The same methods, plus PARSEC, were compared by Masters *et al.* [118] for geometric shape recovery of a dataset of more than 2000 aerofoils. The results varied depending on the error-tolerance values used, but considering a range of numbers of design variables, the SVD approach provided the most efficient design-space coverage, particularly when a tighter tolerance was used. Lee *et al.* [56] compared two geometry control methods, namely B-spline surface and FFD, with respect to their effectiveness for ASO. In general, both methods performed equally well for a range of challenging optimisation problems. The results suggested that B-spline surface approach is better suited for simple geometries, such as wings, whereas FFD is advantageous for more complex geometries, such as unconventional aircraft, and is particularly well suited to a multi-start algorithm and adaptive geometry control strategies. Zhang *et al.* [96] employed both B-spline and NURBS parameterisation to perform ASO of the ONERA M6 wing in transonic inviscid flow. The results showed that NURBS outperformed B-spline approach with fewer control points and produced a smoother shape with smaller variation of curvature, which is beneficial for aerodynamic performance.

Apart from shape parameterisation, the other factors have also been shown to have a significant impact on the optimisation. LeDoux *et al.* [227] studied several ADODG test cases using various optimisation frameworks. The resulting optimised and associated baseline geometries were cross-analysed by four well-validated CFD codes. It was shown that, for the same geometry, different drag counts were obtained by those flow solvers. Poole *et al.* [125, 228] investigated 2D aerofoil optimisation problems using two optimisers, namely a gravitational search algorithm and a feasible SQP (FSQP) algorithm. The global gravitational search algorithm was shown to outperform the gradient-based FSQP algorithm for almost all the cases tested. Regarding the ADODG Case 1, the best solution with lowest drag was found by using a hybrid approach, where the gradient-based method was run on the results obtained from the gradient-free optimisation. Lyu *et al.* [195] evaluated a series of optimisation algorithms (embedded in pyOpt framework) for three test cases of increasing complexity, including a multi-dimensional Rosenbrock function and two wing ASO problems. The majority of the gradient-based optimisers successfully solved all three test cases, while the gradient-free methods required two or three orders of magnitude more computational cost when compared to the gradient-based methods. A similar study was also performed by Yu *et al.* [229], who applied six gradient-based algorithms and three gradient-free algorithms to ASO of the CRM wing. The same findings were observed with the conclusion that gradient-based methods with adjoint gradients are better suited for large-scale ASO problems. Poole *et al.* [230] presented a study of the formulation of the transonic aerofoil optimisation problem and its effect on the performance over a range of operating conditions. In addition to single-point and multi-point optimisations, an improved range-based optimisation problem was

formulated, and it was found to be a more practical approach than drag minimisation and also more indicative of an industrial design objective.

Given an aerodynamic optimisation framework, one can solve the optimisation problem and obtain the optimised geometric shape. The optimisation framework typically involves a large number of parameters, and the selection of appropriate values for these parameters greatly affects the optimisation result. Although many parameters are usually set-up using rules of thumb and are generally suitable for most cases, however, we can not guarantee the optimal solution produced is truly optimal for a specific case. Hence, there is a need to adjust and fine-tune these parameters to gain the best optimisation performance. Currently there are only a few publications regarding this issue. For instance, Keane [231] showed that, with the appropriate choice of control parameters of the GA, the convergence and robustness of optimisation can be significantly improved. Toal *et al.* [98] assessed the performance of several different tuning strategies for the hyperparameters of a kriging model, with the purpose of building an effective surrogate model of high-fidelity computational simulations. Toal [103] also provided some guidelines for the construction of a multi-fidelity kriging model. With respect to geometry parameterisation, Masters *et al.* [232] applied a variety of different implementations to each parameterisation method in an attempt to obtain the best possible results for geometric shape recovery of a large dataset of aerofoils. However, the implementation choice employed in this study may not be well-suited for a specific optimisation problem. There is still the necessity to further quantify the effect of parameter settings in shape parameterisation on the optimisation result for a particular case.

2.4 Multi-fidelity aerodynamic model

When it comes to 3D wing design, an accurate aerodynamic analysis depends on high-fidelity CFD models. However, at the early stages of aircraft design, designers rely heavily on empirical and linear correlations owing to their simplicity and low computational cost. Moreover, optimisation of an aircraft wing often requires a large number of repetitive aerodynamic evaluations of different wing configurations, which further prohibits the routine use of the high-fidelity CFD solvers from an industrial viewpoint. Therefore, it is of great interest to find a medium-fidelity approach, which combines the advantages from both ends of the fidelity spectrum and thus provides a trade-off between computational efficiency and accuracy. Additionally, the involvement of physics-based models into conceptual and preliminary design stages can effectively reduce the number of design iterations and thus shorten the design cycle.

An interesting approach to calculation of the aerodynamic forces with sufficient accuracy and efficiency is the so-called Q3D method, which combines 3D inviscid wing data with 2D viscous aerofoil data. The 3D wing data usually comes from the lifting-line theory

(LLT) or the VLM, whereas the 2D aerofoil data is obtained from a 2D solution of the Navier–Stokes equations. The resulting Q3D approach is nonlinear because the sectional flow nonlinearities are obtained from the 2D CFD analysis. As the LLT or the VLM is inexpensive, the overall cost of a Q3D analysis is comparable to that of a 2D CFD analysis, which makes the Q3D method attractive for use in early phases of aircraft design and optimisation.

Various applications of the Q3D approach have been done in the past [18, 32, 48, 233, 234]. Van Dam *et al.* [233] presented a high-lift system design methodology that combines a modified lifting-line method and 2D viscous data. Piperni *et al.* [18] demonstrated the deployment of a Q3D method in an industrial environment. More recently, Mariens *et al.* [48] developed a Q3D solver for wing drag calculation using the combination of the VLM and a 2D aerofoil analysis tool. The proposed Q3D solver was then applied by Elham [234] to wing ASO and was further extended to aerostructural optimisation [32]. It is worth observing that these references rely on a 2D flow analysis to correct the predictions obtained from a linear 3D aerodynamic model. Flow physics often requires treating swept wing flows. With regard to a swept wing, the aforementioned work either neglected the sweep effects or used the Kuchemann correction [235], where the freestream Mach number is modified by a cosine law to correct 2D solutions. However, this correction is only valid for incompressible inviscid flows. Thus, inaccuracies would arise because experimental results [236] show that the cross-flow effects strongly influence the boundary layer separation as well as the position of shock waves. It was also confirmed by numerical results [237] that the 2D data incorporating cross-flow effects can better predict the maximum lift coefficient and the shock wave position.

The incorporation of sweep effects into 2D aerofoil characteristics results in a model that is equivalent to an infinite-swept wing (ISW), which assumes a wing of infinite span with a constant wing sweep angle. Regarding a wing on a typical commercial aircraft, the solution of the ISW model is valid at a reasonable distance from the fuselage and wing tip. For ISW, the 3D Navier–Stokes equations can be solved using a 2D formulation augmented by an additional crossflow equation. As illustrated in Figure 2.10, the inclusion of sweep effects in an ISW model can better capture the physical effects such as the shock waves and combined streamwise/crossflow trailing-edge boundary-layer flows. To address this issue, Bourgault *et al.* [238] developed an approach that solves 3D flow problems, specifically for the ISW, using an existing 2D solver and 2D grid. The proposed aerodynamic approach was subsequently coupled into a Q3D solver for wing aerodynamic analysis [50] and optimisation [239]. The results show good agreement with wind-tunnel experimental data or high-fidelity numerical data for swept wing configurations; nonetheless, the applications were only limited to steady-state flow problems.

To address this challenge, Franciolini *et al.* [240] developed an ISW solver for both steady and unsteady flow problems. The proposed approach solves the 3D ISW Navier–Stokes

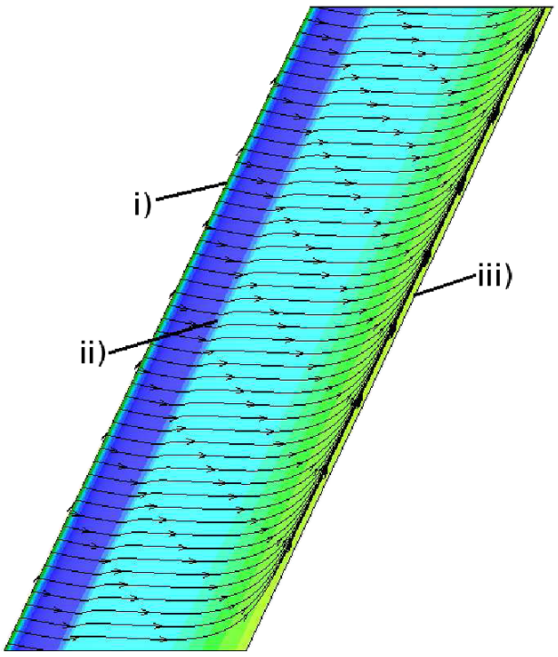


Figure 2.10: Physical effects captured by ISW RANS solution: i) stagnation region; ii) shock waves; iii) trailing-edge crossflow boundary-layer thickening/separation (Source: [50]).

equations using a 2D formulation augmented by an additional cross-flow equation. It is worth noting that the ISW solver achieves a computational cost reduction by at least 75% compared with the cost required by a 3D stencil approach. A multi-fidelity Q3D aerodynamic solver was subsequently developed in the author's group [1, 241], which builds upon the coupling of the ISW solver with a steady/unsteady VLM solver. The hybrid flow solver can not only rapidly calculate the aerodynamic loads based on physical models, but also capture the nonlinear effects and 3D phenomena on finite wings. By far, the multi-fidelity approach has been extensively validated and successfully used for both steady and unsteady flow problems [1, 241], and it has also been applied for the identification of transonic buffet envelope of a 3D wing [242]. Compared to a fully 3D aerodynamic solver, the multi-fidelity solver demonstrates a speed-up value of up to 97% for application to geometries of industrial relevance. In this thesis, we further extend its application to wing twist optimisation by embedding the hybrid solver into an ASO framework. More details of the hybrid solver as well as the optimisation framework are introduced in Section 3.3, and the optimisation results are presented in Chapter 6.

2.5 Chapter summary

ASO is a field of engineering that uses optimisation methods to solve design problems incorporating aerodynamics. In this chapter, a comprehensive but not exhaustive literature review is provided. Firstly, the concept of design optimisation was introduced;

the development and current status within the area of aerospace engineering were then presented, starting from MDO, then aerostructural optimisation, and finally narrowing down to ASO. In particular, the research works of ASO from several representative institutions were summarised, highlighting their contributions to the community. Secondly, a detailed survey and review of the computational techniques for an ASO framework were carried out. For each constitutive part, we outlined some commonly-used approaches and discussed their pros and cons. Thirdly, the previous studies regarding the sensitivity assessment were reviewed, indicating the significance and motivation for part of this research work. Lastly, the development of multi-fidelity aerodynamic models, particularly the Q3D approach, was reviewed, highlighting its computational efficiency and potential application for shape optimisation.

Chapter 3

Computational Methodology

3.1 Introduction

Two aerodynamic solvers, SU2 and FALCon, are used to perform gradient-based aerodynamic optimisations in this work. Specifically, the 2D aerofoil optimisation problems are investigated using SU2; the 3D wing twist optimisations are conducted using both high-fidelity SU2 and multi-fidelity FALCon solvers. In this chapter, the governing equations, numerical methods, and computational approaches within these two optimisation frameworks are presented. The parameter settings and guidelines used in the simulations are also provided.

3.2 SU2

The open-source SU2 code [120, 121] has been developed to perform tasks related to partial differential equations (PDE) analysis and PDE-constrained optimisation on unstructured grids. A key feature of SU2 software suite is that each module is designed for specific functionality and can be executed individually. Most notable is a high-fidelity CFD solver that is capable of solving a wide range of engineering problems. By coupling multiple modules and integrating with other packages, SU2 enables complex activities to be performed, such as adaptive grid refinement [121], UQ [185], and optimal shape design [184]. In the current work, the SU2 solver (v5.0 "Raven") is employed for ASO studies, and the simulations are performed on IRIDIS 4, which is the fourth generation of high-performance computing cluster at the University of Southampton.

3.2.1 Optimisation framework

A typical design process for aerodynamic optimisation is illustrated in Figure 3.1. A baseline geometry and grid are taken as input to the design cycle, along with a chosen objective function, J, to evaluate the optimisation performance and a vector of design variables, $\vec{x}$, to parameterise the shape. When the gradient of the objective function, ∇J, is obtained using the adjoint method, a gradient-based optimiser is then initiated to drive the design cycle and guide the search for the optimum. In this work, the SLSQP (Sequential Least SQuares Programming) optimiser is used. The optimisation process is terminated when the convergence criteria, the Karush–Kuhn–Tucker (KKT) conditions [174, 175], are satisfied or the number of design iterations exceeds a prescribed maximum number.

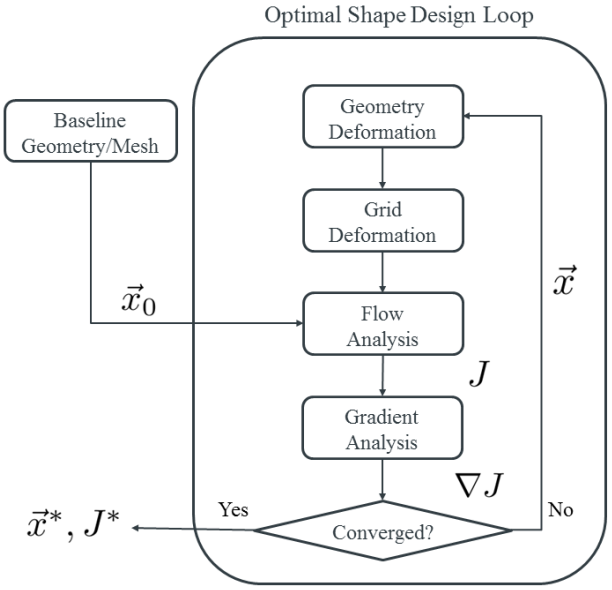


Figure 3.1: Flow chart for gradient-based shape optimisation within SU2.

The primary modules that are called by the optimiser in the design loop are introduced briefly as follows:

- `SU2_CFD` - performs direct and adjoint flow analysis by solving governing PDE equations.
- `SU2_DEF` - perturbs the geometry with a chosen parameterisation and deforms the surrounding volume grid using an approach based on the linear elasticity equations.
- `SU2_DOT` - computes the gradients (i.e. partial derivative of a function with respect to the shape design variables) by projecting the adjoint surface gradients into the design space through a dot product operation.

3.2.2 Flow solver

In this work, the flows around aerodynamic bodies are governed by compressible Euler or Reynolds-averaged Navier–Stokes (RANS) equations. Following the general notation presented in Reference [120], the RANS equations are formulated as

$$\frac{\partial \boldsymbol{U}}{\partial t} + \nabla \cdot \boldsymbol{F}^c - \nabla \cdot (\mu^{vk} \boldsymbol{F}^{vk}) - \boldsymbol{Q} = \mathcal{R} = 0 \tag{3.1}$$

where $\boldsymbol{U}$ represents the vector of conservative variables, $\boldsymbol{F}^c(\boldsymbol{U})$ and $\boldsymbol{F}^{vk}(\boldsymbol{U})$ are convective and viscous fluxes, respectively, $\boldsymbol{Q}(\boldsymbol{U})$ is a generic source term, and $\mathcal{R}(\boldsymbol{U})$ denotes the numerical residual of the governing equations. By setting the vectors $\boldsymbol{Q}$ and $\boldsymbol{F}^{vk}$ to zero, the Euler equations are obtained.

Specifically, the conservative variables are given by $\boldsymbol{U} = [\rho,\, \rho\boldsymbol{v},\, \rho E]^T$; and the convective fluxes, viscous fluxes, and source terms are

$$\begin{aligned}
\boldsymbol{F}^c &= \begin{bmatrix} \rho\boldsymbol{v} \\ \rho\boldsymbol{v} \bigotimes \boldsymbol{v} + \boldsymbol{I}p \\ \rho E\boldsymbol{v} + p\boldsymbol{v} \end{bmatrix}, \quad \boldsymbol{F}^{v1} = \begin{bmatrix} \cdot \\ \boldsymbol{\tau} \\ \boldsymbol{\tau} \cdot \boldsymbol{v} \end{bmatrix}, \\
\boldsymbol{F}^{v2} &= \begin{bmatrix} \cdot \\ \cdot \\ c_p \nabla T \end{bmatrix}, \quad \boldsymbol{Q} = \begin{bmatrix} q_\rho \\ \boldsymbol{q}_{\rho v} \\ q_{\rho E} \end{bmatrix}
\end{aligned} \tag{3.2}$$

where ρ, p, and T are the density, static pressure, and temperature of the fluid, respectively; $\boldsymbol{I}$ is the identity matrix; $\boldsymbol{v} = [u,\, v,\, w]^T$ is the flow velocity in a Cartesian system of reference; E is the total energy per unit mass; c_p is the specific heat at constant pressure; and the viscous stress tensor can be written as

$$\boldsymbol{\tau} = \nabla \boldsymbol{v} + \nabla \boldsymbol{v}^T - \frac{2}{3}\boldsymbol{I}(\nabla \cdot \boldsymbol{v}) \tag{3.3}$$

Assuming a perfect gas with a ratio of specific heats γ and gas constant R, the pressure is determined from

$$p = (\gamma - 1)\Big(\rho E - \frac{1}{2}\rho(\boldsymbol{v} \cdot \boldsymbol{v})\Big), \tag{3.4}$$

the temperature is given by

$$T = \frac{p}{\rho R}, \tag{3.5}$$

and the specific heat at constant pressure is given by

$$c_p = \frac{\gamma R}{\gamma - 1} \tag{3.6}$$

The governing PDE equations are numerically solved using a finite volume method (FVM). The convective fluxes can be discretised using central or upwind methods.

Several numerical schemes have been implemented in SU2, and the classic Jameson–Schmidt–Turkel (JST) [243] scheme is mainly used for spatial discretisation in this study. For steady flows, the solution of the governing equations is marched in time until a steady state is reached. Both explicit and implicit time integration methods are available in SU2, and the implicit Euler scheme is employed in the current work.

The steady-state RANS solver is a key component of the SU2 suite, and the flow solver has been extensively used and rigorously validated and verified [120, 121] across many flow regimes. In this work, the validation of SU2 is carried out for three classic test cases, and the results are provided in Appendix A.

3.2.3 Turbulence modelling

According to the standard approach to turbulence modelling based on the Boussinesq hypothesis [244], the total viscosity is divided into a laminar μ_{dyn} and a turbulent μ_{tur} component. To close the system of RANS equations, the dynamic viscosity μ_{dyn} is assumed to satisfy Sutherland's law [245], and depends only on the temperature; the turbulent viscosity μ_{tur} is computed via a turbulence model. The viscosity terms in Equation (3.1) are given by

$$\mu^{v1} = \mu_{\text{dyn}} + \mu_{\text{tur}}, \quad \mu^{v2} = \frac{\mu_{\text{dyn}}}{Pr_d} + \frac{\mu_{\text{tur}}}{Pr_t} \tag{3.7}$$

where Pr_d and Pr_t are the dynamic and turbulent Prandtl numbers, respectively.

The turbulent viscosity is obtained from a suitable turbulence model involving the flow state and a set of new variables. The one-equation Spalart–Allmaras (SA) [246] model is one of the most common and widely-used turbulence models for the analysis and design of engineering applications in turbulent flows. In this work, the SA model is used and is briefly described in the following.

In the case of SA model, the turbulent viscosity is computed as

$$\mu_{\text{tur}} = \rho\hat{\nu} f_{v1}, \quad f_{v1} = \frac{\chi^3}{\chi^3 + c_{v1}^3}, \quad \chi = \frac{\hat{\nu}}{\nu}, \quad \nu = \frac{\mu_{\text{dyn}}}{\rho} \tag{3.8}$$

The new variable $\hat{\nu}$, known as pseudo eddy viscosity, is obtained by solving a transport equation that includes the following convective, viscous, and source terms:

$$\begin{aligned} &\boldsymbol{F}^c = \boldsymbol{v}\hat{\nu}, \quad \boldsymbol{F}^v = -\frac{\nu + \hat{\nu}}{\sigma}\nabla\hat{\nu}, \\ &\boldsymbol{Q} = c_{b1}(1 - f_{t2})\hat{S}\hat{\nu} - c_{w1} f_w \left(\frac{\hat{\nu}}{d_S}\right)^2 + \frac{c_{b2}}{\sigma}|\nabla\hat{\nu}|^2 \end{aligned} \tag{3.9}$$

where the production term $\hat{S}$ is defined as

$$\hat{S} = |\boldsymbol{\omega}| + \frac{\hat{\nu}}{\kappa^2 d_S^2} f_{v2} \tag{3.10}$$

$\boldsymbol{\omega} = \nabla \times \boldsymbol{v}$ is the fluid vorticity, d_S is the distance to the nearest wall, and

$$f_{v2} = 1 - \frac{\chi}{1 + \chi f_{v1}} \tag{3.11}$$

The function f_{t2} is

$$f_{t2} = c_{t3} \exp(-c_{t4}\, \chi^2) \tag{3.12}$$

and the function f_w is computed as

$$f_w = g\left[\frac{1 + c_{w3}^6}{g^6 + c_{w3}^6}\right]^{1/6} \tag{3.13}$$

where $g = r + c_{w2}(r^6 - r)$ and

$$r = \min\left[\frac{\hat{\nu}}{\hat{S}\kappa^2 d_S^2}, 10\right] \tag{3.14}$$

Finally, the set of closure constants for the SA model is given by

$$\begin{aligned} &\sigma = 2/3, \quad c_{b1} = 0.1355, \quad c_{b2} = 0.622, \quad \kappa = 0.41, \\ &c_{w1} = \frac{c_{b1}}{\kappa^2} + \frac{1 + c_{b2}}{\sigma}, \quad c_{w2} = 0.3, \quad c_{w3} = 2, \quad c_{v1} = 7.1, \\ &c_{t3} = 1.2, \quad c_{t4} = 0.5 \end{aligned} \tag{3.15}$$

The original reference [246] made use of a trip function that most people do not include, because the model is most often employed for fully turbulent applications. The model implementation in SU2 code also neglects the trip terms, c_{t1} and c_{t2}. Herein, the SA model includes 9 closure coefficients, which are used for both RANS simulations and sensitivity assessment in aerodynamic optimisation (see Section 5.5.3). In this work, the convection terms in the SA turbulence model are computed using a first-order upwind scheme. The time advancing method for the turbulence model is the implicit Euler method.

3.2.4 Adjoint solver

In terms of ASO, a typical problem formulation seeks to optimise an objective function J (also termed cost function or merit function) with respect to changes in the shape of the geometry. The objectives, as chosen by the designer, usually refer to the integrated forces (lift, drag, etc.) and moments on the solid surface. Assuming that an infinitesimal deformation is imposed to the surface along the normal direction, the flow state variables

$\boldsymbol{U}$ in the domain will be perturbed and these variations in the state are constrained to satisfy the governing equations, i.e. $\mathcal{R}(\boldsymbol{U}) = 0$ must be satisfied for any candidate shape. As a result, the aerodynamic quantity that we focus on will either improve or deteriorate due to the change of flow solutions. The surface sensitivity (or surface gradients) is a measure of the variation of the objective function with respect to infinitesimal shape perturbations in the local normal direction.

Gradients of a function of interest can be computed in a variety of ways. In the SU2 framework, the surface gradients are obtained using the adjoint methods due to their ability of computing the derivatives at a cost comparable to solving the state PDEs. The SU2 code contains both two implementations of the adjoint methodology: a continuous adjoint by Stanford University and a discrete adjoint via AD by the Technical University of Kaiserslautern. Since debuting in the initial public release of SU2, the continuous adjoint solver has been extensively used and rigorously verified [247, 248, 249] for both inviscid and viscous problems across many flow regimes. On the other hand, the discrete adjoint formulation has been implemented but its incorporation into SU2 was ongoing when this study was conducted. Therefore the continuous adjoint approach is used in this work. Again, following the notation in Reference [120], the adjoint RANS equations are given by

$$-\frac{\partial \boldsymbol{\Psi}^T}{\partial t} - \nabla \boldsymbol{\Psi}^T \cdot \left[\frac{\partial \boldsymbol{F}_i^c}{\partial \boldsymbol{U}} - \mu^{vk}\frac{\partial \boldsymbol{F}_i^{vk}}{\partial \boldsymbol{U}}\right] - \nabla \cdot \left[\nabla \boldsymbol{\Psi}^T \cdot \mu^{vk}\frac{\partial \boldsymbol{F}_i^{vk}}{\partial(\partial_j \boldsymbol{U})}\right] - \boldsymbol{\Psi}^T \frac{\partial \boldsymbol{Q}}{\partial \boldsymbol{U}} = 0 \quad (3.16)$$

where $\boldsymbol{\Psi}$ are the adjoint variables that correspond to each of the conserved variables in the direct flow problem, i, $j = 1, 2, 3$, and $k = 1, 2$. The adjoint Euler equations can be easily obtained by removing the corresponding viscous and source terms. The numerical discretisation and solution of Equation (3.16) follows the methods for solving Equation (3.1). The reader is referred to Reference [247] for more details about the continuous adjoint approach.

3.2.5 Gradient evaluation

In gradient-based optimisation, minimisation of the objective function is achieved through an iterative process in which the gradients are used to guide the design towards the optimal solution. Therefore, it is essential to compute the gradients accurately and efficiently in each design step to provide the optimiser with a new search direction. With regard to the SU2 optimisation framework, the gradient evaluation can be formulated as the

following equation:

$$\underbrace{\begin{bmatrix} \frac{\partial f}{\partial p_1} \\ \frac{\partial f}{\partial p_2} \\ \vdots \\ \frac{\partial f}{\partial p_n} \end{bmatrix}}_{\text{Gradients}} = \underbrace{\begin{bmatrix} \frac{\partial x_1}{\partial p_1} & \frac{\partial x_2}{\partial p_1} & \cdots & \frac{\partial x_m}{\partial p_1} \\ \frac{\partial x_1}{\partial p_2} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ \frac{\partial x_1}{\partial p_n} & \cdots & \cdots & \frac{\partial x_m}{\partial p_n} \end{bmatrix}}_{\text{Geometric gradients}} \underbrace{\begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_m} \end{bmatrix}}_{\text{Surface gradients}} \tag{3.17}$$

where n and m are the number of design variables and surface grid nodes, respectively; f represents the function of interest, being the objective or constraint function; p_i ($i = 1, 2, \ldots, n$) are the design variables; the variables x_j ($j = 1, 2, \ldots, m$) denotes the local surface normal displacements for each discrete grid node on the geometry surface.

The Jacobian matrix $\partial \boldsymbol{x}/\partial \boldsymbol{p}$ is known as geometric gradients, which measure the influence of change of design variables on the positions of surface grid nodes. The geometric gradients are calculated using FDM, of which the computational cost is negligible as it does not involve the solution of the governing PDE equations.Following a perturbation to each of the design variables (such as the amplitude coefficients of HHBFs), the displacement of a grid point on the surface is firstly measured. By projecting this displacement along the normal direction of the surface, we then obtain the geometric sensitivity. Each component of the Jacobian matrix is given by:

$$\left(\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{p}}\right)_{i,\,j} = \frac{\partial x_j}{\partial p_i} n_x + \frac{\partial y_j}{\partial p_i} n_y + \frac{\partial z_j}{\partial p_i} n_z, \quad i = 1, \ldots, n;\ j = 1, \ldots, m \tag{3.18}$$

The term $\partial f/\partial \boldsymbol{x}$, as discussed in Section 3.2.4, is called surface gradients, which represent the variation of the function of interest with respect to infinitesimal perturbations of the geometric shape in the local surface normal direction. The surface gradients at all grid nodes are computed by solving only once the adjoint equations, of which the computational cost is similar to that of one flow solution. The gradients $\partial f/\partial \boldsymbol{p}$ are then computed by the `SU2_DOT` module through a dot product operation between the geometric and surface gradients. Essentially, the low computational cost of gradient evaluation is attributed to the use of the adjoint method, of which a typical derivation is provided herein. Note that we assume the governing equations have been discretised in the following discussion.

Consider some quantities of interest, $\boldsymbol{f}$, that depend implicitly on the independent variables of interest, $\boldsymbol{x}$,

$$\boldsymbol{f} = \boldsymbol{F}\left(\boldsymbol{x}, \boldsymbol{y}\left(\boldsymbol{x}\right)\right) \tag{3.19}$$

where $\boldsymbol{y}$ represent the state variables. The relationship between $\boldsymbol{y}$ and $\boldsymbol{x}$ is defined by the solution of a set of residual equations:

$$\boldsymbol{r} = \boldsymbol{R}(\boldsymbol{x}, \boldsymbol{y}(\boldsymbol{x})) = 0 \tag{3.20}$$

The total derivative of $\boldsymbol{f}$ with respect to $\boldsymbol{x}$ can be calculated using the chain rule,

$$\frac{\mathrm{d}\boldsymbol{f}}{\mathrm{d}\boldsymbol{x}} = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{x}} + \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{y}} \frac{\mathrm{d}\boldsymbol{y}}{\mathrm{d}\boldsymbol{x}} \tag{3.21}$$

and similarly for the residual equations,

$$\frac{\mathrm{d}\boldsymbol{r}}{\mathrm{d}\boldsymbol{x}} = \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{x}} + \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{y}} \frac{\mathrm{d}\boldsymbol{y}}{\mathrm{d}\boldsymbol{x}} = 0 \tag{3.22}$$

The computation of the total derivative $d\boldsymbol{y}/d\boldsymbol{x}$ in Equations (3.21) and (3.22) has a much higher computational cost than the partial derivatives as it requires the solution of the residual equations, whereas the partial derivatives can be computed by the FDM. By rewriting Equation (3.22) we obtain,

$$\frac{\mathrm{d}\boldsymbol{y}}{\mathrm{d}\boldsymbol{x}} = -\left[\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{y}}\right]^{-1} \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{x}} \tag{3.23}$$

Substituting this expression into Equation (3.21) yields,

$$\frac{\mathrm{d}\boldsymbol{f}}{\mathrm{d}\boldsymbol{x}} = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{x}} \underbrace{- \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{y}} \overbrace{\left[\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{y}}\right]^{-1}}^{-\frac{d\boldsymbol{y}}{d\boldsymbol{x}}}}_{\boldsymbol{\Psi}^T} \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{x}} \tag{3.24}$$

Then the adjoint method is introduced to factorise the Jacobian matrix $\partial \boldsymbol{R}/\partial \boldsymbol{y}$ with $\partial \boldsymbol{F}/\partial \boldsymbol{y}$ by solving the following adjoint equations,

$$-\left[\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{y}}\right]^T \boldsymbol{\Psi} = \left[\frac{\partial \boldsymbol{F}}{\partial \boldsymbol{y}}\right]^T \tag{3.25}$$

where $\boldsymbol{\Psi}$ is the adjoint matrix (or adjoint vector for the case where $\boldsymbol{f}$ is a scalar). The solution of $\boldsymbol{\Psi}$ can be determined by solving the linear system and then be substituted into Equation (3.21) to compute the total derivative,

$$\frac{\mathrm{d}\boldsymbol{f}}{\mathrm{d}\boldsymbol{x}} = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{x}} + \boldsymbol{\Psi}^T \frac{\partial \boldsymbol{R}}{\partial \boldsymbol{x}} \tag{3.26}$$

The computational cost of evaluating gradients using the adjoint method is independent of the number of design variables, and instead proportional to the number of quantities of interest. This feature provides great efficiency in practical design problems, where the number of design variables, n_x, are far more than the number of functions of interest,

n_f. Figure 3.2 illustrates why the direct method is preferable when $n_f > n_x$ and the adjoint method is more efficient when $n_x > n_f$. The reader is referred to Reference [121] for more details of the adjoint approach.

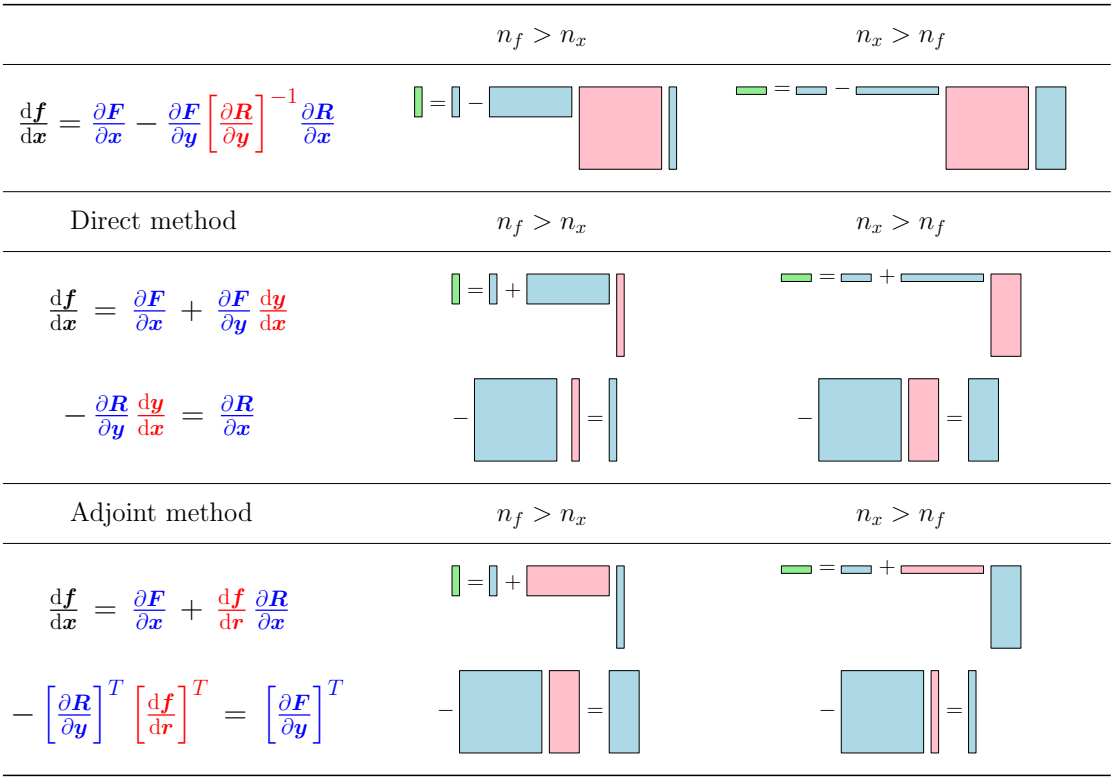


Figure 3.2: Block matrix diagrams illustrating the direct and adjoint methods. The matrices in blue contain partial derivatives and are relatively cheap to compute. The matrices in pink contain total derivatives that are computed by solving linear systems (the third and fifth rows). In both cases, it is assumed that $n_y \gg n_x$, n_f. [2]

3.2.6 Geometry parameterisation

The geometry parameterisation plays an important role in an ASO framework and must be robust and efficient enough for the design process. Two commonly-used parameterisation methods are implemented in SU2, which correspond to HHBF and FFD. Both approaches are employed in this work and are discussed as follows.

3.2.6.1 Hicks–Henne bump function

Hicks and Henne [134] introduced an analytical approach that takes a baseline geometry and adds a linear combination of bump functions to create a new shape. For 2D

problems, the parameterised geometry function can be expressed by:

$$\begin{cases} y = y_{\text{baseline}} + \sum_{i=1}^{n} b_i(x) \\ b_i(x) = a_i \left[\sin\left(\pi x^{\frac{\log 0.5}{\log h_i}} \right) \right]^{t_i}, \quad 0 \leq x \leq 1 \end{cases} \tag{3.27}$$

where n is the number of bump functions, $b_i(x)$ is the bump function (or basis function) proposed by Hicks and Henne, a_i represents the bump amplitude and acts as the weighting coefficient, h_i locates the maximum point of the bump, and t_i controls the width of the bump. By setting all of the coefficients a_i to zero, the baseline geometry is recovered.

By inspecting Equation (3.27), it is apparent that the bump function, $b_i(x)$, is defined by three parameters (i.e. a_i, h_i, and t_i). The bump amplitude coefficients, a_i, are treated as design variables and can be varied during optimisation, whereas the other two parameters, h_i and t_i, are predetermined and fixed in optimisation as this ensures the parameterisation is a linear function of the design variables.

With respect to the locations of bump peak, h_i, two distribution approaches are employed in this study: a) uniform distribution along the aerofoil chordwise direction:

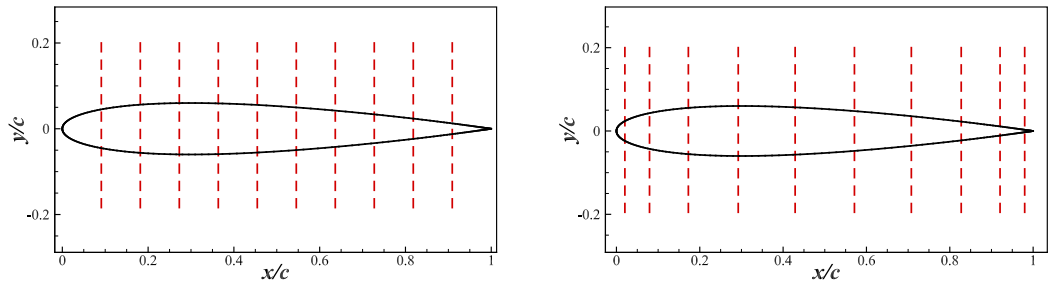
$$h_i = \frac{i}{n+1}, \quad i = 1, \ldots, n. \tag{3.28}$$

and b) uneven distribution described by a "one-minus-cosine" function:

$$h_i = \frac{1}{2}\left[1 - \cos\left(\frac{i\,\pi}{n+1}\right)\right], \quad i = 1, \ldots, n. \tag{3.29}$$

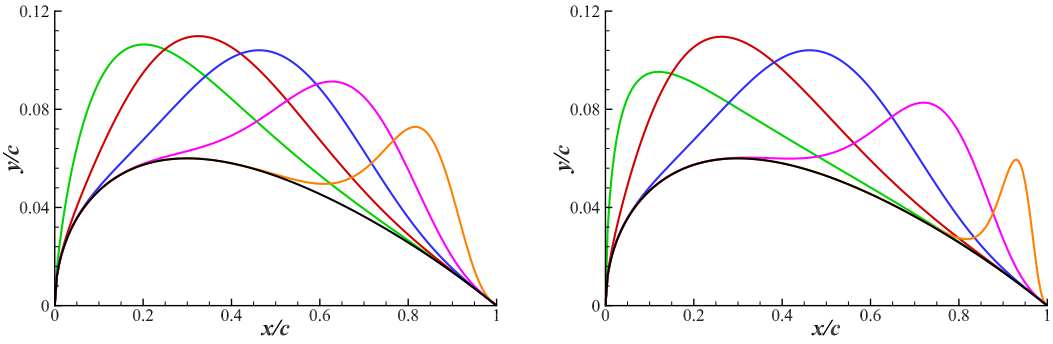
Figure 3.3 gives an illustration of the two distribution approaches for HHBFs. It is worth observing that the "one-minus-cosine" distribution provides a better clustering of design variables at the leading and trailing edge of the aerofoil when compared with the uniform distribution. A comparison of these two distribution approaches is further shown in Figure 3.4, where a set of bump functions are imposed individually on the upper surface of the NACA0012 aerofoil.

Regarding the bump width control parameter, t_i, a constant value is specified for all bump functions within SU2. In this study, in addition to the default setting $t = 3$, a range of integer values are defined, and their impact on the optimisation result is investigated. Figure 3.5 shows three sets of HHBFs with different settings of t. It is observed that the bump width narrows down as the value of t increases, suggesting that a relatively smaller value of t can provide more global shape control whereas a relatively larger value of t generates more local shape control. This fact is demonstrated more evidently in Figure 3.6, where two settings of parameter t are applied to a set of HHBFs that are imposed on the NACA0012 aerofoil separately.



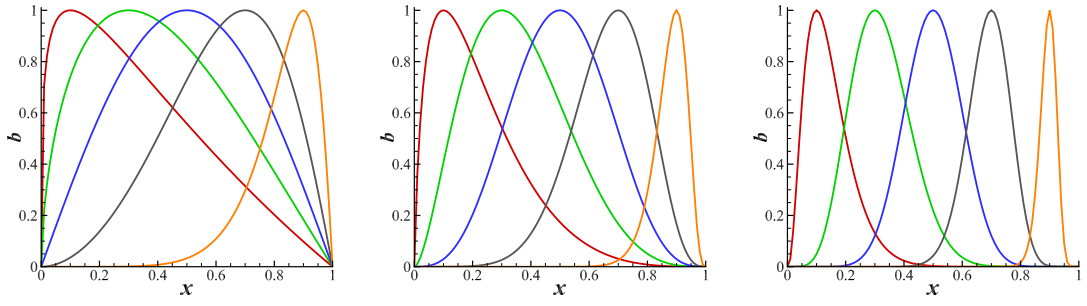
(a) Uniform distribution (b) "One-minus-cosine" distribution

Figure 3.3: Illustration of two distribution approaches for HHBFs ($n = 10$) on the NACA0012 aerofoil. Red dashed lines indicate locations of bump peak.



(a) Uniform distribution (b) "One-minus-cosine" distribution

Figure 3.4: Comparison of two distribution approaches for HHBF parameterisation of the NACA0012 aerofoil ($n = 5$, $a = 0.05$, and $t = 3$).

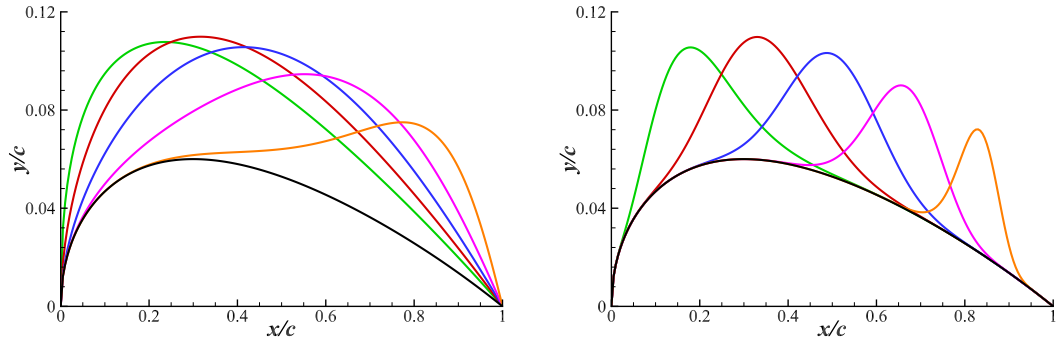


(a) $t = 1$ (b) $t = 3$ (default) (c) $t = 10$

Figure 3.5: Three sets of HHBFs ($n = 5$, $a = 1$, and uniformly distributed in the range of $h_i \in [0.1, 0.9]$) with different value settings of bump width control parameter.

3.2.6.2 Free-form deformation

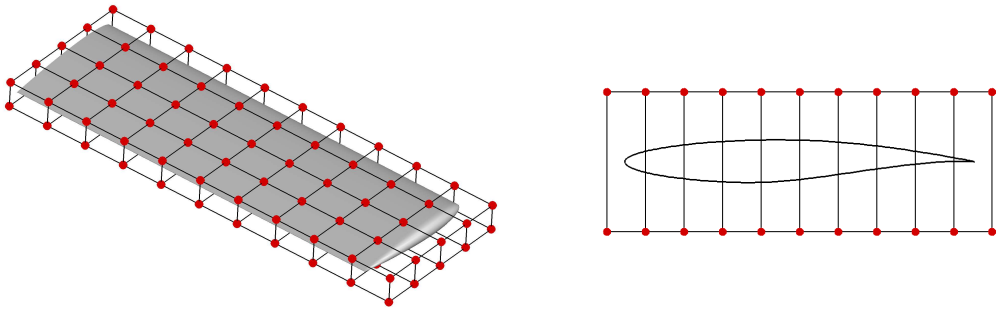
Free-form deformation (FFD), initially proposed by Sederberg and Parry [146], is used as the second parameterisation method. The basic FFD concept can be visualised as



(a) $t = 1$ (b) $t = 10$

Figure 3.6: Influence of bump width control parameter on HHBF parameterisation of the NACA0012 aerofoil ($n = 5$, $a = 0.05$, and uniform distribution).

embedding a flexible object inside a flexible volume and deforming both of them simultaneously by perturbing the lattice of the volume. The FFD control volume (or FFD box) usually has a topology of a cube when deforming 3D objects or a rectangle for 2D geometries, and thus can be parameterised as either a trivariate volume or a bivariate surface. In this study, Bézier curve is used as the FFD blending function. Figure 3.7 illustrates the FFD box encapsulating a 3D wing and the 2D RAE2822 aerofoil, where a lattice of control points are uniformly spaced on the surface of FFD box.



(a) 3D unswept wing (b) 2D RAE2822 aerofoil

Figure 3.7: View of FFD box enclosing the embedded object, including the control points shown as red spheres.

The parameterised Bézier volume can be described using the following equation:

$$\boldsymbol{X}(\xi, \eta, \zeta) = \sum_{i=0}^{l} \sum_{j=0}^{m} \sum_{k=0}^{n} \boldsymbol{P}_{i,j,k} B_i^l(\xi) B_j^m(\eta) B_k^n(\zeta) \tag{3.30}$$

where l, m, n are the degrees of FFD blending function; $\xi, \eta, \zeta \in [0, 1]$ are the parametric coordinates; $\boldsymbol{P}_{i,j,k}$ are the Cartesian coordinates of the control point (i, j, k); $\boldsymbol{X}$ are the corresponding Cartesian coordinates (x, y, z) for a given (ξ, η, ζ) in the Bézier volume; $B_i^l(\xi)$, $B_j^m(\eta)$, and $B_k^n(\zeta)$ are the Bernstein polynomials, which are expressed

as

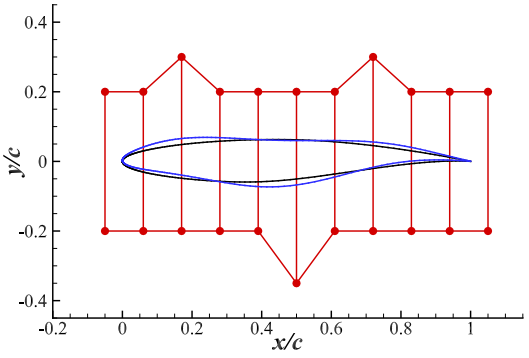
$$\begin{cases} B_i^l(\xi) = \dfrac{l!}{i!\,(l-i)!}\,\xi^i\,(1-\xi)^{l-i} \\ B_j^m(\eta) = \dfrac{m!}{j!\,(m-j)!}\,\eta^j\,(1-\eta)^{m-j} \\ B_k^n(\zeta) = \dfrac{n!}{k!\,(n-k)!}\,\zeta^k\,(1-\zeta)^{n-k} \end{cases} \tag{3.31}$$

The control points of the FFD box are defined as the design variables, the number of which depends on the degree of the chosen Bernstein polynomials. It should be noted that these control points are uniformly spaced in the FFD domain; otherwise the initial geometry of the embedded object can not be recovered with the original positions of the control points.

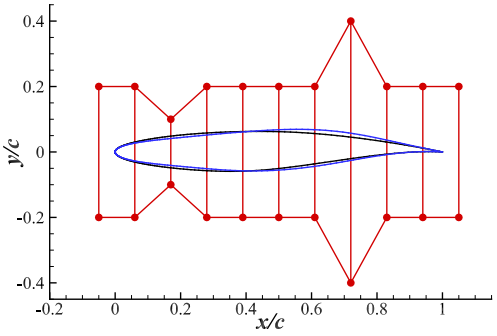
FFD is numerically executed in three steps. Firstly, for the embedded object, a mapping is performed from the physical space to the parametric space of the FFD box. The parametric coordinates (ξ, η, ζ) of each surface grid node are determined and remain unchanged during optimisation. Note that this mapping is evaluated only once. Secondly, the FFD control points are moved according to a given set of design variables, leading to the deformation of the FFD box as well as the perturbation of the embedded object. Thirdly, once the FFD box is deformed, the new Cartesian coordinates (x, y, z) of the embedded object in the physical space are algebraically computed using Equation (3.30).

Figure 3.8(a) shows an example of FFD shape control for the 2D RAE2822 aerofoil by randomly perturbing several control points. One useful and commonly-used feature of FFD technique is that multiple control points can be grouped together to perform specific motions and thus achieve desired shape deformation. Figure 3.8(b) and Figure 3.8(c) illustrate the parameterisation of aerofoil thickness and camber, respectively. In the case of 2D FFD thickness, a pair of control points are moved with the same amplitude but in opposite directions, whereas in the case of 2D FFD camber, the pair of two control points are forced to move with the same amplitude and in the same direction. In the scenario of 3D wing parameterisation, Figure 3.9(a) gives an example of changing wing twist at several spanwise stations. As shown in Figure 3.9(b), the four control points at the same spanwise location rotate simultaneously around the trailing edge of the sectional aerofoil to perform the twist motion. Correspondingly, the local incidence for the enclosed wing changes as well and the sectional aerofoil shape is kept the same.

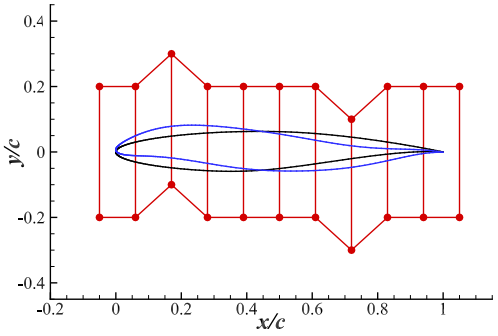
The setting of FFD box is implemented as a pre-processing step prior to performing aerodynamic optimisation. As the position settings of the FFD box are subject to the users' choice and may affect the final optimisation result, the impact of FFD box position on the optimisation performance is thus investigated in this study.



(a) FFD control point

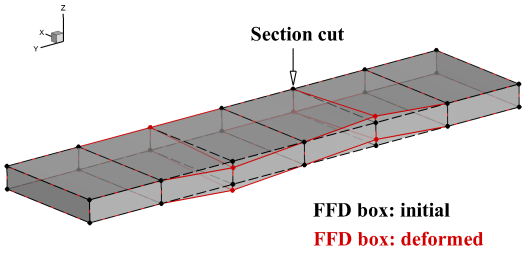


(b) FFD thickness

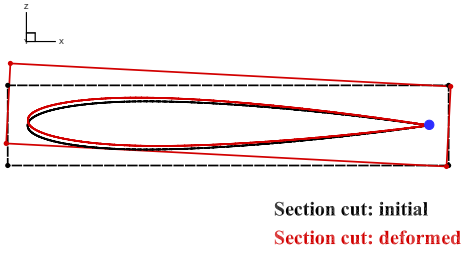


(c) FFD camber

Figure 3.8: Illustration of FFD shape control for the 2D RAE2822 aerofoil. The initial aerofoil geometry is in black, the deformed aerofoil in blue, and the perturbed control points in red.



(a) FFD box



(b) Section cut

Figure 3.9: Illustration of FFD twist shape control for the 3D wing. The blue dot indicates the rotation centre at the trailing edge.

3.2.7 Grid deformation

As the geometric shape is updated at each design step of the aerodynamic optimisation, the computational grid needs to be recomputed. Grid regeneration is often time-consuming and may introduce additional discretisation errors. Hence, the strategy of grid deformation with high efficiency and robustness is preferred.

The grid deformation approach employed in SU2 framework models the computational grid as an elastic body using the equations of linear elasticity [162], which govern small displacements, $\boldsymbol{u} = [u_1, u_2, u_3]^T$, of an elastic solid subject to body forces and surface tractions. Using the summation convention, the linear elasticity equations can be written as

$$\nabla \cdot \boldsymbol{\sigma} = \boldsymbol{f} \quad \text{in}\,\Omega \tag{3.32}$$

where $\boldsymbol{f}$ is the body force, Ω is the computational domain, and $\boldsymbol{\sigma}$ is the stress tensor, given in terms of the strain tensor $\boldsymbol{\epsilon}$ by the constitutive relation

$$\boldsymbol{\sigma} = \lambda\, Tr(\boldsymbol{\epsilon})\, \boldsymbol{I} + 2\mu\boldsymbol{\epsilon} \tag{3.33}$$

where Tr is the trace, λ and μ are the Lamé constants, which are expressed in terms of Young's modulus E and Poisson's ratio ν as

$$\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)}, \quad \mu = \frac{E}{2(1+\nu)} \tag{3.34}$$

$E > 0$ can be thought of as the stiffness of the material, where a large value of E indicates rigidity. Poisson's ratio ν is a measure of how much the material shrinks in the lateral direction as it extends in the axial direction. The strain tensor $\boldsymbol{\epsilon}$ is evaluated using the linear kinematic law

$$\boldsymbol{\epsilon} = \frac{1}{2}\left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T\right) \tag{3.35}$$

An advantage of applying these equations to grid deformation is that various features required in practice can be readily and simply realised in a manner consistent with the modelling of the grid as an elastic body. In SU2, the Young's modulus of elasticity in each grid cell may be treated in three ways: a) inversely proportional to the cell volume; b) inversely proportional to the distance from the wall; and c) being a constant value. In this study, the first option is adopted. With this strategy, near-body cells undergo nearly rigid motion, whereas larger cells further from the body adapt to the deformation of the surface. Moreover, regarding viscous flow, this strategy can avoid generating negative cell volumes in the deformation of boundary layer grid, where there are cells with large aspect ratio. Therefore, the grid quality can be preserved in near-wall area and regions of high resolution with the selected implementation. More details on the grid deformation method can be found in Reference [162].

3.3 FALCon

The hybrid aerodynamic solver, FALCon, was developed within a national project aimed at the development, implementation and demonstration of a multi-fidelity aerodynamic

solver for rapid load calculations. The multi-fidelity solver mainly consists of three building blocks: a) a linear VLM solver; b) an efficient ISW solver; and c) a hybrid coupling algorithm. The 3D effects on finite wings are modelled by the VLM solver, which is corrected by an ISW RANS/URANS solver to capture sectional viscous flow effects. The coupling algorithm combines two fidelity levels in flow predictions to obtain fast estimates of aerodynamic loads in subsonic and transonic regimes for both steady and unsteady problems. The hybrid solver is also integrated with the open-source optimisation package pyOpt to formulate an ASO framework.

3.3.1 Vortex lattice method

The implementation of the VLM closely follows the description of Katz and Plotkin [126]. As shown in Figure 3.10, the thin wing planform is divided into a lattice of panels containing vortex ring elements. The leading segment of the vortex ring is placed on the panel's quarter-chord line and the collocation point is at the centre of the three-quarter-chord line, where the normal vector is defined as well.

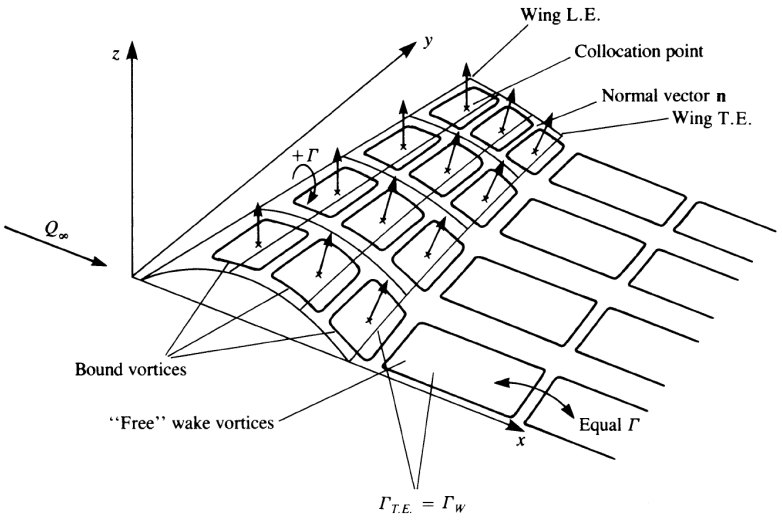


Figure 3.10: Vortex ring model for a thin lifting surface. (source: [126])

The linear system of equations is solved for each collocation point:

$$\boldsymbol{A} \cdot \boldsymbol{\Gamma} = \boldsymbol{R}(\boldsymbol{\alpha}) \tag{3.36}$$

where $\boldsymbol{A}$ is the aerodynamic influence coefficient (AIC) matrix representing the mutual influence of VLM panels, $\boldsymbol{\Gamma}$ is the vector of unknown circulation intensity of wing bounded VLM panels, $\boldsymbol{R}(\boldsymbol{\alpha})$ is the right-hand-side (RHS) vector containing non-circulatory velocity contributions acting on each collocation point, and $\boldsymbol{\alpha}$ is the vector of angle of attack at each spanwise VLM panel. For steady-state calculations, the AIC matrix $\boldsymbol{A}$ is evaluated only once and remains constant during the coupling procedure as discussed in Section 3.3.3. It is also worth noting that, in VLM formulation, the vector

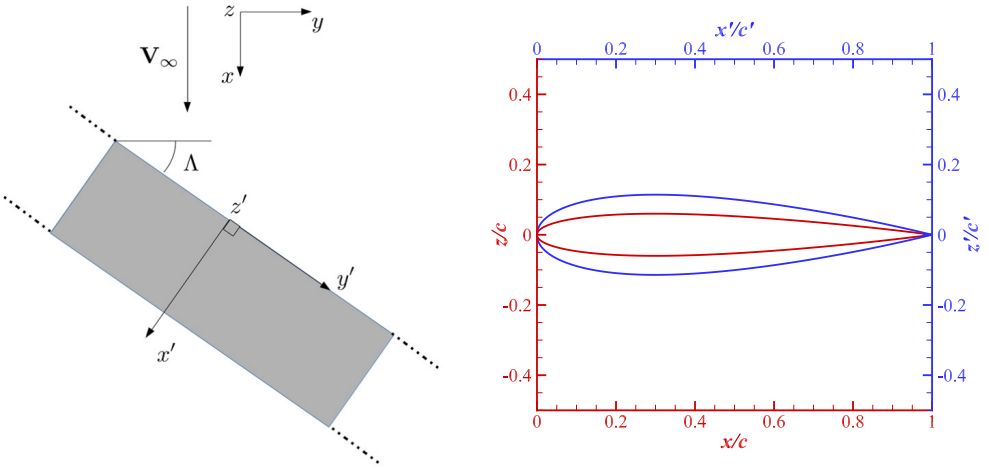
of angle of attack $\boldsymbol{\alpha}$ generally remains unchanged but requires modification in our work. As described in Section 3.3.3, $\boldsymbol{\alpha}$ is corrected at each iteration of the coupling loop to account for the nonlinear effects introduced by the ISW solver.

Both steady and unsteady VLM solvers have been developed in the author's group, and only the steady-state VLM is employed in this work. The reader is referred to Reference [1] for more details of the VLM formulation.

The VLM solves inviscid, incompressible and attached flows, limiting its validity to a very narrow flow regime. To capture the nonlinear effects existing in viscous, compressible flows, the VLM needs to be corrected with sectional viscous data.

3.3.2 Infinite swept wing solver

The ISW aerodynamic model is tailored for the analysis of lifting surfaces where cross-flow effects are evident due to sweep [236, 237]. The ISW model assumes a wing of infinite span with a moderate wing sweep angle, Λ. A schematic of an ISW is illustrated in Figure 3.11(a). Two typical frames of reference (FoRs) are shown. Compared to the global FoR with axes (x, y, z), the body-attached FoR with axes (x', y', z') provides a more suitable choice for the ISW model. The x' axis is perpendicular to the quarter-chord axis of the local chord, pointing at the trailing-edge, and the y' axis is parallel to the quarter-chord axis of the local chord, pointing at the wing tip. Note that the chord lengths within the two FoRs are not the same. To keep the chord length consistent for both FoRs, the thickness of the sectional aerofoil in the local FoR (x'-z' plane) needs to be scaled by the factor of $1/\cos\Lambda$, as shown in Figure 3.11(b).



(a) Schematic of an ISW model [240] (b) NACA0012 aerofoil in two FoRs ($\Lambda = 45$ deg)

Figure 3.11: Illustration of infinite swept wing and sectional aerofoil in global and local frames of reference.

In the body-attached FoR, we assume that the cross-flow is fully developed. Thus, the flow has the statistical homogeneity in the y' direction, leading to $\partial(\cdot)/\partial y' = 0$. The governing 3D Navier–Stokes equations are therefore simplified and written as

$$\frac{\partial}{\partial x'}(\rho u') + \frac{\partial}{\partial z'}(\rho w') = 0 \tag{3.37a}$$

$$\frac{\partial}{\partial x'}(\rho u'u') + \frac{\partial}{\partial z'}(\rho u'w') = -\frac{\partial p}{\partial x'} + \frac{\partial \tau'_{xx}}{\partial x'} + \frac{\partial \tau'_{xz}}{\partial z'} \tag{3.37b}$$

$$\frac{\partial}{\partial x'}(\rho v'u') + \frac{\partial}{\partial z'}(\rho v'w') = \frac{\partial \tau'_{yx}}{\partial x'} + \frac{\partial \tau'_{yz}}{\partial z'} \tag{3.37c}$$

$$\frac{\partial}{\partial x'}(\rho w'u') + \frac{\partial}{\partial z'}(\rho w'w') = -\frac{\partial p}{\partial z'} + \frac{\partial \tau'_{zx}}{\partial x'} + \frac{\partial \tau'_{zz}}{\partial z'} \tag{3.37d}$$

$$\frac{\partial}{\partial x'}(\rho h u') + \frac{\partial}{\partial z'}(\rho h w') = \frac{\partial}{\partial x'}(u'\tau'_{xx} + v'\tau'_{xy} + w'\tau'_{xz}) + \frac{\partial}{\partial z'}(u'\tau'_{zx} + v'\tau'_{zy} + w'\tau'_{zz}) \tag{3.37e}$$

Note that dependence on time is neglected, and heat flux components are removed herein for conciseness. Equations (3.37a), (3.37b) and (3.37d) are independent from the cross-flow component, v', and are equivalent to the 2D Navier–Stokes equations. Equation (3.37c) represents a transport equation for the cross-flow component v', and can be solved at each iteration once the flow field variables u' and w' are known. The coupling between cross-flow and the other variables is provided by the energy conservation equation, Equation (3.37e).

Following the same method, the equations for turbulent viscosity within a RANS scheme can also be derived for the ISW problem. Additionally, the solution of the governing equations requires appropriate boundary conditions to be defined. These relevant equations and derivations are not reported herein for brevity, and the reader is referred to Reference [240] for more details.

The existing state-of-the-art approach to model the ISW involves a one-cell width 3D stencil and can be implemented in two ways, see Figure 3.12, with periodic boundary conditions set on both planes. These two approaches are equivalent in terms of computational cost and can generate identical results. The proposed novel approach, on the other hand, relies purely on a 2D stencil and solves the ISW problem with the computational cost equivalent to that of a 2D analysis. The proposed methodology has been implemented within the DLR-Tau flow solver, where it is referred to as the 2.5D+ solver [250]. The numerical solution obtained from the 2.5D+ solver has the same properties as that obtained using either the "sheared" or "beta" approach. The reader is referred to Reference [240] for more details of the efficient ISW solver and its applications to both steady and unsteady compressible flows around a variety of geometries.

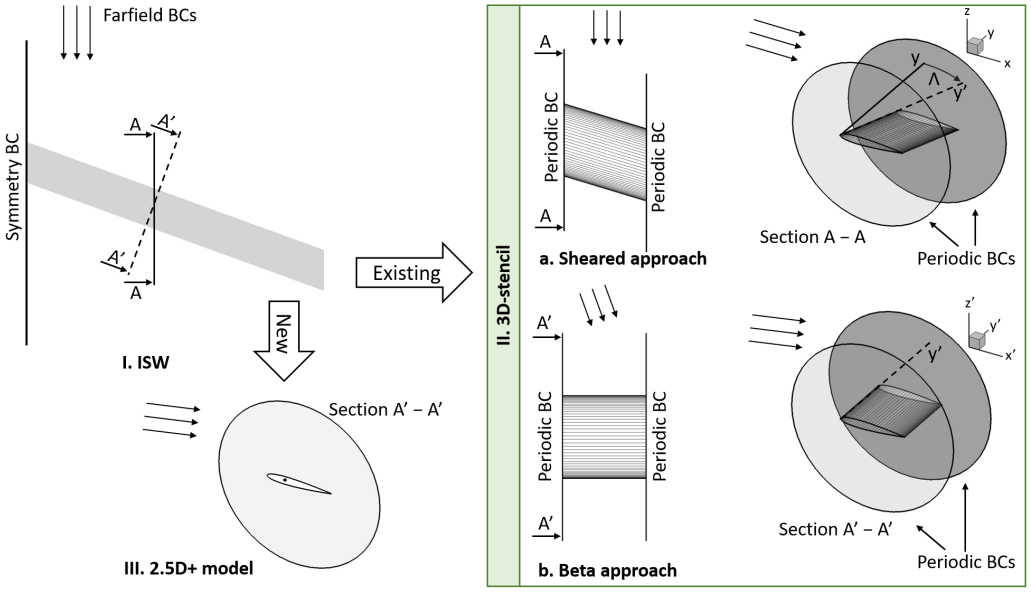


Figure 3.12: Summary of approaches to model the ISW problem. (Source: [240])

3.3.3 Coupling algorithm

The steady-state hybrid solver is based on the α-based coupling algorithm [50, 233, 251]. This algorithm corrects the local angle of attack, α, at every wing bounded VLM panel along the wing span using nonlinear sectional data. The procedure of the coupling method is described in Algorithm 2.

Algorithm 2 α-based coupling method for steady-state hybrid solver.

1: Initialise: $\alpha(j) = \alpha_\infty$ and $\Delta\alpha(j) = 0$ for all spanwise sections $j = 1, 2, \ldots, N_y$
2: Run linear VLM solver to calculate the sectional inviscid lift coefficient:
$\boldsymbol{\alpha} \Rightarrow C_{l,\,\mathrm{inv}}(j)$
3: **for** Every spanwise section j **do**
4: Calculate the effective angle of attack:
$$\alpha_e(j) = \alpha_\infty - \alpha_i(j) = \frac{C_{l,\,\mathrm{inv}}(j)}{2\pi} - \Delta\alpha(j)$$
5: Obtain the sectional viscous lift coefficient from 2.5D+ database at the effective angle of attack:
$\alpha_e(j) \Rightarrow C_{l,\,\mathrm{visc}}(j)$
6: Calculate the angle of attack correction:
$$\Delta\alpha(j) = \Delta\alpha(j) + \nu\,\frac{C_{l,\,\mathrm{visc}}(j) - C_{l,\,\mathrm{inv}}(j)}{2\pi}$$
7: Update the local angle of attack:
$\alpha(j) = \alpha_\infty + \Delta\alpha(j)$
8: **end for**
9: Repeat Steps 2-8 until $|C_{l,\,\mathrm{visc}}(j) - C_{l,\,\mathrm{inv}}(j)| < \epsilon$ for every spanwise section j

Figure 3.13 depicts the coupling process to help understand the coupling procedure. For a typical wing, the sectional nonlinear database is either obtained from wind tunnel experiments or pre-computed using the 2.5D+ solver at span locations that are carefully

selected, such as the kink. It is worth noting that the database is computed once and for all. The initialisation includes the estimation of the AIC matrix, $\boldsymbol{A}$. In the iteration loop, the RHS non-circulatory velocity vector in Equation (3.36), $\boldsymbol{R}(\boldsymbol{\alpha})$, is updated at each iteration according to the correction step, $\Delta\alpha$. Once the inviscid lift coefficient has been obtained, the effective angle of attack, α_e, is calculated from the following formula

$$\begin{aligned} C_{l,\,\mathrm{inv}} &= 2\pi\,(\alpha_\infty - \alpha_i - \alpha_0 + \Delta\alpha) \\ &= 2\pi\,(\alpha_e - \alpha_0 + \Delta\alpha) \end{aligned} \tag{3.38}$$

where α_∞ is the freestream angle of attack at farfield, α_i is the induced angle of attack calculated from the VLM solver, and α_0 is the zero-lift angle of attack.

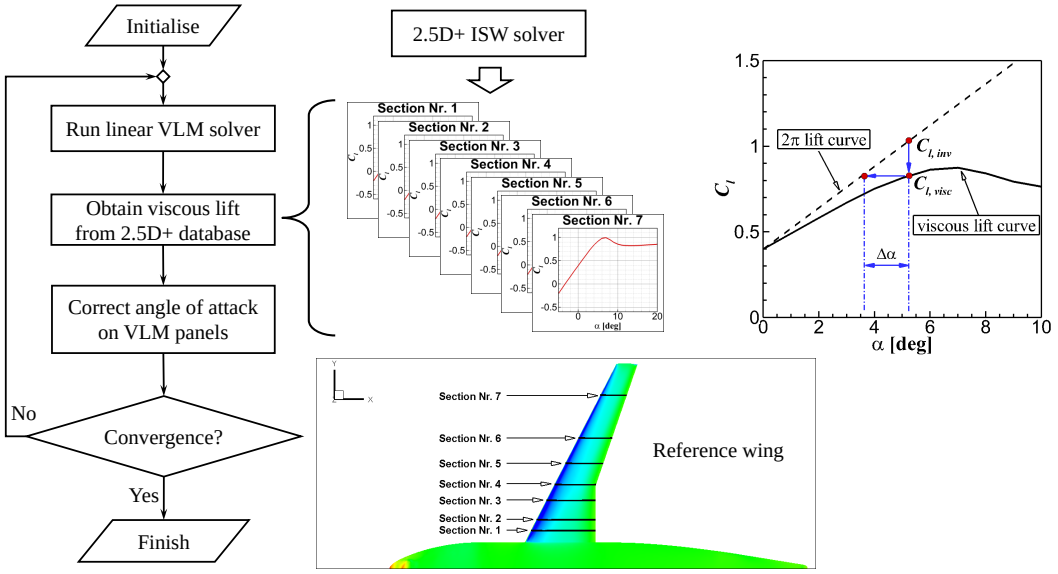


Figure 3.13: Steady-state hybrid coupling algorithm.

Once the effective angle of attack is known at discrete VLM panels, the sectional viscous lift coefficient is interpolated from the pre-computed database, and the viscous lift distribution along the span can be obtained using linear or higher order interpolation methods. Once both the inviscid and viscous lift coefficients are obtained, the angle of attack correction is calculated for every spanwise panel. An under-relaxation factor, ν, is used to help stabilise the iteration, especially at high angles of attack close to stall where nonlinear behaviour becomes dominant. The relaxation factor typically has a default value of 0.5. As there is only one physical model for the coupling system, the calculation steps need to be iterated to obtain a converged solution. The iteration loop proceeds until the gap between the inviscid and viscous sectional lift for every VLM panel is within a prescribed tolerance ϵ.

The unsteady hybrid coupling approach is based on the steady-state coupling algorithm and extends it with an outer time-marching loop. The reader is referred to References [1,

241] for more details of the unsteady coupling methodology and validation of the hybrid solver.

3.3.4 Forces calculation

The aerodynamic forces are obtained as a post-processing operation once the hybrid coupling iterations are completed. The total lift is calculated by summing the individual panel contributions using the Kutta–Joukowski theorem:

$$L = \sum_{k=1}^{N} \rho V_\infty \Gamma_k \Delta y_k \tag{3.39}$$

where N is the total number of vortex ring elements, V_∞ is the freestream flow velocity, Γ_k is the circulation around an incremental vortex, and Δy_k is the panel bound vortex projection normal to the freestream. The total drag of a wing is decomposed into two components, induced drag and viscous drag.

$$D = D_{\text{ind}} + D_{\text{visc}} \tag{3.40}$$

Generally there are two methods for calculating the induced drag of a wing: near field analysis and far field analysis, alternatively surface integration method and wake integration method. It should be noted that the induced drag is included in the pressure drag in a 3D wing drag decomposition using near field analysis. However, in 2.5D+ drag analysis, the sectional pressure drag does not include the induced drag. Therefore, the near field approach can not be used for the hybrid solver. In this work, the induced drag is evaluated in the VLM code through the Trefftz plane analysis [126], which is a type of far field analysis method. The induced drag can be obtained by integrating the kinetic energy on the Trefftz plane (see Figure 3.14), which is far behind the trailing edge and normal to the freestream. Moreover, because of the symmetry of the induced velocity above and under the vortex sheet, the surface integral on the Trefftz plane can be reduced to a single spanwise line integral over the wake trailing vortices, which can be further discretised as follows

$$\begin{aligned} D_{\text{ind}} &= -\frac{\rho}{2} \int_{-b_{\text{w}}/2}^{b_{\text{w}}/2} \Gamma(y) w dy \\ &= -\frac{\rho}{2} \sum_{k=1}^{N_W} \Gamma_k \, w_{\text{ind}_k} \Delta y_k \end{aligned} \tag{3.41}$$

where b_{w} is the local wake span, N_W is the number of trailing vortex lines, and w_{ind} is the wake-induced downwash at the trailing vortex line.

The viscous drag, which includes the friction and pressure drag, is calculated from the pre-computed sectional database. For each spanwise section, the viscous drag coefficient,

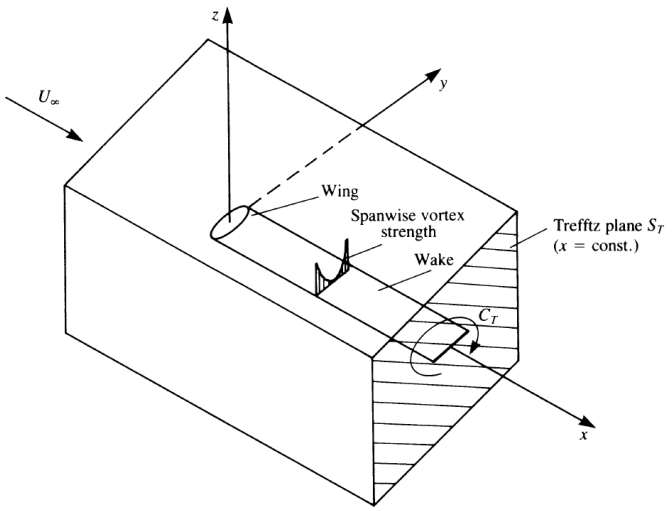


Figure 3.14: Trefftz plane used for the calculation of induced drag [126].

$C_{d,\,\mathrm{visc}}$, is interpolated from the available tabulated dataset at the corresponding effective angle of attack. Afterwards, the total viscous drag is obtained by applying a spanwise integration

$$D_{\mathrm{visc}} = \frac{1}{2}\,\rho\,{V_\infty}^2 \int_{-b/2}^{b/2} C_{d,\,\mathrm{visc}}\,c\,dy \tag{3.42}$$

where b is the wing span and c is the local chord length.

3.3.5 Optimisation framework

An aerodynamic optimisation framework, as shown in Figure 3.15, is formulated to tackle wing twist optimisation problems. A more detailed description is given in Algorithm 3. The multi-fidelity FALCon solver is used to provide aerodynamic forces for evaluation of the objective function and constraints. As the twist design variables are linked to the local angle of attack for VLM panels, there is no need for geometry parameterisation method. Moreover, the angle of attack correction $\Delta\alpha$ and the twist angle θ of the VLM panels are artificial concepts in the FALCon solver, and only have impact on the VLM solution of Equation (3.36); during the iterative process the wing geometry is not perturbed and the VLM grid remains unchanged. Therefore the grid deformation technique is also not required. In this study, the SLSQP algorithm is employed as the gradient-based optimiser, and is easily accessible through the common interface within the open-source pyOpt package. As the multi-fidelity aerodynamic solver is computationally efficient, the gradients are computed using FDM.

It is worth noting that, for every design iteration of the optimisation loop, the design variables of geometric twist need to be mapped onto the twist angles of wing-bound panels, as shown in Figure 3.16. This can be done using interpolation or other fitting techniques. In this study, a simple linear interpolation is used unless otherwise stated.

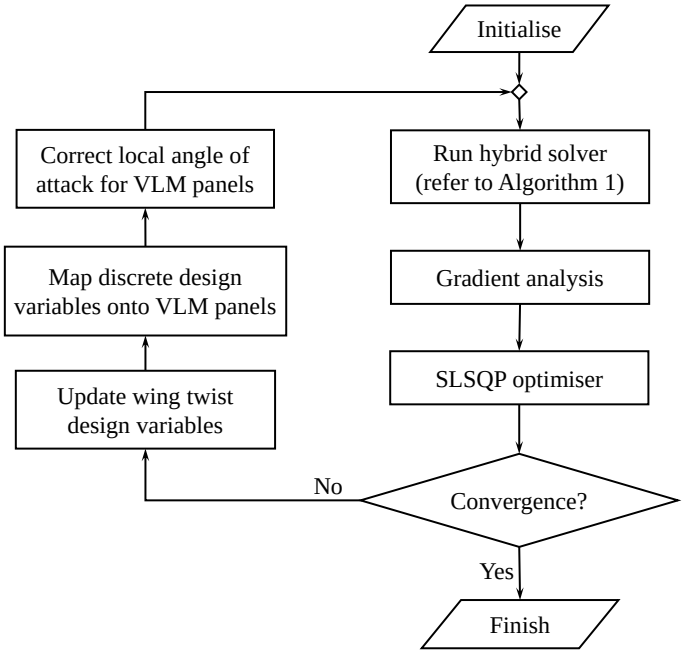


Figure 3.15: Wing twist optimisation framework based on hybrid solver.

Algorithm 3 Wing twist optimisation using hybrid solver.

1: Initialise: $\alpha(j) = \alpha_\infty$ and $\Delta\alpha(j) = 0$ for all spanwise sections $j = 1, 2, \ldots, N_y$
2: Run hybrid solver for aerodynamic analysis (Step 2-8 in Algorithm 2)
3: Compute gradients via FDM
4: Update wing twist design variables $\gamma(i)$ where $i = 1, 2, \ldots, N_{\text{dv}}$
5: Map twist design variables onto VLM spanwise sections:
$\gamma(i) \Rightarrow \theta(j)$
6: Update the local angle of attack:
$\alpha(j) = \alpha_\infty + \Delta\alpha(j) + \theta(j)$
7: Repeat Steps 2-6 until convergence criteria are met

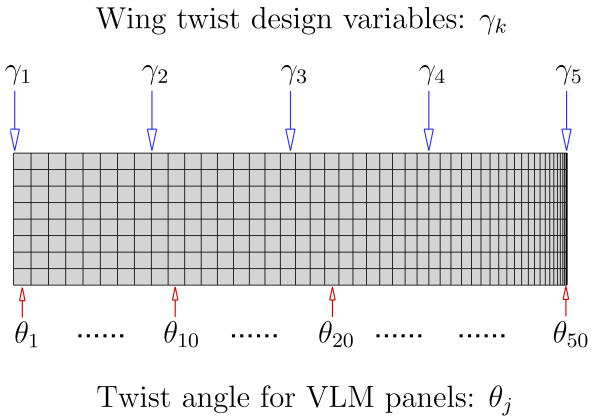


Figure 3.16: Schematic of twist mapping on VLM panels.

3.4 Chapter summary

This chapter details the methodology for two aerodynamic solvers, namely SU2 and FALCon, and the associated optimisation frameworks that are employed in this research. The numerical methodologies and computational techniques for SU2 are provided first, including the governing equations for flow and adjoint solver, turbulence modelling, gradient evaluation, geometry parameterisation and grid deformation. The multi-fidelity aerodynamic solver FALCon consists of three building blocks: a VLM solver, an ISW solver and a coupling algorithm. Concerning the FALCon solver, the forces calculation methods and the optimisation structure are also provided.

Chapter 4

Two-dimensional NACA0012 Aerofoil Optimisation

4.1 Introduction

The wing is the primary lifting surface of an aircraft, and it is built with aerofoil-shaped cross sections. As such, aerofoil design and optimisation remains a critical area of research for aerodynamic optimisation. Of the benchmark problems defined by AIAA ADODG, the first case specifies the drag minimisation of a NACA0012 aerofoil in a constrained design space. This optimisation case has been widely exercised by a number of researchers. Various optimised shapes were obtained with very different parameterisation types, flow solvers, optimisation algorithms, and number of design variables. For example, methods of choice for geometry parameterisation were FFD [57], Bézier curves [131], B-splines [57, 131, 143], RBFs [143, 252], CST [253], and SVD [125], among others. Masters *et al.* [226] selected six shape parameterisation techniques and investigated their impact on the optimisation results. The choice of flow solver [227] or optimisation algorithm [125] has also been shown to have influence on the optimal solution. In addition, Méheut *et al.* [254] and Destarac *et al.* [255] carried out a cross-validation of different optimal shapes obtained from several participants. The assessment process is identical for all the optimised aerofoils.

It should be noted that the available methodologies for each constituent of an ASO framework have their own merits and drawbacks. Thus, the most appropriate set of methods for a specific optimisation case may not be suitable for another case. More importantly, for a given ASO framework, there are a number of parameters that affect the final optimal solution. It requires fine-tuning of those parameters in order to achieve the best optimisation performance for a specific case. To date, however, many parameters have been largely neglected in practice, and their effects on the optimal solution are therefore unknown.

In this chapter, gradient-based optimisations are performed using SU2 for the benchmark NACA0012 case. Several key parameters are carefully selected, particularly in geometry parameterisation, which formulates the design space and provides design variables as input for optimisation. Instead of using the default (or common) value, a range of settings are specified and applied. The primary aim is to investigate the sensitivity of optimal solution to these parameters and gain the best practice from the sensitivity assessment, which can provide suggestions of parameter settings for future aerodynamic design and optimisation.

This chapter is organised in the following manner. The optimisation problem is firstly described in Section 4.2. The computational grid and optimisation set-up are then given in Section 4.3 and 4.4, respectively. Section 4.5 provides a detailed sensitivity assessment. The optimisation results are presented in Section 4.6. Section 4.7 finally summarises the key findings in this chapter.

4.2 Optimisation problem

The optimisation problem is the drag minimisation of a modified NACA0012 aerofoil in inviscid, transonic flow. The freestream Mach number, M, is 0.85, and the angle of attack, α, is fixed at 0 degree. The thickness is constrained to be greater than or equal to that of the initial aerofoil along the entire chord. The optimisation problem is written as

$$\begin{aligned} \text{Minimise:} \quad & C_d \\ \text{w.r.t:} \quad & y \\ \text{Subject to:} \quad & y\left(x\right) \geq y_{\text{baseline}}\left(x\right), \forall\, x \in [0,\, 1] \end{aligned}$$

where C_d is the drag coefficient, x is the coordinate along the aerofoil chord, and y is the coordinate describing the thickness of the symmetric aerofoil.

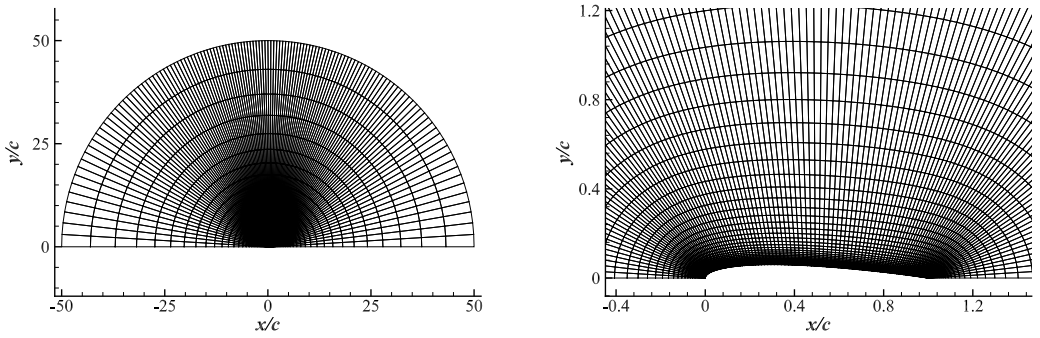
The optimisation case is based on the work by Vassberg *et al.* [256]. Differing from the reference paper, a modified NACA0012 aerofoil with zero-thickness trailing edge is defined as

$$y = \pm 0.6(0.2969\sqrt{x} - 0.1260x - 0.3516x^2 + 0.2843x^3 - 0.1036x^4) \tag{4.1}$$

where $x \in [0,\, 1]$. The zero-thickness trailing edge is achieved through a modification of the x^4 coefficient.

4.3 Computational grid

As the flow is symmetric around the aerofoil at the prescribed flow conditions, only the upper aerofoil surface is modelled unless otherwise stated. The structured O-grid was generated using *Pointwise* grid generator and is shown in Figure 4.1. The farfield boundary is located at a distance of 50 chord lengths from the aerofoil. The grid is clustered towards the leading and trailing edge in order to accurately represent the aerofoil geometry. The coarse grid has 129 points in the circumferential direction and 65 in the normal direction. To establish grid convergence, four levels of grid are considered, and the key parameters are summarised in Table 4.1.



(a) Far view

(b) Near view

Figure 4.1: Computational domain and coarse grid (129 × 65) for the NACA0012 aerofoil.

Table 4.1: Grid parameters for the NACA0012 aerofoil grid convergence study.

Grid Level	Grid Size	Grid Points	LE/TE Spacing
Coarse	129 × 65	8,385	$1.0 \cdot 10^{-3}$
Medium	257 × 129	33,153	$5.0 \cdot 10^{-4}$
Fine	513 × 257	131,841	$2.5 \cdot 10^{-4}$
Superfine	1025 × 513	525,825	$1.25 \cdot 10^{-4}$

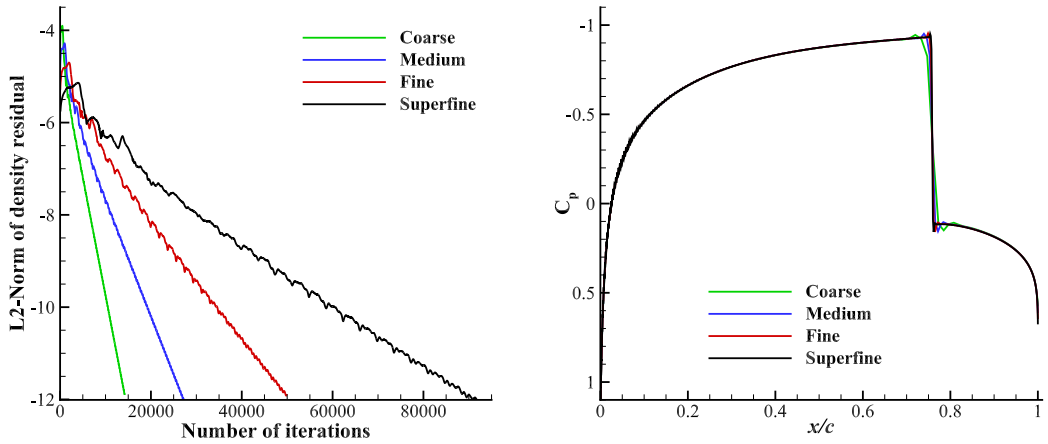
An inviscid flow analysis was carried out on the baseline aerofoil, and the drag results are listed in Table 4.2. Note that the drag values correspond to the complete aerofoil geometry and 1 drag count is equal to a C_d of $1 \cdot 10^{-4}$. The convergence history and the resulting pressure distribution are also shown in Figure 4.2. The coarse grid was found adequate to provide flow solutions with sufficient accuracy, and was thus used for optimisation.

4.4 Optimisation set-up

The open-source code SU2 is used to perform flow analysis and gradient-based optimisation for the NACA0012 case. A summary of the computational methods and numerical

Table 4.2: Drag results for the NACA0012 aerofoil grid convergence study ($M = 0.85$, $\alpha = 0$ deg).

Grid Level	C_d (Counts)
Coarse	468.0
Medium	469.0
Fine	469.3
Superfine	469.4



(a) Convergence history

(b) Pressure coefficient distribution

Figure 4.2: Grid convergence study for the baseline NACA0012 aerofoil ($M = 0.85$, $\alpha = 0$ deg).

settings is given in Table 4.3.

Table 4.3: Computational methods and numerical settings for the NACA0012 optimisation.

Parameter	Methodology or Value
Geometry parameterisation	HHBF/FFD
Grid deformation	Linear elasticity equations
Flow governing equations	Euler
Spatial discretisation	2nd order JST
Time discretisation	Euler implicit
CFL number	5
Optimiser	SLSQP
Gradient evaluation	Continuous adjoint
Scaling factor for objective function	0.001
Maximum number of iterations	100
Tolerance of KKT conditions	$1 \cdot 10^{-6}$

Two types of shape parametrisation, HHBF and FFD, are employed for the NACA0012 case. Prior to optimisation, a number of parameter values need to be determined. The

HHBF has two coefficients that need to be specified. For locations of bump peak, h_i, two distribution approaches are employed. For bump width control coefficient, t, a range of integer values are used. In terms of FFD, various settings of FFD box position are defined. Additionally, the number of design variables, N_{dv}, is varied within a reasonable range to perform the dimensionality study. The main numerical parameters used in this case are summarised in Table 4.4.

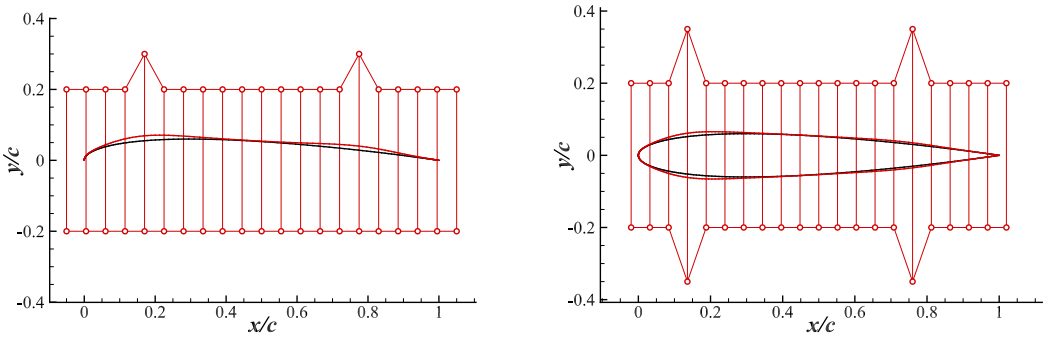
Table 4.4: Numerical parameters for two geometry parameterisation methods; the values in square brackets indicate the range for a specific parameter.

Parameter	Methodology or Value
HHBF	
Distribution approach	Uniform/One-minus-cosine
Bump width control parameter	[1, 15]
Number of design variables	[5, 40]
FFD	
FFD box: upper/lower	[±0.0601, ±0.3]
FFD box: left	[-0.01, -0.0001]
FFD box: right	[1.0001, 1.01]
Number of design variables	[5, 40]

For the NACA0012 aerofoil, an example of the commonly-used FFD control point parameterisation is illustrated in Figure 4.3(a). The control points on the upper surface of FFD box are specified as the design variables, whereas those on the lower surface are held fixed during optimisation because only the upper half aerofoil geometry is used. Due to the symmetric characteristic of the flowfield, the FFD thickness approach is also employed for this optimisation problem, which is illustrated in Figure 4.3(b). It should be noted that, for FFD thickness approach, the whole aerofoil geometry as well as corresponding computational grid are used. The thickness at specific chordwise location of the aerofoil can be modified by manipulating a pair of control points, which move with the same magnitude but in opposite directions. For both FFD methods, the thickness constraint in this optimisation problem can be satisfied implicitly by allowing the control points to move only in the outward direction as shown in Figure 4.3.

4.5 Sensitivity assessment

This section contains results obtained from the sensitivity assessment for the NACA0012 optimisation case. This work primarily focuses on investigating the sensitivity of the optimal solution to a series of model parameters, with the purpose of finding a suite of parameter values that can produce the best optimisation result.



(a) FFD control point method

(b) FFD thickness method

Figure 4.3: Two FFD parameterisation approaches for the NACA0012 aerofoil optimisation case. Baseline aerofoil is shown in black colour and deformed aerofoil in red.

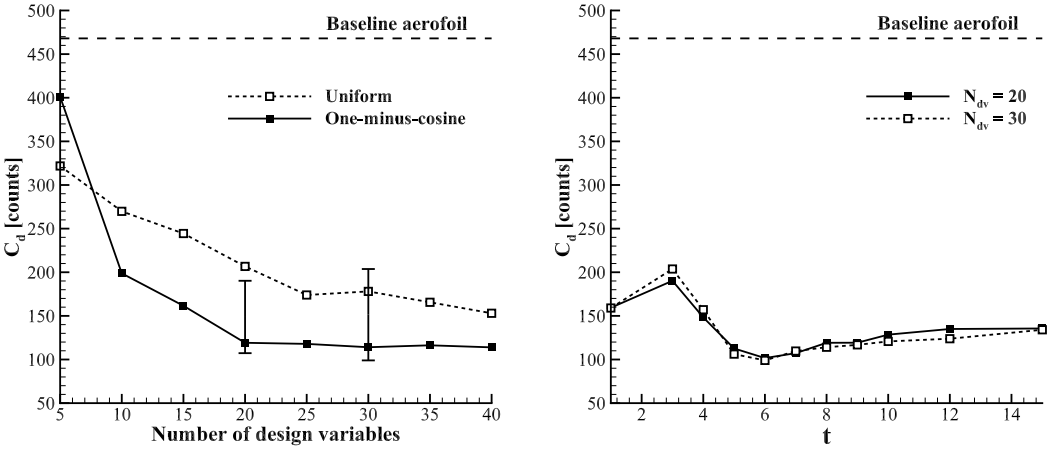
4.5.1 Impact of parameter settings in HHBF

The design variables (i.e. bump amplitude coefficients a_i) are restricted to have non-negative values in this case. Consequently, the deformed aerofoil is guaranteed to have a larger thickness along the chord than the initial aerofoil. Therefore, the constrained optimisation problem is transformed into an unconstrained one by satisfying the thickness constraint implicitly. Optimisations were carried out using the parameter settings as shown in Table 4.4. A 3D carpet plot of drag coefficient versus N_{dv} and t for both uniform and "one-minus-cosine" distributions can be obtained. Some sections are extracted from the 3D plot, and the corresponding optimisation results are shown and discussed as follows.

By applying two HHBF distribution approaches, the final drag results are plotted versus the number of design variables in Figure 4.4(a). Note that the presented results correspond to the setting of $t = 8$. With respect to other values of t, the results show a similar pattern and are not reported herein for brevity. It is apparent that the "one-minus-cosine" distribution outperforms the uniform distribution by producing significantly lower drag values, suggesting a large dependence of the optimal solution on the distribution of the bump functions. As shown later, the geometric shape deformations are mainly concentrated in the fore and aft section of the aerofoil, indicating that the "one-minus-cosine" distribution is preferably used for the NACA0012 optimisation case.

The exception only occurs at $N_{\mathrm{dv}} = 5$, where the uniform distribution method performs better. It was found that the optimised geometry exhibits a flatter aft section when using the uniform distribution and a weaker shock is generated. Nonetheless, the number of design variables in this case is too small to fully cover the design space. This exception is thus not representative of the overall trend.

As the number of design variables is increased, the drag coefficient exhibits a convergence feature, especially for "one-minus-cosine" distribution, which suggests the design space is gradually explored. Moreover, the gap between these two groups of drag values becomes smaller when using more design variables. The drag difference is 39.1 counts for the case of $N_{\mathrm{dv}} = 40$, and the difference is further decreased to 21.6 counts when N_{dv} is doubled from 40 to 80. In the context hereafter, the "one-minus-cosine" distribution approach is used for the NACA0012 aerofoil optimisation case.



(a) Bump function distribution ($t = 8$)

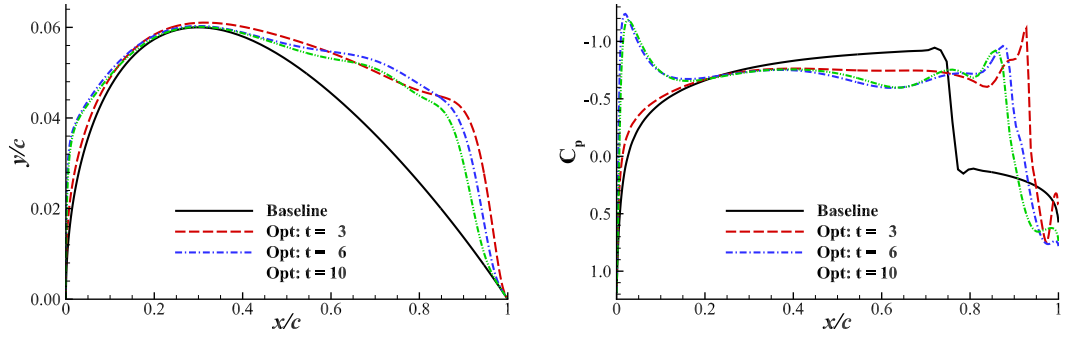
(b) Bump width control parameter

Figure 4.4: Influence of HHBF parameters on drag coefficient ($M = 0.85$, $\alpha = 0$ deg).

With respect to $N_{\mathrm{dv}} = 20$ and $N_{\mathrm{dv}} = 30$, Figure 4.4(b) shows the final drag results plotted versus the bump width control parameter. An evident observation is that the drag value drops sharply when t is increased from 3 to 5. Regarding the optimisation performance, the results obtained from $t = 3$ and $t = 6$ correspond to the two ends of the spectrum (shown as error bars in Figure 4.4(a)), where the drag difference for $N_{\mathrm{dv}} = 30$ is over 100 counts. This indicates that the bump width control parameter also has a large impact on the optimal solution in this optimisation case.

To find out the cause for this fact, three representative groups of optimisation results are compared in Figure 4.5. An evident distinction is observed in the leading edge area of the optimised aerofoil shape: a significantly blunt leading edge is generated with setting of $t = 6$ or $t = 10$, whereas the surface perturbation is trivial for the setting of $t = 3$. As mentioned earlier in Section 3.2.6.1, when the bump width control parameter t is set up with a larger value, more local shape control is achieved in geometry parameterisation. This property accounts for the fact that the setting of $t = 6$ or $t = 10$ effectively deforms the aerofoil in the narrow region near the leading edge, while the setting of $t = 3$ did not exhibit the same behaviour. Consequently, a suction peak in the pressure coefficient distribution is generated for $t = 6$ and $t = 10$, whereas the C_p distribution for $t = 3$ remains almost unchanged from the baseline aerofoil near the leading edge. Due to the

existence of pressure recovery after the suction peak, the two cases with larger values of t exhibit a weaker shock near the trailing edge and thus generate a lower wave drag. Since the wave drag contributes most to the total drag in this optimisation problem, it is not unexpected that the two larger values of t result in much better optimisation performance than that of $t = 3$. With consideration of the fact that $t = 6$ produces the lowest drag among all settings, this value is used hereafter for the NACA0012 aerofoil optimisation case.



(a) Aerofoil shape

(b) Pressure coefficient distribution

Figure 4.5: Influence of Hicks–Henne bump width control parameter on optimisation results ($N_{\mathrm{dv}} = 30$, $M = 0.85$, and $\alpha = 0$ deg).

It is worth noting that most studies in the literature successfully perturbed both the leading edge and aft section to minimise the wave drag while a few did not. For example, Fabiano and Mavriplis [253] employed 10 CST design variables to perform adjoint-based optimisation; however, the optimised aerofoil geometry near the leading edge remained almost unchanged. As a consequence, the drag value only reduced from 467 counts to 297 counts, which is much higher than that obtained from other participants. A sensitivity assessment is therefore required to tune the CST implementation to obtain the true optimal solution.

4.5.2 Impact of parameter settings in FFD

Two FFD parameterisation approaches embedded in SU2 framework are used in the NACA0012 optimisation case, which are FFD control point and FFD thickness method. The FFD box has a rectangular shape for the 2D aerofoil case and is defined by four boundaries. Figure 4.6 illustrates the schematic diagram of the FFD box that encloses the upper half of the NACA0012 aerofoil.

Optimisation can be successfully performed as long as these boundaries do not intersect with the embedded geometry and are located not too far from the aerofoil. Nonetheless, it was found that the drag result for the optimised aerofoil has a dependence on different settings of the FFD box position. A combination of the four boundary positions that

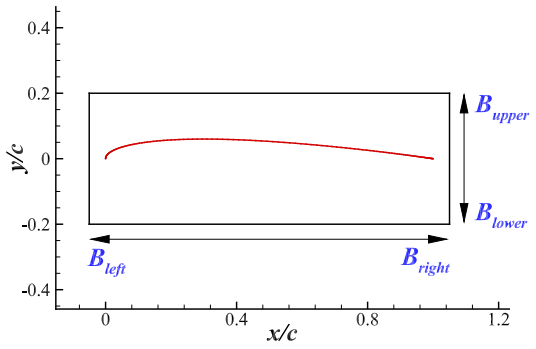


Figure 4.6: Schematic of FFD box set-up for the NACA0012 optimisation case.

can produce the best optimisation performance are listed in Table 4.5, and these FFD box settings are used hereafter for the NACA0012 aerofoil optimisation case.

Table 4.5: Settings of FFD box position with best practice for the NACA0012 aerofoil optimisation case.

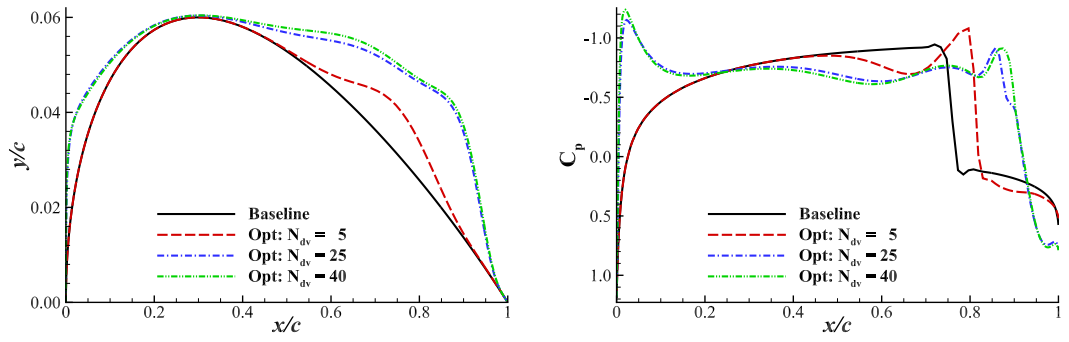
Parameterisation	B_{upper}	B_{lower}	B_{left}	B_{right}
FFD control point	$y = 0.2000$	$y = -0.2000$	$x = -0.0010$	$x = 1.0010$
FFD thickness	$y = 0.0601$	$y = -0.0601$	$x = -0.0001$	$x = 1.0001$

4.5.3 Dimensionality study

The dimensionality study was conducted using the best practice obtained from above investigation. Figure 4.7 plots the aerofoil shapes and pressure distributions for the optimised designs using HHBF parameterisation approach. As more design variables are used, the leading edge becomes blunter and the aft section of the aerofoil gets thicker, indicating that a flatter aerofoil surface is created. Correspondingly, a suction peak is generated in the C_p distribution near the leading edge and becomes steeper as N_{dv} is increased. The shock position moves further downstream towards the trailing edge.

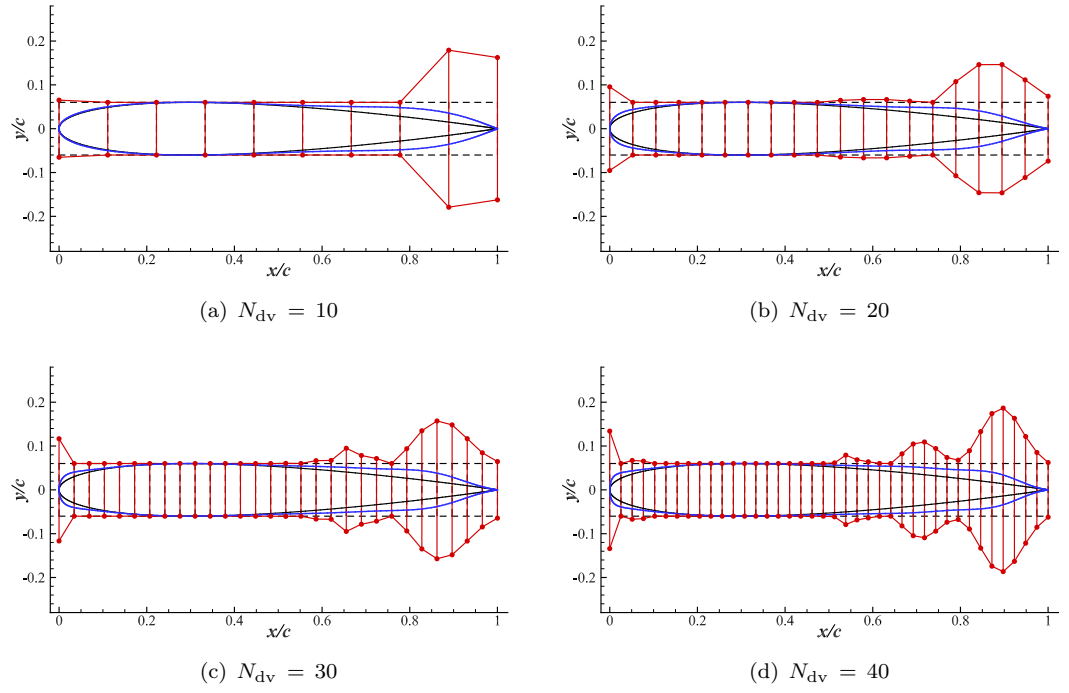
Figure 4.8 displays FFD box perturbation and aerofoil deformation using FFD thickness parameterisation method. As more control points are placed on the surface of FFD box, the optimiser is provided with more freedom to explore the design space, and thus better optimisation results are obtained.

The final drag results of the dimensionality study are shown in Figure 4.9. For both FFD methods, the drag coefficient monotonically decreases as more design variables are added into optimisation; for HHBF approach, however, the drag value initially drops sharply and then maintains a nearly constant level. This indicates that around 15 Hicks–Henne design variables are sufficient to cover the design space, while more FFD design variables are needed to do so. This fact is possibly caused by the difference of design variable distribution. The bump functions are distributed using "one-minus-cosine" function and



(a) Aerofoil shape (b) Pressure coefficient distribution

Figure 4.7: Influence of design variable dimensionality on optimisation results using HHBF approach ($M = 0.85$, $\alpha = 0$ deg).



(a) $N_{\mathrm{dv}} = 10$ (b) $N_{\mathrm{dv}} = 20$

(c) $N_{\mathrm{dv}} = 30$ (d) $N_{\mathrm{dv}} = 40$

Figure 4.8: FFD box perturbation and geometry deformation for dimensionality study using FFD thickness parameterisation method (original FFD box and aerofoil geometry in black, deformed FFD box in red and deformed aerofoil geometry in blue).

are clustered in the area where the surface sensitivities are relatively large. By contrast, the control points are placed uniformly on the surface of FFD box, which means that more design variables are needed for the optimiser to fully explore the design space. In terms of optimisation performance, the best result for each parameterisation method achieves drag reduction magnitudes of around 80%. Specifically, FFD control point method produces the lowest drag with 80.5 counts in the optimisation case using 40 design variables, which corresponds to 82.8% of drag reduction.

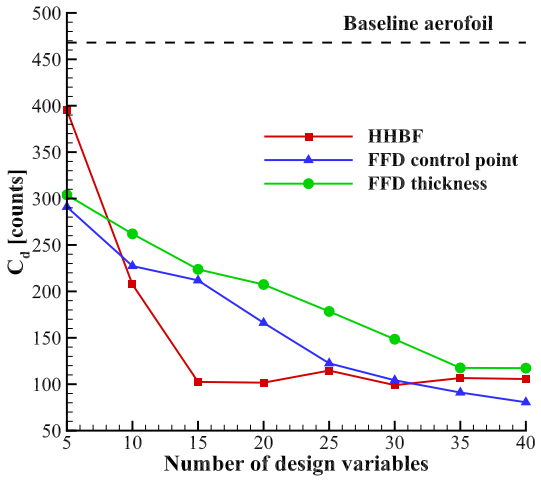
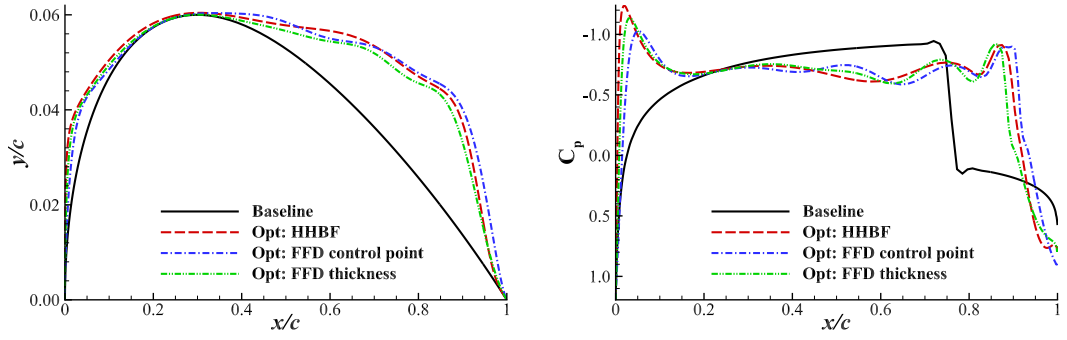


Figure 4.9: Drag coefficient results obtained from dimensionality study using three parameterisation methods ($M = 0.85$, $\alpha = 0$ deg).

For the case with 40 design variables, the optimised aerofoil shapes and pressure coefficient distributions are compared in Figure 4.10 with respect to three parameterisation methods. It is worth observing that very similar optimisation results are obtained, reflecting that the parameterisation methods employed in this work are equivalently effective for this optimisation problem.



(a) Aerofoil shape

(b) Pressure coefficient distribution

Figure 4.10: Case 1: comparison of optimisation results obtained from using three parameterisation methods ($N_{\text{dv}} = 40$, $M = 0.85$, and $\alpha = 0$ deg).

The convergence histories of the objective function are plotted in Figure 4.11. In terms of the convergence rate of optimisation, HHBF approach performs better than two FFD methods as it requires much less design iterations to meet the KKT conditions. This is possibly due to the fact that HHBF places the design variables in areas with high sensitivities through using "one-minus-cosine" distribution. For each parameterisation method, as the number of design variables is increased, the design space is enlarged, and more design cycles are needed for the optimiser to find the local minimum. As shown later in Section 4.6, the existence of non-unique flow solutions causes difficulty for the

optimiser to find the optimum, especially in the cases with more design variables, which also leads to the slow convergence of optimisation.

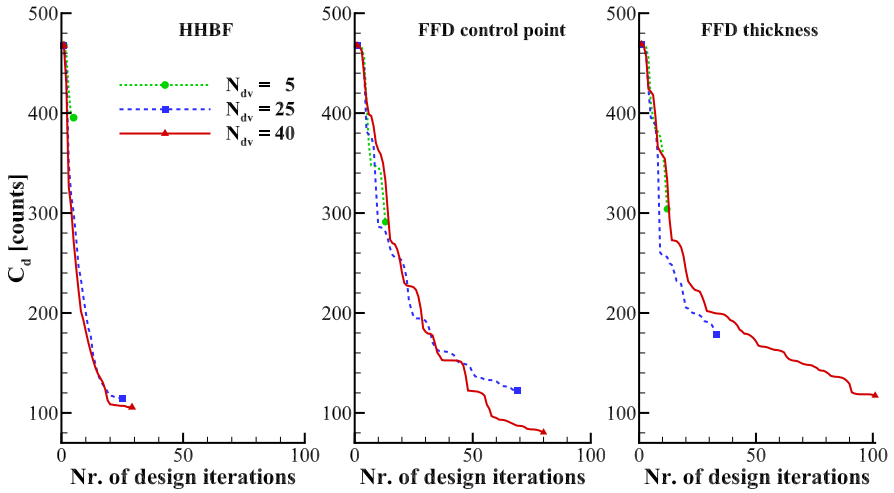
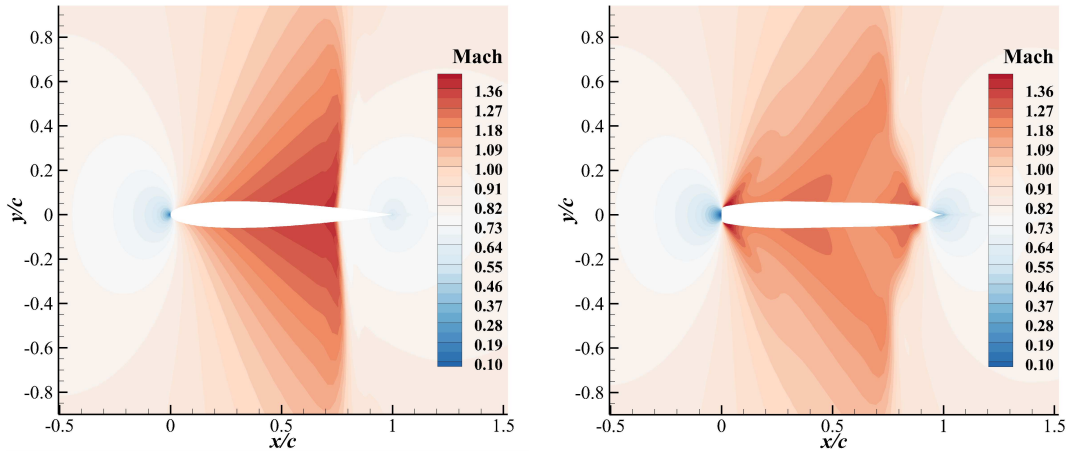


Figure 4.11: Convergence histories for the NACA0012 aerofoil optimisation using three parameterisation methods.

4.6 Optimisation results

The drag reduction mechanism in this optimisation problem is to minimise the strength of the shock wave. A representative optimisation case with 30 Hicks–Henne design variables is taken for analysis. Figure 4.12 displays the Mach contours for both the baseline and optimised aerofoil. An aerofoil with a flat surface is created by the optimiser through thickening the leading edge as well as the aft section. This is reflected by the surface sensitivities, shown in Figure 4.13, which are computed by the adjoint solver within SU2 and then serve as search direction information to guide the optimiser. For the baseline aerofoil, a strong shock exists at about three quarter-chord position, whereas for the optimised aerofoil, the shock is substantially weakened and is pushed further downstream, locating at around 90% chordwise position. Hence, the total drag is substantially reduced by minimising the wave drag.

Figure 4.14(a) plots the drag coefficient convergence history versus function evaluations for the case of $N_{\mathrm{dv}} = 30$. It is evident that the optimiser consistently produces designs with sharply increased drag values, especially in late stage of optimisation process when the optimal point is nearly reached. This behaviour is also observed from other cases, particularly for those with large number of design variables. To understand the cause for this phenomenon, two representative designs with different drag values were selected, and their optimisation results were compared. It was found that the aerofoil shapes are almost identical, however, the steady flow solutions exhibit different features. Figure 4.14(b) shows the comparison of Mach contour results. It is apparent that the



(a) Baseline aerofoil (b) Optimised aerofoil

Figure 4.12: Mach contours for baseline and optimised NACA0012 aerofoil using HHBF parameterisation method ($N_{\text{dv}} = 30$, $M = 0.85$, and $\alpha = 0$ deg).

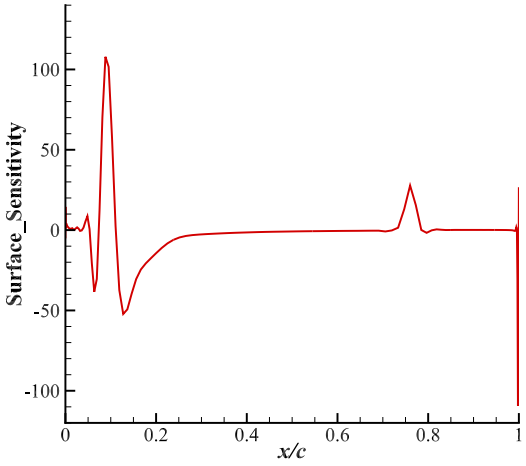
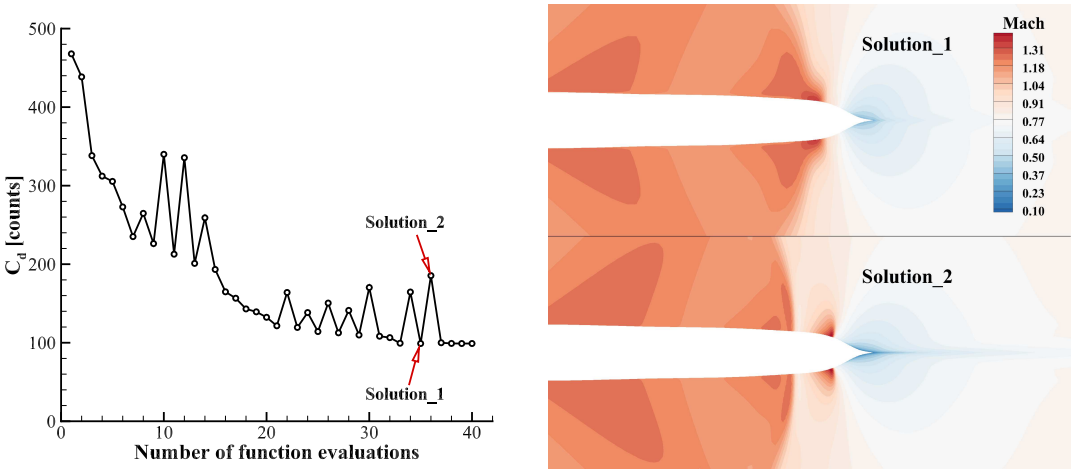


Figure 4.13: Surface sensitivity for the baseline NACA0012 aerofoil.

primary difference between these two flow solutions is the shock wave structure near the trailing edge. One flow solution shows a single shock, whereas the other solution demonstrates double shock structure. This indicates that non-unique flow solutions exist with respect to the optimised NACA0012 aerofoil, which accounts for the convergence difficulty shown in Figure 4.11. In addition, the presence of non-unique solutions in this ADODG optimisation case has also been reported by other researchers [57, 226, 227, 255]. Note that a symmetry boundary condition was applied in this case for the half model of the aerofoil geometry. As a result, the flowfield solution is symmetric as well, as shown in Figure 4.14(b). However, by using the whole aerofoil geometry, the flowfield solutions obtained from other references [57, 227] are asymmetric, generating a single shock on one side and a double shock on the other side. The reason was explained by Ou *et al.* [257], who states as follows: for a symmetric aerofoil in critical transonic

flow, a small disturbance in angle of attack increases the Mach number on one side and reduces the Mach number on the other; consequently, a single shock is formulated on one side with increasing speed and a double shock on the other side; a lift force is thus generated following the perturbation, and a circulation is generated around the aerofoil, which reinforces the initial local speed perturbation and stabilises the mechanism.



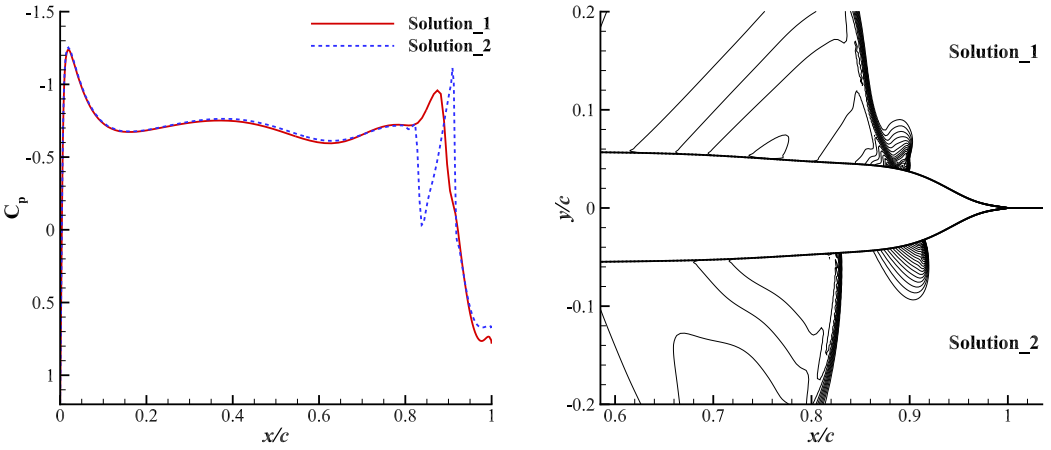
(a) Convergence history

(b) Mach contour comparison

Figure 4.14: Non-unique flow solutions for the NACA0012 optimised aerofoil using HHBF parameterisation method ($N_{\text{dv}} = 30$, $M = 0.85$, and $\alpha = 0$ deg).

Figure 4.15(a) plots pressure coefficient distributions on the optimised aerofoil for the two flow solutions. It is apparent that the difference mainly exists for the shock topology near the trailing edge. An inspection of the flowfield is provided in Figure 4.15(b), displaying supersonic iso-Mach number line patterns. Solution_1 shows a single supersonic area (for a half-aerofoil), and solution_2 shows a double supersonic area. Specifically, solution_1 produces a supersonic/subsonic shock at a distance in the field; and near the aerofoil, a more complex structure is formulated, consisting of a supersonic/supersonic shock followed by a short supersonic expansion and ending with a supersonic/subsonic shock. Solution_2 produces an extended supersonic/subsonic shock in the field and a subsonic-to-supersonic expansion leading to a second less extended supersonic/subsonic shock. The shock structure is closely related to drag production with aerofoils and hence explains the drag difference observed between the two flow solutions.

To further investigate the issue of non-uniqueness, a series of flow analysis were conducted on the optimised aerofoil. The Mach number was swept up and down in a small range around the design point ($M = 0.85$), and each flow simulation was initialised using the converged solution obtained from the previous simulation. The drag results are then shown in Figure 4.16. It is worth observing that hysteresis behaviour occurs for Mach sweeps, and the drag coefficient results exhibit bifurcation in a narrow region. As the design point falls into this bifurcation zone, two distinct flow solutions are thus expected. The single-shock solution was generated with relatively lower drag in



(a) Pressure coefficient distribution

(b) Supersonic iso-Mach number lines ($\Delta M = 0.025$)

Figure 4.15: Comparison of non-unique flow solutions for the NACA0012 optimised aerofoil ($N_{\mathrm{dv}} = 30$, $M = 0.85$, and $\alpha = 0$ deg).

downward Mach sweep, whereas the double-shock solution was produced with relatively higher drag in upward Mach sweep. These results may indicate that the NACA0012 optimised aerofoil is ill-posed in inviscid transonic flow. Moreover, it was found by Jameson *et al.* [258] and Ou *et al.* [257] that, for aerofoil geometries with a nearly flat section, non-unique transonic flow solutions not only exist for steady Euler flows but also for unsteady RANS flows.

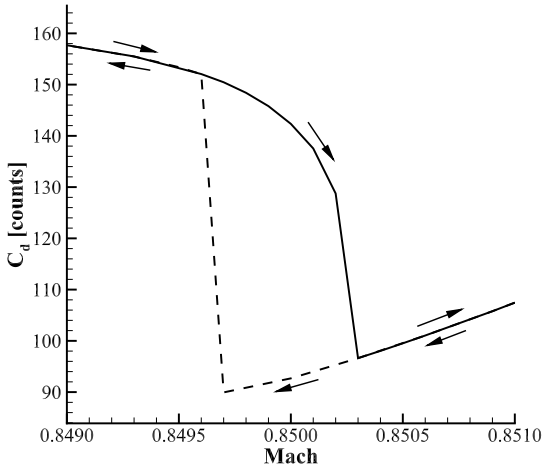


Figure 4.16: Drag coefficient hysteresis for Mach sweep analysis on the NACA0012 optimised aerofoil ($N_{\mathrm{dv}} = 30$, $\alpha = 0$ deg). Arrows indicate the sweep direction.

4.7 Chapter summary

In this chapter, the benchmark NACA0012 aerofoil optimisation case was investigated using SU2. Three shape parameterisation methods are employed in this study, which are HHBF, FFD control point and FFD thickness approach. The aim is to investigate the sensitivity of the optimal solution to a number of model parameters. The key findings of the numerical optimisations are summarised as follows:

1. **Optimised geometric shape**
   The optimised aerofoil exhibits a blunter leading edge and a thicker aft section compared with the baseline geometry. The strong shock wave at about three quarter-chord location is substantially weakened and pushed further downstream toward the trailing edge.

2. **Sensitivity assessment**
   The optimal solution shows sensitivity to several parameters in Hicks–Henne and FFD parameterisation methods, such as bump function distribution, bump width control parameter and FFD box position. For this optimisation problem, the "one-minus-cosine" distribution for bump functions outperforms the uniform distribution approach, and the setting of $t = 6$ produces the lowest drag results among a range of values.

   The impact of design variable dimensionality on optimisation performance is investigated. It was found that 15 HHBFs are sufficient to cover the design space, whereas more design variables are needed for FFD approaches. Each parameterisation method achieves approximately 80% of drag reduction with respect to the baseline value, and HHBF outperforms FFD approaches in terms of optimisation convergence speed.

3. **Non-uniqueness for optimal solution**
   Non-unique flow solutions were found to exist on the optimised aerofoil, which demonstrate single-shock and double-shock structure, respectively. In Mach number sweep analysis, hysteresis behaviour of the drag coefficient occurs in a narrow region around the design point, indicating that the optimisation problem is in fact ill-posed.

Chapter 5

Two-dimensional RAE2822 Aerofoil Optimisation

5.1 Introduction

Nowadays, most commercial aircraft fly at transonic speed during the cruise stage. As the flow speed of the aircraft approaches the speed of sound, the air accelerating around the wing reaches Mach 1 and shock waves begin to form, which produces wave drag and thus deteriorates the overall aerodynamic performance. Supercritical aerofoils are designed specifically to delay the onset of shock wave in the transonic speed range and are commonly used in modern aircraft. Compared with the conventional aerofoils, supercritical aerofoils are characterised by the flattened upper surface and highly cambered aft section.

The RAE2822 aerofoil, shown below in Figure 5.1, is a supercritical aerofoil [259], which has become a standard test case for turbulence modelling validation. This aerofoil is made up with a max camber of 2%, camber position of 80%, and maximum thickness-to-chord ratio of 22%. The RAE2822 aerofoil is often used in CFD in order to model shock waves and other phenomena in 2D transonic flow.

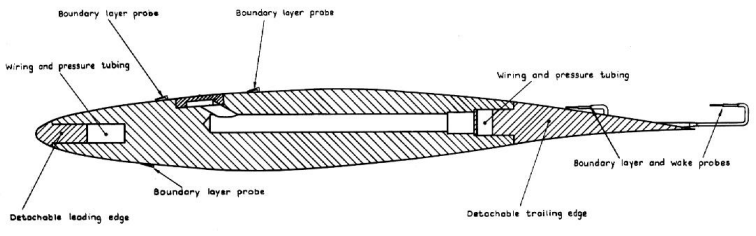


Figure 5.1: Schematic of RAE2822 aerofoil (Source: [259]).

The second case of AIAA ADODG benchmark problems is the drag minimisation of the RAE2822 aerofoil subject to several constraints of geometric and aerodynamic quantities. This optimisation case has previously been investigated by a number of participants [57, 125, 131, 143, 227], and various methodologies and tools have been applied to each component of the ASO framework. Differing from the first case of NACA0012 optimisation, which is more of an academic problem, the RAE2822 optimisation is a case of more practical value and considers realistic constraints.

In this chapter, gradient-based numerical optimisations are performed using SU2 for the benchmark RAE2822 optimisation. Similar to the NACA0012 case, two types of geometry parameterisation, HHBF and FFD, are employed for the RAE2822 case, and the influence of several key parameters on the optimal solution is investigated. In addition, two sets of parameter values in turbulence modelling are used for the viscous optimisation case.

This chapter is structured in the following manner. The optimisation problem is firstly described in Section 5.2. The computational grid and optimisation set-up are then given in Section 5.3 and 5.4, respectively. Section 5.5 provides a detailed sensitivity assessment. The optimisation results are presented in Section 5.6. Section 5.7 finally summarises the key findings in this chapter.

5.2 Optimisation problem

The optimisation problem is the drag minimisation of the RAE2822 aerofoil in viscous, transonic flow. The freestream Mach number is 0.734, and the Reynolds number, Re, is $6.5 \cdot 10^6$. The lift coefficient is constrained to 0.824, the pitching moment coefficient (evaluated at the quarter-chord) must be no less than -0.092, and the aerofoil area must be greater than or equal to the initial aerofoil area. The optimisation problem is written as

$$\begin{aligned} \text{Minimise:} \quad & C_d \\ \text{w.r.t:} \quad & y \\ \text{Subject to:} \quad & C_l = 0.824 \\ & C_m \geq -0.092 \\ & S \geq S_0 \end{aligned}$$

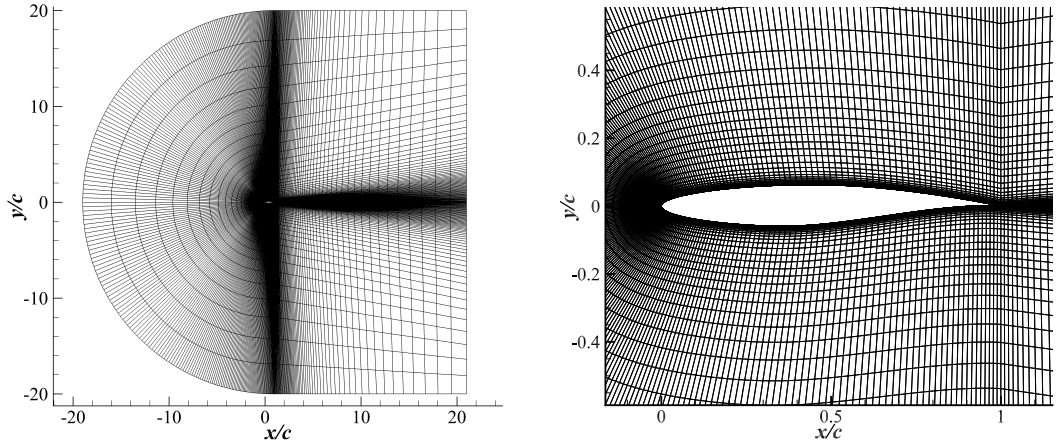
where C_d, C_l and C_m are the drag, lift, and pitching moment coefficients, respectively; and S and S_0 are the optimised and initial aerofoil areas, respectively. The coordinates of the RAE2822 aerofoil are obtained from the UIUC Aerofoil Coordinates Database[1].

[1] https://m-selig.ae.illinois.edu/ads/coord_database.html [retrieved 2019]

In order to satisfy the lift constraint, the angle of attack is set up as an additional design variable in this optimisation case.

5.3 Computational grid

For the RAE2822 aerofoil, the structured C-grid was generated using *Pointwise* grid generator and is shown in Figure 5.2. The computational domain is bounded by a no-slip aerofoil surface and a farfield located at a distance of 20 chord lengths from the aerofoil. The coarse grid consists of 385 × 65 grid points in the wrap-around and normal directions respectively, where 257 points are distributed along the aerofoil and 65 points in the grid cut. The off-wall spacing was also set to ensure that the y^+ [2] is smaller than 1. To establish grid convergence, four levels of grid were generated with the same topology, and the key parameters are summarised in Table 5.1.



(a) Far view

(b) Near view

Figure 5.2: Computational domain and coarse grid (385 × 65) for the RAE2822 aerofoil.

Table 5.1: Grid parameters for the RAE2822 aerofoil grid convergence study.

Grid Level	Grid Size	Grid Points	LE Spacing	TE Spacing	Off-wall Spacing
Coarse	385 × 65	24,960	$2.0 \cdot 10^{-4}$	$1.0 \cdot 10^{-2}$	$1.0 \cdot 10^{-5}$
Medium	549 × 93	50,964	$1.4 \cdot 10^{-4}$	$7.1 \cdot 10^{-3}$	$7.1 \cdot 10^{-6}$
Fine	769 × 129	99,072	$1.0 \cdot 10^{-4}$	$5.0 \cdot 10^{-3}$	$5.0 \cdot 10^{-6}$
Superfine	1097 × 185	202,760	$7.1 \cdot 10^{-5}$	$3.5 \cdot 10^{-3}$	$3.5 \cdot 10^{-6}$

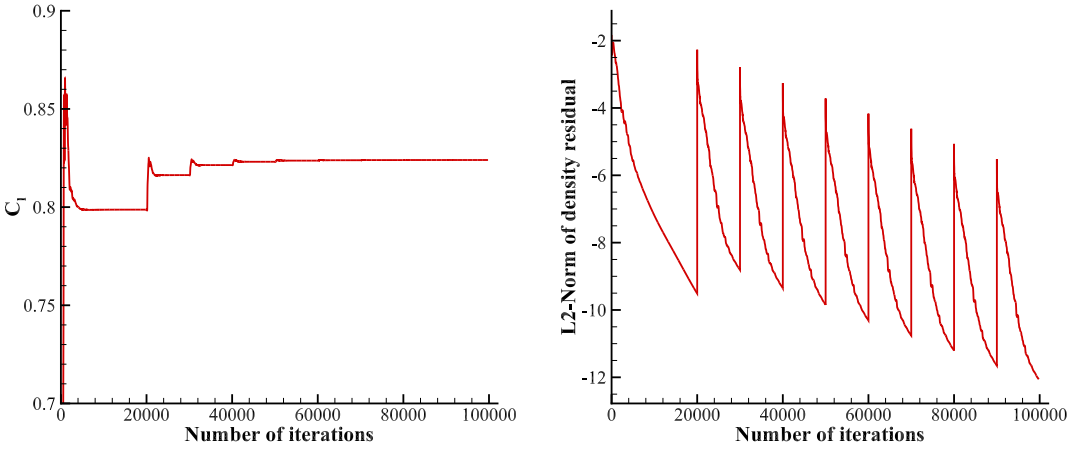
For the baseline aerofoil, the compressible RANS simulation was conducted using the standard single-equation SA turbulence model. In order to meet the lift target, the angle

[2]The non-dimensional off-wall distance y^+ is given by $y^+ = \frac{u_\tau y}{\nu}$, where $u_\tau = \sqrt{\tau/\rho}$ is the friction velocity at the wall, $\tau = \mu \frac{\partial u}{\partial y}$ is the wall shear stress, y is the distance to the wall, and ν is the local kinematic viscosity of fluid. Typically, the y^+ value is required to be smaller than 5 for SA turbulence model in order to resolve the viscous sublayer in the RANS simulations.

of attack was iteratively updated during the flow analysis using the following formula

$$\alpha(i+1) = \alpha(i) + \frac{C_{l,\,\text{target}} - C_l(i)}{\mathrm{d}C_l \,/\, \mathrm{d}\alpha}, \qquad i = 0,\, 1,\, 2\,... \tag{5.1}$$

where $\alpha(0) = 3$ deg, $C_{l,\,\text{target}} = 0.824$, and $dC_l \,/\, d\alpha = 0.2$ [deg^{-1}] for this case. Figure 5.3 shows an example of the convergence history for the CFD simulation. It is evident that the angle of attack was updated 8 times to obtain the desired lift coefficient.



(a) Lift coefficient

(b) Density residual

Figure 5.3: Convergence history for CFD simulation of the baseline RAE2822 aerofoil on coarse grid (385 × 65).

The aerodynamic results of the baseline aerofoil are listed in Table 5.2. The initial aerofoil area was evaluated to be 0.07787. The pressure distributions for grid convergence study are plotted in Figure 5.4. A shock wave is located at approximately 57% chordwise position on the upper surface of the aerofoil. The coarse grid was found adequate to guarantee grid independent solutions, and was thus used for optimisation.

Table 5.2: Aerodynamic results for the RAE2822 aerofoil grid convergence study at $M = 0.734$ and $Re = 6.5 \cdot 10^6$.

Grid Level	C_l	C_d (Counts)	C_m	α [deg]
Coarse	0.82400	241.24	-0.089	3.1848
Medium	0.82400	236.35	-0.091	3.1071
Fine	0.82400	232.27	-0.092	3.0654
Superfine	0.82400	228.81	-0.093	3.0450

5.4 Optimisation set-up

The SU2 suite is used to perform flow analysis and gradient-based optimisation for the RAE2822 case. A summary of the computational methods and numerical settings is

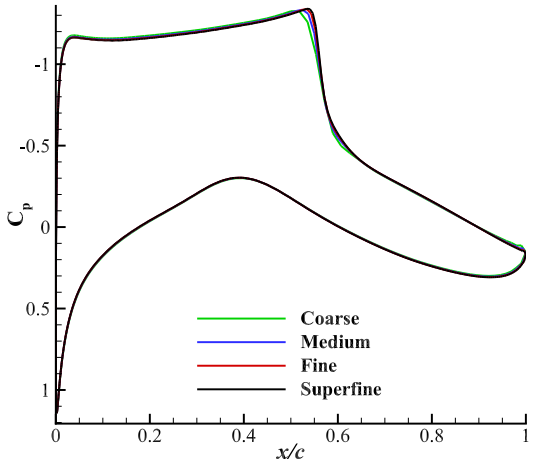


Figure 5.4: Pressure coefficient distributions on the baseline RAE2822 aerofoil in the grid convergence study ($M = 0.734$, $C_l = 0.824$, and $Re = 6.5 \cdot 10^6$).

given in Table 5.3.

Table 5.3: Computational methods and numerical settings for the RAE2822 optimisation.

Parameter	Methodology or Value
Geometry parameterisation	HHBF/FFD
Grid deformation	Linear elasticity equations
Flow governing equations	RANS
Turbulence model	SA
Spatial discretisation	2nd order JST
Time discretisation	Euler implicit
CFL number	5
Optimiser	SLSQP
Gradient evaluation	Continuous adjoint
Scaling factor for objective function	0.001
Scaling factor for constraint functions	0.001
Maximum number of iterations	100
Tolerance of KKT conditions	$1 \cdot 10^{-6}$

Two types of shape parametrisation, HHBF and FFD, are employed for the RAE2822 optimisation. Same with the NACA0012 case, a number of parameter values need to be defined prior to optimisation. In addition, two sets of closure coefficients for SA turbulence model are used for the RAE2822 case. The main numerical parameters used in this case are summarised in Table 5.4.

Table 5.4: Numerical parameters for two geometry parameterisation methods; the values in square brackets indicate the range for a specific parameter.

Parameter	Methodology or Value
HHBF	
Distribution approach	Uniform/One-minus-cosine
Bump width control parameter	[1, 10]
Number of design variables	[5, 40]
FFD	
FFD box: upper/lower	[±0.08, ±0.4]
FFD box: left	[-0.01, -0.0001]
FFD box: right	[1.0001, 1.01]
Number of design variables	[5, 40]
Turbulence model	Standard SA/Calibrated SA

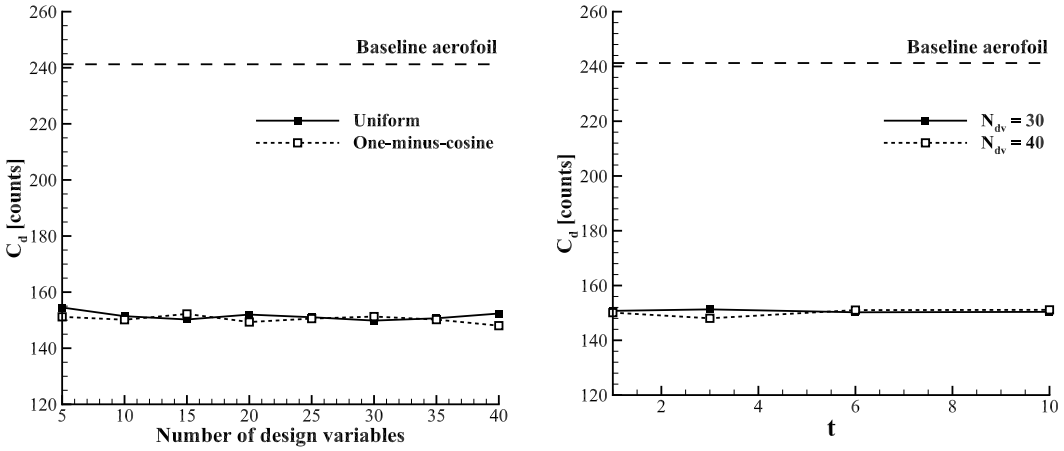
5.5 Sensitivity assessment

This section contains results obtained from the sensitivity assessment for the RAE2822 optimisation case. This work primarily focuses on investigating the sensitivity of the optimal solution to a series of model parameters, with the purpose of finding a suite of parameter values that can produce the best optimisation result. Additionally, the influence of SA turbulence model coefficients on the flow solution and optimisation performance is also investigated.

5.5.1 Impact of parameter settings in geometry parameterisation

Both the HHBF and FFD control point approach are employed for geometry parameterisation. The impact of parameter settings within HHBF on optimisation performance is firstly investigated. The final drag results are shown in Figure 5.5. It is apparent that the same level of optimisation performance is achieved using different parameter settings, implying that the optimal solution in this case is insensitive to the settings of both bump function distribution and bump width control parameter. In this study hereafter, the uniform distribution is selected and the default setting of $t = 3$ is used.

With respect to FFD parameterisation, the FFD box is defined by four boundaries and has a rectangular shape, as illustrated in Figure 5.6. A number of control points are placed on the upper and lower surfaces of the FFD box, and their movement is restricted in the y-direction. Different settings of the FFD box position are applied. Nonetheless, the optimisation performance shows independence of FFD box position, and the results are not reported herein for brevity. In this work hereafter, the FFD box settings listed in Table 5.5 are used.



(a) Bump function distribution ($t = 3$)

(b) Bump width control parameter

Figure 5.5: Influence of HHBF parameters on drag coefficient ($M = 0.734$, $C_l = 0.824$, and $Re = 6.5 \cdot 10^6$).

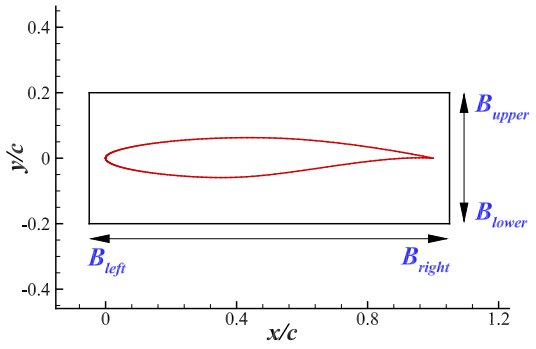


Figure 5.6: Schematic of FFD box set-up for the RAE2822 optimisation case.

Table 5.5: Settings of FFD box position for the RAE2822 aerofoil optimisation case.

Parameterisation	B_{upper}	B_{lower}	B_{left}	B_{right}
FFD control point	$y = 0.100$	$y = -0.100$	$x = -0.001$	$x = 1.001$

5.5.2 Dimensionality study

The effect of dimensionality on the optimal solution is then investigated, and the final drag results are shown in Figure 5.7. It is observed that the drag values vary in a very small range and the optimisation performance does not improve when using more design variables. Figure 5.8 displays FFD box perturbation and aerofoil deformation. The design space is easily explored in this case as only 5 design variables are sufficient to locate the optimum. Additionally, HHBF and FFD control point methods are found to be equivalently effective for this optimisation problem, achieving a drag reduction of approximately 38% in both cases.

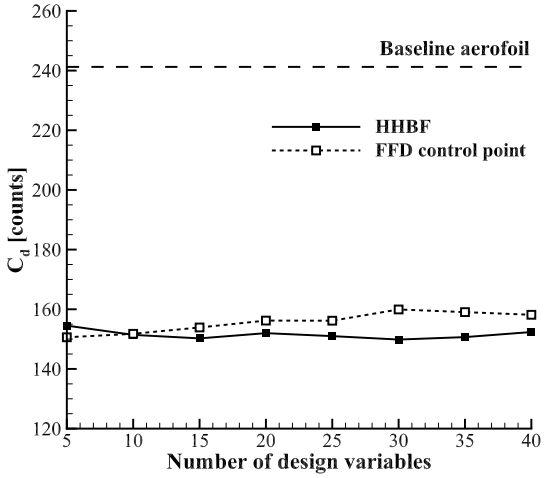
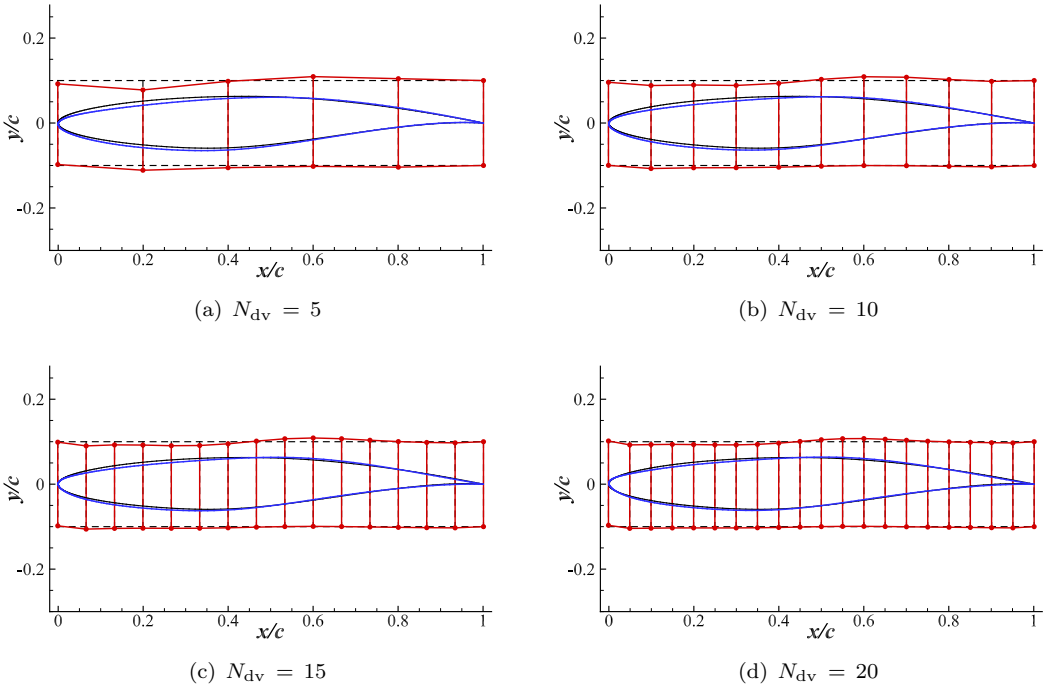


Figure 5.7: Drag coefficient results obtained from dimensionality study using two parameterisation methods ($M = 0.734$, $C_l = 0.824$, and $Re = 6.5 \cdot 10^6$).



(a) $N_{\mathrm{dv}} = 5$ (b) $N_{\mathrm{dv}} = 10$

(c) $N_{\mathrm{dv}} = 15$ (d) $N_{\mathrm{dv}} = 20$

Figure 5.8: FFD box perturbation and geometry deformation for dimensionality study using FFD control point parameterisation method (original FFD box and aerofoil geometry in black, deformed FFD box in red and deformed aerofoil geometry in blue).

The convergence histories of the objective function are shown in Figure 5.9. The drag coefficient drops substantially in the first few design iterations, and then reduces slowly until the required KKT conditions are met. The RAE2822 optimisation converges faster when using HHBF parameterisation method over the FFD approach.

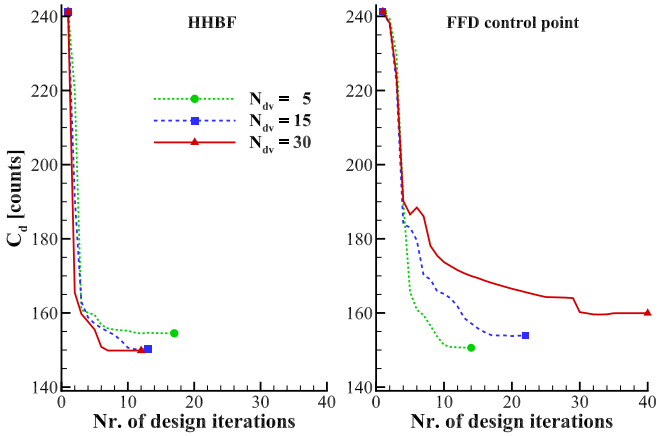


Figure 5.9: Convergence histories for the RAE2822 aerofoil optimisation using two parameterisation methods.

5.5.3 Impact of turbulence model closure coefficients

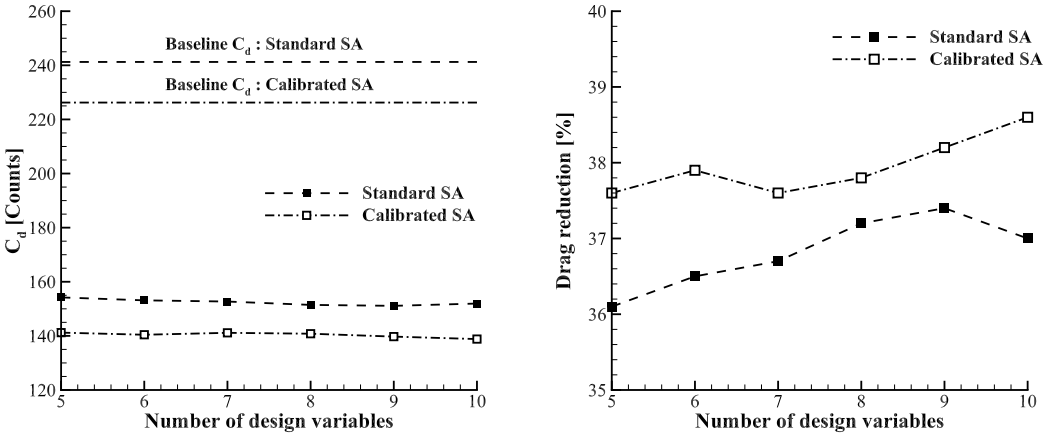
For the RAE2822 aerofoil, the solution of the RANS equations employs the SA turbulence model, and the closure coefficients are set to standard values. However, there is no guarantee that standard values are universal for all test cases. Da Ronch *et al.* [260] recently revisited the calibration of the SA closure coefficients for the RAE2822 aerofoil in transonic flow. The calibration was performed with the aid of machine learning and adaptive DoE techniques by minimising the deviation of numerical pressure coefficient results from available experimental data. Both standard and calibrated values of SA closure coefficients are listed in Table 5.6. It is worth noting that the last five parameters (c_{w2} through c_{t4}) were kept at their nominal values in the case tested as they have nearly zero influence on the outputs. The reader is referred to Reference [260] for more details about the calibration process.

Table 5.6: Standard and calibrated values of SA turbulence model closure coefficients; data from [260].

Parameter	Standard value	Calibrated value
κ	0.4100	0.3600
c_{v1}	7.1000	7.5000
σ	0.6667	1.0030
c_{b1}	0.1355	0.1400
c_{w2}	0.3000	0.3000
c_{b2}	0.6220	0.6220
c_{t3}	1.2000	1.2000
c_{w3}	2.0000	2.0000
c_{t4}	0.5000	0.5000

The above two sets of parameter values for SA turbulence model were employed in the RAE2822 optimisation case. The HHBF was used as the parameterisation approach.

Since the design space is easily explored in this case, the number of Hicks–Henne design variables ranges from 5 to 10 herein. Optimisations were then carried out using both standard and calibrated SA models, and the corresponding drag results are shown in Figure 5.10. Regarding the case with 10 design variables on each surface, the drag gradients on the baseline geometry are shown in Figure 5.11.



(a) Drag coefficient

(b) Percentage of drag reduction

Figure 5.10: Comparison of drag results obtained from optimisation using two sets of SA turbulence model closure coefficients ($M = 0.734$, $C_l = 0.824$ and $Re = 6.5 \cdot 10^6$).

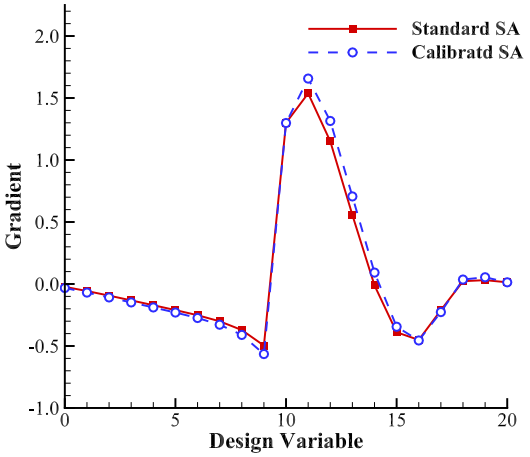
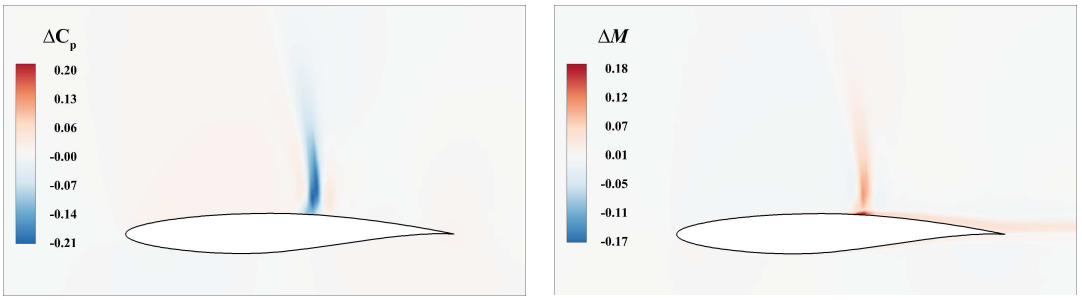


Figure 5.11: Drag gradients on the baseline geometry using both the standard and calibrated SA turbulence models.

For the baseline RAE2822 aerofoil, it is observed from Figure 5.10(a) that the drag results obtained from using the standard and calibrated SA model differ by approximately 15 counts, which is close to the value reported in Reference [260]. This is indicative of a certain sensitivity of the RANS solution on the SA turbulence model coefficients. Figure 5.12 shows the difference in the flowfield solutions obtained using the standard and calibrated SA model. The differences mainly exist at the shock region. The reason is

attributed to the fact that the pressure coefficient distribution obtained with calibrated SA model improves the agreement with the experimental data, particularly near the leading edge and at the shock front. Concerning the drag gradient, the results obtained from using calibrated coefficients match closely with those obtained from standard coefficients, and the largest difference is approximately 7% compared to the peak value of gradients. This fact indicates that the calibration process has trivial impact on the drag reduction mechanism for this case.



(a) Pressure coefficient (b) Mach number

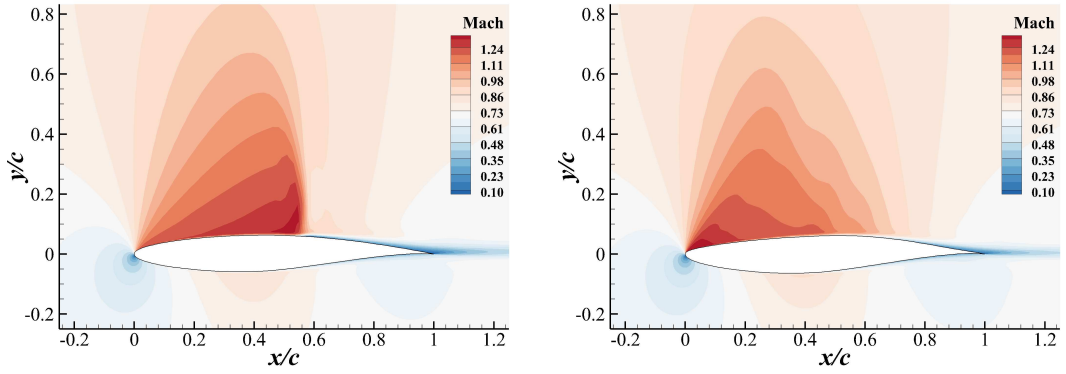
Figure 5.12: Difference in baseline RAE2822 flowfield solutions obtained using standard and calibrated SA turbulence model ($M = 0.734$, $C_l = 0.824$ and $Re = 6.5 \cdot 10^6$).

For the optimised RAE2822 aerofoil, the drag results obtained when using calibrated SA model are, on average, 12 counts lower than those obtained when using the standard model. As shown in Figure 5.10(b), the drag reduction is, on average, 1.1% higher when replacing standard SA model by calibrated model. In the area of aircraft design, a high level of accuracy with respect to drag prediction is required, and this need is confirmed by Meredith [261], who showed that one drag count is equal to the weight of one passenger in a long-haul aircraft. This highlights the importance of turbulence modelling, and the need for more extensive calibration campaigns to reduce modelling uncertainties. It is worth noting that although the calibration of SA coefficients was carried out on the RAE2822 aerofoil using the DLR-Tau solver, the calibrated SA turbulence model slightly outperforms the standard version in analysing the complex flow features around the ONERA M6 wing; and the expected prediction accuracy holds across different flow solvers [260]. A further dependence of the optimal solution would be on the turbulence model, which is not done in this work, but the reader may have an indication of its importance by looking at other references [262, 263].

5.6 Optimisation results

The optimisation case selected for analysis has 20 Hicks–Henne design variables and uses standard SA turbulence model. For both the baseline and optimised design, the Mach contour results are shown in Figure 5.13. The strong shock wave on the upper surface

of the baseline aerofoil is eliminated after optimisation. The total drag is thus reduced by removing the component of wave drag. The drag reduction is achieved through generating a flat fore section on the upper surface, which is also reflected by the surface sensitivity, as shown in Figure 5.14.



(a) Baseline aerofoil (b) Optimised aerofoil

Figure 5.13: Mach contours for baseline and optimised RAE2822 aerofoil using HHBF parameterisation method ($M = 0.734$, $C_l = 0.824$, $Re = 6.5 \cdot 10^6$, and $N_{\text{dv}} = 20$).

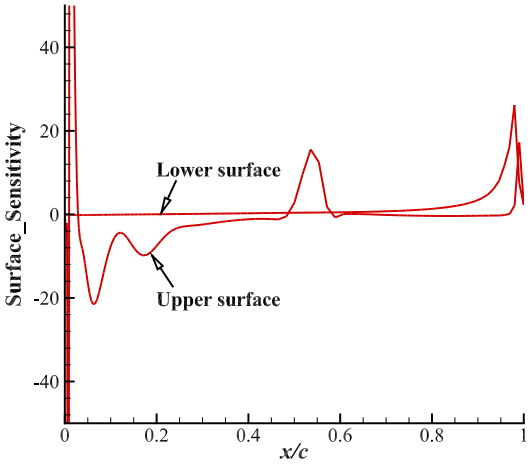


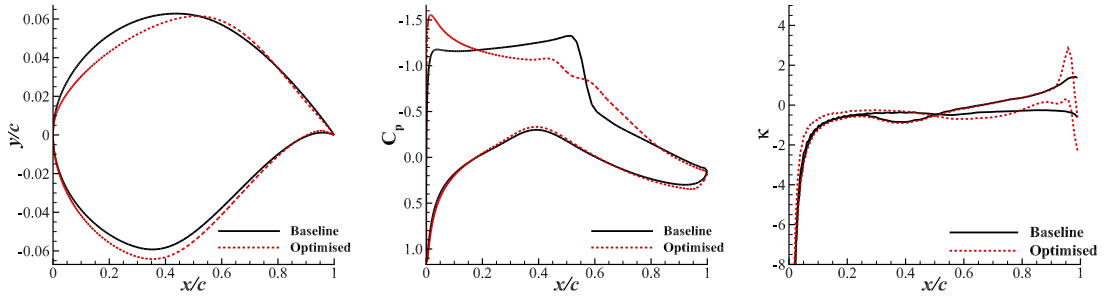
Figure 5.14: Surface sensitivity for the baseline RAE2822 aerofoil.

In order to further understand the drag reduction mechanism, the aerofoil shapes, pressure distributions as well as surface curvature distributions are plotted in Figure 5.15 for comparison. It should be noted that the curvature of aerofoil surface, κ, is defined to be the reciprocal of the local radius r, and is expressed as

$$\kappa = \frac{1}{r} = \frac{x'y'' - y'x''}{(x'^2 + y'^2)^{\frac{3}{2}}} = \frac{y''}{(1 + y'^2)^{\frac{3}{2}}} \tag{5.2}$$

The geometry deformation largely occurs in the fore section of the aerofoil. On the upper side, the curvature is reduced to create a relatively flat surface, which alleviates

the flow acceleration and hence delays or eliminates the formulation of shock wave; whereas on the lower side, the aerofoil becomes thicker, which is primarily to satisfy the area constraint. Correspondingly, the pressure discontinuity at around 56% chordwise position is replaced by a smooth pressure recovery, thus eliminating the shock wave and reducing the drag. Additionally, the surface curvature near the trailing edge is enlarged after optimisation, and the local camber is increased accordingly.



(a) Aerofoil shape (b) Pressure coefficient distribution (c) Surface curvature

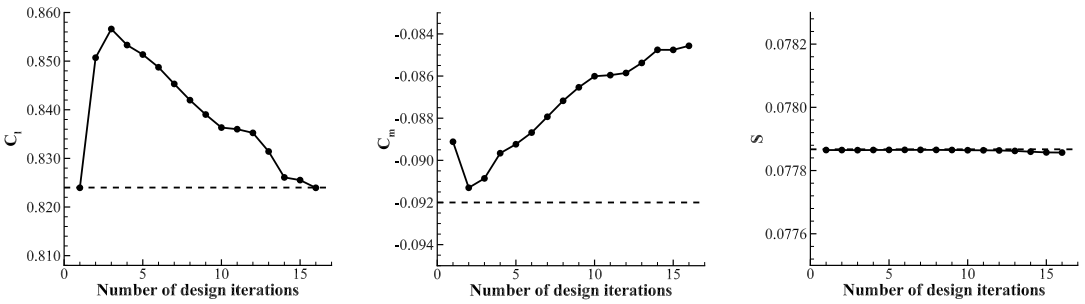
Figure 5.15: Comparison of optimisation results for the RAE2822 aerofoil optimisation using HHBF parameterisation method ($M = 0.734$, $C_l = 0.824$, $Re = 6.5 \cdot 10^6$, and $N_{\text{dv}} = 20$).

We have further investigated the impact that the optimal shape has on the aerodynamic derivatives. Results are summarised in Table 5.7. It was found that the influence on the drag and pitching moment curve slopes, C_{d_α} and C_{m_α} respectively, is minimal but significant on the lift curve slope, C_{l_α}. This may potentially affect the aerodynamic performance such as the gust response of the aerofoil.

Table 5.7: Aerodynamic derivatives of the RAE2822 aerofoil at the design point ($M = 0.734$, $C_l = 0.824$, $Re = 6.5 \cdot 10^6$, and $N_{\text{dv}} = 20$).

	C_{l_α}	C_{d_α}	C_{m_α}
Baseline	0.0806	0.01500	0.0022
Optimal	0.1241	0.01474	0.0021

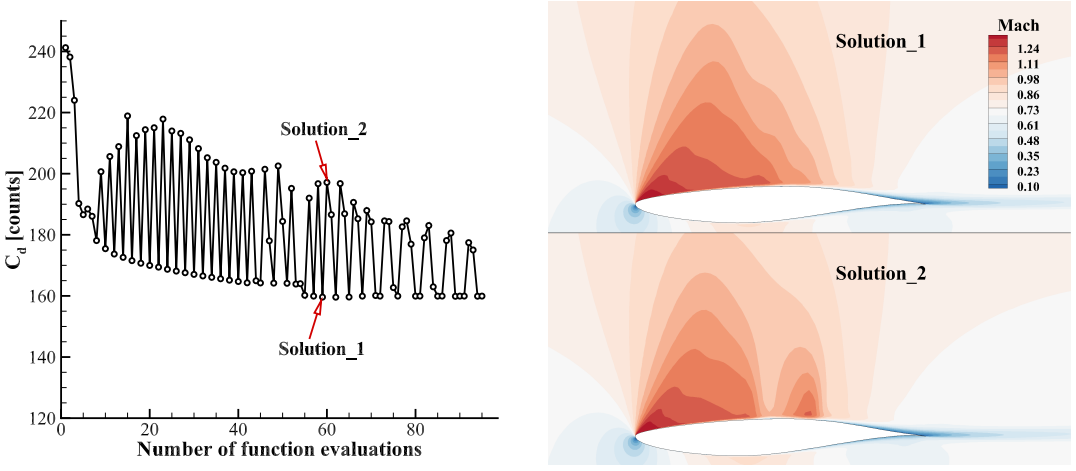
For this optimisation case, the convergence histories of several constraints are plotted in Figure 5.16. Firstly, the lift coefficient initially deviates from the value of 0.824 because the shape deformation is mainly performed in the first few design steps. The lift coefficient then gradually recovers to the target value and finally satisfies the constraint. Secondly, despite an initial decrease of the pitching moment coefficient, the optimisation seeks to generate aerofoils with higher C_m values in the following design cycles, which leaves more margin for this constraint. Thirdly, the aerofoil area basically remains the same value throughout the optimisation process, and hence does not violate the area constraint. Therefore, the optimisation provides a feasible design that meets the requirement of this benchmark problem.



(a) Lift (b) Pitching moment (c) Surface area

Figure 5.16: Convergence histories of several constraints in the RAE2822 optimisation using HHBF parameterisation method ($N_{\text{dv}} = 20$). Dashed line indicates the desired value for a specific constraint.

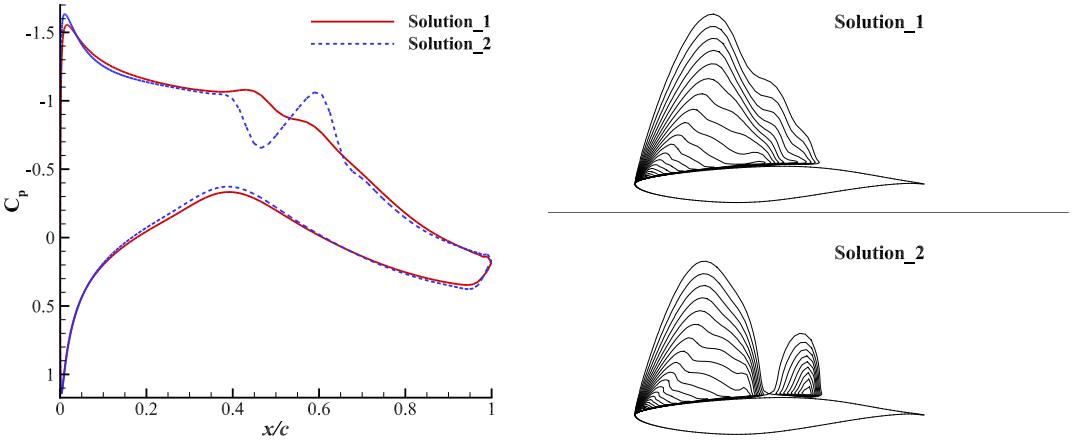
The convergence history of the objective function versus function evaluations is plotted in Figure 5.17(a). After decreasing significantly in the early few evaluation steps, the drag coefficient in the RAE2822 optimisation case starts to exhibit oscillatory behaviour, and designs with abruptly increased drag values are constantly generated, which is very similar to the NACA0012 case. Two representative designs with different drag values are then selected for result comparison. It was again found that the aerofoils have nearly identical geometries but the flow solutions are different. Figure 5.17(b) shows comparison of the Mach contour results for these two steady solutions. The discrepancy occurs in the mid-section area and is localised. One solution exhibits single pressure recovery feature, whereas the other solution displays double pressure recovery. Consequently, the presence of non-unique flow solutions results in a design space that is not smooth. This, in turn, causes difficulty for gradient-based optimisation to search for the local minimum.



(a) Convergence history (b) Mach contour comparison

Figure 5.17: Non-unique flow solutions for the RAE2822 optimised aerofoil using HHBF parameterisation method ($M = 0.734$, $C_l = 0.824$, $Re = 6.5 \cdot 10^6$, and $N_{\text{dv}} = 20$).

For the two selected flow solutions, chordwise pressure coefficient distributions on the aerofoil are plotted in Figure 5.18(a). The primary difference exists in the mid-sectional area where pressure recovery occurs. To compare the flowfield, the supersonic iso-Mach number lines are displayed in Figure 5.18(b). Solution_1 has a single supersonic zone, whereas solution_2 shows two supersonic zones, with a large one in the front section followed by a small one in the mid-section. The difference of flowfield structure contributes significantly to the drag production of the optimised aerofoil. The fact that two aerofoils with nearly identical geometries have very different drag values further explains the oscillations in the optimisation process, which have a negative effect on the convergence of the optimisation algorithm.



(a) Pressure coefficient distribution

(b) Supersonic iso-Mach number lines ($\Delta M = 0.02$)

Figure 5.18: Comparison of non-unique flow solutions for the RAE2822 optimised aerofoil ($M = 0.734$, $C_l = 0.824$, $Re = 6.5 \cdot 10^6$, and $N_{\text{dv}} = 20$).

Further investigations were undertaken where angle of attack, instead of Mach number, was chosen as the perturbation factor. In Figure 5.19, the hysteresis loop is shown for the RAE2822 optimised aerofoil. The angle of attack was swept up and down in a small range with an increment of 0.01, and each simulation was restarted from the previous converged solution. Since the design point for the optimised aerofoil ($\alpha = 3.1848$ deg) falls into the drag bifurcation zone, two distinct flow solutions are thus obtained. The flow solution with single pressure recovery feature was generated in downward sweep and corresponds to relatively lower drag; whereas the flow solution with double pressure recovery feature was produced in upward sweep and corresponds to relatively higher drag. The non-uniqueness in the RAE2822 optimisation case was also documented by Lee *et al.* [57] and LeDoux *et al.* [227]. However, the reference [57] hypothesized that the occurrence of non-unique solutions was triggered by the treatment of block interfaces of computational grid. The hypothesis is disproved in this study by finding a hysteresis loop around the design point. Nonetheless, the cause and mechanism for this phenomenon are not fully understood and thus require further study. The reader may also refer

to the work of Jameson *et al.* [257, 258] and Kuzmin [264, 265] for studies of similar phenomena, i.e. non-unique solutions in transonic flow over aerofoils with flat section.

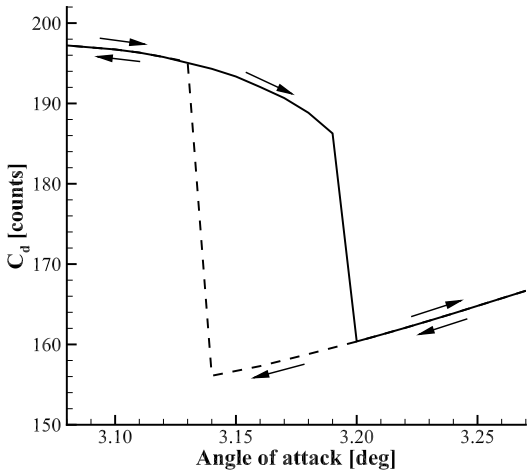


Figure 5.19: Drag coefficient hysteresis for angle of attack sweep analysis on the RAE2822 optimised aerofoil ($M = 0.734$, $Re = 6.5 \cdot 10^6$, and $N_{\mathrm{dv}} = 20$). Arrows indicate the sweep direction.

5.7 Chapter summary

In this chapter, the benchmark RAE2822 aerofoil optimisation case was investigated using SU2. Two shape parameterisation methods, HHBF and FFD control point approach, are employed in this study. For turbulence modelling, two sets of SA model coefficients are applied. The aim is to investigate the sensitivity of the optimal solution to a number of model parameters. The key findings of the numerical optimisations are summarised as follows:

1. **Optimised geometric shape**
   The aerofoil shape deformation primarily occurs in the fore section. Compared to the baseline geometry, a relatively flat surface with smaller curvature is created on the upper aerofoil to eliminate the shock, and the thickness on the lower side is increased to satisfy the area constraint.

2. **Sensitivity assessment**
   The optimal solution is insensitive to the parameter settings in both parameterisation methods, including bump function distribution, bump width control parameter and FFD box position. The drag coefficient for the optimised aerofoil is around 150 counts, which corresponds to approximately 38% of drag reduction.

   The influence of dimensionality on optimisation performance is very trivial as the same level of drag results are obtained using various numbers of design variables. Moreover, as few as 5 design variables are sufficient to fully cover the design space.

The optimisation results show sensitivity to SA turbulence model coefficients. The drag coefficient obtained from using calibrated SA model is lower than that obtained from using standard SA model. Consequently, the drag reduction is approximately one percent higher when using calibrated SA model.

3. **Non-uniqueness for optimal solution**
   Non-unique flow solutions were obtained for the optimised aerofoil, displaying both single and double pressure recovery features. Hysteresis loop of the drag coefficient was observed in angle of attack sweep analysis and a narrow drag bifurcation zone exists surrounding the design point.

Chapter 6

Three-dimensional Wing Twist Optimisation

6.1 Introduction

On top of 2D aerofoil optimisation, 3D wing shape optimisation is also of vital importance to improve the aerodynamic performance of an aircraft. A substantial amount of research has been conducted towards wing design and optimisation for both conventional and novel aircraft configurations. The primary goal is to minimise the drag while maintaining sufficient lift. In particular, induced drag, as known as vortex drag, accounts for approximately 40% of the total drag on a conventional aircraft in cruise flight [266]. Therefore, induced drag reduction remains a crucial area of research in the aerospace community.

While retrofitting an existing wing with winglets is a common practice to reduce the vortex drag, it also raises issues affecting other disciplines. Winglets incur a weight penalty that requires further structural reinforcement of the wing, ultimately adding more mass to the aircraft. The additional mass at the tip also creates more rolling inertia and hence reduces maneuverability. One more problem for short wings with winglets is that they are more prone to flutter. An alternative approach is to treat wing twist as a design parameter, which is a method to tailor the local incidence to achieve the ideal lift distribution over span. Given a specific wing planform, one can optimise the spanwise twist, which aims to reduce the lift-induced drag by shifting the aerodynamic loads inboard. In addition, the benefits are multi-fold by treating wing twist as a design variable and incorporating it into optimisation. Specifically, with the built-in wing twist in the jig shape, the aerodynamic forces can deform the wing into the desired shape in flight. The predefined wing twist can also be used to prevent tip stall through washout.

Regarding wing twist optimisation, a number of studies have been performed using aerodynamic solvers of different levels of fidelity. Phillips [267] used a low-fidelity model based on Prandtl's LLT to find the optimal design for finite twisted wings, and the optimised geometry was subsequently analysed with an Euler flow solver [268]. Similarly, for a BWB geometry, Qin *et al.* [79] used a low-fidelity panel method aerodynamic model for the inverse design optimisation. The specified lift distribution was achieved through the variation of spanwise twist distribution on the baseline geometry. A high-fidelity RANS model was later applied to investigate whether the targeted improvement was achieved. Nonetheless, the aerodynamic methods used above are limited by inviscid and incompressible flow assumptions. Recently, Hicken and Zingg [27] employed an Euler-based optimisation to minimise induced drag on several nonplanar configurations. A number of interesting results were produced. However, although the nonlinear impact on induced drag was considered, the viscous and turbulence effects were missing.

In this chapter, two wing twist optimisation problems are investigated. The first case is a benchmark problem suggested by AIAA ADODG. This case involves twist optimisation of an unswept NACA0012 wing in subsonic, inviscid flow. A number of researchers [27, 57, 122, 124, 125, 143] have studied this case using high-fidelity CFD solvers. The second case deals with twist optimisation of a swept NACA2412 wing in subsonic, viscous flow. The NACA2412 case exhibits more complexity than the NACA0012 case as it involves viscous and sweep effects on finite wings.

In this work, two aerodynamic solvers are used to perform gradient-based optimisations. The first solver is the high-fidelity SU2 code, and the second is the multi-fidelity FALCon solver. The primary aim is to demonstrate the applicability and efficiency of the hybrid solver in wing twist optimisation. In addition, we further investigate the sensitivity of optimisation results to a number of parameters, including design variable dimensionality, twist interpolation approach, and geometry control method.

This chapter is organised in the following manner. Section 6.2 and 6.3 present the results obtained for twist optimisation of NACA0012 wing and NACA2412 wing, respectively. For each case, the optimisation problem is briefly introduced, following by a description of the computational grid; a verification of the FALCon solver against SU2 is then provided for the baseline geometry; the optimisation results are finally presented, including the dimensionality study. Section 6.4 provides a discussion revisiting the geometric twist distribution based on the results from the two cases. Section 6.5 summarises the key findings in this chapter.

6.2 Case 1: Optimisation of an unswept wing in inviscid subsonic flow

6.2.1 Optimisation problem

The optimisation problem is the induced drag minimisation of an unswept wing with zero-thickness trailing-edge NACA0012 sections in inviscid, subsonic flow [27]. The freestream Mach number is 0.5, and the target lift coefficient is 0.375. The design variables are the twist of sections along the span and about the trailing edge. In this work, the twist at the root section is allowed to vary, whereas the freestream angle of attack is fixed under the flow conditions. The purpose of this case is to produce a lift distribution that is close to elliptical and a span efficiency factor approximately equal to unity. The optimisation problem is written as follows:

$$\begin{aligned} \text{Minimise:} \quad & C_D \\ \text{w.r.t:} \quad & \gamma\left(y\right) \\ \text{Subject to:} \quad & C_L = 0.375 \end{aligned}$$

where C_D and C_L are the drag and lift coefficients of the wing, respectively, and $\gamma\left(y\right)$ is the twist distribution along the wing span. The initial geometry is a rectangular, planar wing with NACA0012 sections and sharp trailing edge. The semispan is $3.06c$ where c is the chord length; the wing planform is rectangular over the first $3c$ and the last $0.06c$ is enclosed by a round wing-tip cap. Note that the wing-tip cap can not be modelled in FALCon due to its nature.

6.2.2 Computational grid

For the SU2 solver, an unstructured 3D grid was generated by *Pointwise* grid generator using the tetrahedral extrusion (T-Rex) technique. The computational domain is bounded by a no-penetration wing surface, a symmetry plane, and a farfield located at a distance of 20 chord lengths from the wing. The wing geometry as well as the grid on the symmetry plane are shown in Figure 6.1. The computational grid points are clustered towards the leading and trailing edge and also the wing tip to accurately represent the geometry. To establish grid convergence, four levels of grid were generated, and the key parameters are summarised in Table 6.1.

An inviscid flow analysis was conducted on the baseline geometry. Note that the angle of attack was iteratively updated during the flow analysis in order to meet the lift target. The aerodynamic results are listed in Table 6.2. The fine grid was found adequate to guarantee grid independent results and was used for optimisation.

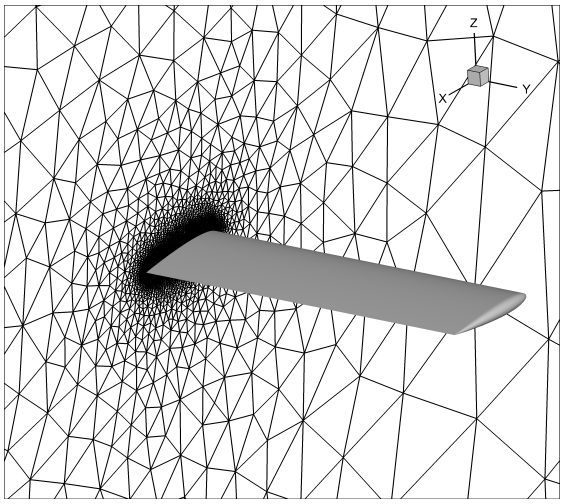


Figure 6.1: Case 1: SU2 computational grid for the unswept NACA0012 wing (Fine grid: 1.27 million elements).

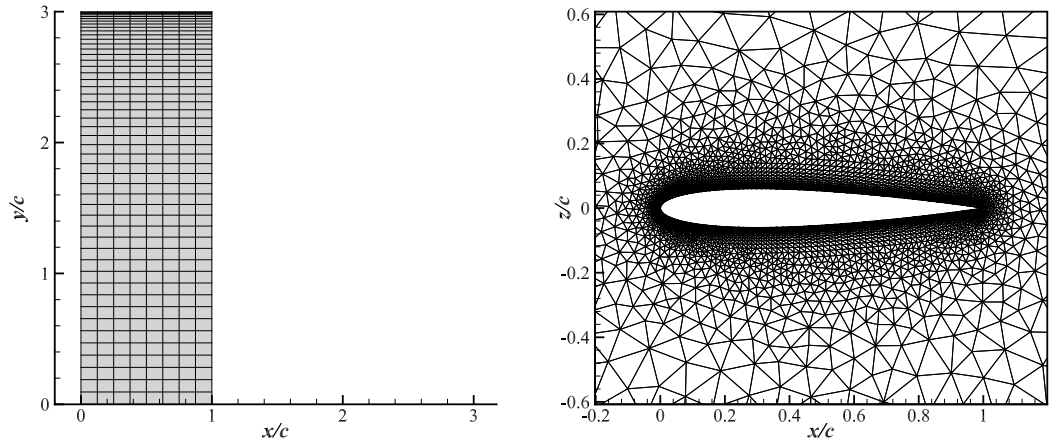
Table 6.1: Case 1: SU2 grid parameters for grid convergence study of the unswept NACA0012 wing.

Grid Level	Grid Elements	LE Spacing	TE Spacing	Off-wall Spacing
Coarse	468,620	$2.5 \cdot 10^{-3}$	$5.0 \cdot 10^{-3}$	$5.0 \cdot 10^{-3}$
Medium	858,099	$2.5 \cdot 10^{-3}$	$5.0 \cdot 10^{-3}$	$2.5 \cdot 10^{-3}$
Fine	1,269,840	$1.8 \cdot 10^{-3}$	$3.5 \cdot 10^{-3}$	$2.0 \cdot 10^{-3}$
Superfine	2,291,570	$1.2 \cdot 10^{-3}$	$2.5 \cdot 10^{-3}$	$1.5 \cdot 10^{-3}$

Table 6.2: Case 1: SU2 aerodynamic results for grid convergence study of the unswept NACA0012 wing ($M = 0.5$).

Grid Level	C_L	C_D (Counts)	α [deg]
Coarse	0.3750	87.5	4.2930
Medium	0.3750	82.6	4.2508
Fine	0.3750	81.7	4.2580
Superfine	0.3750	81.4	4.2611

For FALCon solver, a lattice of wing bound panels is defined and shown in Figure 6.2(a). The VLM panels are uniformly spaced in the chordwise direction and unevenly distributed in the spanwise direction using "cosine" function, which provides sufficient resolution at the wing tip. A grid convergence study was conducted and the results are listed in Table 6.3. For the following optimisation, the number of chordwise and spanwise panels used are $N_x = 8$ and $N_y = 50$, respectively. As the aerofoil section remains the same along the span, only one section is extracted to generate the database. As shown in Figure 6.2(b), the 2D stencil on symmetry plane of the 3D SU2 grid is used by the 2.5D+ solver. For the NACA0012 case, the 2.5D+ computation is reduced to a pure 2D analysis as the sweep angle is zero.



(a) VLM grid ($N_x = 8$ and $N_y = 50$)

(b) 2D grid for 2.5D+ solver

Figure 6.2: Case 1: FALCon computational grid for the unswept NACA0012 wing.

Table 6.3: Case 1: FALCon aerodynamic results for grid convergence study of the unswept NACA0012 wing ($M = 0.5$).

$N_x \times N_y$	C_L	C_D (Counts)	α [deg]
2 × 13	0.3750	80.6	4.2278
4 × 25	0.3750	82.1	4.2732
8 × 50	0.3750	82.9	4.3018
16 × 100	0.3750	83.3	4.3185

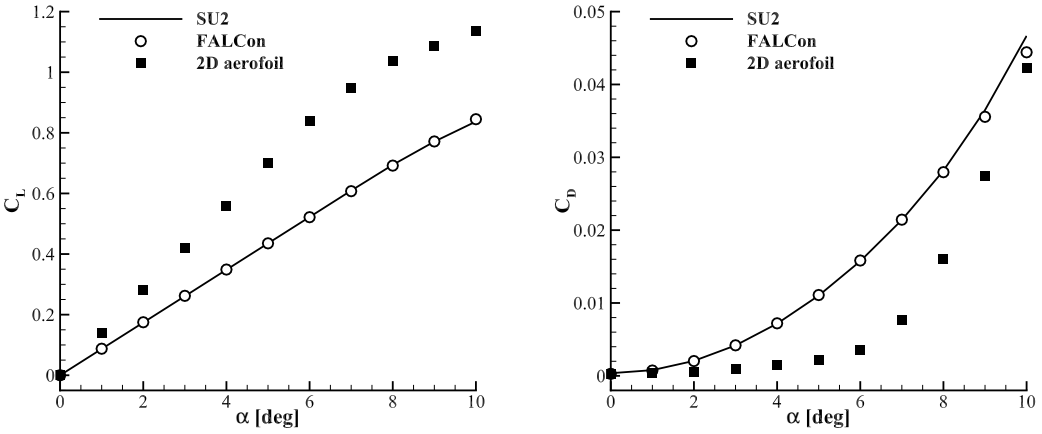
6.2.3 Verification

To assess the computational accuracy of aerodynamic results, the hybrid flow solver is verified against the SU2 solver for the baseline geometry. Figure 6.3 shows the lift and drag curves obtained from both solvers. The results obtained from a 2D simulation of the NACA0012 aerofoil is also included for comparison. It is apparent that FALCon can accurately predict the lift curve slope and also the drag of the wing. Note that the lift curve is linear at low angles of attack. The difference of lift curve slope between 2D and 3D results is caused by the finite span of the 3D wing. The lift curve slope for wings of finite span is given by

$$C_{L_\alpha} = \frac{C_{l_\alpha}}{1 + \frac{C_{l_\alpha}}{\pi\, e_1\, AR}} = \frac{C_{l_\alpha}}{1 + \frac{C_{l_\alpha}}{\pi\, AR}(1+\tau)} \tag{6.1}$$

where C_{L_α} and C_{l_α} are the 3D wing and 2D aerofoil lift curve slope, respectively; $AR = b^2/(2S)$ is the wing aspect ratio, b is the span, $S = cb/2 = 3c^2$ is the semispan area; e_1 is a factor that depends on the geometric shape of the wing, including the aspect ratio and taper ratio; τ is the lift curve slope parameter in Glauert's formulation [269]. In

this case, the small aspect ratio accounts for the reduction of lift curve slope from 2D to 3D, with a ratio of $C_{L_\alpha} / C_{l_\alpha} = 0.625$.



(a) Lift curve (b) Drag curve

Figure 6.3: Case 1: Lift and drag curves for baseline geometry ($M = 0.5$).

Distributions of the spanwise local lift for the various incidences are shown in Figure 6.4. It is observed that FALCon is able to reproduce the aerodynamic loading across a range of angles of attack. Overall, a good agreement is observed between FALCon and SU2 results, which confirms the prediction accuracy of aerodynamic properties for FALCon.

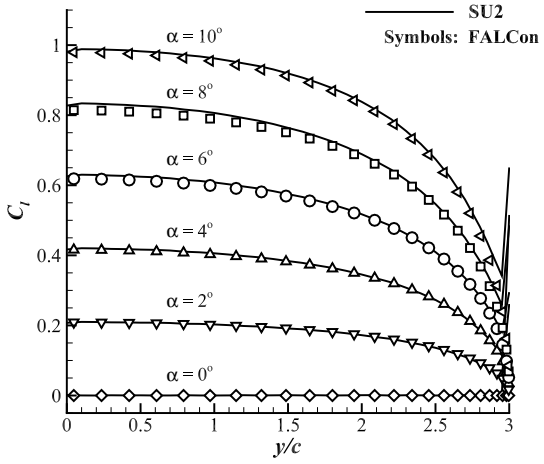


Figure 6.4: Case 1: Spanwise lift distributions for baseline geometry ($M = 0.5$).

6.2.4 Optimisation set-up

Both SU2 and FALCon are used to perform aerodynamic analysis and gradient-based optimisation for the unswept NACA0012 wing. The computational methods and numerical settings for SU2 and FALCon are summarised in Table 6.4 and Table 6.5, respectively. Note that the design variables are limited to the geometric twist, that is, through the

variation of the local aerofoil section incidence along the span while maintaining the aerofoil shape. It should also be mentioned that the discrete design variables are uniformly distributed along the wing span from root to tip. With regard to FALCon, a linear interpolation method, unless otherwise stated, is adopted to map design variables onto the twist angles of VLM panels.

Table 6.4: Case 1: Computational methods and numerical settings for SU2 optimisation.

Parameter	Methodology or Value
Geometry parameterisation	FFD twist
Grid deformation	Linear elasticity equations
Number of design variables	[3, 11]
Flow governing equations	3D Euler
Spatial discretisation	2nd order JST
Time discretisation	Euler implicit
CFL number	4
Optimiser	SLSQP
Gradient evaluation	Continuous adjoint
Scaling factor for objective function	0.001
Scaling factor for constraint function	0.001
Maximum number of iterations	100
Tolerance of KKT conditions	$1 \cdot 10^{-6}$

Table 6.5: Case 1: Computational methods and numerical settings for FALCon optimisation.

Parameter	Methodology or Value
Solver for 2.5D+ database	SU2
Governing equations	2D Euler
Range of angle of attack [deg]	[-5, 10]
Increment of incidence [deg]	1 deg
Coupling relaxation factor	0.5
Tolerance of coupling convergence	$1 \cdot 10^{-11}$
Number of design variables	[3, 11]
Twist interpolation method	Linear (unless otherwise stated)
Optimiser	SLSQP
Gradient evaluation	FDM
Scaling factor for objective function	0.001
Scaling factor for constraint function	0.001
Maximum number of iterations	100
Tolerance of KKT conditions	$1 \cdot 10^{-6}$

6.2.5 Dimensionality study

For both FALCon and SU2, the optimisations were conducted with different numbers of design variables. The drag coefficients of the baseline and optimised geometries are plotted in Figure 6.5. For the baseline geometry, the difference in drag between FALCon and SU2 solver is insignificant, only about 1.3 counts. This fact again reflects that the multi-fidelity solver can estimate drag accurately for this case with no viscous or sweep effects. With regard to the optimised geometries, the variation of drag coefficient is very trivial, indicating that the optimal solution is insensitive to the design variable dimensionality. In addition, the drag reduction for both optimisation frameworks is approximately 1 count, implying that the initial design is very close to the optimal solution.

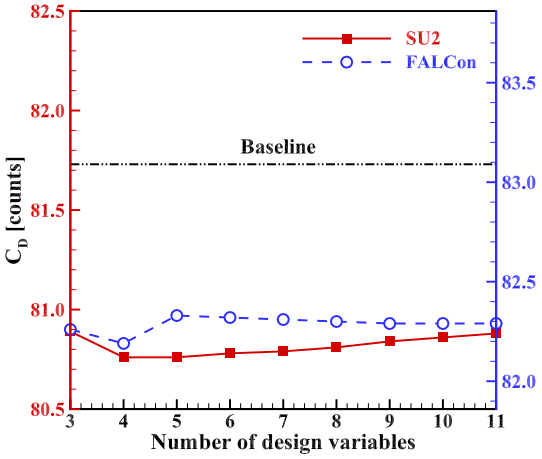


Figure 6.5: Case 1: Baseline and optimised drag coefficient.

When only vortex drag is present, the drag coefficient can be expressed as

$$C_D = \frac{C_L^2}{\pi\, AR\, e} = \frac{C_{D,\text{ellip}}}{e} \tag{6.2}$$

where $C_{D,\text{ellip}}$ is the minimum induced drag predicted by LLT for a planar wake; the parameter e is the span efficiency, which is a useful and popular means of comparing the induced drag of different configurations. Rearranging Equation (6.2) we have

$$e = \frac{C_L^2}{\pi\, AR\, C_D} = \frac{C_{D,\text{ellip}}}{C_D} \tag{6.3}$$

This suggests that the span efficiency is 1 for an ideal wing having the same aspect ratio and an elliptical lift distribution. As suggested in Reference [143], a modified span efficiency is used in this study, which is defined as

$$e_0 = \frac{C_L^2}{\pi\, AR\, (C_D - C_{D_0})} \tag{6.4}$$

where C_{D_0} is the zero-lift drag coefficient. The modified factor e_0, also known as Oswald efficiency, may provide a better estimate of the actual span efficiency by eliminating the spurious drag at $\alpha = 0$ deg, which is attributed to the numerics of the flow solver.

Figure 6.6 plots the span efficiency for the baseline and optimised wing geometries. It is observed that the span efficiency for the initial untwisted geometry is over 0.9. Both SU2 and FALCon optimisations yield about 1.2% improvement, pushing wing span efficiency closer to the theoretical optimal value of unity.

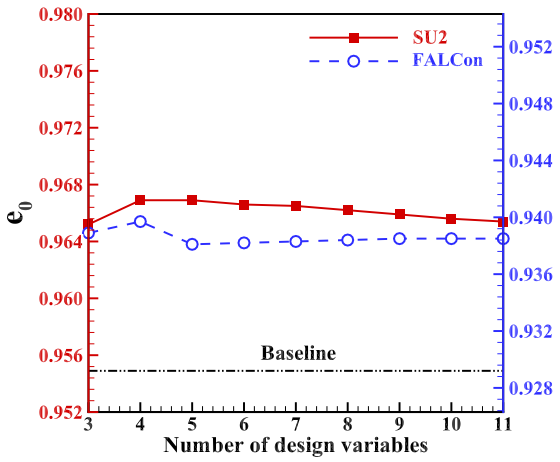


Figure 6.6: Case 1: Baseline and optimised span efficiency.

6.2.6 Optimisation results

The optimisation with 5 design variables is taken as example for further analysis. The convergence history of the objective function is plotted in Figure 6.7. It is evident that only a few design iterations are required to meet convergence criteria of KKT conditions, indicating that the optimiser can easily locate the minimum point for this optimisation problem.

The sectional lift distributions are shown in Figure 6.8. For the baseline geometry, the load distributions generated by the two aerodynamic solvers are indistinguishable, and they are close to elliptical, which is the theoretical optimal solution. As expected, the lift distributions on the optimised geometry closely match the elliptical distribution, and the sum of squared errors between the optimised result and the theoretical solution is within the value of $3.0 \cdot 10^{-4}$ for both solvers. The two optimisation frameworks perform equivalently well in minimising drag for this benchmark case.

The spanwise twist distributions for the optimised design are shown in Figure 6.9. Note that the twist values are measured relative to the freestream angle of attack. A similar pattern is observed from the two solutions obtained from FALCon and SU2. The outboard sections produce negative twist to reduce the vortex drag, whereas the inboard

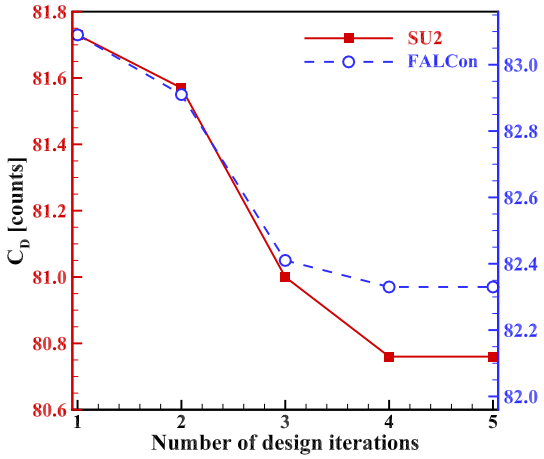
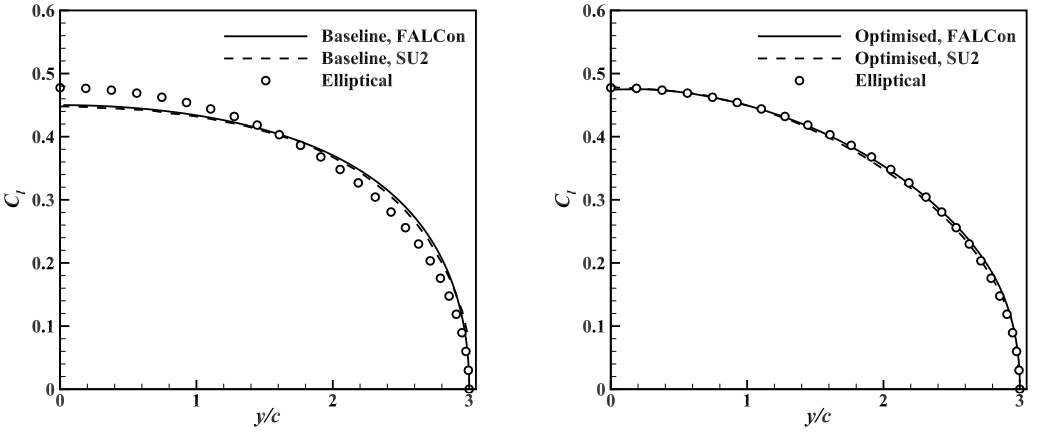


Figure 6.7: Case 1: Convergence history of the drag coefficient ($N_{\mathrm{dv}} = 5$).



(a) Baseline geometry

(b) Optimised geometry

Figure 6.8: Case 1: Sectional lift distributions for initial and final geometries ($M = 0.5$, $\alpha = 4.258$ deg, and $N_{\mathrm{dv}} = 5$).

sections generate positive twist to increase the sectional lift and thus satisfy the equality constraint for the total lift.

Figure 6.10 shows the FFD parameterisation, FFD box deformation as well as wing geometry deformation within SU2 optimisation. A lattice of control points are uniformly spaced on the surface of FFD box, and the control points located at the same spanwise position are grouped to perform the twist motion simultaneously. Consequently, the wing geometry encapsulated in the FFD box undergoes the twist deformation as well. The wing surface maintains second-order derivative continuity after deformation. Note that the twist at two ends of the FFD box are not defined as design variables in this optimisation case.

The pressure contours on the upper surface for the baseline and optimised wings are

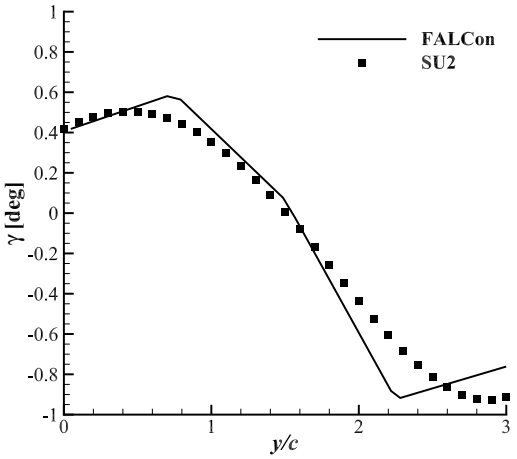


Figure 6.9: Case 1: Sectional twist distributions of the optimised geometry ($N_{\mathrm{dv}} = 5$).

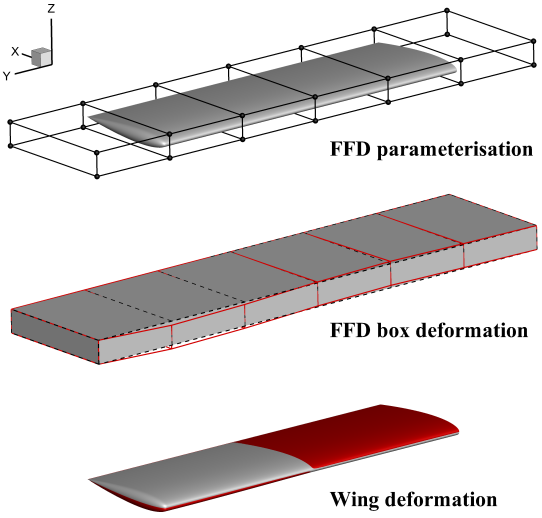


Figure 6.10: Case 1: FFD parameterisation, FFD box deformation, and wing geometry deformation ($N_{\mathrm{dv}} = 5$).

compared in Figure 6.11. To visualise the change more straightforwardly, Figure 6.12 shows the difference of pressure coefficient, ΔC_p, between the initial and optimised geometry. The variation of aerodynamic loading is mainly located at the leading edge area for both inboard and outboard sections. At the wing tip, the pressure coefficient is also different due to the decrease of vortex intensity after optimisation. The contour lines exhibit a nearly symmetric pattern about the midspan, which is due to the fact that the outboard wing generates less lift to reduce the vortex drag while the inboard wing produces more lift to satisfy the equality constraint of total lift.

The aerofoils and corresponding pressure distributions at three representative wing sections are plotted in Figure 6.13. The change of aerodynamic loading is clearly found at

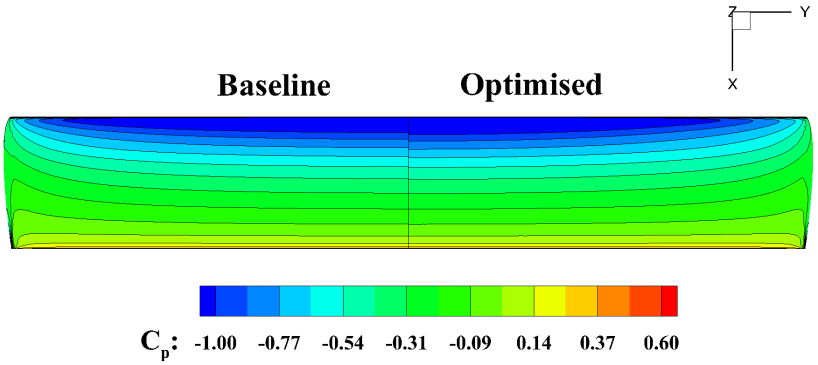


Figure 6.11: Case 1: Upper-surface pressure contours for baseline and optimised wings ($N_{\mathrm{dv}} = 5$).

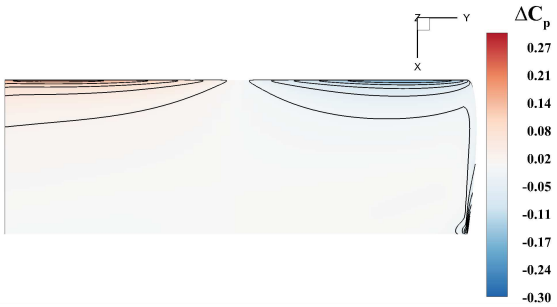


Figure 6.12: Case 1: Difference of pressure coefficient on upper surface between baseline and optimised wings ($N_{\mathrm{dv}} = 5$).

20% and 80% span positions by comparing the included area of pressure distribution, whereas the loading at midspan position remains nearly the same. To further illustrate the mechanism for drag reduction, we extracted a 2D plane of the flowfield, which is located $0.1c$ behind the trailing edge, for analysis. The vorticity magnitude contours, as shown in Figure 6.14, are compared for the baseline and optimised designs. It is evident that the peak value at the vortex core decreases through twist optimisation, leading to the reduction of energy loss and hence the vortex drag.

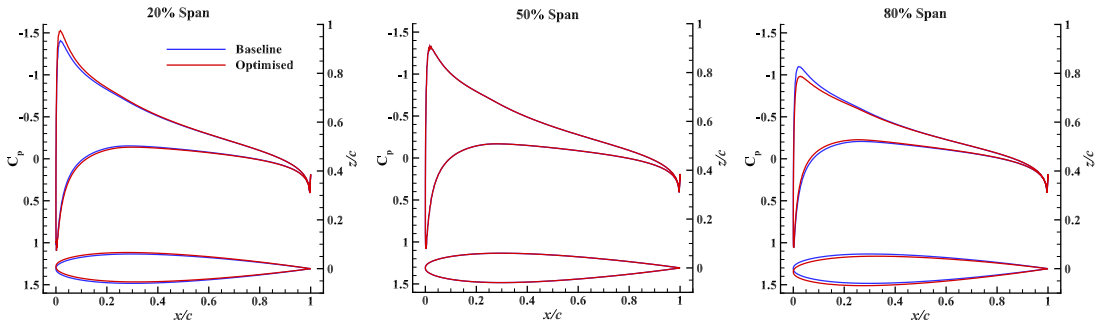


Figure 6.13: Case 1: Sectional pressure plots and aerofoil sections for baseline and optimised wings ($N_{\mathrm{dv}} = 5$).

In terms of computational efficiency, the CPU time for the dimensionality study is

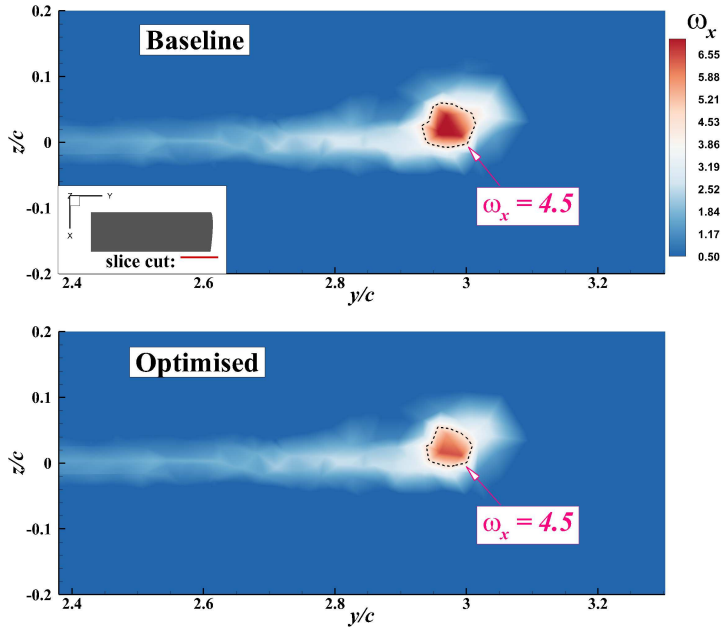


Figure 6.14: Case 1: Vorticity magnitude contours on extracted plane ($x/c = 1.1$) of the flowfield ($N_{\mathrm{dv}} = 5$).

recorded in Table 6.6. As of FALCon, the 2.5D+ database was obtained at a range of angles of attack from -5 to 10 deg with an increment of 1 deg. Note that the database is pre-computed only once and used thereafter for all cases. The computational time for optimisation increases gradually with more design variables due to the gradient evaluation using FDM. It is worth noting that all optimisations complete within 1 minute, demonstrating high efficiency of the multi-fidelity approach. With respect to SU2, the computational cost does not scale with the number of design variables due to the merit of the adjoint method for efficiently computing the gradient. Nonetheless, a single optimisation costs over 200 CPU hours to converge. Overall, FALCon costs only a fraction of computational time compared with that of SU2, achieving a speed-up of two orders of magnitude. Furthermore, with the 2.5D+ database at hand, one can run a parametric study within a short space of time.

6.2.7 Case summary

The twist optimisation of the NACA0012 wing was investigated using both SU2 and FALCon solvers. The aerodynamic results obtained from the hybrid solver are verified against SU2 on the baseline geometry. Both optimisation frameworks perform equally well for the benchmark case, and an elliptical loading distribution is recovered.

The optimal drag result is insensitive to the number of design variables, and approximately 1 drag count reduction is achieved. The outboard sections exhibit negative twist to reduce the strength of tip vorticity and hence the vortex drag; the inboard sections

Table 6.6: Case 1: Computational time

N_{dv}	FALCon		SU2
	2.5D+ Database (CPU hour)	Optimisation (second)	(CPU hour)
3	1.2 (only once)	13.5	220
4		19.8	235
5		23.3	235
6		25.6	235
7		28.7	235
8		32.9	235
9		38.3	235
10		39.3	235
11		41.1	235

show positive twist to maintain the required lift. In terms of computational cost, FALCon shows a much higher efficiency compared to SU2, demonstrating its potential in early stages of aircraft design and optimisation.

6.3 Case 2: Optimisation of a swept wing in viscous subsonic flow

6.3.1 Optimisation problem

The second optimisation problem is the drag minimisation of a swept wing with NACA2412 sections in viscous, subsonic flow. The freestream Mach number is 0.5, the freestream angle of attack is 3 deg, and the Reynolds number is 5.6 million based on the chord length. The lift coefficient is constrained to be equal to the initial value. The design variables are the twist of sections along the span and about the trailing edge. The twist at the root section is allowed to vary, whereas the freestream angle of attack is kept constant. Compared to the first optimisation problem, this case uses an unsymmetrical aerofoil section and considers the sweep and viscous effects on the finite wing. The purpose of this case is to minimise the induced drag out of the total drag, and to further demonstrate the applicability of FALCon. The optimisation problem is summarised as follows:

$$\begin{aligned} \text{Minimise:} \quad & C_D \\ \text{w.r.t:} \quad & \gamma\,(y) \\ \text{Subject to:} \quad & C_L = C_{L,\,\text{baseline}} \end{aligned}$$

The initial geometry is a swept, planar wing with NACA2412 sections and sharp trailing edge. The sweep angle, Λ, is 30 deg; the semispan is $5c$; and the wing planform shape is a parallelogram. Note that the wing tip in this case has a sharp side edge.

6.3.2 Computational grid

With respect to SU2, an unstructured 3D grid was generated using the T-Rex technique. The computational domain is bounded by a no-slip wing surface, a symmetry plane, and a farfield located at a distance of 100 chord lengths from the wing. The wing geometry as well as the grid on the symmetry plane are shown in Figure 6.15. The computational grid points are clustered towards the leading and trailing edge and also the wing tip to accurately represent the geometry. A sufficient number of anisotropic layers were generated surrounding the wing body to capture the boundary layer. The wall distance for the first grid layer was set to ensure that y^+ is below 1. To establish grid convergence, four levels of grid were generated, and the key parameters are summarised in Table 6.7.

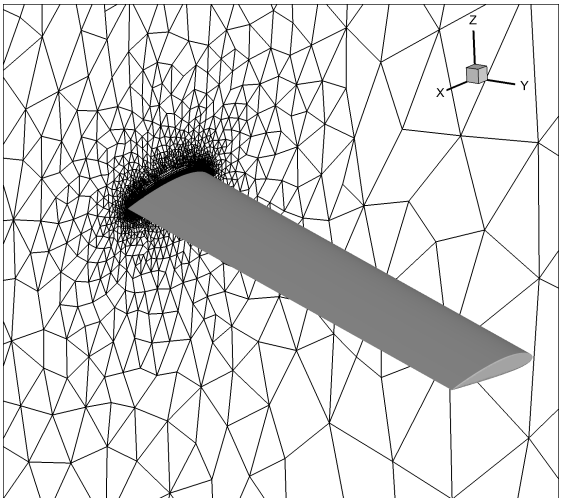


Figure 6.15: Case 2: SU2 computational grid for the swept NACA2412 wing (Fine grid: 3.58 million elements).

Table 6.7: Case 2: SU2 grid parameters for grid convergence study of the swept NACA2412 wing.

Grid Level	Grid Elements	LE Spacing	TE Spacing	Off-wall Spacing
Coarse	922,373	$2.0 \cdot 10^{-3}$	$5.0 \cdot 10^{-3}$	$4.0 \cdot 10^{-6}$
Medium	1,816,963	$1.4 \cdot 10^{-3}$	$5.0 \cdot 10^{-3}$	$2.8 \cdot 10^{-6}$
Fine	3,578,365	$1.0 \cdot 10^{-3}$	$5.0 \cdot 10^{-3}$	$2.0 \cdot 10^{-6}$
Superfine	6,997,485	$7.1 \cdot 10^{-4}$	$2.5 \cdot 10^{-3}$	$1.4 \cdot 10^{-6}$

The compressible RANS simulation was carried out using the SA turbulence model, and the aerodynamic results are listed in Table 6.8. From the preliminary study, the superfine grid was found adequate to guarantee grid independent results and was thus used for optimisation.

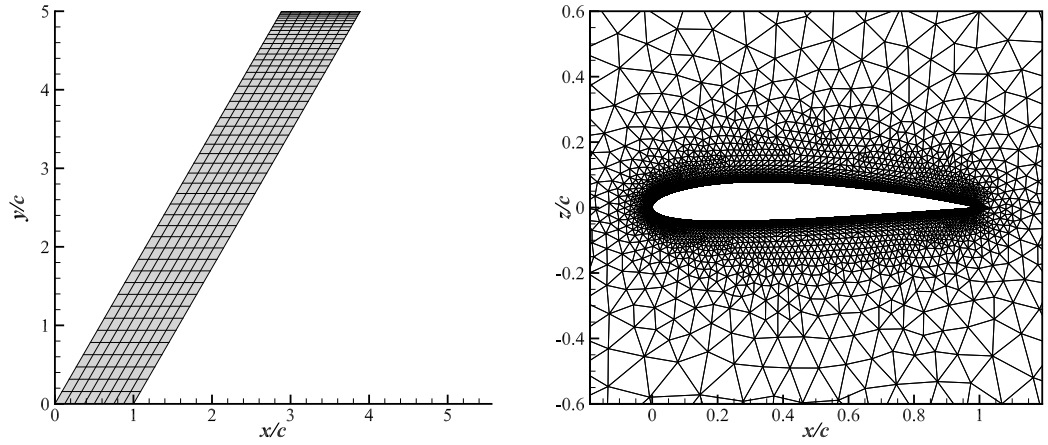
A structured grid of wing bound panels for the FALCon solver is defined and shown in Figure 6.16(a). The VLM panels are uniformly spaced in the chordwise direction and unevenly distributed in the spanwise direction using “cosine” function, aiming to

Table 6.8: Case 2: SU2 aerodynamic results for grid convergence study of the swept NACA2412 wing ($M = 0.5$, $\alpha = 3$ deg, $Re = 5.6 \cdot 10^6$).

Grid Level	C_L	C_D (Counts)
Coarse	0.4470	176.6
Medium	0.4518	176.8
Fine	0.4424	169.8
Superfine	0.4363	165.1

provide sufficient modelling resolution at the wing tip. A grid convergence study was conducted and the results are listed in Table 6.9. For the following optimisation, the number of chordwise and spanwise panels used are $N_x = 8$ and $N_y = 50$, respectively. As the aerofoil section remains the same along the span, only one section is extracted to generate the database. In this work, the 2D stencil on symmetry plane of the 3D SU2 grid, as shown in Figure 6.16(b), is used. As the 2.5D+ solver is based on the local FoR, a scaling transformation between the two FoRs needs to be performed for the 2D grid using the following expression:

$$\frac{z'}{z} = \frac{1}{\cos\Lambda} \tag{6.5}$$



(a) VLM grid ($N_x = 8$ and $N_y = 50$)

(b) 2D grid for 2.5D+ solver

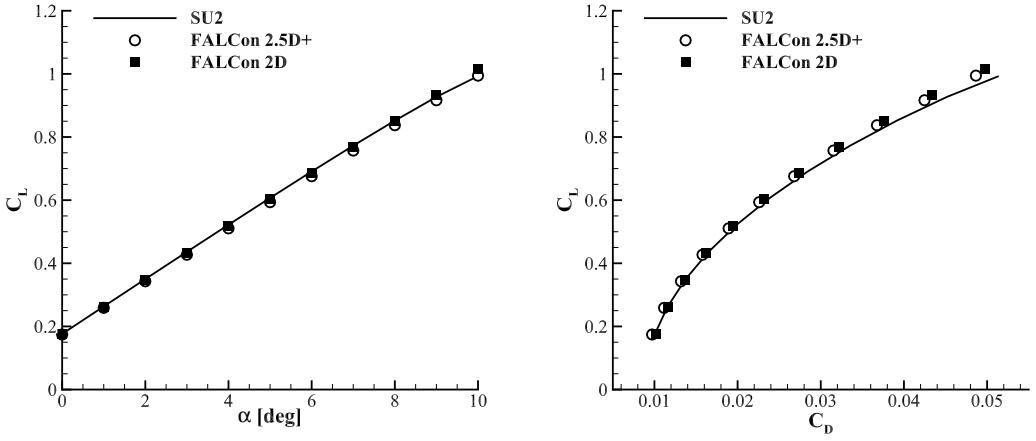
Figure 6.16: Case 2: FALCon computational grid for the swept NACA2412 wing.

Table 6.9: Case 2: FALCon aerodynamic results for grid convergence study of the swept NACA2412 wing ($M = 0.5$, $\alpha = 3$ deg, $Re = 5.6 \cdot 10^6$).

$N_x \times N_y$	C_L	C_D (Counts)
2×13	0.4341	134.9
4×25	0.4296	134.7
8×50	0.4271	134.6
16×100	0.4256	134.5

6.3.3 Verification

In terms of aerodynamic quantities, the hybrid flow solver is verified against SU2 for the baseline geometry. Figure 6.17 shows the lift curve and drag polar obtained from both solvers. With respect to FALCon, the 2.5D+ database, which incorporates the crossflow effect, was obtained from an ISW model. The information related to the sweep angle, i.e. $\Lambda = 30$ deg, is introduced as an appropriate boundary condition and imposed at the farfield on a 2D stencil. For the purpose of comparison, a 2D database was also generated and used, which does not consider the sweep effect. It is demonstrated that FALCon can accurately predict the total lift coefficient using either the 2.5D+ or the 2D database. From the drag polar, a satisfactory agreement is found between FALCon results and SU2 data. Minor difference exists at higher angles of attack, where FALCon underpredicts the drag coefficient compared to SU2 solver. This is possibly due to that the flow gradually loses homogeneity in the y' direction when the angle of attack becomes large. It is worth noting that FALCon generates very similar results using either 2.5D+ or 2D database, as the flow conditions are moderate and hence the sweep effect does not have much influence on the integrated aerodynamic quantities.



(a) Lift curve (b) Drag polar

Figure 6.17: Case 2: Lift curve and drag polar for baseline geometry ($M = 0.5$, $Re = 5.6 \cdot 10^6$).

The spanwise lift distributions for a range of angles of attack are shown in Figure 6.18. Overall, a good agreement is observed, except that there is minor difference near the wing tip. The SU2 results show an abrupt rise at the wing tip, particularly for cases with large angles of attack. This is caused by the flow acceleration on the upper surface due to the tip vortex. However, this phenomenon can not be captured by the multi-fidelity aerodynamic model. In terms of 3D physics, the VLM solver only considers the downwash velocity in the spanwise direction, which in turn affects the effective angle of attack for the 2.5D+ database. As such, the surface flow variations are not properly modelled by the FALCon solver, which limits its prediction accuracy. Moreover, as

shown in Figure 6.18, the discrepancy near the wing tip becomes larger when increasing the freestream angle of attack. This fact is due to the increased vortex intensity and hence the pronounced 3D flow physics.

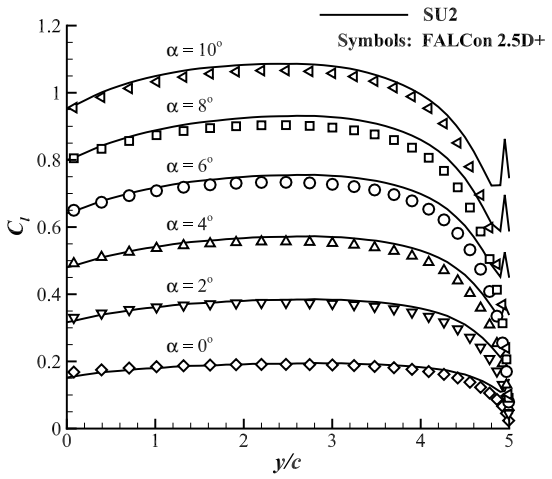


Figure 6.18: Case 2: Spanwise lift distributions for baseline geometry ($M = 0.5$, $Re = 5.6 \cdot 10^6$).

6.3.4 Optimisation set-up

Two aerodynamic solvers, SU2 and FALCon, are used to perform aerodynamic analysis and gradient-based optimisation for the swept NACA2412 wing. The computational methods and numerical settings with respect to SU2 and FALCon are summarised in Table 6.10 and Table 6.11, respectively. It should be noted that the twist design variables are uniformly distributed along the wing span from root to tip. A linear interpolation method, unless otherwise stated, is adopted by FALCon to map the discrete design variables onto the twist angles of VLM panels.

6.3.5 Dimensionality study

The optimisations were conducted using both SU2 and FALCon with different numbers of design variables. The drag results of the dimensionality study are plotted in Figure 6.19. For both optimisation frameworks, the optimal solution is nearly insensitive to the number of design variables, and approximately 5.5% drag reduction is achieved.

A full drag breakdown for both solvers is provided in Table 6.12. Note that two ways of drag decomposition are used. In the SU2 solver, a near field analysis is employed, and the total drag is computed by integrating the pressure and the friction drag around the body. In this way the induced drag is included in the pressure drag. Regarding the FALCon solver, as stated earlier in Section 3.3.4, a far field analysis is used and the total drag is decomposed into the viscous and the induced drag. It is apparent that, through

Table 6.10: Case 2: Computational methods and numerical settings for SU2 optimisation.

Parameter	Methodology or Value
Geometry parameterisation	FFD twist
Grid deformation	Linear elasticity equations
Number of design variables	[3, 11]
Flow governing equations	3D RANS
Turbulence model	SA
Spatial discretisation	2nd order JST
Time discretisation	Euler implicit
CFL number	4
Optimiser	SLSQP
Gradient evaluation	Continuous adjoint
Scaling factor for objective function	0.001
Scaling factor for constraint function	0.001
Maximum number of iterations	100
Tolerance of KKT conditions	$1 \cdot 10^{-6}$

Table 6.11: Case 2: Computational methods and numerical settings for FALCon optimisation.

Parameter	Methodology or Value
Solver for 2.5D+ database	DLR-Tau
Governing equations	2D RANS
Turbulence model	SA
Range of angle of attack [deg]	[-5, 10]
Increment of incidence [deg]	1
Coupling relaxation factor	0.5
Tolerance of coupling convergence	$1 \cdot 10^{-11}$
Number of design variables	[3, 11]
Twist interpolation method	Linear (unless otherwise stated)
Optimiser	SLSQP
Gradient evaluation	FDM
Scaling factor for objective function	0.001
Scaling factor for constraint function	0.001
Maximum number of iterations	100
Tolerance of KKT conditions	$1 \cdot 10^{-6}$

twist optimisation, the viscous drag in FALCon (or friction drag in SU2) remains nearly unchanged, whereas the induced drag in FALCon (or pressure drag in SU2) undergoes a reduction, revealing the fact that the optimisation seeks to reduce the induced drag out of the total drag.

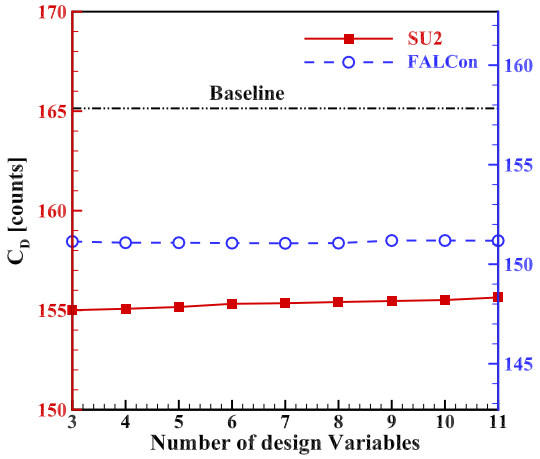


Figure 6.19: Case 2: Baseline and optimised drag coefficient.

Table 6.12: Case 2: Drag breakdown (unit in drag counts)

N_{dv}	FALCon				SU2			
	Viscous drag		Induced drag		Friction drag		Pressure drag	
	Init.	Opt.	Init.	Opt.	Init.	Opt.	Init.	Opt.
3	93.0	93.4	64.8	57.7	66.0	66.7	97.4	86.8
4		93.4		57.7		66.7		86.8
5		93.4		57.7		66.7		86.9
6		93.4		57.6		66.7		87.0
7		93.4		57.6		66.8		87.0
8		93.4		57.7		66.8		87.0
9		93.4		57.8		66.8		87.1
10		93.4		57.8		66.9		87.1
11		93.4		57.8		66.9		87.2

6.3.6 Optimisation results

The optimisation with 6 design variables is chosen for further analysis. The convergence history of the objective function is plotted in Figure 6.20. It is apparent that the optimal solution is reached within a few design iterations, indicating the easiness of locating the minimum for this optimisation problem.

The sectional lift distributions for baseline and optimised geometries are shown in Figure 6.21. As expected, the spanwise loading shifts inboard after optimisation, minimising the induced drag at the wing tip. Note that there is no theoretical optimal solution for this case. It is also observed that the optimised lift distributions obtained from SU2 and FALCon show a similar pattern; the minor difference is possibly due to the different geometry control methods that are used.

Figure 6.22 shows the spanwise twist distributions. A similar pattern is found between the two sets of results. Positive twist occurs for the inboard sections while negative twist

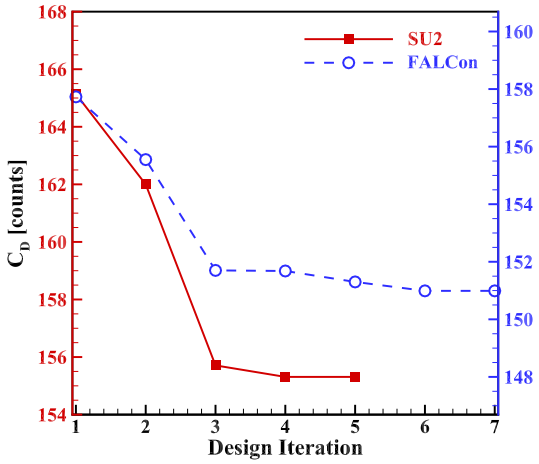


Figure 6.20: Case 2: Convergence history of the drag coefficient ($N_{\mathrm{dv}} = 6$).

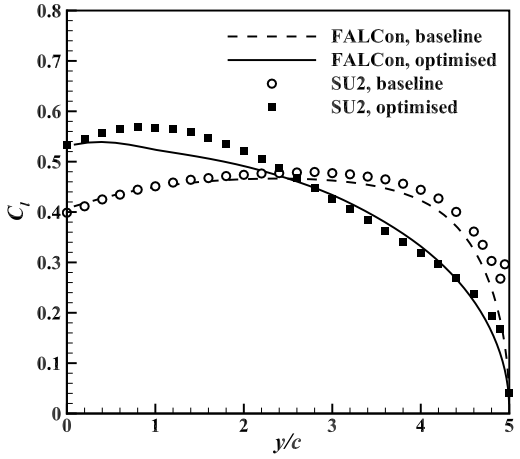


Figure 6.21: Case 2: Sectional lift distributions for initial and final geometries ($M = 0.5$, $\alpha = 3$ deg, $Re = 5.6 \cdot 10^6$, and $N_{\mathrm{dv}} = 6$).

for the outboard sections. Figure 6.23 gives the illustration of FFD parameterisation, FFD box perturbation and wing deformation in SU2 optimisation. It is worth noting that the optimisation strategy is basically the same for both NACA0012 and NACA2412 optimisation cases.

As shown in Figure 6.24, the pressure contours on the upper surface are compared for the initial and optimised geometries. It is apparent that the aerodynamic loading moves inboard towards the wing root after twist optimisation. For better visualisation effect, the difference of pressure coefficient on the wing surface is displayed in Figure 6.25. As is the case with NACA0012 optimisation, the pressure difference is primarily concentrated at the leading edge region for both inboard and outboard sections. The variation of tip vortex also causes some difference at the wing tip.

Figure 6.26 shows the aerofoil sections and pressure distributions at three characteristic

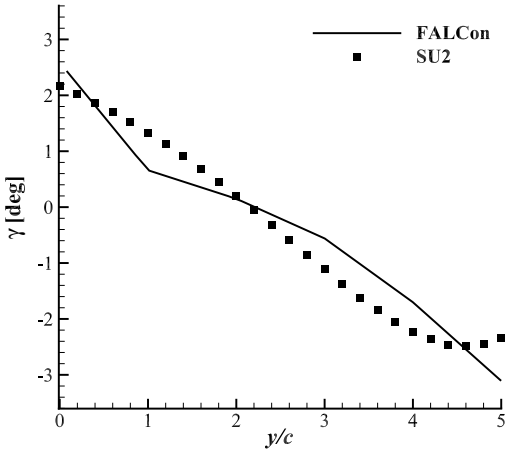


Figure 6.22: Case 2: Sectional twist distributions of the optimised geometry ($N_{\mathrm{dv}} = 6$).

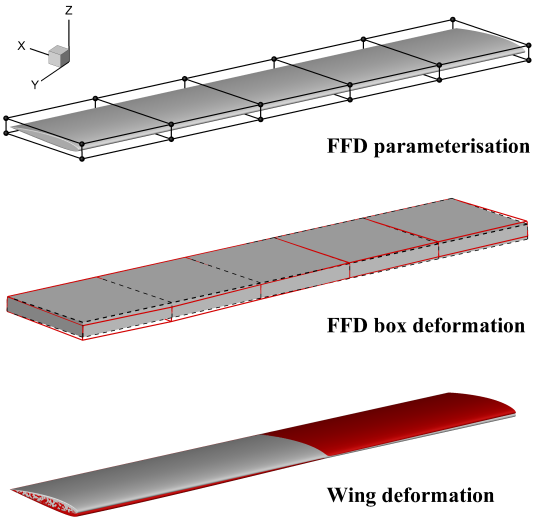


Figure 6.23: Case 2: FFD parameterisation, FFD box deformation, and wing geometry deformation ($N_{\mathrm{dv}} = 6$).

spanwise stations. The change of sectional twist angle as well as aerodynamic loading is obviously found at 20% and 80% span, whilst the midspan section is almost unmoved. In addition, two representative slices of the flowfield are extracted near the trailing edge, and the vorticity contours are compared, as shown in Figure 6.27, between the baseline and optimised wing. It is observed on these two slices that the peak vortex intensity decreases and the iso-contour line of vorticity magnitude shrinks, indicating the reduction of the vortex drag by twist optimisation.

To compare the computational efficiency between two optimisation frameworks, the CPU time for the dimensionality study is recorded in Table 6.13. The 2.5D+ database for FALCon is obtained at a range of angles of attack from -5 to 10 deg with an

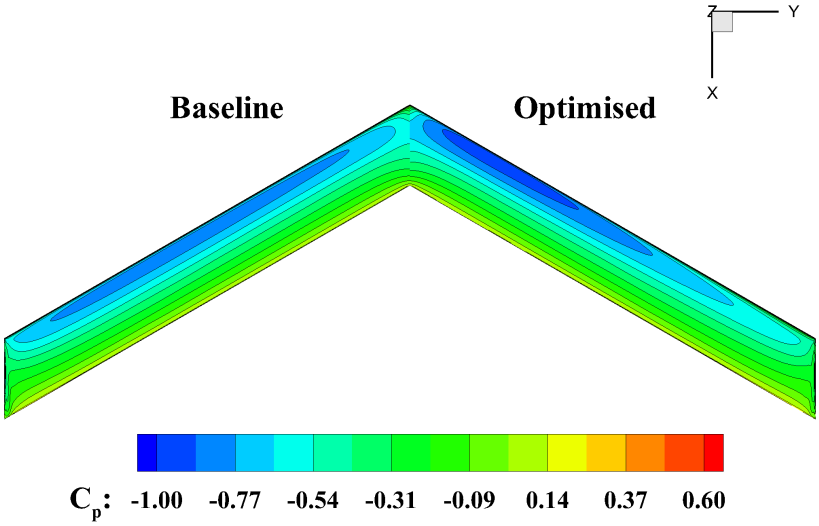


Figure 6.24: Case 2: Upper-surface pressure contours for baseline and optimised wings ($N_{\mathrm{dv}} = 6$).

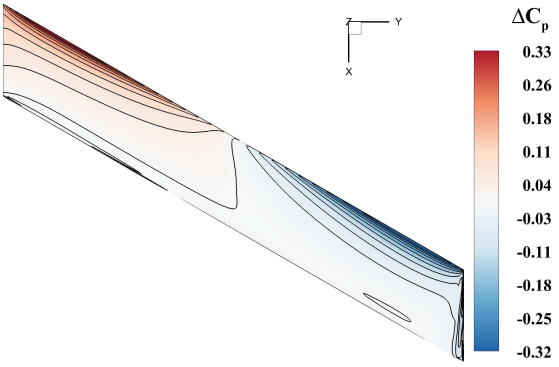


Figure 6.25: Case 2: Difference of pressure coefficient on upper surface between baseline and optimised wings ($N_{\mathrm{dv}} = 6$).

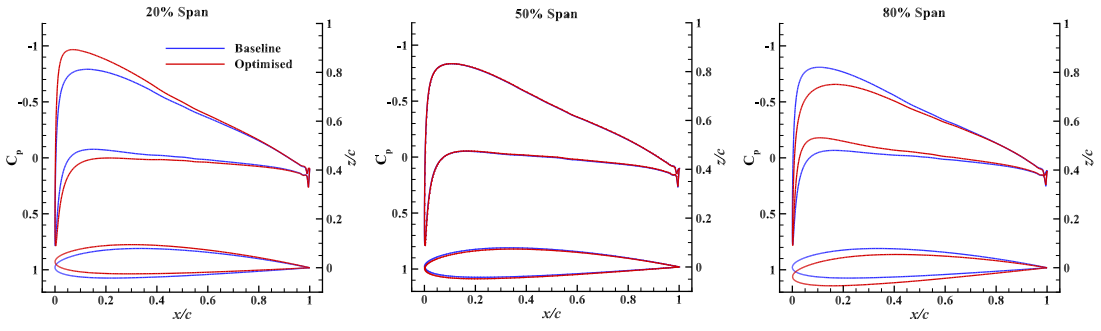


Figure 6.26: Case 2: Sectional pressure plots and aerofoil sections for baseline and optimised wings ($N_{\mathrm{dv}} = 6$).

increment of 1 deg. It takes just over two hours for the computation of the database and just one minute for the optimisation. It is apparent that the FALCon optimisations with 6 to 11 design variables have much more computational cost than other cases,

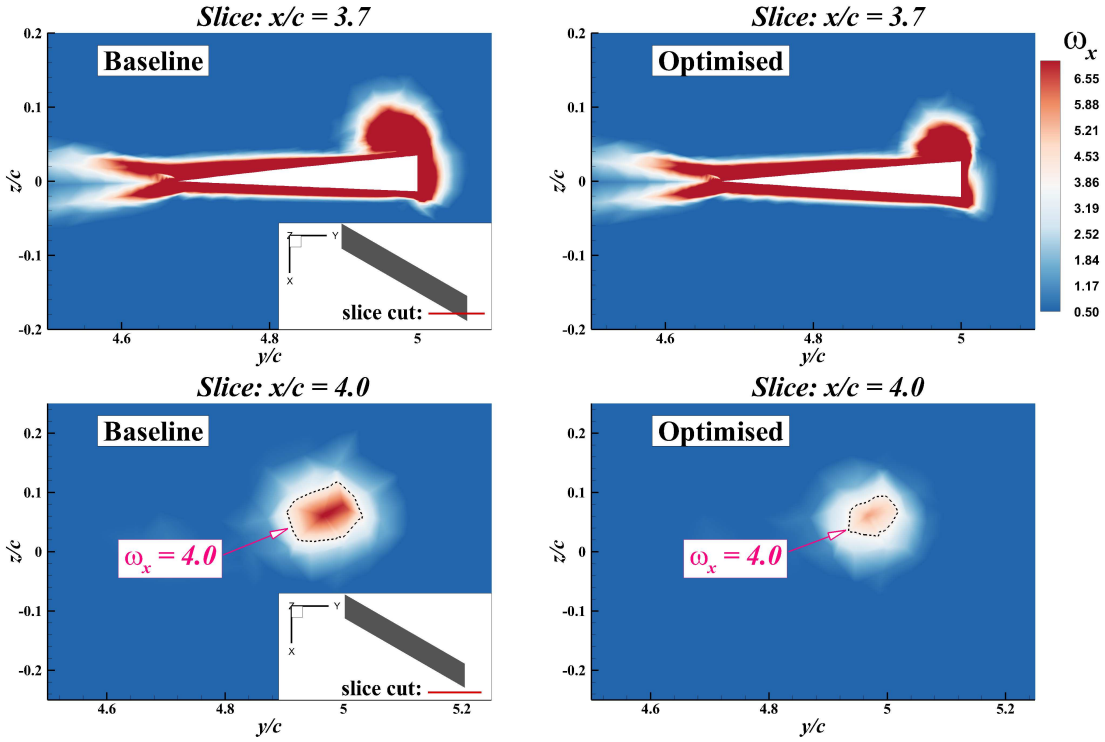


Figure 6.27: Case 2: Vorticity magnitude contours on extracted planes (x/c = 3.7 and x/c = 4.0) of the flowfield (N_{dv} = 6).

which is due to two factors. First, the optimisations, particularly with 6 to 8 design variables, have a relatively larger number of function and gradient evaluations. Second, the number of FALCon computations for gradient evaluation scales with the number of design variables as FDM is used herein. Similarly, the SU2 optimisation cases with 4 to 9 design variables exhibit higher computational cost, which is primarily due to the relatively larger number of function calls in these cases. Note that nearly 3000 CPU hours is required for SU2 optimisation, which is a significant amount of computational effort even with parallel computational capability. Overall, FALCon provides a physically consistent solution when compared to SU2, but reduces the computational time from days to only two hours, demonstrating its high efficiency and suitability for preliminary aircraft design. Moreover, once the 2.5D+ database is available for a specific wing, one can perform further studies within seconds, such as estimating the aerodynamic loads for aeroelasticity analysis.

6.3.7 Case summary

The twist optimisation of the NACA2412 wing was investigated using both the SU2 and FALCon solvers. For the finite swept wing, the computational accuracy of the multi-fidelity solver for predicting aerodynamic loads is verified against SU2. Both optimisation frameworks perform equivalently well in this case, achieving drag reduction of around 5.5%.

Table 6.13: Case 2: Computational time and number of evaluations

N_{dv}	FALCon				SU2		
	2.5D+ Database (CPU hour)	Optimisation (second)	F.C. (-)	G.C. (-)	Optimisation (CPU hour)	F.C. (-)	G.C. (-)
3	2.25 (only once)	13.8	6	6	2202	12	4
4		19.2	7	6	2937	16	5
5		22.9	7	6	2937	16	5
6		51.5	13	12	2937	16	5
7		60.1	13	13	2937	16	5
8		67.2	13	13	2937	16	5
9		40.7	8	7	2937	16	5
10		44.7	8	7	2202	12	4
11		48.2	8	7	2202	12	4

F.C. – Function Calls; G.C. – Gradient Calls.

The final drag result shows insensitivity to the number of design variables. The drag reduction mechanism is same as the NACA0012 case, which is to shift the aerodynamic loading from outboard sections to inboard sections and thus reduce the vorticity at wing tip. With regard to computational efficiency, the advantage of FALCon over SU2 is more apparent than the NACA0012 case, reducing the CPU hours from thousands to only a few.

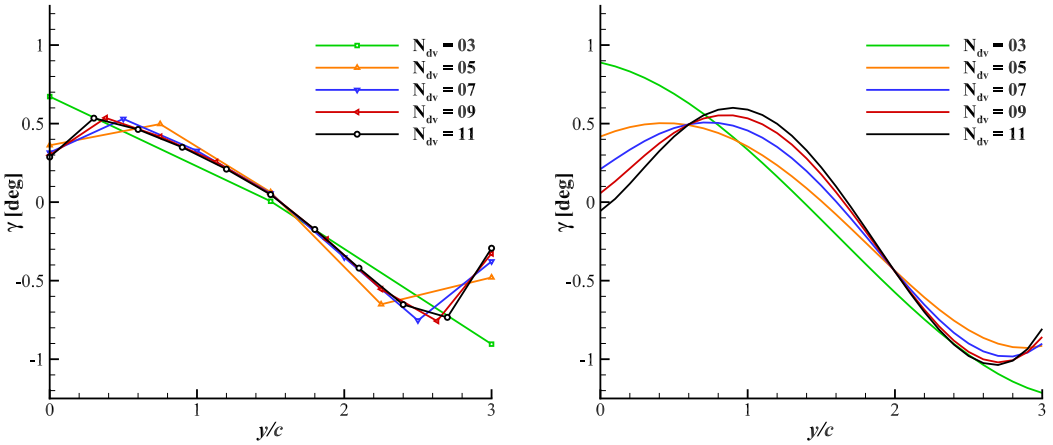
6.4 Sensitivity assessment for twist distribution

Geometry parameterisation and the set-up of design variables play an important role in an ASO framework. The impact of shape parameterisation and its numerical settings on AIAA ADODG benchmark problems is discussed in References [123, 226]. Recall that, in this context, the geometric twist distribution of the wing is defined as the design variable. In the scenario of aircraft design, this information is directly transferred from aerodynamic engineers to structural and manufacturing departments. It is thus worth revisiting the twist distributions for the two optimisation cases.

6.4.1 Impact of dimensionality

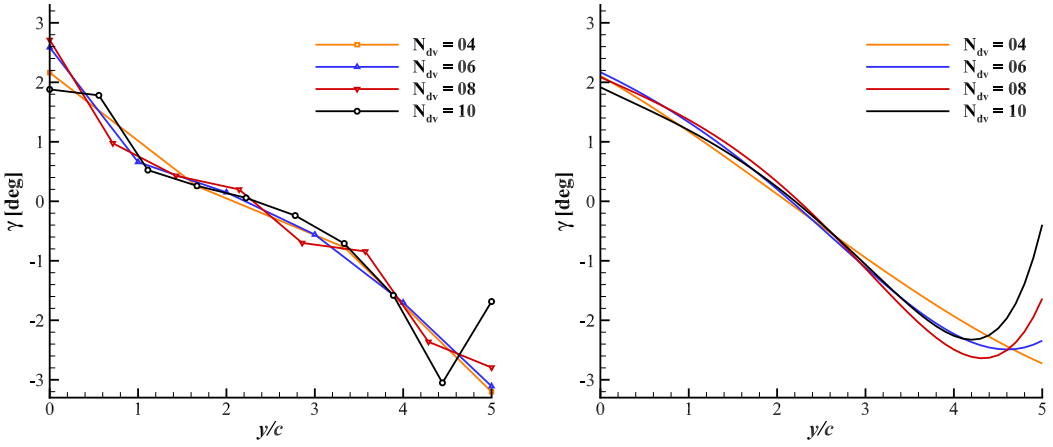
For both NACA0012 and NACA2412 cases, the same amount of drag reduction is achieved in the dimensionality study, suggesting the insensitivity of optimal aerodynamic performance to the number of design variables. However, the results of twist distribution, as shown in Figure 6.28 and Figure 6.29, vary with increasing number of design variables. This indicates that the optimal aerodynamic shape is not unique for both cases. With respect to NACA0012 wing, the twist distributions, as shown in Figure 6.28, exhibit a converging tendency with more design variables, and show a shape

similar to the sine function. For the case of NACA2412 wing, the twist angle monotonically decreases from root to tip for cases with small N_{dv}, whereas for cases with relatively larger N_{dv}, the twist shows a steady washout before experiencing an abrupt turn close to the wing tip. Overall, the number and locations of discrete twist control variables have an influence on the continuous twist distribution along the wing span.



(a) FALCon (b) SU2

Figure 6.28: Case 1: Sectional twist distributions in dimensionality study.



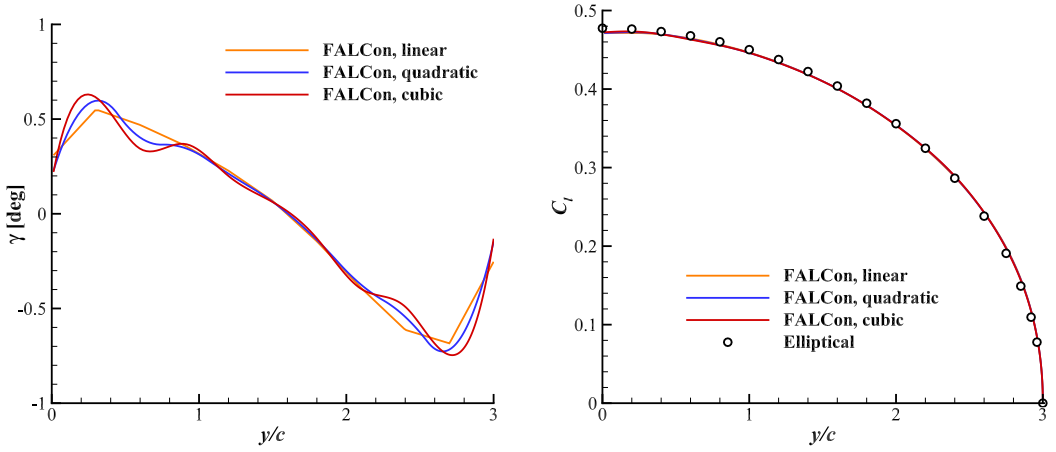
(a) FALCon (b) SU2

Figure 6.29: Case 2: Sectional twist distributions in dimensionality study.

6.4.2 Impact of interpolation

Recall that, within FALCon optimisation procedure, the linear interpolation method is used to map twist design variables onto twist angles of VLM panels. The main drawback of this approach is that the resulting twist distribution can not maintain the

second-order continuity, which provides the smoothness on the wing surface. This is unfavourable from aerodynamic viewpoint and will also cause difficulties for manufacturing process. Therefore, nonlinear polynomial functions, including quadratic and cubic, are then employed as interpolation method for optimisations with FALCon. The sectional distribution results with respect to 11 design variables for both cases are plotted in Figure 6.30 and Figure 6.31. It is worth noting that the same lift distribution is obtained in spite of the degree of polynomials. Although the geometric smoothness of wing surface is guaranteed when using interpolation with polynomials of high degree, an undesirable by-product is the problem of oscillations in twist distribution, which is observed in Figure 6.30(a) and Figure 6.31(a). For optimisation cases with more design variables or polynomials of higher degree, the oscillations would be more apparent. As oscillations can lead to unacceptable wing designs in reality, an alternative to polynomial interpolation is required to overcome this shortcoming.



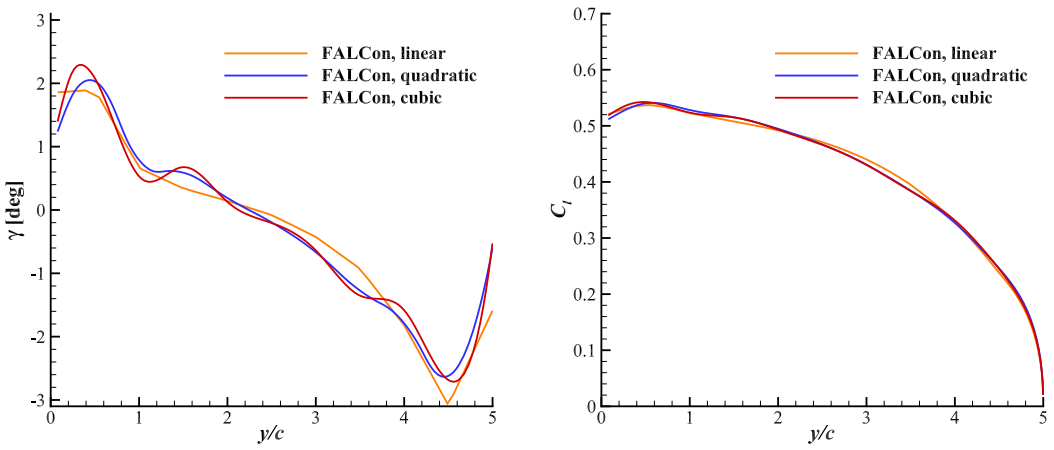
(a) Twist distribution

(b) Lift distribution

Figure 6.30: Case 1: Sectional twist and lift distributions of optimised design using different interpolation methods in FALCon ($N_{\mathrm{dv}} = 11$).

6.4.3 Cross validation

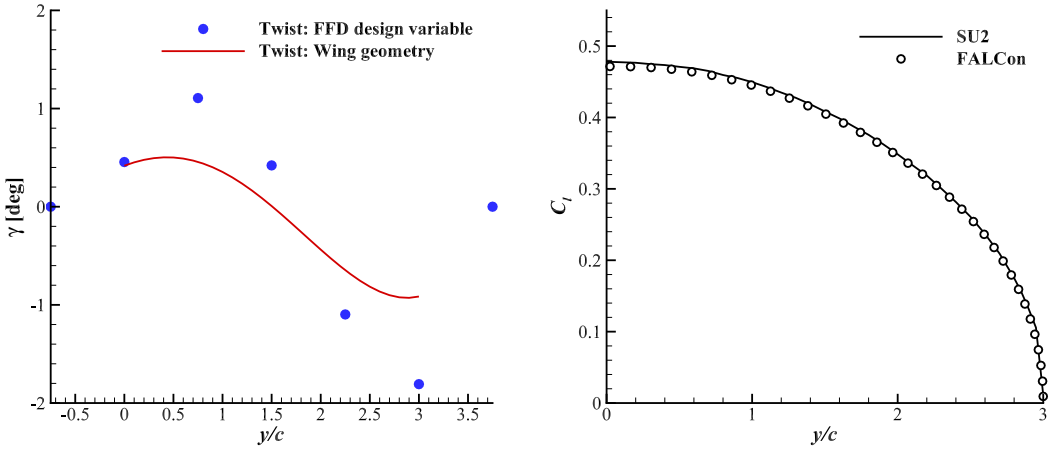
Within the SU2 optimisation, a Bézier curve is used as the FFD blending function to link the perturbation of FFD box with the deformation of embedded wing geometry. Figure 6.32(a) and Figure 6.33(a) give an illustration of the relationship between FFD twist design variables and wing twist distribution for both optimisation problems. Note that a fine shape control is achieved by FFD technique with a small set of design variables. The optimal twist distribution identified from SU2 optimisation is also imposed in FALCon solver for aerodynamic analysis. As a result, the same lift distribution is generated, as shown in Figure 6.32(b) and Figure 6.33(b). This again validates the computational accuracy of FALCon compared to SU2, that is, the same lift distribution can



(a) Twist distribution (b) Lift distribution

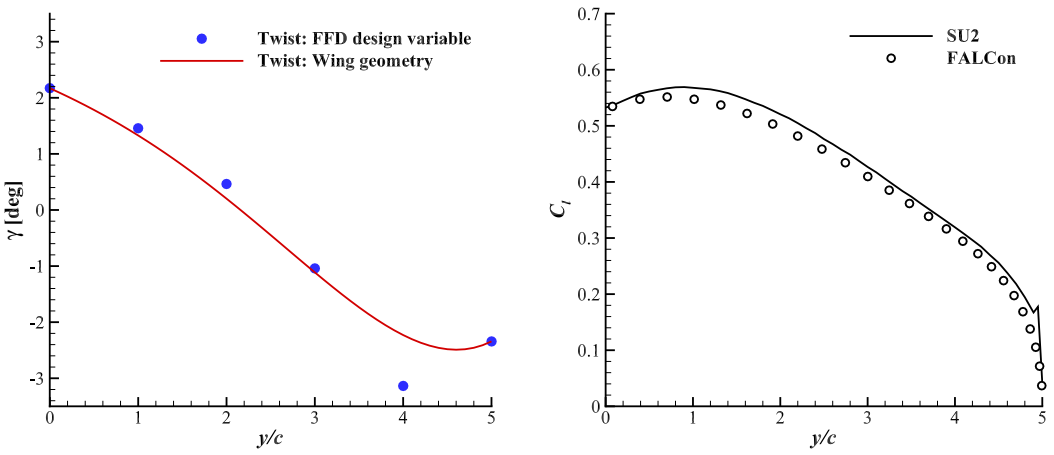
Figure 6.31: Case 2: Sectional twist and lift distributions of optimised design using different interpolation methods in FALCon ($N_{\mathrm{dv}} = 11$).

be produced given a specific geometry. Furthermore, this implies the possibility of applying more practical geometry control methods in FALCon. Recently, B-spline surface control and FFD approach have been widely used in a range of challenging optimisation problems, and they are compared in Reference [56] in terms of their effectiveness for ASO. This indicates a direction for future work in order to improve the robustness and suitability of FALCon in ASO problems.



(a) Twist distribution (b) Lift distribution

Figure 6.32: Case 1: Lift distributions under the geometrical twist computed from SU2 ($N_{\mathrm{dv}} = 5$).



(a) Twist distribution

(b) Lift distribution

Figure 6.33: Case 2: Lift distributions under the geometrical twist computed from SU2 ($N_{\mathrm{dv}} = 6$).

6.4.4 Section summary

The optimisation results of twist distribution show sensitivity to the number of design variables and twist interpolation method. For the NACA0012 case, the twist distributions exhibit a converging pattern with more design variables, and show a shape similar to the sine function. For the NACA2412 case, the twist angle decreases monotonically from root to tip when N_{dv} is small; whereas with relatively larger N_{dv}, the twist distribution has an abrupt turn near the wing tip.

Three polynomial interpolation approaches were employed for twist mapping in FALCon optimisation, which are linear, quadratic and cubic interpolation, respectively. The same lift distribution is obtained despite the interpolation method. Although nonlinear interpolations can guarantee the geometric smoothness of wing surface, they produce the side-product of oscillations for twist distribution, which is unfavourable and impractical for wing design. Regarding the SU2 optimisation, FFD parameterisation is employed and a Bézier curve is used as the FFD blending function. The resulting twist distribution not only maintains second-order continuity but also avoids the issue of oscillations. By applying the twist distribution (obtained from SU2) to the wing geometry, the same lift distribution is generated from FALCon and SU2, which further confirms the computational accuracy of the hybrid solver.

6.5 Chapter summary

In this chapter, two wing twist optimisation problems were investigated: an unswept wing with NACA0012 sections in inviscid, subsonic flow, and a swept-back wing with

NACA2412 sections in viscous, subsonic flow. Two aerodynamic solvers were employed in this study. The first solver is the high-fidelity SU2, which solves 3D Euler or RANS equations. The second is a multi-fidelity solver, FALCon, which builds upon three constituent parts: a VLM solver, an ISW solver and an α-based coupling algorithm. For the multi-fidelity aerodynamic approach, the 3D effects on finite wings are modelled by the linear VLM solver, and the sectional nonlinear effects are captured by the ISW solver.

With respect to the baseline geometry for both optimisation cases, the aerodynamic results are verified between FALCon and SU2 at a range of angles of attack, including the lift and drag coefficient, and sectional lift distributions. Overall, both optimisation frameworks perform equally well for these two cases. The key findings of the numerical optimisations are summarised as follows:

1. **Optimisation mechanism**
   The mechanism for drag reduction is the same for these two optimisation problems. The spanwise loading moves from outboard to inboard sections in order to alleviate the intensity of the tip vortex and thus reduce the lift-induced drag. For the benchmark NACA0012 case, an elliptical lift distribution is recovered on the optimised geometry, which validates the effectiveness of two optimisation frameworks.

2. **Efficiency assessment**
   Compared with SU2, FALCon demonstrates a much higher computational efficiency, saving up to 98% of the computational cost. The advantage primarily benefits from the high efficiency of the ISW computation on a 2D stencil.

3. **Sensitivity assessment**
   The optimal drag coefficient in both cases shows insensitivity to the number of design variables. Around 1 drag count is reduced in the NACA0012 case, and approximately 5.5% drag reduction is achieved in the NACA2412 case.

   The number of twist design variables shows influence on the optimal twist distribution. As more design variables are used, the twist distributions in the NACA0012 case exhibit a convergence trend towards a shape similar to “sine” function. For the NACA2412 case, a steady washout is observed with small number of design variables; an abrupt increase of twist angle occurs near the wing tip when more design variables are added.

   The twist interpolation method within FALCon optimisation framework has an impact on the optimisation results. Although the smoothness of wing surface can be achieved by applying nonlinear polynomial interpolations, an evident side-product is the issue of local oscillations, which is impractical for wing design.

4. **Geometry control**

   For the SU2 optimisation, the FFD technique has shown its advantages in geometry control and shape parameterisation. By imposing the optimal twist distribution (identified from SU2) to both aerodynamic solvers, the same spanwise loading is obtained, which further confirms the computational accuracy of the multi-fidelity solver. For FALCon optimisation, polynomial interpolation methods were employed to map twist design variables onto twist angle of VLM panels. However, these approaches are not suitable for practical wing design. More robust and well-behaved geometry control methods may be considered in future work, such as FFD and spline methods that have the desirable properties of flexibility, efficiency, and smoothness.

Chapter 7

Conclusions and Future Work

7.1 Summary of findings

This research presents investigations into the assessment of optimal design sensitivity and computational efficiency in aerodynamic shape optimisation (ASO). The research methodology consists of gradient-based numerical optimisation using two aerodynamic solvers of different levels of fidelity. The applications include several benchmark cases defined by the AIAA Aerodynamic Design Optimisation Discussion Group (ADODG). The thesis primarily focuses on two research streams. The first branch investigates the sensitivity of optimal solution to a series of numerical parameters for both two-dimensional (2D) aerofoil and three-dimensional (3D) wing optimisation problems. The high-fidelity SU2 code based on full 3D Euler/Navier–Stokes equations is used to undertake this study. The gradients are obtained through the continuous adjoint approach and the simulations are performed on the high-performance computing facility IRIDIS 4. The second is the application of an efficient multi-fidelity aerodynamic solver, FALCon, to wing twist optimisation problems. The SU2 code is also used, with the purpose of verifying the computational accuracy and evaluating the optimisation efficiency.

The current study includes a comprehensive overview of the existing literature concerning ASO in the field of aeronautical applications, and this is carried out from two perspectives. First, the research activities conducted from a number of representative groups are summarised, highlighting their contributions to the community. Second, a review is provided with respect to the constitutive computational techniques for an ASO framework, discussing their advantages and drawbacks. Additionally, a survey is provided regarding sensitivity assessment in the previous studies, which mainly focused on the influence of different techniques (e.g. geometry parameterisation methods) on the optimisation results; a review is also presented concerning the development of multi-fidelity aerodynamic models, particularly the quasi-three-dimensional (Q3D) solver, with a focus on its efficiency in aerodynamic analysis. For a given optimisation framework,

the sensitivity of the final optimal solution to numerical parameters, however, has been largely underestimated to date, indicating the significance of this study. Moreover, current trend of ASO includes adjoint-based large-scale optimisation utilising high-fidelity aerodynamic solvers. However, this is impractical in the real world engineering environment, particularly in early phases of aircraft design with limited computational resources and time budgets, raising the necessity for developing efficient computational methodologies.

This research primarily contains two parts, addressing the two issues identified in the area of ASO, namely revisiting the optimal solution and improving optimisation efficiency. The detailed summary and conclusions have been provided earlier in Sections 4.7, 5.7, and 6.5. This section highlights the principal achievements, the major findings, and the most significant conclusions of each part.

7.1.1 Drag minimisation of two-dimensional aerofoils

Gradient-based aerodynamic optimisations of 2D aerofoils were performed using the open-source SU2 code, and gradients were computed using the continuous adjoint method. Two benchmark problems defined by the ADODG were exercised. The first case is the drag minimisation of the NACA0012 aerofoil in transonic, inviscid flow, with a minimum thickness constraint; the second case is the drag minimisation of the RAE2822 aerofoil in transonic, viscous flow, subject to lift, pitching moment and area constraints. Hicks–Henne bump function (HHBF) and free-form deformation (FFD) were employed as geometry parameterisation methods.

In the NACA0012 case, the drag reduction mechanism is to minimise the shock wave strength, hence the wave drag. The optimised aerofoil exhibits a blunter leading edge and a thicker aft section compared to the baseline geometry, formulating a rather flat aerofoil surface. As a consequence, the strong shock at around three-quarter chord position is substantially weakened and is pushed further downstream, locating at approximately 90% chordwise position. Regarding HHBF parameterisation, the "one-minus-cosine" distribution approach outperforms uniform distribution, and the setting of $t = 6$ for bump width control parameter produces the lowest drag among a range of values. With best practice adopted in the systematic study, the drag is reduced from 468.0 counts to 80.5 counts, which corresponds to 82.8% of drag reduction.

Differing from the NACA0012 case that is an academic problem, the RAE2822 case reflects a more practical problem of transonic aerofoil design with realistic constraints. The drag reduction principle is to eliminate the shock, thus removing the wave drag from the total drag. Compared with the baseline aerofoil, the optimised shape shows a relatively flat surface with smaller curvature on the upper aerofoil, creating a shock-free

geometry. The drag coefficient is reduced from 241 counts to 150 counts, achieving a drag reduction of about 38%.

The primary purpose of this part of work is to investigate the influence that numerical parameters have on the optimisation results. The numerical parameters include: a) two parameterisation methods, HHBF and FFD; b) numerical settings related to the tuning of each parameterisation method; and c) closure coefficients of Spalart–Allmaras (SA) turbulence model. Several conclusions were formulated from this study, which are listed as follows:

1. HHBF and FFD were shown to be equivalently effective as geometry parameterisation method for both optimisation problems.
2. The optimisation of the NACA0012 aerofoil exhibits strong dependency on virtually all numerical parameters investigated. This dependency reflects the high curvature observed locally on the optimised shape. Fine-tuning of these parameters is thus required to provide sufficient local shape control.
3. In the RAE2822 case, the insensitivity of the optimal solution to numerical parameters arises from low to mild curvatures on the final shape. The design space is easily explored by moderately perturbing the geometry.
4. The optimisation performance is influenced by turbulence modelling as the Reynolds-averaged Navier–Stokes (RANS) solution shows a certain sensitivity to the closure coefficients of SA model, raising the need for a good calibration of the turbulence model.

Additionally, for the optimised aerofoil in both optimisation cases, non-unique flow solutions were found to exist at the design point. Further investigations confirmed the non-uniqueness by discovering the hysteresis loop of drag coefficient in a narrow region around the design point. This may indicate that the optimisation problems are ill-posed.

Overall, although the sensitivity maps provide us with the information of key design variables, there is no priori knowledge that we can use to know the degree of sensitivity of the optimal design to numerical settings. This can only be estimated a posteriori by running a number of analyses. The sensitivity assessment conducted for the two aerofoil cases may provide guidelines of best practice for complex optimisation problems.

7.1.2 Twist optimisation of three-dimensional wings

An efficient optimisation framework based on a multi-fidelity aerodynamic solver, FALCon, was developed. The multi-fidelity aerodynamic approach consists of three constituent parts: a linear vortex lattice method (VLM) solver, an infinite swept wing

(ISW) solver of steady/unsteady RANS equations, and an α-based coupling algorithm. The 3D effects on finite wings are modelled by the linear VLM solver, and the sectional nonlinear effects are captured by the ISW solver. The coupling algorithm combines the two solvers to reproduce the aerodynamic loads with high computational efficiency. Gradient-based wing twist optimisations were performed using the SLSQP optimiser available from the open-source pyOpt package, and the gradients were computed using finite difference method (FDM) as the aerodynamic approach has very low computational cost. Two optimisation problems were investigated to illustrate the capability of the proposed aerodynamic model: an unswept wing with NACA0012 sections under viscous, subsonic flow, and a swept-back wing with NACA2412 sections under viscous, subsonic flow. For reference, 3D data were obtained from SU2.

With respect to the baseline geometry for both cases, the aerodynamic results (including the lift and drag coefficient, and sectional lift distributions) obtained from FALCon were verified against SU2 at a range of angles of attack. An overall good agreement was found between these two aerodynamic solvers. However, in terms of spanwise lift distribution, the FALCon solver is unable to predict the abrupt rise at the wing tip, which is captured by the SU2 solver. This may limit the prediction accuracy of FALCon for flows with strong vortex intensity and thus the pronounced 3D physics.

Regarding these two optimisation problems, the mechanism for drag reduction is the same: the spanwise aerodynamic loading moves from outboard to inboard sections in order to alleviate the intensity of the tip vortex and thus reduce the lift-induced drag. For the benchmark NACA0012 case, an elliptical lift distribution, which is the theoretical optimal solution, is recovered on the optimised geometry, which validates the effectiveness of these two optimisation frameworks.

The primary objective of this part of research is to analyse the computational efficiency of the multi-fidelity solver in wing twist optimisation. Additionally, the sensitivity of optimisation results to a number of parameters is also investigated. The key findings and conclusions are listed as follows:

1. Both optimisation frameworks performed equally well for these two cases. In the NACA0012 case, the analytical optimal solution was achieved with around 1 drag count reduction; in the NACA2412 case, approximately 5.5% drag reduction was obtained.

2. The aerodynamic optimisation using the multi-fidelity approach was performed at a negligible computational cost compared to the full 3D SU2 solver. The advantage primarily benefits from the high efficiency of the ISW computation on a 2D stencil.

3. Although the final drag coefficient shows insensitivity to the number of design variables, the design variable dimensionality and the twist interpolation method have an influence on the optimal twist distribution.

4. Compared with the polynomial interpolation methods that are employed for twist mapping in FALCon, the FFD technique in SU2 optimisation demonstrates its advantages in geometry control and shape parameterisation, such as smoothness, efficiency and flexibility, which are desirable features in practical wing design.

Overall, the proposed approach is suitable for accurate prediction of aerodynamic loads at early stages of aircraft design. Moreover, as demonstrated in this study, the multi-fidelity aerodynamic solver may find easy application to other areas by a straightforward extension and coupling of the three constitutive blocks.

7.2 Future work arising from this study

There are a number of studies that can be performed following this research. This section provides some recommendations of potential future work, which are listed as follows:

1. **Sensitivity assessment regarding other techniques**
   In this research, initial attempts were made towards sensitivity assessment to revisit the optimal solution in ASO. The parameters investigated mainly included the numerical settings within HHBF and FFD. Since geometry parameterisation plays a significant role in ASO and there exist a variety of parameterisation approaches as described in Section 2.2.2, there is a need for sensitivity assessment concerning other shape parameterisation methods, such as the commonly-used class shape transformation (CST) and non-uniform rational B-splines (NURBS). As discussed in Section 4.5.1, the CST parameterisation in Reference [253] did not successfully increase the leading edge bluntness of NACA0012 aerofoil and hence failed in locating the true optimal solution. This is due to a lack of tuning of CST implementation for this optimisation problem. Moreover, as discussed in Section 2.2.4.2, many gradient-free optimisation algorithms, such as genetic algorithm (GA) and particle swarm optimisation (PSO), contain a number of parameters, which are usually limited in a range with lower and upper bounds and are generally used by default settings. For a specific case, the sensitivity assessment is required to adjust these parameters in order to obtain the best optimisation performance. Additionally, an extension from sensitivity assessment will step into the realm of uncertainty quantification (UQ), which targets at identifying quantitatively the uncertainties from various aspects of ASO.

2. **Further investigation into non-uniqueness**
   For both aerofoil optimisation problems in this study, non-unique flow solutions were found to exist in a narrow band near the design point. Hysteresis was observed when performing an upward and downward sweep of the flow condition (Mach number or angle of attack). Although some researchers [257, 258, 264, 265] have investigated the issue of non-unique solutions in transonic flow, the aerofoils were mainly of the

same kind with the optimised NACA0012 aerofoil, and the studies were largely limited to steady-state simulations. The mechanism for the non-uniqueness phenomenon remains an important question. To gain a better understanding of the evolution and stability of these flows, unsteady simulations are required to be carried out, particularly for the aerofoils similar to optimised RAE2822 geometry. Further analysis shall not only investigate steady and unsteady methods but also in general examine the effects of the spatial resolution and the numerical schemes (and the combination of both).

On the other hand, the non-unique solutions may indicate the single-point optimisation problems are ill-posed. To enhance the robustness of the optimal solution and its performance at off-design flow conditions, well-posed optimisation problems need to be proposed, including multi-point aerodynamic optimisations and imposing more specific design constraints such as curvature.

3. **Extension of FALCon to other applications**
   This work demonstrated the applicability and efficiency of the multi-fidelity aerodynamic solver FALCon in wing twist optimisation. In this study, two subsonic optimisation cases with clean wing configurations were considered. For commercial aircraft with complex configurations, the fuselage has influence on the aerodynamic loading near the wing root [50]. The fuselage contributions are not modelled in the current framework. Further development is thus needed to appropriately take the influence of the fuselage into account. As shown in Reference [50], the fuselage correction can be performed through the aerodynamic influence coefficients (AICs) in the VLM instead of geometrically modelling the fuselage. Moreover, to highlight the capability of FALCon, future studies shall consider applications to a wider range of problems, such as commercial aircraft wings in transonic flow where the nonlinear effects are dominant. Additionally, as discussed in Section 6.4.3, more suitable geometry control methods, such as FFD and B-spline, can be employed as the twist mapping approach, which ensures a practical wing design.

   In this research, a rigid planar wing model was used for the twist optimisation. For flexible wings with high aspect ratio, large structural deformations may occur. Therefore, a structural model is required to be built and coupled with the multi-fidelity aerodynamic solver. Ongoing work in the author's group is tackling this problem.

   Furthermore, this work illustrated the integration of FALCon into wing twist optimisation, where the twist angle was treated as an additional angle of attack correction and was embedded directly into the local angle of attack of VLM panels. FALCon may find potential applications in other fields of ASO, such as wing planform optimisation as demonstrated in Reference [239].

Appendix A

CFD Validation of SU2

The steady-state RANS solver of SU2 is validated against experimental measurements for three test cases, which are NACA0012 aerofoil, RAE2822 aerofoil, and ONERA M6 wing, respectively. For reference, the CFD results obtained from the commercial software ANSYS Fluent are also added for comparison. The freestream conditions for the three validation cases are listed in Table A.1. The computational set-up is nearly identical for these cases and is given in Table A.2. The computational results are presented in the following sections. An overall good agreement is found between the CFD results and experimental data, and the SU2 results are consistent with those obtained from Fluent. This fact confirms the computational accuracy of SU2 in RANS simulations.

Table A.1: Freestream conditions for the three validation cases.

Test case	M	α [deg]	Re
NACA0012	0.15	10	$6.0 \cdot 10^6$
RAE2822	0.729	2.31	$6.5 \cdot 10^6$
ONERA M6	0.84	3.06	$11.72 \cdot 10^6$

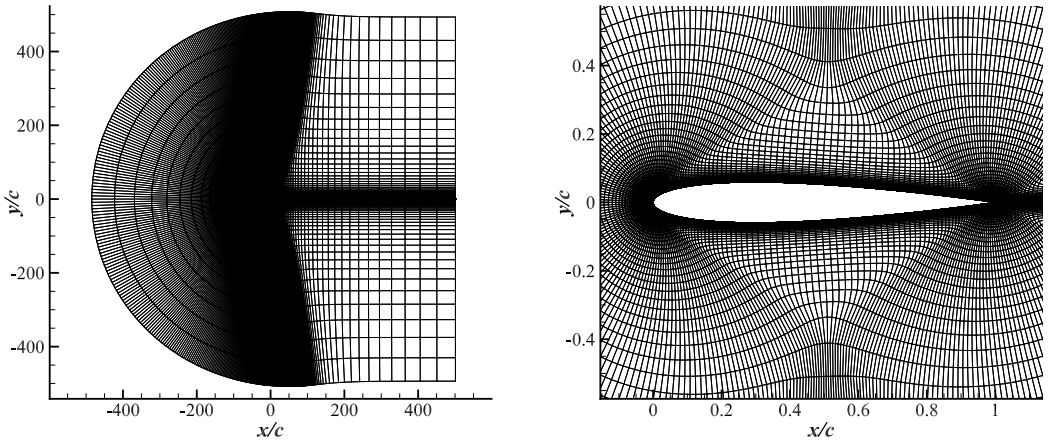
Table A.2: Computation set-up for the three validation cases.

Parameter	Method or value
Governing equations	RANS
Turbulence model	SA
Spatial discretisation	2nd order Roe
Time discretisation	Implicit Euler
CFL number	5

A.1 NACA0012 aerofoil

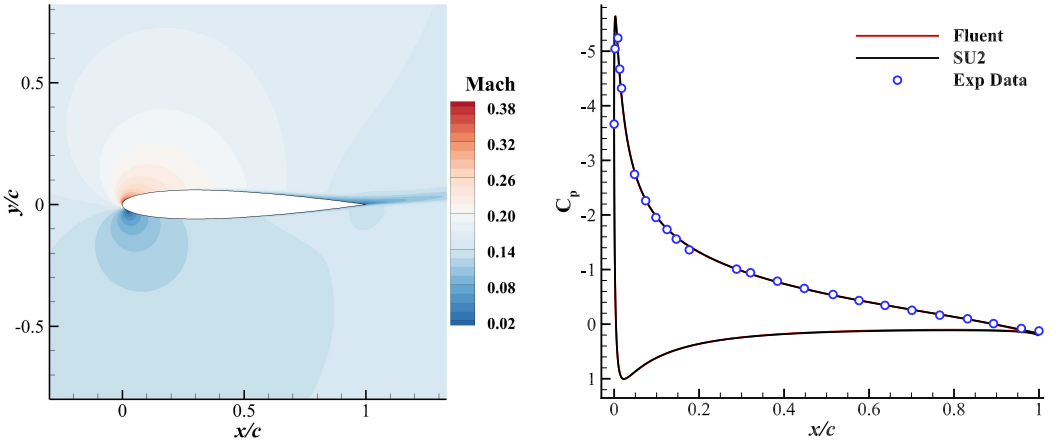
The NACA0012 aerofoil is a classic low-speed aerofoil that has been extensively used for CFD validation of turbulence models. In this study, a C-topology structured grid

available from NASA website [1] is used. The computational grid, as shown in Figure A.1, consists of 449 × 129 grid points in the wrap-around and normal directions, respectively, where 257 points are distributed along the aerofoil and 97 points in the grid cut. The off-wall spacing is set to ensure the y^+ is below 1. Under the prescribed subsonic flow conditions, boundary layers should be fully turbulent over most of the aerofoil. The Mach contours of the flow field are shown in Figure A.2(a). Figure A.2(b) shows the pressure distribution compared with the experimental data documented by Gregory and O'Reilly [270]. Identical results are found between SU2 and Fluent, and an excellent agreement is observed between CFD results and experimental data.



(a) Far view (b) Near view

Figure A.1: Computational grid for the NACA0012 aerofoil.



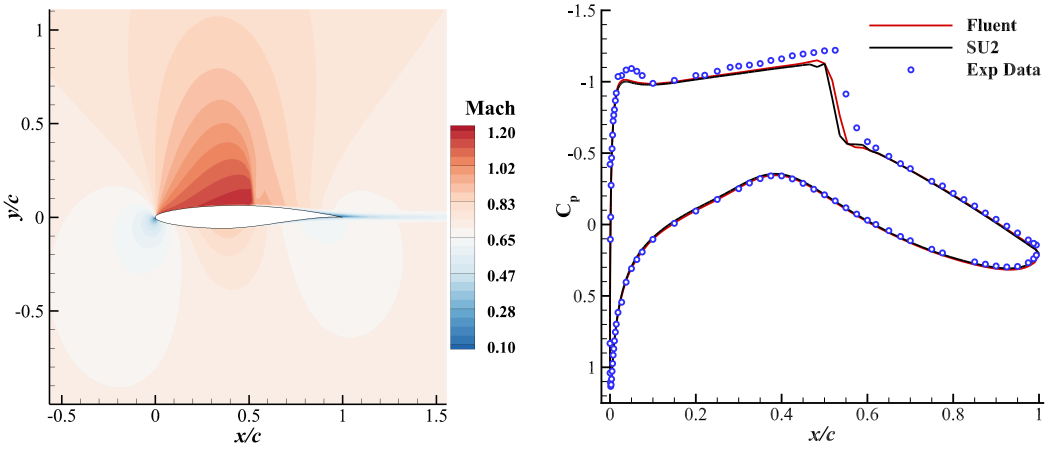
(a) Mach contour (SU2) (b) Comparison of pressure coefficient distribution

Figure A.2: Mach contours and comparison of pressure coefficient distribution on the NACA0012 aerofoil ($M = 0.15$, $\alpha = 10$ deg, and $Re = 6.0 \cdot 10^6$); Experimental data from [270].

[1] https://turbmodels.larc.nasa.gov/naca0012_val.html [retrieved 2019]

A.2 RAE2822 aerofoil

The RAE2822 aerofoil is a supercritical aerofoil commonly used for validation of turbulence models. For this test case the flow is turbulent and transonic. Additionally, conditions are such that no separation occurs downstream of the shock position. The computational grid (shown in Figure 5.2) used in this work follows that in Section 5.3 and is not reported herein for brevity. The test case is based on the AGARD Report by Cook *et al.* [259]. The flow field results are shown in Figure A.3(a), and the comparison of pressure coefficient distribution is shown in Figure A.3(b). A good match is observed between the two CFD solvers, and a good agreement is found between the CFD results and experimental data, except at the suction peak and shock location, which is also documented in other literature [121, 260].



(a) Mach contour (SU2)

(b) Comparison of pressure coefficient distribution

Figure A.3: Mach contours and comparison of pressure coefficient distribution on the RAE2822 aerofoil ($M = 0.729$, $\alpha = 2.31$ deg, and $Re = 6.5 \cdot 10^6$); Experimental data from [259].

A.3 ONERA M6 wing

The ONERA M6 wing is a swept, semispan wing with no twist that uses a symmetric aerofoil (ONERA D sections). The aspect ratio is $AR = 3.8$ and the leading edge sweep angle is $\Lambda = 30$ deg. The Reynolds number is based on the mean aerodynamic chord, $c = 0.64607\,m$. The ONERA M6 was tested in a wind tunnel at multiple Mach numbers and various angles of attack, and the experimental results are documented by Schmitt and Charpin [271]. The ONERA M6 is a commonly used validation case for external flows because of its simple geometry combined with complexities of transonic flow, i.e. local supersonic flow, shocks and turbulent boundary layer separation. The simulation

in this study uses the flowfield conditions of Test 2308, where $M = 0.84$, $\alpha = 3.06$ deg, and $Re = 11.72 \cdot 10^6$.

As shown in Figure A.4, the computational grid is a C-C topology structured grid and is available from NASA website [2]. Figure A.5 shows the pressure contours on the upper surface of the wing. A key flow feature for this case is that there are two shocks generated, which is commonly denoted as the λ shock. In Figure A.6 the comparison with experimental data is shown for the pressure coefficient distributions at different sections along the span of the wing. An overall good agreement between numerical and experimental data is observed, and the SU2 results are largely in line with those produced from Fluent. At $y/b = 0.80$ CFD analyses hardly capture the double shock, which is possibly due to the coarse grid resolution. At $y/b = 0.99$ both flow solvers give less accurate results due to the wingtip vortex created in this region of wing. The rotation of the tip vortices generates strong 3D effects, making it difficult to predict accurately the pressure coefficients in this region.

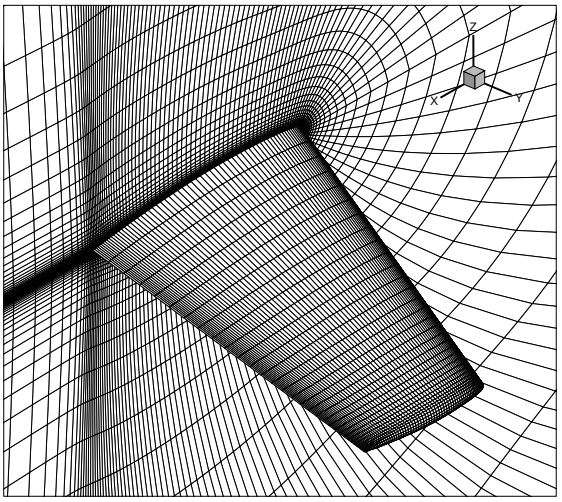


Figure A.4: Computational grid for the ONERA M6 wing.

[2] https://www.grc.nasa.gov/WWW/wind/valid/m6wing/m6wing01/m6wing01.html [retrieved 2019]

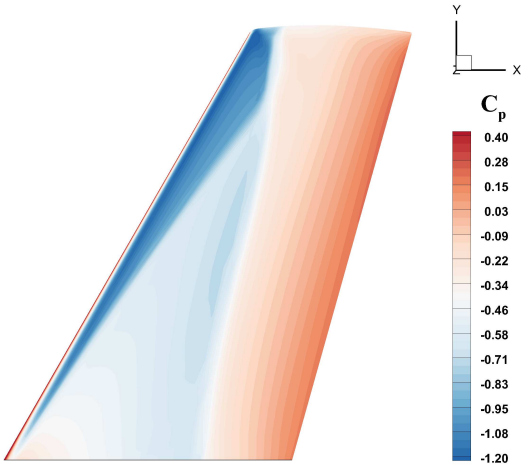
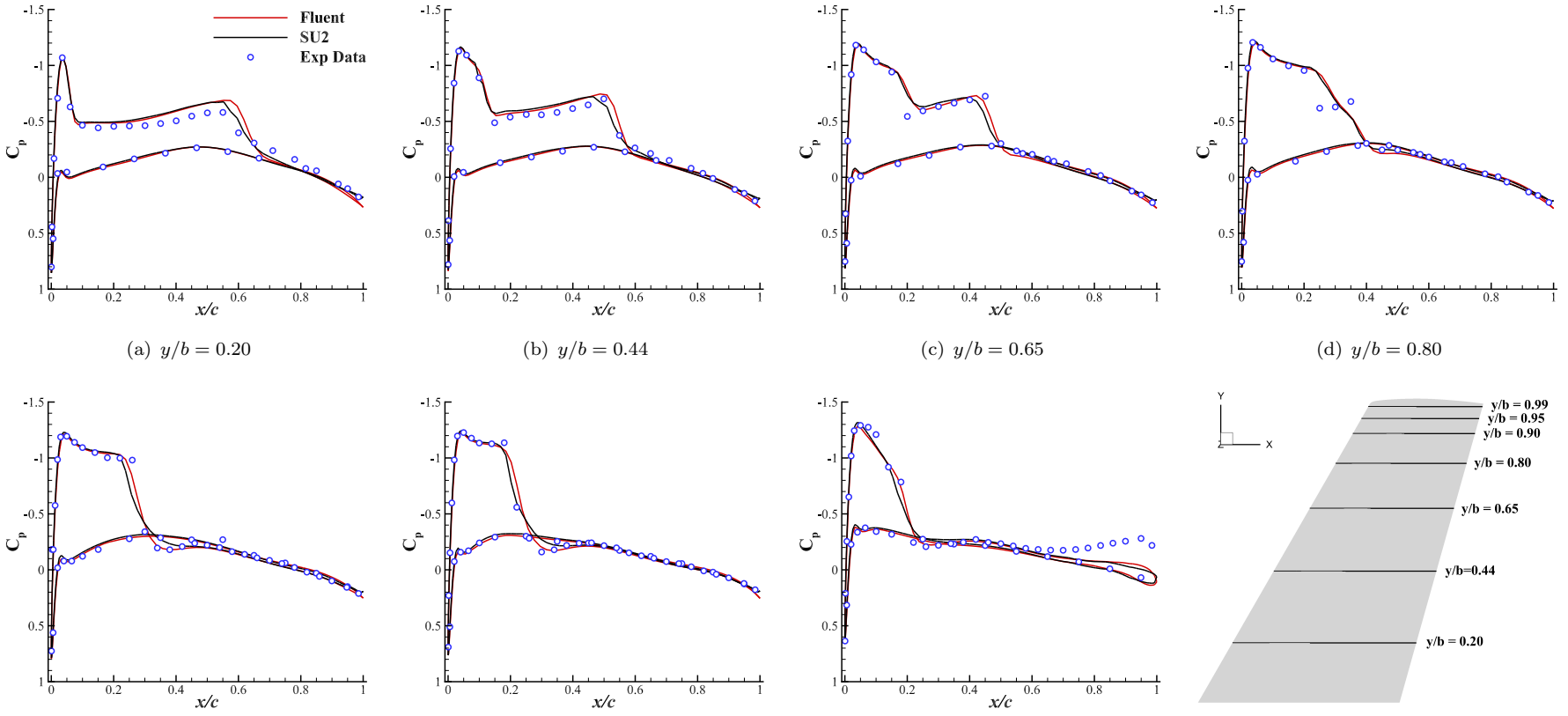


Figure A.5: Pressure contours (obtained from SU2) on the upper surface of the ONERA M6 wing ($M = 0.84$, $\alpha = 3.06$ deg, and $Re = 11.72 \cdot 10^6$).



(a) $y/b = 0.20$ (b) $y/b = 0.44$ (c) $y/b = 0.65$ (d) $y/b = 0.80$

(e) $y/b = 0.90$ (f) $y/b = 0.95$ (g) $y/b = 0.99$ (h) Spanwise locations

Figure A.6: Comparison of C_p profiles of numerical results (SU2 and Fluent) against experimental data (documented by Schmitt and Charpin [271]) at 7 spanwise locations on the ONERA M6 wing ($M = 0.84$, $\alpha = 3.06$ deg, and $Re = 11.72 \cdot 10^6$).

References

[1] Kharlamov, D., Drofelnik, J., Da Ronch, A., and Walker, S., "Rapid load calculations using an efficient unsteady aerodynamic solver," *2018 Atmospheric Flight Mechanics Conference*, 2018, p. 3621.

[2] Martins, J. R., "Multidisciplinary Design Optimization," *University of Michigan*, 2017.

[3] Haftka, R. T., "Optimization of flexible wing structures subject to strength and induced drag constraints," *AIAA Journal*, Vol. 15, No. 8, 1977, pp. 1101–1106.

[4] Kroo, I., Altus, S., Braun, R., Gage, P., and Sobieski, I., "Multidisciplinary optimization methods for aircraft preliminary design," *5th symposium on multidisciplinary analysis and optimization*, 1994, p. 4325.

[5] Antoine, N. E., and Kroo, I. M., "Framework for aircraft conceptual design and environmental performance studies," *AIAA Journal*, Vol. 43, No. 10, 2005, pp. 2100–2109.

[6] Alonso, J. J., and Colonno, M. R., "Multidisciplinary optimization with applications to sonic-boom minimization," *Annual Review of Fluid Mechanics*, Vol. 44, 2012, pp. 505–526.

[7] McAllister, C. D., and Simpson, T. W., "Multidisciplinary robust design optimization of an internal combustion engine," *Journal of mechanical design*, Vol. 125, No. 1, 2003, pp. 124–130.

[8] Ganguli, R., "A survey of recent developments in rotorcraft design optimization," *Journal of Aircraft*, Vol. 41, No. 3, 2004, pp. 493–510.

[9] Fuglsang, P., and Madsen, H. A., "Optimization method for wind turbine rotors," *Journal of Wind Engineering and Industrial Aerodynamics*, Vol. 80, No. 1-2, 1999, pp. 191–206.

[10] Kenway, G., and Martins, J., "Aerostructural shape optimization of wind turbine blades considering site-specific winds," *12th AIAA/ISSMO multidisciplinary analysis and optimization conference*, 2008, p. 6025.

[11] Wang, X., Shen, W., Zhu, W., Sørensen, J. N., and Chen, J., "Shape optimization of wind turbine blades," *Wind Energy: An International Journal for Progress and Applications in Wind Power Conversion Technology*, Vol. 12, No. 8, 2009, pp. 781–803.

[12] Sobieszczanski-Sobieski, J., and Haftka, R. T., "Multidisciplinary aerospace design optimization: survey of recent developments," *Structural optimization*, Vol. 14, No. 1, 1997, pp. 1–23.

[13] Martins, J. R., and Lambe, A. B., "Multidisciplinary design optimization: a survey of architectures," *AIAA Journal*, Vol. 51, No. 9, 2013, pp. 2049–2075.

[14] Willcox, K., and Wakayama, S., "Simultaneous optimization of a multiple-aircraft family," *Journal of Aircraft*, Vol. 40, No. 4, 2003, pp. 616–622.

[15] Piperni, P., Abdo, M., Kafyeke, F., and Isikveren, A. T., "Preliminary aerostructural optimization of a large business jet," *Journal of Aircraft*, Vol. 44, No. 5, 2007, pp. 1422–1438.

[16] Agte, J., de Weck, O., Sobieszczanski-Sobieski, J., Arendsen, P., Morris, A., and Spieck, M., "MDO: assessment and direction for advancement—an opinion of one international group," *Structural and Multidisciplinary Optimization*, Vol. 40, No. 1–6, 2009, pp. 17–33.

[17] Brezillon, J., Ronzheimer, A., Haar, D., Abu-Zurayk, M., Lummer, K., Kruger, W., and Nattere, F., "Development and application of multi-disciplinary optimization capabilities based on high-fidelity methods," *AIAA Paper*, Vol. 1757, 2012, p. 2012.

[18] Piperni, P., DeBlois, A., and Henderson, R., "Development of a multilevel multidisciplinary-optimization capability for an industrial environment," *AIAA Journal*, Vol. 51, No. 10, 2013, pp. 2335–2352.

[19] Gray, J. S., Hwang, J. T., Martins, J. R., Moore, K. T., and Naylor, B. A., "OpenMDAO: An open-source framework for multidisciplinary design, analysis, and optimization," *Structural and Multidisciplinary Optimization*, 2019, pp. 1–30.

[20] Grossman, B., Gurdal, Z., Haftka, R., Strauch, G., and Eppard, W., "Integrated aerodynamic/structural design of a sailplane wing," *Journal of Aircraft*, Vol. 25, No. 9, 1988, pp. 855–860.

[21] Grossman, B., Haftka, R., Sobieszczanski-Sobieski, J., Kao, P.-J., Polen, D., and Rais-Rohani, M., "Integrated aerodynamic-structural design of a transport wing," *Journal of Aircraft*, Vol. 27, No. 12, 1990, pp. 1050–1056.

[22] Chittick, I. R., and Martins, J. R., "An asymmetric suboptimization approach to aerostructural optimization," *Optimization and Engineering*, Vol. 10, No. 1, 2009, pp. 133–152.

[23] Schmit, L. A., "Structural design by systematic synthesis," *Proceedings, 2nd conference on Electronic computation, ASCE, New York*, 1960, pp. 105–122.

[24] Venkayya, V. B., "Structural optimization: a review and some recommendations," *International Journal for Numerical Methods in Engineering*, Vol. 13, No. 2, 1978, pp. 203–228.

[25] Venkataraman, S., and Haftka, R. T., "Structural optimization complexity: what has Moore's law done for us?" *Structural and Multidisciplinary Optimization*, Vol. 28, No. 6, 2004, pp. 375–387.

[26] Jameson, A., "Aerodynamic design via control theory," *Journal of scientific computing*, Vol. 3, No. 3, 1988, pp. 233–260.

[27] Hicken, J. E., and Zingg, D. W., "Induced-drag minimization of nonplanar geometries based on the Euler equations," *AIAA Journal*, Vol. 48, No. 11, 2010, pp. 2564–2575.

[28] Mader, C. A., and Martins, J. R., "Stability-constrained aerodynamic shape optimization of flying wings," *Journal of Aircraft*, Vol. 50, No. 5, 2013, pp. 1431–1449.

[29] Dwight, R. P., and Brezillon, J., "Efficient and robust algorithms for solution of the adjoint compressible Navier–Stokes equations with applications," *International journal for numerical methods in fluids*, Vol. 60, No. 4, 2009, pp. 365–389.

[30] Lyu, Z., Kenway, G. K., and Martins, J. R., "Aerodynamic shape optimization investigations of the common research model wing benchmark," *AIAA Journal*, Vol. 53, No. 4, 2014, pp. 968–985.

[31] Osusky, L., Buckley, H., Reist, T., and Zingg, D. W., "Drag minimization based on the Navier–Stokes equations using a Newton–Krylov approach," *AIAA Journal*, Vol. 53, No. 6, 2015, pp. 1555–1577.

[32] Elham, A., and van Tooren, M. J., "Coupled adjoint aerostructural wing optimization using quasi-three-dimensional aerodynamic analysis," *Structural and Multidisciplinary Optimization*, Vol. 54, No. 4, 2016, pp. 889–906.

[33] Maman, N., and Farhat, C., "Matching fluid and structure meshes for aeroelastic computations: a parallel approach," *Computers & Structures*, Vol. 54, No. 4, 1995, pp. 779–785.

[34] Brown, S., "Displacement extrapolation for CFD+CSM aeroelastic analysis," *AIAA paper*, Vol. 1090, 1997, p. 1997.

[35] Farhat, C., Lesoinne, M., and Le Tallec, P., "Load and motion transfer algorithms for fluid/structure interaction problems with non-matching discrete interfaces: Momentum and energy conservation, optimal discretization and application

to aeroelasticity," *Computer methods in applied mechanics and engineering*, Vol. 157, No. 1, 1998, pp. 95–114.

[36] Kamakoti, R., and Shyy, W., "Fluid-structure interaction for aeroelastic applications," *Progress in Aerospace Sciences*, Vol. 40, No. 8, 2004, pp. 535–558.

[37] Felippa, C. A., Park, K., and Farhat, C., "Partitioned analysis of coupled mechanical systems," *Computer methods in applied mechanics and engineering*, Vol. 190, No. 24, 2001, pp. 3247–3270.

[38] Kim, J., Aluru, N., and Tortorelli, D., "Improved multi-level Newton solvers for fully coupled multi-physics problems," *International journal for numerical methods in engineering*, Vol. 58, No. 3, 2003, pp. 463–480.

[39] Maute, K., Nikbay, M., and Farhat, C., "Coupled analytical sensitivity analysis and optimization of three-dimensional nonlinear aeroelastic systems," *AIAA Journal*, Vol. 39, No. 11, 2001, pp. 2051–2061.

[40] Martins, J. R., Alonso, J. J., and Reuther, J. J., "A coupled-adjoint sensitivity analysis method for high-fidelity aero-structural design," *Optimization and Engineering*, Vol. 6, No. 1, 2005, pp. 33–62.

[41] Martins, J. R., Alonso, J. J., and Reuther, J. J., "High-fidelity aerostructural design optimization of a supersonic business jet," *Journal of Aircraft*, Vol. 41, No. 3, 2004, pp. 523–530.

[42] Kenway, G. K., Kennedy, G. J., and Martins, J. R., "Scalable parallel approach for high-fidelity steady-state aeroelastic analysis and adjoint derivative computations," *AIAA Journal*, Vol. 52, No. 5, 2014, pp. 935–951.

[43] Kenway, G. K., and Martins, J. R., "Multi-point high-fidelity aerostructural optimization of a transport aircraft configuration," *Journal of Aircraft*, Vol. 51, No. 1, 2014, pp. 144–160.

[44] Kenway, G. K., Kennedy, G. J., and Martins, J. R., "Aerostructural optimization of the Common Research Model configuration," *15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2014, p. 3274.

[45] Zhang, Z. J., Khosravi, S., and Zingg, D. W., "High-fidelity aerostructural optimization with integrated geometry parameterization and mesh movement," *Structural and Multidisciplinary Optimization*, Vol. 55, No. 4, 2017, pp. 1217–1235.

[46] Ghazlane, I., Carrier, G., Dumont, A., and Desideri, J., "Aerostructural adjoint method for flexible wing optimization," *AIAA Paper*, Vol. 1924, 2012, p. 2012.

[47] Jansen, P. W., Perez, R. E., and RA Martins, J. R., "Aerostructural optimization of nonplanar lifting surfaces," *Journal of Aircraft*, Vol. 47, No. 5, 2010, pp. 1490–1503.

[48] Mariens, J., Elham, A., and Van Tooren, M., "Quasi-three-dimensional aerodynamic solver for multidisciplinary design optimization of lifting surfaces," *Journal of Aircraft*, Vol. 51, No. 2, 2014, pp. 547–558.

[49] Elham, A., and van Tooren, M. J., "Weight indexing for wing-shape multi-objective optimization," *AIAA Journal*, Vol. 52, No. 2, 2014, pp. 320–337.

[50] Gallay, S., and Laurendeau, E., "Preliminary-design aerodynamic model for complex configurations using lifting-line coupling algorithm," *Journal of Aircraft*, Vol. 53, No. 4, 2016, pp. 1145–1159.

[51] Variyar, A., Economon, T. D., and Alonso, J. J., "Design and Optimization of Unconventional Aircraft Configurations with Aeroelastic Constraints," *55th AIAA Aerospace Sciences Meeting*, 2017, p. 0463.

[52] Kontogiannis, S. G., Kipouros, T., and Savill, A. M., "Multiobjective aerostructural optimization for efficient transport wing conceptual design," *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2018, p. 0104.

[53] Jasa, J. P., Hwang, J. T., and Martins, J. R., "Open-source coupled aerostructural optimization using Python," *Structural and Multidisciplinary Optimization*, Vol. 57, No. 4, 2018, pp. 1815–1827.

[54] Hicken, J. E., and Zingg, D. W., "Aerodynamic optimization algorithm with integrated geometry parameterization and mesh movement," *AIAA Journal*, Vol. 48, No. 2, 2010, pp. 400–413.

[55] Gagnon, H., and Zingg, D. W., "Two-level free-form and axial deformation for exploratory aerodynamic shape optimization," *AIAA Journal*, Vol. 53, No. 7, 2015, pp. 2015–2026.

[56] Lee, C., Koo, D., and Zingg, D. W., "Comparison of B-spline surface and free-form deformation geometry control for aerodynamic optimization," *AIAA Journal*, Vol. 55, No. 1, 2017, pp. 228–240.

[57] 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, p. 0262.

[58] Koo, D., and Zingg, D. W., "Progress in aerodynamic shape optimization based on the Reynolds-averaged Navier–Stokes equations," *54th AIAA Aerospace Sciences Meeting*, 2016, p. 1292.

[59] Chernukhin, O., and Zingg, D. W., "Multimodality and global optimization in aerodynamic design," *AIAA Journal*, Vol. 51, No. 6, 2013, pp. 1342–1354.

[60] Koo, D., and Zingg, D. W., "Investigation into aerodynamic shape optimization of planar and nonplanar wings," *AIAA Journal*, Vol. 56, No. 1, 2018, pp. 250–263.

[61] Mader, C. A., Martins, J. R., Alonso, J. J., and Der Weide, E. V., "ADjoint: An approach for the rapid development of discrete adjoint solvers," *AIAA Journal*, Vol. 46, No. 4, 2008, pp. 863–873.

[62] Lyu, Z., Kenway, G. K., Paige, C., and Martins, J., "Automatic differentiation adjoint of the Reynolds-averaged Navier–Stokes equations with a turbulence model," *21st AIAA Computational Fluid Dynamics Conference*, 2013, p. 2581.

[63] Kenway, G. K., and Martins, J. R., "Multipoint aerodynamic shape optimization investigations of the common research model wing," *AIAA Journal*, Vol. 54, No. 1, 2016, pp. 113–128.

[64] Chen, S., Lyu, Z., Kenway, G. K., and Martins, J. R., "Aerodynamic shape optimization of common research model wing–body–tail configuration," *Journal of Aircraft*, Vol. 53, No. 1, 2015, pp. 276–293.

[65] Lyu, Z., and Martins, J. R., "Aerodynamic design optimization studies of a blended-wing-body aircraft," *Journal of Aircraft*, Vol. 51, No. 5, 2014, pp. 1604–1617.

[66] Lyu, Z., and Martins, J. R., "Aerodynamic shape optimization of an adaptive morphing trailing-edge wing," *Journal of Aircraft*, Vol. 52, No. 6, 2015, pp. 1951–1970.

[67] Kenway, G. K., and Martins, J. R., "Buffet-onset constraint formulation for aerodynamic shape optimization," *AIAA Journal*, Vol. 55, No. 6, 2017, pp. 1930–1947.

[68] Morris, A., Allen, C., and Rendall, T., "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, 2008, pp. 827–860.

[69] Morris, A., Allen, C., and S. Rendall, T., "Domain-element method for aerodynamic shape optimization applied to modern transport wing," *AIAA Journal*, Vol. 47, No. 7, 2009, pp. 1647–1659.

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

[71] Poole, D., Allen, C., and Rendall, T., "High-fidelity aerodynamic shape optimization using efficient orthogonal modal design variables with a constrained global optimizer," *Computers & Fluids*, Vol. 143, 2017, pp. 1–15.

[72] Allen, C. B., Poole, D. J., and Rendall, T. C., "Wing aerodynamic optimization using efficient mathematically-extracted modal design variables," *Optimization and Engineering*, Vol. 19, No. 2, 2018, pp. 453–477.

[73] Masters, D., Taylor, N., Rendall, T., and Allen, C., "Multilevel subdivision parameterization scheme for aerodynamic shape optimization," *AIAA Journal*, 2017, pp. 3288–3303.

[74] Payot, A., Rendall, T., and Allen, C., "Restricted snakes volume of solid (RSVS): A parameterisation method for topology optimisation of external aerodynamics," *Computers & Fluids*, Vol. 182, 2019, pp. 60–84.

[75] Kedward, L., Allen, C. B., and Rendall, T. C., "Efficient and exact mesh deformation using multiscale RBF interpolation," *Journal of Computational Physics*, Vol. 345, 2017, pp. 732–751.

[76] Poole, D., Allen, C., and Rendall, T., "Global optimization of wing aerodynamic optimization case exhibiting multimodality," *Journal of Aircraft*, Vol. 55, No. 4, 2018, pp. 1576–1591.

[77] Le Moigne, A., and Qin, N., "Variable-fidelity aerodynamic optimization for turbulent flows using a discrete adjoint formulation," *AIAA Journal*, Vol. 42, No. 7, 2004, pp. 1281–1292.

[78] Qin, N., Vavalle, A., Le Moigne, A., Laban, M., Hackett, K., and Weinerfelt, P., "Aerodynamic considerations of blended wing body aircraft," *Progress in Aerospace Sciences*, Vol. 40, No. 6, 2004, pp. 321–343.

[79] Qin, N., Vavalle, A., and Moigne, A. L., "Spanwise lift distribution for blended wing body aircraft." *Journal of Aircraft*, Vol. 42, No. 2, 2005, pp. 356–365.

[80] Le Moigne, A., and Qin, N., "Aerofoil profile and sweep optimisation for a blended wing-body aircraft using a discrete adjoint method," *The Aeronautical Journal*, Vol. 110, No. 1111, 2006, pp. 589–604.

[81] Wong, W., Le Moigne, A., and Qin, N., "Parallel adjoint-based optimisation of a blended wing body aircraft with shock control bumps," *The Aeronautical Journal*, Vol. 111, No. 1117, 2007, pp. 165–174.

[82] Liu, X., Qin, N., and Xia, H., "Fast dynamic grid deformation based on Delaunay graph mapping," *Journal of Computational Physics*, Vol. 211, No. 2, 2006, pp. 405–423.

[83] Wang, Y., Qin, N., and Zhao, N., "Delaunay graph and radial basis function for fast quality mesh deformation," *Journal of Computational Physics*, Vol. 294, 2015, pp. 149–172.

[84] Wang, Y., Qin, N., and Zhao, N., "Delaunay graph based inverse distance weighting for fast dynamic meshing," *Communications in Computational Physics*, Vol. 21, No. 5, 2017, pp. 1282–1309.

[85] Vavalle, A., and Qin, N., "Iterative response surface based optimization scheme for transonic airfoil design," *Journal of Aircraft*, Vol. 44, No. 2, 2007, pp. 365–376.

[86] Zhu, F., and Qin, N., "Intuitive class/shape function parameterization for airfoils," *AIAA Journal*, Vol. 52, No. 1, 2013, pp. 17–25.

[87] Hinchliffe, B., and Qin, N., "Using surface sensitivity from mesh adjoint for transonic wing drag reduction," *AIAA Journal*, Vol. 55, No. 3, 2017, pp. 818–831.

[88] Mura, G. L., Hinchliffe, B. L., Qin, N., and Brezillon, J., "Efficient method to eliminate mesh sensitivity in adjoint-based optimization," *AIAA Journal*, Vol. 55, No. 4, 2017, pp. 1140–1151.

[89] Mura, G. L., Hinchliffe, B. L., Qin, N., and Brezillon, J., "Nonconsistent mesh movement and sensitivity calculation on adjoint aerodynamic optimization," *AIAA Journal*, Vol. 56, No. 4, 2018, pp. 1541–1553.

[90] Jones, D., Müller, J.-D., and Christakopoulos, F., "Preparation and assembly of discrete adjoint CFD codes," *Computers & Fluids*, Vol. 46, No. 1, 2011, pp. 282–286.

[91] Müller, J.-D., Gugala, M., Xu, S., Hüchelheim, J., Mohanamuraly, P., and Imam-Lawal, O. R., "Introducing STAMPS: an open-source discrete adjoint CFD solver using source-transformation AD," *11th ASMO UK/ISSMO/NOED2016: International Conference on Numerical Optimisation Methods for Engineering Design*, 2016.

[92] Xu, S., Jahn, W., and Müller, J.-D., "CAD-based shape optimisation with CFD using a discrete adjoint," *International Journal for Numerical Methods in Fluids*, Vol. 74, No. 3, 2014, pp. 153–168.

[93] Zhang, X., Wang, Y., Gugala, M., and Müller, J.-D., "Geometric continuity constraints for adjacent NURBS patches in shape optimisation," *ECCOMAS Congress*, Vol. 2, 2016, p. 9316.

[94] Yu, G., Müller, J.-D., Jones, D., and Christakopoulos, F., "CAD-based shape optimisation using adjoint sensitivities," *Computers & Fluids*, Vol. 46, No. 1, 2011, pp. 512–516.

[95] Xu, S., Timme, S., Mykhaskiv, O., and Müller, J.-D., "Wing-body junction optimisation with CAD-based parametrisation including a moving intersection," *Aerospace Science and Technology*, Vol. 68, 2017, pp. 543–551.

[96] Zhang, X., Jesudasan, R., and Müller, J.-D., "Adjoint-based aerodynamic optimisation of wing shape using Non-Uniform Rational B-splines," *Evolutionary and Deterministic Methods for Design Optimization and Control With Applications to Industrial and Societal Problems*, Springer, 2019, pp. 143–158.

[97] Forrester, A., Sobester, A., and Keane, A., *Engineering design via surrogate modelling: a practical guide*, John Wiley & Sons, 2008.

[98] Toal, D., Bressloff, N., and Keane, A., "Kriging hyperparameter tuning strategies," *AIAA Journal*, Vol. 46, No. 5, 2008, pp. 1240–1252.

[99] Toal, D., Forrester, A., Bressloff, N., Keane, A., and Holden, C., "An adjoint for likelihood maximization," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, Vol. 465, No. 2111, 2009, pp. 3267–3287.

[100] Toal, D., Bressloff, N., Keane, A., and Holden, C., "Geometric filtration using proper orthogonal decomposition for aerodynamic design optimization," *AIAA Journal*, Vol. 48, No. 5, 2010, pp. 916–928.

[101] Forrester, A., Sóbester, A., and Keane, A., "Multi-fidelity optimization via surrogate modelling," *Proceedings of the royal society a: mathematical, physical and engineering sciences*, Vol. 463, No. 2088, 2007, pp. 3251–3269.

[102] Toal, D., and Keane, A., "Efficient multipoint aerodynamic design optimization via cokriging," *Journal of Aircraft*, Vol. 48, No. 5, 2011, pp. 1685–1695.

[103] Toal, D., "Some considerations regarding the use of multi-fidelity Kriging in the construction of surrogate models," *Structural and Multidisciplinary Optimization*, Vol. 51, No. 6, 2015, pp. 1223–1245.

[104] Viswanath, A., Forrester, A., and Keane, A., "Dimension reduction for aerodynamic design optimization," *AIAA Journal*, Vol. 49, No. 6, 2011, pp. 1256–1266.

[105] Viswanath, A., Forrester, A., and Keane, A., "Constrained design optimization using generative topographic mapping," *AIAA Journal*, Vol. 52, No. 5, 2014, pp. 1010–1023.

[106] Parr, J., Keane, A., Forrester, A., and Holden, C., "Infill sampling criteria for surrogate-based optimization with constraint handling," *Engineering Optimization*, Vol. 44, No. 10, 2012, pp. 1147–1166.

[107] Parr, J., Forrester, A., Keane, A., and Holden, C., "Enhancing infill sampling criteria for surrogate-based constrained optimization," *Journal of Computational Methods in Sciences and Engineering*, Vol. 12, No. 1, 2, 2012, pp. 25–45.

[108] Sóbester, A., "Concise airfoil representation via case-based knowledge capture," *AIAA Journal*, Vol. 47, No. 5, 2009, pp. 1209–1218.

[109] Sóbester, A., and Powell, S., "Design space dimensionality reduction through physics-based geometry re-parameterization," *Optimization and Engineering*, Vol. 14, No. 1, 2013, pp. 37–59.

[110] Sóbester, A., Forrester, A., Toal, D., Tresidder, E., and Tucker, S., "Engineering design applications of surrogate-assisted optimization techniques," *Optimization and Engineering*, Vol. 15, No. 1, 2014, pp. 243–265.

[111] Kontogiannis, K., Sóbester, A., and Taylor, N., "Efficient parameterization of waverider geometries," *Journal of Aircraft*, Vol. 54, No. 3, 2016, pp. 890–901.

[112] Dulikravich, G. S., "Aerodynamic shape design and optimization: status and trends," *Journal of Aircraft*, Vol. 29, No. 6, 1992, pp. 1020–1026.

[113] Newman III, J. C., Taylor III, A. C., Barnwell, R. W., Newman, P. A., and Hou, G. J.-W., "Overview of sensitivity analysis and shape optimization for complex aerodynamic configurations," *Journal of Aircraft*, Vol. 36, No. 1, 1999, pp. 87–96.

[114] Skinner, S. N., and Zare-Behtash, H., "State-of-the-art in aerodynamic shape optimisation methods," *Applied Soft Computing*, Vol. 62, 2018, pp. 933–962.

[115] Forrester, A., and Keane, A., "Recent advances in surrogate-based optimization," *Progress in Aerospace Sciences*, Vol. 45, No. 1-3, 2009, pp. 50–79.

[116] Lian, Y., Oyama, A., and Liou, M.-S., "Progress in design optimization using evolutionary algorithms for aerodynamic problems," *Progress in Aerospace Sciences*, Vol. 46, No. 5-6, 2010, pp. 199–223.

[117] Han, Z., "Kriging surrogate model and its application to design optimization: A review of recent progress," *Acta Aeronautica et Astronautica Sinica*, Vol. 37, No. 11, 2016, pp. 3197–3225.

[118] Masters, D., Taylor, N., Rendall, T., Allen, C., and Poole, D., "Geometric comparison of aerofoil shape parameterization methods," *AIAA Journal*, Vol. 55, No. 5, 2017, pp. 1575–1589.

[119] Zhao, H., Gao, Z., Xu, F., and Zhang, Y., "Review of robust aerodynamic design optimization for air vehicles," *Archives of Computational Methods in Engineering*, 2018, pp. 1–48.

[120] Economon, T. D., Palacios, F., Copeland, S. R., Lukaczyk, T. W., and Alonso, J. J., "SU2: an open-source suite for multiphysics simulation and design," *AIAA Journal*, Vol. 54, No. 3, 2016, pp. 828–846.

[121] Palacios, F., Alonso, J., Duraisamy, K., Colonno, M., Hicken, J., Aranake, A., Campos, A., Copeland, S., Economon, T., Lonkar, A., et al., "Stanford University Unstructured (SU2): an open-source integrated computational environment

for multi-physics simulation and design," *51st AIAA Aerospace Sciences Meeting*, 2013, p. 287.

[122] Yang, G., and Da Ronch, A., "Aerodynamic shape optimisation of benchmark problems using SU2," *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2018, p. 0412.

[123] Yang, G., Da Ronch, A., Drofelnik, J., and Xie, Z.-T., "Sensitivity assessment of optimal solution in aerodynamic design optimisation using SU2," *Aerospace Science and Technology*, Vol. 81, 2018, pp. 362–374.

[124] Yang, G., Da Ronch, A., Kharlamov, D., and Drofelnik, J., "Wing twist optimisation using aerodynamic solvers of different fidelity," *31st Congress of the International Council of the Aeronautical Sciences*, 2018, p. 0440.

[125] Poole, D. J., Allen, C. B., and Rendall, T., "Control point-based aerodynamic shape optimization applied to AIAA ADODG test cases," *53rd AIAA Aerospace Sciences Meeting*, 2015, p. 1947.

[126] Katz, J., and Plotkin, A., *Low-speed aerodynamics*, Vol. 13, Cambridge university press, 2001.

[127] Albano, E., and Rodden, W. P., "A doublet-lattice method for calculating lift distributions on oscillating surfaces in subsonic flows," *AIAA Journal*, Vol. 7, No. 2, 1969, pp. 279–285.

[128] Kennedy, G. J., and Martins, J. R., "A parallel aerostructural optimization framework for aircraft design studies," *Structural and Multidisciplinary Optimization*, Vol. 50, No. 6, 2014, pp. 1079–1101.

[129] Drela, M., "Three-dimensional integral boundary layer formulation for general configurations," *21st AIAA Computational Fluid Dynamics Conference*, 2013, p. 2437.

[130] van Dam, C., "Aircraft design and the importance of drag prediction," *CFD-Based Aircraft Drag Prediction and Reduction*, Vol. 2, 2003, pp. 1–37.

[131] 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*, 2014, p. 0568.

[132] Kulfan, B. M., "Universal parametric geometry representation method," *Journal of Aircraft*, Vol. 45, No. 1, 2008, pp. 142–158.

[133] Sobieczky, H., "Parametric airfoils and wings," *Recent development of aerodynamic design methodologies*, Springer, 1999, pp. 71–87.

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

[135] Samareh, J. A., "Novel multidisciplinary shape parameterization approach," *Journal of Aircraft*, Vol. 38, No. 6, 2001, pp. 1015–1024.

[136] Samareh, J. A., "Survey of shape parameterization techniques for high-fidelity multidisciplinary shape optimization," *AIAA Journal*, Vol. 39, No. 5, 2001, pp. 877–884.

[137] Castonguay, P., and Nadarajah, S., "Effect of shape parameterization on aerodynamic shape optimization," *45th AIAA Aerospace Sciences Meeting and Exhibit*, 2007, p. 59.

[138] Mousavi, A., Castonguay, P., and Nadarajah, S., "Survey of shape parameterization techniques and its effect on three-dimensional aerodynamic shape optimization," *18th AIAA computational fluid dynamics conference*, 2007, p. 3837.

[139] Kulfan, B., and Bussoletti, J., "" Fundamental" parameteric geometry representations for aircraft component shapes," *11th AIAA/ISSMO multidisciplinary analysis and optimization conference*, 2006, p. 6948.

[140] Straathof, M. H., and L. Van Tooren, M. J., "Extension to the class-shape-transformation method based on B-splines," *AIAA Journal*, Vol. 49, No. 4, 2011, pp. 780–790.

[141] Zhu, F., "Geometric parameterisation and aerodynamic shape optimisation," Ph.D. thesis, University of Sheffield, 2014.

[142] Nemec, M., and Zingg, D. W., "Newton–Krylov algorithm for aerodynamic design using the Navier–Stokes equations," *AIAA Journal*, Vol. 40, No. 6, 2002, pp. 1146–1154.

[143] Bisson, F., Nadarajah, S., and Shi-Dong, D., "Adjoint-based aerodynamic optimization of benchmark problems," *52nd Aerospace Sciences Meeting*, 2014, p. 0412.

[144] Robinson, T. T., Armstrong, C. G., Chua, H. S., Othmer, C., and Grahs, T., "Optimizing parameterized CAD geometries using sensitivities based on adjoint functions," *Computer-Aided Design and Applications*, Vol. 9, No. 3, 2012, pp. 253–268.

[145] Agarwal, D., Robinson, T. T., Armstrong, C. G., Marques, S., Vasilopoulos, I., and Meyer, M., "Parametric design velocity computation for CAD-based design optimization using adjoint methods," *Engineering with Computers*, Vol. 34, No. 2, 2018, pp. 225–239.

[146] Sederberg, T. W., and Parry, S. R., "Free-form deformation of solid geometric models," *ACM SIGGRAPH computer graphics*, Vol. 20, No. 4, 1986, pp. 151–160.

[147] Samareh, J. A., "Aerodynamic shape optimization based on free-form deformation," *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2004, p. 4630.

[148] Chatterjee, A., "An introduction to the proper orthogonal decomposition," *Current science*, 2000, pp. 808–817.

[149] Ghoman, S., Wang, Z., Chen, P., and Kapania, R., "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, p. 1808.

[150] Gordon, W. J., and Hall, C. A., "Construction of curvilinear co-ordinate systems and applications to mesh generation," *International Journal for Numerical Methods in Engineering*, Vol. 7, No. 4, 1973, pp. 461–477.

[151] Eriksson, L., "Generation of boundary-conforming grids around wing-body configurations using transfinite interpolation," *AIAA Journal*, Vol. 20, No. 10, 1982, pp. 1313–1320.

[152] Soni, B., "Two-and three-dimensional grid generation for internal flow applications of computational fluid dynamics," *7th Computational Physics Conference*, 1985, p. 1526.

[153] Reuther, J., Jameson, A., Farmer, J., Martinelli, L., and Saunders, D., "Aerodynamic shape optimization of complex aircraft configurations via an adjoint formulation," *34th Aerospace Sciences Meeting and Exhibit*, 1996, p. 94.

[154] Jones, W., and Samareh-Abolhassani, J., "A grid generation system for multidisciplinary design optimization," *12th Computational Fluid Dynamics Conference*, 1995, p. 1689.

[155] Batina, J. T., "Unsteady Euler algorithm with unstructured dynamic mesh for complex-aircraft aerodynamic analysis," *AIAA Journal*, Vol. 29, No. 3, 1991, pp. 327–333.

[156] Nielsen, E. J., and Anderson, W. K., "Aerodynamic design optimization on unstructured meshes using the Navier–Stokes equations," *AIAA Journal*, Vol. 37, No. 11, 1999, pp. 1411–1419.

[157] Nielsen, E. J., and Anderson, W. K., "Recent improvements in aerodynamic design optimization on unstructured meshes," *AIAA Journal*, Vol. 40, No. 6, 2002, pp. 1155–1163.

[158] Farhat, C., Degand, C., Koobus, B., and Lesoinne, M., "Torsional springs for two-dimensional dynamic unstructured fluid meshes," *Computer methods in applied mechanics and engineering*, Vol. 163, No. 1-4, 1998, pp. 231–245.

[159] Degand, C., and Farhat, C., "A three-dimensional torsional spring analogy method for unstructured dynamic meshes," *Computers & structures*, Vol. 80, No. 3-4, 2002, pp. 305–316.

[160] Johnson, A. A., and Tezduyar, T. E., "Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces," *Computer methods in applied mechanics and engineering*, Vol. 119, No. 1-2, 1994, pp. 73–94.

[161] Truong, A. H., Oldfield, C. A., and Zingg, D. W., "Mesh movement for a discrete-adjoint Newton–Krylov algorithm for aerodynamic optimization," *AIAA Journal*, Vol. 46, No. 7, 2008, pp. 1695–1704.

[162] Dwight, R. P., "Robust mesh deformation using the linear elasticity equations," *Computational Fluid Dynamics 2006*, Springer, 2009, pp. 401–406.

[163] De Boer, A., Van der Schoot, M., and Bijl, H., "Mesh deformation based on radial basis function interpolation," *Computers & structures*, Vol. 85, No. 11-14, 2007, pp. 784–795.

[164] Rendall, T., and Allen, C., "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.

[165] Jakobsson, S., and Amoignon, O., "Mesh deformation using radial basis functions for gradient-based aerodynamic shape optimization," *Computers & Fluids*, Vol. 36, No. 6, 2007, pp. 1119–1136.

[166] Rendall, T. C., and Allen, C. B., "Efficient mesh motion using radial basis functions with data reduction algorithms," *Journal of Computational Physics*, Vol. 228, No. 17, 2009, pp. 6231–6249.

[167] Rendall, T., and Allen, C., "Parallel efficient mesh motion using radial basis functions with application to multi-bladed rotors," *International journal for numerical methods in engineering*, Vol. 81, No. 1, 2010, pp. 89–105.

[168] Rendall, T. C., and Allen, C. B., "Reduced surface point selection options for efficient mesh deformation using radial basis functions," *Journal of Computational Physics*, Vol. 229, No. 8, 2010, pp. 2810–2820.

[169] Wang, G., Mian, H. H., Ye, Z.-Y., and Lee, J.-D., "Improved point selection method for hybrid-unstructured mesh deformation using radial basis functions," *AIAA Journal*, Vol. 53, No. 4, 2014, pp. 1016–1025.

[170] Gillebaart, T., Blom, D., Van Zuijlen, A., and Bijl, H., "Adaptive radial basis function mesh deformation using data reduction," *Journal of Computational Physics*, Vol. 321, 2016, pp. 997–1025.

[171] Poirier, V., and Nadarajah, S., "Efficient reduced-radial basis function-based mesh deformation within an adjoint-based aerodynamic optimization framework," *Journal of Aircraft*, Vol. 53, No. 6, 2016, pp. 1905–1921.

[172] Allen, C. B., and Rendall, T. C., "CFD-based optimization of hovering rotors using radial basis functions for shape parameterization and mesh deformation," *Optimization and Engineering*, Vol. 14, No. 1, 2013, pp. 97–118.

[173] Perez, R. E., Jansen, P. W., and Martins, J. R., "pyOpt: a Python-based object-oriented framework for nonlinear constrained optimization," *Structural and Multidisciplinary Optimization*, Vol. 45, No. 1, 2012, pp. 101–118.

[174] Karush, W., "Minima of functions of several variables with inequalities as side conditions," *Traces and Emergence of Nonlinear Programming*, Springer, 2014, pp. 217–245.

[175] Kuhn, H., and Tucker, A., "Nonlinear programming," *Proceedings of 2nd Berkeley Symposium. Berkeley: University of California Press*, 1951, pp. 481–492.

[176] Gill, P. E., Murray, W., and Saunders, M. A., "SNOPT: An SQP algorithm for large-scale constrained optimization," *SIAM review*, Vol. 47, No. 1, 2005, pp. 99–131.

[177] Kraft, D., "A software package for sequential quadratic programming," *Forschungsbericht- Deutsche Forschungs- und Versuchsanstalt fur Luft- und Raumfahrt*, 1988.

[178] Martins, J. R., and Hwang, J. T., "Review and unification of methods for computing derivatives of multidisciplinary computational models," *AIAA Journal*, Vol. 51, No. 11, 2013, pp. 2582–2599.

[179] Squire, W., and Trapp, G., "Using complex variables to estimate derivatives of real functions," *SIAM review*, Vol. 40, No. 1, 1998, pp. 110–112.

[180] Martins, J. R., Sturdza, P., and Alonso, J. J., "The complex-step derivative approximation," *ACM Transactions on Mathematical Software (TOMS)*, Vol. 29, No. 3, 2003, pp. 245–262.

[181] Griewank, A., *Evaluating derivatives: principles and techniques of algorithmic differentiation*, SIAM, Philadelphia, 2000.

[182] Pironneau, O., "On optimum design in fluid mechanics," *Journal of Fluid Mechanics*, Vol. 64, No. 1, 1974, pp. 97–110.

[183] Giles, M. B., Duta, M. C., M-uacute, J.-D., and Pierce, N. A., "Algorithm developments for discrete adjoint methods," *AIAA Journal*, Vol. 41, No. 2, 2003, pp. 198–205.

[184] Albring, T. A., Sagebaum, M., and Gauger, N. R., "Efficient aerodynamic design using the discrete adjoint method in SU2," *17th AIAA/ISSMO multidisciplinary analysis and optimization conference*, 2016, p. 3518.

[185] Mishra, A. A., Mukhopadhaya, J., Iaccarino, G., and Alonso, J., "Uncertainty estimation module for turbulence model predictions in SU2," *AIAA Journal*, Vol. 57, No. 3, 2018, pp. 1066–1077.

[186] Holland, J. H., "Outline for a logical theory of adaptive systems," *Journal of the ACM*, Vol. 9, No. 3, 1962, pp. 297–314.

[187] Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P., "Optimization by simulated annealing," *Science*, Vol. 220, No. 4598, 1983, pp. 671–680.

[188] Colorni, A., Dorigo, M., Maniezzo, V., et al., "Distributed optimization by ant colonies," *Proceedings of the first European conference on artificial life*, Vol. 142, Cambridge, MA, 1992, pp. 134–142.

[189] Dorigo, M., Maniezzo, V., Colorni, A., et al., "Ant system: optimization by a colony of cooperating agents," *IEEE Transactions on Systems, man, and cybernetics, Part B: Cybernetics*, Vol. 26, No. 1, 1996, pp. 29–41.

[190] Eberhart, R., and Kennedy, J., "A new optimizer using particle swarm theory," *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, Ieee, 1995, pp. 39–43.

[191] Nelder, J. A., and Mead, R., "A simplex method for function minimization," *The computer journal*, Vol. 7, No. 4, 1965, pp. 308–313.

[192] Widhalm, M., Ronzheimer, A., and Hepperle, M., "Comparison between gradient-free and adjoint based aerodynamic optimization of a flying wing transport aircraft in the preliminary design," *25th AIAA applied aerodynamics conference*, 2007, p. 4060.

[193] Quagliarella, D., and Della Cioppa, A., "Genetic algorithms applied to the aerodynamic design of transonic airfoils," *Journal of Aircraft*, Vol. 32, No. 4, 1995, pp. 889–891.

[194] Antunes, A. P., and Azevedo, J. L. F., "Studies in aerodynamic optimization based on genetic algorithms," *Journal of Aircraft*, Vol. 51, No. 3, 2014, pp. 1002–1012.

[195] Lyu, Z., Xu, Z., and Martins, J., "Benchmarking optimization algorithms for wing aerodynamic design optimization," *Proceedings of the 8th International Conference on Computational Fluid Dynamics, Chengdu, Sichuan, China*, Vol. 11, 2014.

[196] Lee, J., and Hajela, P., "Parallel genetic algorithm implementation in multidisciplinary rotor blade design," *Journal of Aircraft*, Vol. 33, No. 5, 1996, pp. 962–969.

[197] Oyama, A., and Liou, M.-S., "Multiobjective optimization of rocket engine pumps using evolutionary algorithm," *Journal of Propulsion and Power*, Vol. 18, No. 3, 2002, pp. 528–535.

[198] Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T., "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE transactions on evolutionary computation*, Vol. 6, No. 2, 2002, pp. 182–197.

[199] Gudla, P. K., and Ganguli, R., "An automated hybrid genetic-conjugate gradient algorithm for multimodal optimization problems," *Applied Mathematics and Computation*, Vol. 167, No. 2, 2005, pp. 1457–1474.

[200] Muyl, F., Dumas, L., and Herbert, V., "Hybrid method for aerodynamic shape optimization in automotive industry," *Computers & Fluids*, Vol. 33, No. 5-6, 2004, pp. 849–858.

[201] Chelouah, R., and Siarry, P., "Genetic and Nelder–Mead algorithms hybridized for a more accurate global optimization of continuous multiminima functions," *European Journal of Operational Research*, Vol. 148, No. 2, 2003, pp. 335–348.

[202] Shi, Y., and Eberhart, R. C., "Parameter selection in particle swarm optimization," *International conference on evolutionary programming*, Springer, 1998, pp. 591–600.

[203] Eberhart, R. C., and Shi, Y., "Comparing inertia weights and constriction factors in particle swarm optimization," *Proceedings of the 2000 congress on evolutionary computation. CEC00 (Cat. No. 00TH8512)*, Vol. 1, IEEE, 2000, pp. 84–88.

[204] Venter, G., and Sobieszczanski-Sobieski, J., "Multidisciplinary optimization of a transport aircraft wing using particle swarm optimization," *structural and Multidisciplinary optimization*, Vol. 26, No. 1-2, 2004, pp. 121–131.

[205] Blasi, L., and Core, G. D., "Particle swarm approach in finding optimum aircraft configuration," *Journal of Aircraft*, Vol. 44, No. 2, 2007, pp. 679–683.

[206] Hassan, R., Cohanim, B., De Weck, O., and Venter, G., "A comparison of particle swarm optimization and the genetic algorithm," *46th AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference*, 2005, p. 1897.

[207] Mukesh, R., Pandiyarajan, R., Selvakumar, U., and Lingadurai, K., "Influence of search algorithms on aerodynamic design optimisation of aircraft wings," *Procedia engineering*, Vol. 38, 2012, pp. 2155–2163.

[208] Giunta, A., and Watson, L., "A comparison of approximation modeling techniques-Polynomial versus interpolating models," *7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, 1998, p. 4758.

[209] Krige, D. G., "A statistical approach to some basic mine valuation problems on the Witwatersrand," *Journal of the Southern African Institute of Mining and Metallurgy*, Vol. 52, No. 6, 1951, pp. 119–139.

[210] Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P., "Design and analysis of computer experiments," *Statistical science*, 1989, pp. 409–423.

[211] Mullur, A. A., and Messac, A., "Extended radial basis functions: more flexible and effective metamodeling," *AIAA Journal*, Vol. 43, No. 6, 2005, pp. 1306–1315.

[212] Elanayar, V. T., and Shin, Y. C., "Radial basis function neural network for approximation and estimation of nonlinear stochastic dynamic systems," *IEEE transactions on neural networks*, Vol. 5, No. 4, 1994, pp. 594–603.

[213] Smola, A. J., and Schölkopf, B., "A tutorial on support vector regression," *Statistics and computing*, Vol. 14, No. 3, 2004, pp. 199–222.

[214] Xiu, D., *Numerical methods for stochastic computations: a spectral method approach*, Princeton University Press, 2010.

[215] Queipo, N. V., Haftka, R. T., Shyy, W., Goel, T., Vaidyanathan, R., and Tucker, P. K., "Surrogate-based analysis and optimization," *Progress in Aerospace Sciences*, Vol. 41, No. 1, 2005, pp. 1–28.

[216] Giunta, A., Wojtkiewicz, S., and Eldred, M., "Overview of modern design of experiments methods for computational simulations," *41st Aerospace Sciences Meeting and Exhibit*, 2003, p. 649.

[217] Chung, H.-S., and Alonso, J., "Using gradients to construct cokriging approximation models for high-dimensional design optimization problems," *40th AIAA Aerospace Sciences Meeting & Exhibit*, 2002, p. 317.

[218] Laurenceau, J., and Sagaut, P., "Building efficient response surfaces of aerodynamic functions with kriging and cokriging," *AIAA Journal*, Vol. 46, No. 2, 2008, pp. 498–507.

[219] Han, Z.-H., Görtz, S., and Zimmermann, R., "Improving variable-fidelity surrogate modeling via gradient-enhanced kriging and a generalized hybrid bridge function," *Aerospace Science and Technology*, Vol. 25, No. 1, 2013, pp. 177–189.

[220] Han, Z.-H., Zimmermann, R., and Goretz, S., "A new cokriging method for variable-fidelity surrogate modeling of aerodynamic data," *48th AIAA Aerospace Sciences Meeting*, 2010, p. 1225.

[221] Han, Z.-H., and Görtz, S., "Hierarchical kriging model for variable-fidelity surrogate modeling," *AIAA Journal*, Vol. 50, No. 9, 2012, pp. 1885–1896.

[222] Song, W., and Keane, A., "A study of shape parameterisation methods for airfoil optimisation," *10th AIAA/ISSMO multidisciplinary analysis and optimization conference*, 2004, p. 4482.

[223] Amoiralis, E. I., and Nikolos, I. K., "Freeform deformation versus B-spline representation in inverse airfoil design," *Journal of Computing and Information Science in Engineering*, Vol. 8, No. 2, 2008, p. 024001.

[224] 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, p. 9050.

[225] Zhu, F., Qin, N., Burnaev, E., Bernstein, A., and Chernova, S., "Comparison of three geometric parameterization methods and their effect on aerodynamic optimization," *Eurogen*, 2011, pp. 758–772.

[226] Masters, D., Poole, D., Taylor, N., Rendall, T., and Allen, C., "Influence of shape parameterization on a benchmark aerodynamic optimization problem," *Journal of Aircraft*, Vol. 54, No. 6, 2017, pp. 2242–2256.

[227] LeDoux, S. T., Vassberg, J. C., Young, D. P., Fugal, S., Kamenetskiy, D., Huffman, W. P., Melvin, R. G., and Smith, M. F., "Study based on the AIAA aerodynamic design optimization discussion group test cases," *AIAA Journal*, Vol. 53, No. 7, 2015, pp. 1910–1935.

[228] Poole, D. J., Allen, C. B., and Rendall, T., "Comparison of local and global constrained aerodynamic shape optimization," *32nd AIAA Applied Aerodynamics Conference*, 2014, p. 3223.

[229] Yu, Y., Lyu, Z., Xu, Z., and Martins, J. R., "On the influence of optimization algorithm and initial design on wing aerodynamic shape optimization," *Aerospace Science and Technology*, Vol. 75, 2018, pp. 183–199.

[230] Poole, D., Allen, C., and Rendall, T., "Comparison of point design and range-based objectives for transonic aerofoil optimization," *AIAA Journal*, Vol. 56, No. 8, 2018, pp. 3240–3256.

[231] Keane, A. J., "Genetic algorithm optimization of multi-peak problems: studies in convergence and robustness," *Artificial Intelligence in Engineering*, Vol. 9, No. 2, 1995, pp. 75–83.

[232] Masters, D. A., Taylor, N. J., Rendall, T., Allen, C. B., and Poole, D. J., "Review of aerofoil parameterisation methods for aerodynamic shape optimisation," *53rd AIAA Aerospace Sciences Meeting*, 2015, p. 0761.

[233] Van Dam, C., Vander Kam, J., and Paris, J., "Design-oriented high-lift methodology for general aviation and civil transport aircraft," *Journal of Aircraft*, Vol. 38, No. 6, 2001, pp. 1076–1084.

[234] Elham, A., "Adjoint quasi-three-dimensional aerodynamic solver for multi-fidelity wing aerodynamic shape optimization," *Aerospace Science and Technology*, Vol. 41, 2015, pp. 241–249.

[235] Kuchemann, D., "The aerodynamic design of aircraft," *Progress in aeronautical sciences, 1965, 6,271 (Pergamon, London)*, 1978.

[236] Dagenhart, J., and Saric, W. S., "Crossflow stability and transition experiments in swept-wing flow," *NASA/TP-1999-209344*, 1999.

[237] Gallay, S., Ghasemi, S., and Laurendeau, E., "Sweep effects on non-linear lifting line theory near stall," *52nd Aerospace Sciences Meeting*, 2014, p. 1105.

[238] Bourgault-Côté, S., Ghasemi, S., Mosahebi, A., and Laurendeau, É., "Extension of a two-dimensional Navier–Stokes solver for infinite swept flow," *AIAA Journal*, Vol. 55, No. 2, 2017, pp. 662–667.

[239] Parenteau, M., Laurendeau, É., and Carrier, G., "Combined high-speed and high-lift wing aerodynamic optimization using a coupled VLM-2.5D RANS approach," *Aerospace Science and Technology*, Vol. 76, 2018, pp. 484–496.

[240] Franciolini, M., Da Ronch, A., Drofelnik, J., Raveh, D., and Crivellini, A., "Efficient infinite-swept wing solver for steady and unsteady compressible flows," *Aerospace Science and Technology*, Vol. 72, 2018, pp. 217–229.

[241] Drofelnik, J., Da Ronch, A., Kharlamov, D., and Sun, Y., "Rapid calculation of unsteady aircraft loads," *7th European Conference on Computational Fluid Dynamics*, 2018, p. 2108.

[242] Drofelnik, J., Da Ronch, A., Franciolini, M., and Crivellini, A., "Fast identification of transonic buffet envelope using computational fluid dynamics," *Aircraft Engineering and Aerospace Technology*, 2018.

[243] Jameson, A., Schmidt, W., and Turkel, E., "Numerical solution of the Euler equations by finite volume methods using Runge Kutta time stepping schemes," *14th fluid and plasma dynamics conference*, 1981, p. 1259.

[244] Wilcox, D. C., et al., *Turbulence modeling for CFD*, Vol. 2, DCW industries La Canada, CA, 1998.

[245] White, F. M., and Corfield, I., *Viscous fluid flow*, Vol. 3, McGraw-Hill New York, 2006.

[246] Spalart, P., and Allmaras, S., "A one-equation turbulence model for aerodynamic flows," *La Recherche Aerospatiale*, , No. 1, 1994, pp. 5–21.

[247] Economon, T. D., Palacios, F., and Alonso, J. J., "A viscous continuous adjoint approach for the design of rotating engineering applications," *21st AIAA Computational Fluid Dynamics Conference*, 2013, p. 2580.

[248] Economon, T. D., Palacios, F., and Alonso, J. J., "Unsteady continuous adjoint approach for aerodynamic design on dynamic meshes," *AIAA Journal*, Vol. 53, No. 9, 2015, pp. 2437–2453.

[249] Palacios, F., Economon, T. D., and Alonso, J. J., "Large-scale aircraft design using SU2," *53rd AIAA Aerospace Sciences Meeting*, 2015, p. 1946.

[250] Drofelnik, J., and Da Ronch, A., "2.5D+ User Guide," *DLR-Tau Documentation*, 2017.

[251] Gallay, S., and Laurendeau, E., "Nonlinear generalized lifting-line coupling algorithms for pre/poststall flows," *AIAA Journal*, Vol. 53, No. 7, 2015, pp. 1784–1792.

[252] Anderson, G. R., Nemec, M., and Aftosmis, M. J., "Aerodynamic shape optimization benchmarks with error control and automatic parameterization," *53rd AIAA Aerospace Sciences Meeting*, 2015, p. 1719.

[253] Fabiano, E., and Mavriplis, D. J., "Adjoint-based aerodynamic design on unstructured meshes," *54th AIAA Aerospace Sciences Meeting*, 2016, p. 1295.

[254] Méheut, M., Destarac, D., Ben Khelil, S., Carrier, G., Dumont, A., and Peter, J., "Gradient-based single and multi-points aerodynamic optimizations with the elsA software," *53rd AIAA Aerospace Sciences Meeting*, 2015, p. 0263.

[255] Destarac, D., Carrier, G., Anderson, G. R., Nadarajah, S., Poole, D. J., Vassberg, J. C., and Zingg, D. W., "Example of a pitfall in aerodynamic shape optimization," *AIAA Journal*, Vol. 56, No. 4, 2018, pp. 1532–1540.

[256] 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*, 2011, p. 3176.

[257] Ou, K., Jameson, A., and Vassberg, J. C., "Airfoils supporting non-unique transonic solutions for unsteady viscous flows," *7th AIAA Theoretical Fluid Mechanics Conference*, 2014, p. 2927.

[258] Jameson, A., Vassberg, J. C., and Ou, K., "Further studies of airfoils supporting non-unique solutions in transonic flow," *AIAA Journal*, Vol. 50, No. 12, 2012, pp. 2865–2881.

[259] Cook, P., Firmin, M., and McDonald, M., "Aerofoil RAE 2822 - Pressure Distributions, and Boundary Layer and Wake Measurements," AGARD-AR-138, NATO, 1979.

[260] Da Ronch, A., Panzeri, M., Drofelnik, J., and d'Ippolito, R., "Data-driven optimisation of closure coefficients of a turbulence model," *6th CEAS Air & Space Conference*, 2017, p. 912.

[261] Meredith, P., "Viscous phenomena affecting high-lift systems and suggestions for future CFD development," AGARD-CP-515, 1993.

[262] Barakos, G., and Drikakis, D., "Numerical simulation of transonic buffet flows using various turbulence closures," *International Journal of Heat and Fluid Flow*, Vol. 21, No. 5, 2000, pp. 620–626.

[263] Giannelis, N. F., Levinski, O., and Vio, G. A., "Influence of Mach number and angle of attack on the two-dimensional transonic buffet phenomenon," *Aerospace Science and Technology*, Vol. 78, 2018, pp. 89–101.

[264] Kuzmin, A., "Non-unique transonic flows over airfoils," *Computers & Fluids*, Vol. 63, 2012, pp. 1–8.

[265] Kuzmin, A., and Ryabinin, A., "Transonic aerofoils admitting anomalous behaviour of lift coefficient," *The Aeronautical Journal*, Vol. 118, No. 1202, 2014, pp. 425–433.

[266] Kroo, I., "Drag due to lift: concepts for prediction and reduction," *Annual Review of Fluid Mechanics*, Vol. 33, No. 1, 2001, pp. 587–617.

[267] Phillips, W., "Lifting-line analysis for twisted wings and washout-optimized wings," *Journal of Aircraft*, Vol. 41, No. 1, 2004, pp. 128–136.

[268] Phillips, W. F., Fugal, S. R., and Spall, R. E., "Minimizing induced drag with wing twist, computational-fluid-dynamics validation," *Journal of Aircraft*, Vol. 43, No. 2, 2006, pp. 437–444.

[269] Glauert, H., *The elements of aerofoil and airscrew theory*, Cambridge University Press, 1983.

[270] Gregory, N., and O'Reilly, C. L., "Low-speed aerodynamic characteristics of NACA 0012 aerofoil section, including the effects of upper-surface roughness simulating hoar frost," Reports and memoranda, 1970.

[271] Schmitt, V., and Charpin, F., "Pressure distributions on the ONERA-M6-Wing at transonic Mach numbers," *Experimental data base for computer program assessment*, Vol. 4, 1979.