



**QUEEN'S
UNIVERSITY
BELFAST**

Aerodynamic Shape Optimization Using Feature based CAD Systems and Adjoint Methods

Agarwal, D., Marques, S., Robinson, T., Armstrong, C., & Hewitt, P. (2017). Aerodynamic Shape Optimization Using Feature based CAD Systems and Adjoint Methods. In 18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, AIAA AVIATION Forum: Proceedings (AIAA Aviation Forum). DOI: 10.2514/6.2017-3999

Published in:

18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, AIAA AVIATION Forum: Proceedings

Document Version:

Peer reviewed version

Queen's University Belfast - Research Portal:

[Link to publication record in Queen's University Belfast Research Portal](#)

Publisher rights

© 2017 American Institute of Aeronautics and Astronautics.

This work is made available online in accordance with the publisher's policies. Please refer to any applicable terms of use of the publisher.

General rights

Copyright for the publications made accessible via the Queen's University Belfast Research Portal is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy

The Research Portal is Queen's institutional repository that provides access to Queen's research output. Every effort has been made to ensure that content in the Research Portal does not infringe any person's rights, or applicable UK laws. If you discover content in the Research Portal that you believe breaches copyright or violates any law, please contact openaccess@qub.ac.uk.

Aerodynamic Shape Optimization Using Feature based CAD Systems and Adjoint Methods

Dheeraj Agarwal*, Simão Marques[†], Trevor T. Robinson[‡], Cecil G. Armstrong[§]

Philip Hewitt [¶]

School of Mechanical and Aerospace Engineering, Queen's University Belfast, Belfast, BT9 5GX, U.K.

This paper presents a CAD-based optimization framework using adjoint functions for aerodynamic design. In this work, the SU^2 code is used to obtain high-fidelity flow solutions and surface sensitivities using adjoint methods. This work proposes methodologies to exploit CAD models created using standard commercial modelling software like CATIA V5 in the optimization workflow. A formulation to obtain geometric sensitivities is introduced, enabling the calculation of gradients with respect to these CAD variables. The performance and robustness of the optimization framework is assessed using a range of inviscid and viscous problems. The results show the CAD parameterisation can be efficiently used in obtaining reliable optimums, while operating directly on feature based CAD systems.

Nomenclature

c	chord (m)
C_d	Drag coefficient
C_l	Lift coefficient
P	pressure
C_p	Pressure coefficient
CAD	Computer Aided Design
\mathbf{F}^c	convective flux
\mathbf{F}^v	viscous flux
g	inequality constraint function
h	equality constraint function
J	objective function
L/D	lift to drag ratio
\mathbf{U}	flow variables
\mathbf{R}	residual
\mathbf{X}	mesh coordinates
\mathbf{Q}	Generic source term
t	non-dimensional time variable
V_n	Design velocity (normal component)
α	angle of attack
Ψ	vector of adjoint variables
θ	design variable array
ϕ	adjoint surface sensitivity

*PhD Student, School of Mechanical and Aerospace Engineering and AIAA Student Member

[†]Lecturer, School of Mechanical and Aerospace Engineering

[‡]Lecturer, School of Mechanical and Aerospace Engineering

[§]Professor, School of Mechanical and Aerospace Engineering

[¶]PhD Student, School of Mechanical and Aerospace Engineering

I. Introduction

IN the field of computational fluid dynamics (CFD) analyses can be computationally expensive, with typical run-times of many hours or days on high-performance systems. Thus, for problems involving large parameter spaces, the use of traditional numerical optimization techniques such as Design of Experiments or Evolutionary algorithms with CFD problems becomes prohibitive for routine design. Another approach would be to use gradient-based methods which require fewer iterations to achieve the desired convergence, but this would require the gradient of the cost function, which is presently not directly provided by standard industrial CFD solvers. A promising method for efficient CFD optimization methods involving large parameter spaces is the adjoint approach, which allows the computation of gradients at cost that is independent of the number of design variables.¹ In the case of aerodynamic optimization, problems where the number of design variables is very large compared to a very few desired objectives functions (lift, drag etc.), adjoint based techniques have shown considerable efficiency gains and have been an area of extensive research in last two decades.²⁻⁵ The advantage of an adjoint method arises from the fact that the gradient computation becomes independent of the number of design variables. This is achieved by solving a linear problem of computational cost comparable to that of the primal analysis. Most early developments in adjoint optimization used the model's surface mesh node coordinates as design variables,^{2,6} which represents the richest design space the CFD can evaluate. On the other hand, since all grid surface nodes can move independently, the implementation of a smoothing algorithm is required to prevent the appearance of oscillatory shapes during the optimization process. Moreover, such CAD-free methods produce as their output an optimized mesh, which then has to be converted back to CAD in order to obtain the optimized geometry for further design applications. This mesh-to-CAD step is a non-trivial task and it may require extensive user interaction. For these reasons, CAD-based optimization (i.e. modifying geometry) is considered more desirable and thus a link between CFD optimization framework and CAD is of utmost importance, and is the main focus of this paper. Recent work has attempted to achieve this to some extent through development of optimization processes based exclusively on non-uniform rational B-splines (NURBS).⁷ Whilst NURBS are capable of representing a wide variety of shapes, and represents a rich design space due to the number of control points which define a NURBS, the model may be too coarse at some places, and since approach does not work with the CAD parameters used to create the model, thus losing any design intent captured in those parameters when the design was created.

The approach presented in this work aims to enable shape optimization by using a parametric CAD model created in any feature based CAD modelling software (e.g. CATIA V5,⁸ SIEMENS NX⁹). In this approach the shape of a part model can be updated by changing the values of the parameters which define it. The main advantage of the parametric approach is that the optimized model produced can be directly used for downstream applications including manufacturing and process planning. The SU^2 solver is a state-of-the-art, open source, tool capable of high-fidelity aerodynamic analysis and adjoint based design optimization, made available from Stanford University. It includes specific tools for aerodynamic shape optimization based on Hicks-Henne functions and free-form deformation methods, among others. This paper will show the implementation of an alternative parameterisation scheme developed in CATIA V5, to be used with the SU^2 framework, which fully exploits its adjoint solver. The CAD based adjoint optimization framework will be assessed for robustness and performance for analysing flows over an NACA0012 aerofoil, the ONERA M6 wing and the NLR 7301 High-Lift test cases.

This paper is structured as follows. Section II the optimization framework used in this work, the design parameterisation developed in CATIA V5 for the design of aerofoils and 3D wing. Section III provides the key details of the CFD flow and adjoint solver SU^2 . Section IV presents the optimization results for the test cases, and Section VI summarizes the results and the conclusions drawn from this work.

II. Design Optimization Framework

Numerical aerodynamic optimization involves the minimization of a chosen objective function, e.g. drag, through the manipulation of a set of design variables. Within gradient based optimization methods, the gradients are used to guide the design towards an optimum design. With each new step a new set of values for each design variables is produced, causing a change in the objective function.

A general optimization problem would take the form seen below,

$$\begin{aligned} \underset{\boldsymbol{\theta}}{\text{Minimize}} : & f(\boldsymbol{\theta}), \\ \text{Subject to} : & g(\boldsymbol{\theta}) \geq 0, \\ & h(\boldsymbol{\theta}) = 0 \end{aligned} \tag{1}$$

$f(\boldsymbol{\theta})$ is the objective function to be minimised (maximised), $g(\boldsymbol{\theta})$ is the inequality constraint and $h(\boldsymbol{\theta})$ represents equality constraints. The optimization algorithm used in this work is the Sequential Least Squares Programming (SLSQP) implemented in Python-Scipy.

A. Design Parameterisation

For the process of shape optimization, SU^2 can employ different methods to parameterize aerofoil shapes such as: NACA series, cosine bumps and Hicks-Henne bumps for aerofoil deformations and the Free-Form deformation method for three dimensional designs. The Hicks-Henne bumps functions involve the linear superposition of several shape functions added to the base aerofoil to deform the surface. The magnitude and therefore influence of each shape function is manipulated by weights applied to each function respectively. These weights in turn are taken to be the design variables within the optimization process. The parameterisation is considered very effective and ensures the resulting deformed mesh is smooth.⁶

An alternative way for aerofoil parameterisation is the use of Bézier curves to define model in a parametric CAD system. The aerofoil could be described by two Bézier curves each defined by a number of control points: one defining the upper surface and another defining the lower surface. The design variables are then the position of each control point of these Bézier curves.

Figure 1 shows the original NACA0012 aerofoil and the deformed aerofoil configuration for a combination of parametric perturbations. The airfoil is constructed using Bézier curves with five control points adhering with the following constraints: the leading edge and trailing edge points are fixed, and the first control point on each surface after the leading edge are constrained to move in equal and opposite directions, vertically offset from the leading edge point. This is done in order to preserve C2 continuity at the leading edge and gives a total of nine parameters to be used for optimization. These parameters can be accessed in the CAD model feature tree through the CAD system API.

In the ONERA M6 test case, the CAD model is constructed in CATIA V5 using three different cross-sections along the wing span. Each cross-section is defined as above using Bézier curves, with a total of 27 CAD parameters. The 3D wing is then constructed by sweeping a surface through the section curves as shown in figure 2.

The geometry for NLR-High lift case is constructed in CATIA V5 using Bézier curves to define the main aerofoil and the flap. In this work the wing body is considered fixed and only the aerofoil flap is parametrized by considering the positions of Bézier control points defining the upper and lower surface, with a total of 14 CAD parameters to be used in optimization.

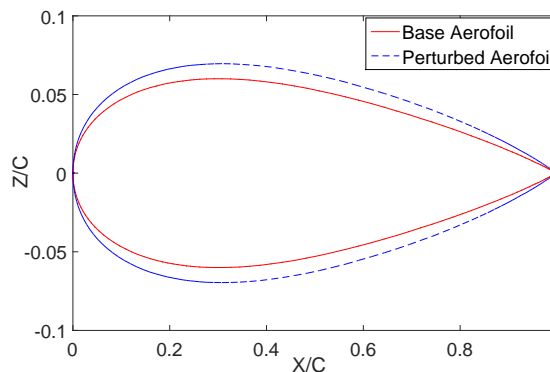


Figure 1: NACA0012 aerofoil with original and perturbed CAD parameters

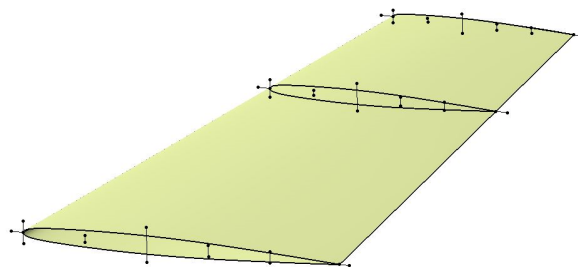


Figure 2: ONERA M6 CAD model showing Bézier control points for three section profiles

B. Gradient Evaluation

For the optimizer to establish a new search direction it is necessary for the gradient to be evaluated with respect to each design variable. Within SU^2 the calculation of the gradient is performed through the use of the chain rule. This is shown in Eq. 2 and provides flexibility for the use of other parameterisation methods:

$$\begin{bmatrix} \frac{\partial J}{\partial \theta_1} \\ \frac{\partial J}{\partial \theta_2} \\ \vdots \\ \frac{\partial J}{\partial \theta_n} \end{bmatrix} = \begin{bmatrix} \frac{\partial x_1}{\partial \theta_1} & \cdots & \frac{\partial x_m}{\partial \theta_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial x_1}{\partial \theta_n} & \cdots & \frac{\partial x_m}{\partial \theta_n} \end{bmatrix} \begin{bmatrix} \frac{\partial J}{\partial x_1} \\ \frac{\partial J}{\partial x_2} \\ \vdots \\ \frac{\partial J}{\partial x_m} \end{bmatrix} \quad (2)$$

where n and m are the number of design variables and surface mesh points, respectively; J represents the current function of interest, be it the objective or constraint functions. The variables x_i represent the normal displacement of a discrete point on the surface. The Jacobian $\frac{\partial x}{\partial \theta}$ is known as the geometric sensitivity matrix and measures the influence that each design variable (θ_i) has on the position of each grid point on the surface mesh. The third term represents the surface sensitivities with respect to a change in the function of interest with a change on the surface grid points. The next sections will explain how these terms are calculated.

1. Surface Sensitivities - Adjoint Method

Within SU^2 the surface sensitivities are calculated through the use of the continuous adjoint method. Consider a semi-discrete system of fluid conservation laws described by

$$\frac{d\mathbf{U}}{dt} = \mathbf{R}(\mathbf{U}, \mathbf{X}). \quad (3)$$

Eq. 3 is referred to as the *primal* solution, where \mathbf{X} represents the mesh coordinates and \mathbf{U} is the vector of the fluid system variables. During the convergence of the primal solution, the non-linear residual \mathbf{R} for each equation is driven to zero. The objective function J depends on the system variables,

$$J = J(\mathbf{U}, \mathbf{X}(\theta)). \quad (4)$$

The change in performance dJ due to a change in the value of the design parameter, $d\theta$, can be defined in terms of the surface mesh coordinates \mathbf{X}_s :

$$\frac{dJ}{d\theta} = \frac{dJ}{d\mathbf{X}} \frac{d\mathbf{X}}{d\mathbf{X}_s} \frac{d\mathbf{X}_s}{d\theta}. \quad (5)$$

The volumetric sensitivity term $dJ/d\mathbf{X}$ can be obtained by differentiating Eq. 4 with respect to \mathbf{X} as

$$\frac{dJ}{d\mathbf{X}} = \frac{\partial J}{\partial \mathbf{X}} + \frac{\partial J}{\partial \mathbf{U}} \frac{d\mathbf{U}}{d\mathbf{X}}. \quad (6)$$

The solution of Eq. 6 using finite differences requires the solution of Eq. 3 for each design variable. Alternatively, it can be shown that by selecting an arbitrary vector Ψ , Eq. 6 can be re-written as:¹⁰

$$\frac{dJ}{d\mathbf{X}} = \frac{\partial J}{\partial \mathbf{X}} + \Psi^T \frac{\partial \mathbf{R}}{\partial \mathbf{X}}. \quad (7)$$

Using this method, only one set of additional equations needs to be solved for each objective function (known as the *adjoint* solution), regardless of the number of design parameters. The adjoint volumetric sensitivities are then combined with the inverse operation of a mesh moving algorithm to yield the adjoint surface sensitivities ϕ as

$$\phi = \frac{dJ}{d\mathbf{X}_s} = \frac{dJ}{d\mathbf{X}} \frac{d\mathbf{X}}{d\mathbf{X}_s}. \quad (8)$$

The surface sensitivity is a measure of how sensitive the objective function is to the surface movement in the normal direction, calculated for every mesh point on the surface. The main benefit for choosing this method is that the cost of evaluating the surface sensitivity becomes independent of the number of design variables, leading to a significant reduction in computational expense. The surface sensitivities obtained when using drag minimization as the objective function is shown in Fig 3 for the NACA0012 aerofoil at a Mach number of 0.8 and at an angle of attack of 1.25° .

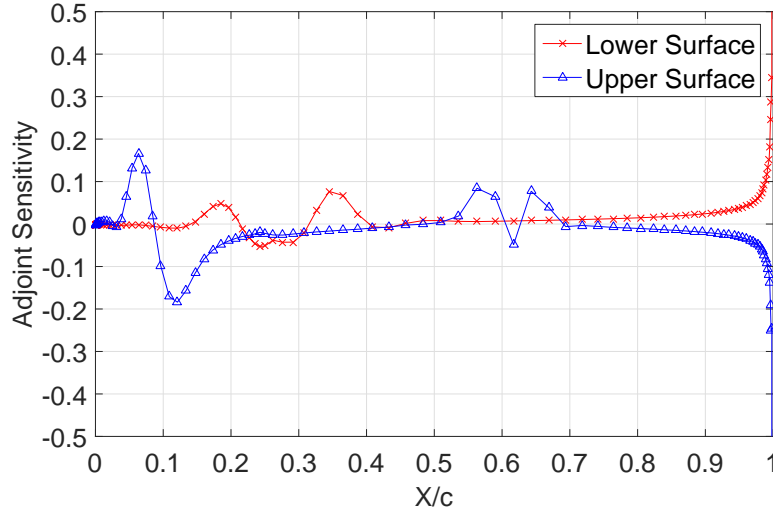


Figure 3: Adjoint surface sensitivity on NACA0012 aerofoil with drag as objective function

2. Geometric Sensitivities or Design Velocity

Design velocity can be defined as a measure of boundary movement resulting from a shape change. In particular this paper is concerned with computing the normal component of design velocity on the boundary of the model as

$$V_n = \delta \mathbf{X}_s \cdot \hat{n} \quad (9)$$

where $\delta \mathbf{X}_s$ is the movement of surface nodes and \hat{n} is the normal direction. In figure 4, the arrows represent the design velocity as the boundary changes from that of the solid line to that of the dashed line.

A number of approaches have been identified by other researchers for computing design velocity fields for shape sensitivity optimization.¹¹⁻¹³ The approach used in this paper is described in reference¹⁴ and is based on faceted approximations of CAD geometry generated by using an open-source surface mesh generator GMSH.¹⁵

The displacement of the model due to a parametric perturbation is calculated by projecting a point at the centroid of each facet (C_0) in the unperturbed model onto the facets in the perturbed model in the normal direction to get the projection point P_p . Once the projection point is obtained, the design velocity is calculated using Eq. 10.

$$V_n = (P_p - C_0) \cdot \hat{n}_{C_0} \quad (10)$$

The approach is applicable to any feature-based CAD modelling package. It places no constraints on boundary topology changes due to parametric changes, and it requires no special access to the CAD modeller source code. This makes it suitable for implementation within an industrial context, where closed-source, commercial CAD packages are widely used. The design velocities for NACA0012 aerofoil CAD parameters are presented in figure 5, with different colors representing different parametric perturbations.

To validate the gradient values produced using the developed framework coupling the adjoint sensitivities (figure 3) and CAD geometric sensitivities (figure 5) for NACA0012 aerofoil, a comparison was done with the gradients calculated by finite differences (FD) using flow analysis for geometries with parametric perturbations of 0.1%. The results are shown in figure 6, a strong correlation between the two sets is clear, giving confidence in the implemented process.

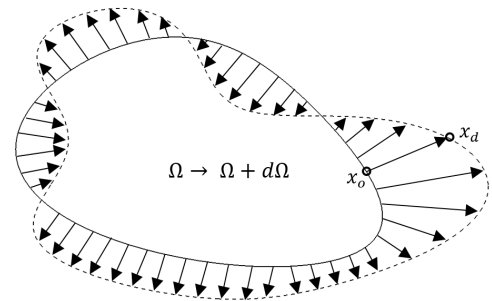


Figure 4: A simple 2D design velocity field

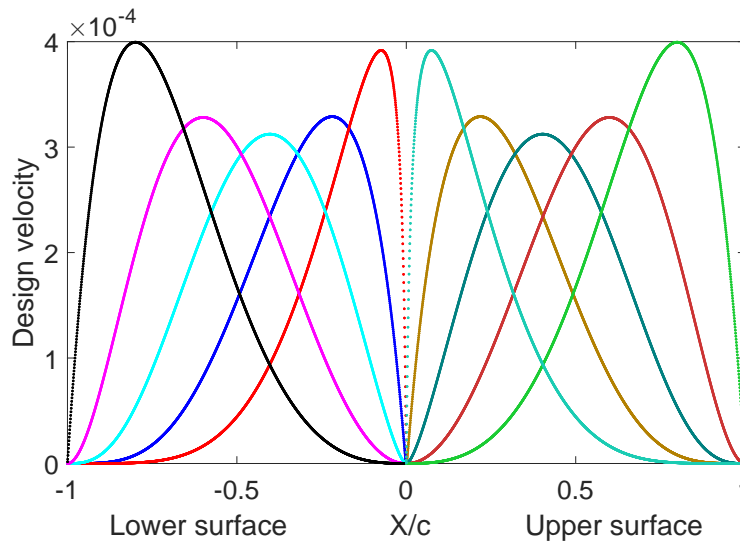


Figure 5: Design Velocities for NACA0012 aerofoil: $\frac{X}{c} > 0$ values correspond to the upper surface, whereas $\frac{X}{c} < 0$ correspond to the lower surface; the leading edge is located at the center of the plot.

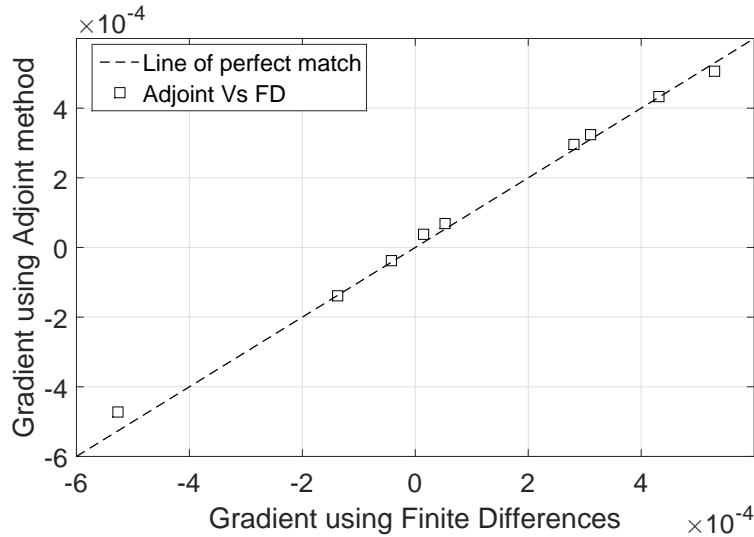


Figure 6: Gradient Validation for NACA0012 aerofoil with drag as objective function

III. Flow and Adjoint Solver

The SU^2 code was developed primarily for the purpose of providing an open source tool for use in aerodynamic shape optimization. Additionally, the framework also has the capability to solve various governing equations such as electrodynamics and chemically reacting flows. The philosophy employed by the developers, as stated in Palacios et al.¹⁶ is to provide:

- An open source model
- Portability
- Reusability and encapsulation
- Performance
- Gradient Availability

With the integration of these characteristics in SU^2 , the code allows access to all aspects of the implementation and is ready to be modified and expanded upon. To this effect, the code has been created as several primary modules, each used to handle a particular task so as to minimize interference and promote

ease of development. The suite consists of seven primary *C++* modules including the ones for solving partial differential equations (*SU2_CFD*), for mesh deformation (*SU2_MDC*), for gradient projection (*SU2_GPC*) etc. The execution of these individual modules is managed through Python scripts wrapped around *C++* modules, which handle the interaction between the modules and the transfer of data. In this work, the *SU2_CFD* is used for solving the flow equations, while the gradient projection is performed using the developed approach linking the CAD design parameters with the surface sensitivity obtained from *SU²*.

As described in reference,¹⁶ *SU²* uses a finite volume method for the spatial discretizing the partial differential equations (PDEs) with a standard edge-based structure on a dual grid with control volumes constructed using a median-dual, vertex-based scheme. The convective and viscous fluxes are evaluated at the midpoint of each edge in the primal mesh and then integrates them to evaluate the residual at every node in the numerical grid. Several discretization schemes are implemented in *SU²* which includes, among others the Jameson-Schmidt-Turkel (JST), Lax-Friedrich and Roe schemes. The solution can be marched forward in time, with or without Multi-Grid acceleration, using an implicit Euler scheme or an explicit Runge-Kutta method, until a steady state is reached.

The *SU²* suite is also able to solve the continuous adjoint Euler/RANS equations; following the work by Economou *et al.*¹⁷ the adjoint equations are given by:

$$-\frac{\partial \Psi}{\partial t} - \Delta \Psi \cdot \left[\left(\frac{\partial \mathbf{F}^c}{\partial \mathbf{U}} \right) - \mu_{tot} \left(\frac{\partial \mathbf{F}^v}{\partial \mathbf{U}} \right) \right] - \nabla \cdot \left[\nabla \Psi \cdot \mu_{tot} \frac{\partial}{\partial \mathbf{x}} \left(\frac{\partial \mathbf{F}^v}{\partial \mathbf{U}} \right) \right] - \Psi \frac{\partial \mathbf{Q}}{\partial \mathbf{U}} = 0, \quad (11)$$

where Ψ are the adjoint variables. The solution of Eq. 11 allows the efficient computation of the surface sensitivity, ϕ , from Eq. 7-8.

IV. Results

The developed framework is evaluated using the test cases as discussed in following sections. The first test case is an un-constrained optimization of the NACA0012 aerofoil in the transonic flow regime. The second test case is a constrained optimization of the ONERA M6 wing; the objective function is to minimize drag constraining the lift to be greater than a certain limit. The third test case analysed is for the optimization of the NLR 7301 high-lift aerofoil in fully turbulent flow; the objective function is to maximise the lift to drag ratio i.e. L/D for the high lift configuration. All problems are modelled using *SU²* CFD solver and the respective adjoint solver.

A. NACA0012 Aerofoil

The geometry for NACA0012 aerofoil is constructed in CATIA V5 as described in section II. The flow conditions and optimization problem is defined as:

- Freestream Temperature = 273.15K
- Freestream Mach number = 0.8
- Angle of attack (*AoA*) = 1.25°
- Objective Function = *min*(*C_d*)
- No. of design variables = 27

The parameters used for the optimization are defined in a commercial CAD modelling software CATIA V5. The mesh used for this test case is created in ICEM CFD using multi-block strategy, which is then exported as CFD General Notation System (CGNS) to be used within *SU²*. The airfoil surface is discretized using 200 points, and there are 31 points along the normal direction to the surface. A detailed view of the mesh around the aerofoil is shown in figure 7(a). A strong shock-wave is formed at upper surface of the aerofoil as shown in figure 7(b) which contributes to increased drag on the aerofoil.

Using the CAD parameters as design variables, an optimization framework is setup to use the adjoint function to calculate the gradients with respect to CAD parameters and use them in Scipy SLSQP optimization algorithm. For each optimization step, a new CFD mesh is created in ICEM CFD using an automated script. The drag coefficient (*C_d*) is reduced from 0.022223 to 0.000959. A comparison between the original and optimized aerofoil geometries along with the corresponding *C_p* values is shown in figure 8 and the optimization history in figure 9.

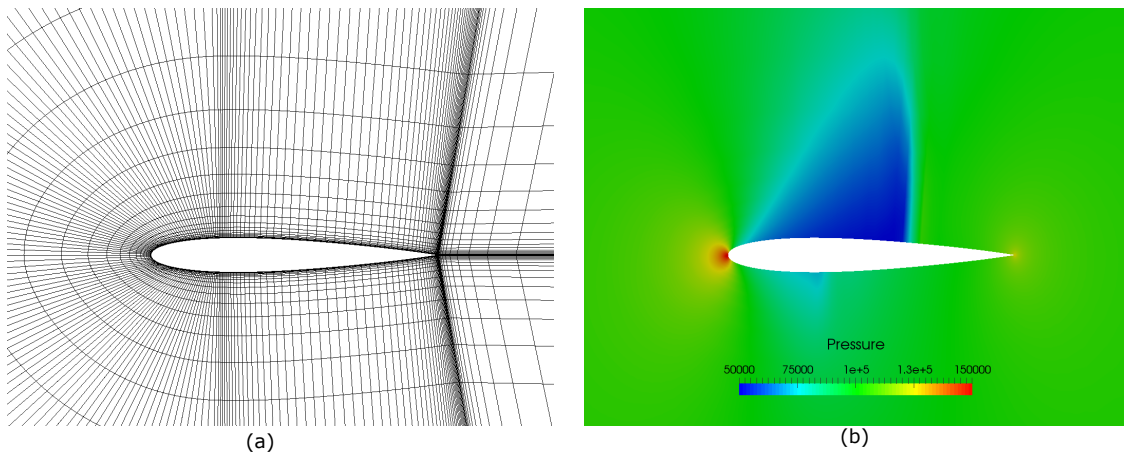


Figure 7: (a) Mesh around the aerofoil surface. (b) Pressure flow field

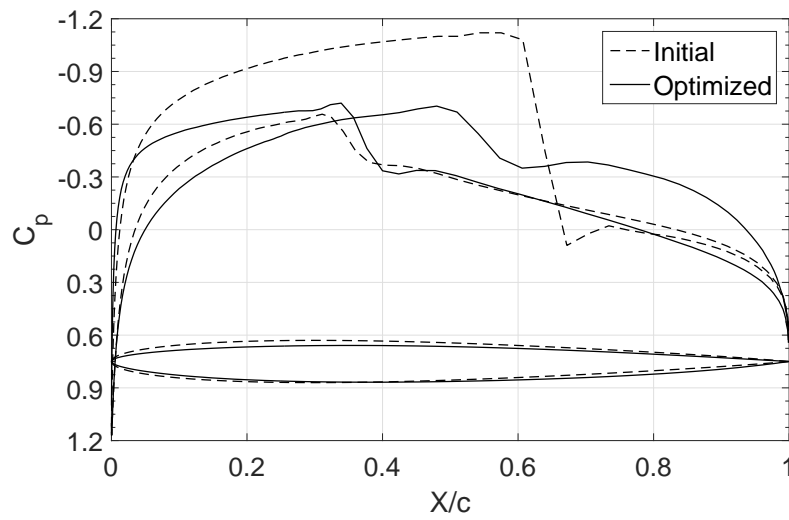


Figure 8: C_p and shape comparison of initial and optimized NACA0012 aerofoil

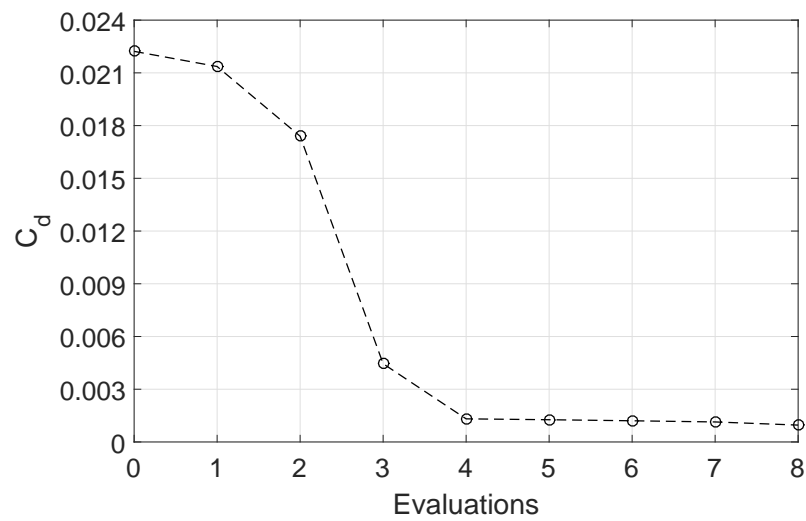


Figure 9: Function evaluations during optimization of NACA0012 aerofoil

B. Inviscid ONERA M6

The starting geometry for this case is 3D ONERA M6 wing constructed in CATIA V5 as shown in figure 2. The flow conditions and optimization problem is defined as:

- Freestream Temperature = $288.15K$
- Freestream Mach number = 0.8395
- Angle of attack (α) = 3.06°
- Objective Function = $\min(C_d)$
- Lift Constraint: $C_L > 0.2864$
- No. of design variables = 27

An unstructured mesh generated in GMSH with 154,617 nodes and 707,115 tetrahedral elements was used for the analysis. The details regarding the calculation of geometric sensitivities, convergence of primal and adjoint CFD solutions can be found in reference,¹⁴ where an unconstrained optimization of ONERA M6 wing was performed. The gradient of lift constraint with respect to CAD parameters is calculated by performing one additional adjoint analysis in SU^2 with *Lift* as objective. The optimization results can be visualized in figure 10, where pressure contours on the initial and optimized wing are shown. A reduction in drag coefficient from 0.011795 to 0.010153 (approximately 14%) is achieved, respecting the lift coefficient constraint ($C_l = 0.2864$). A comparison between the original and optimized aerofoil geometries along with the corresponding C_p values is shown in figure 11 and the optimization history in figure 12.

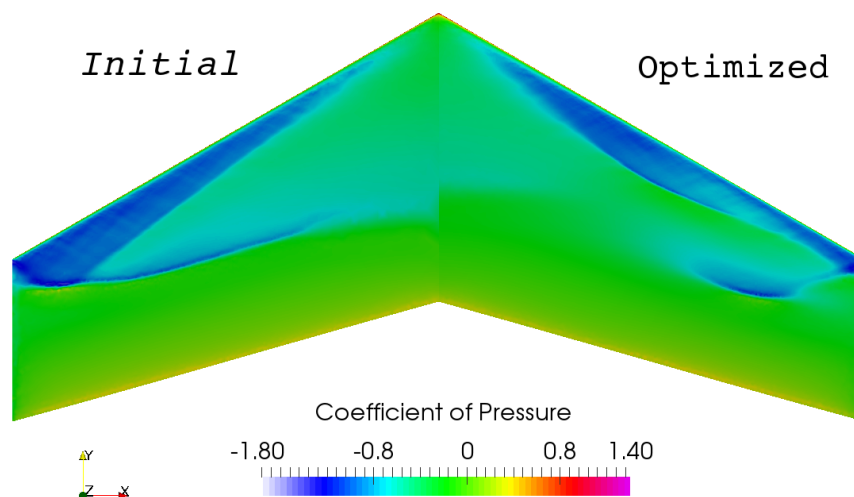


Figure 10: Pressure contours for initial and optimized ONERA M6

C. NLR7301 High-Lift Aerofoil

To further exercise the proposed methods, a viscous case was examined using the NLR7301 aerofoil with trailing edge flap.¹⁸ The flow conditions and optimization problem for the NLR 7301 high lift test case is defined as:

- Freestream Mach number = 0.185
- Angle of attack (α) = 6.0°
- Reynolds Number = 2.51×10^6
- Objective Function = $\max(L/D)$
- Turbulence model: Spalart-Allmaras turbulence model

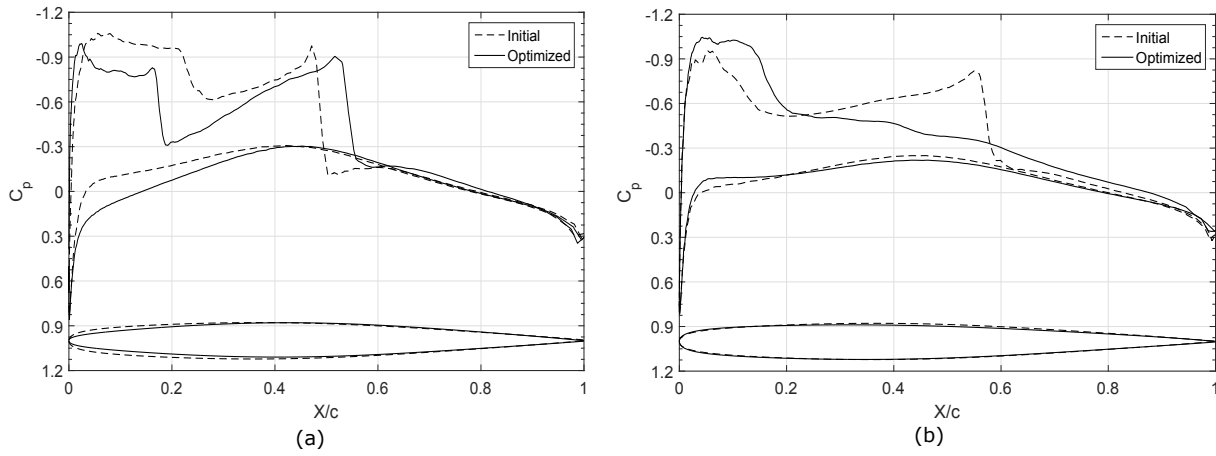


Figure 11: C_p distribution along the wing span: (a) $Y = 0.40$, (b) $Y = 0.80$

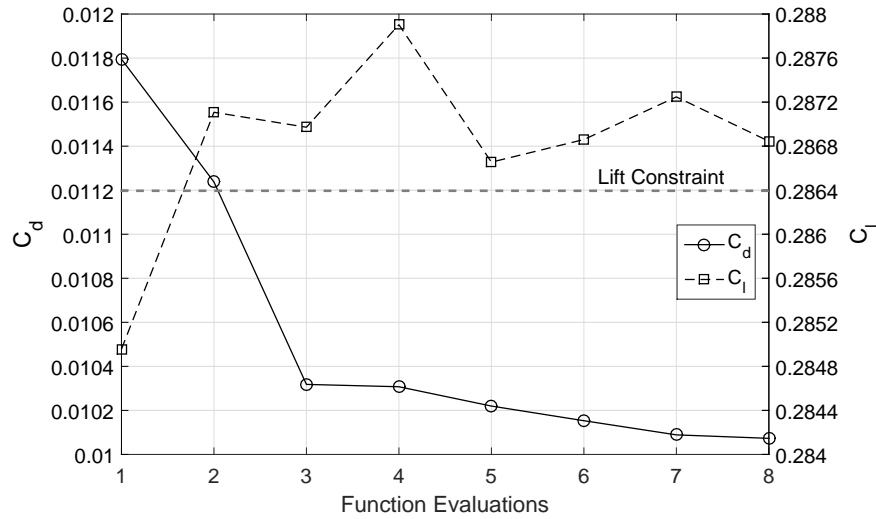


Figure 12: Function evaluations for optimization of ONERA wing

The CFD mesh used to start the optimization process is created as a multi-block mesh in ICEM CFD as shown in figure 13, and then exported in CGNS format to be used in SU^2 . The two-dimensional discretized volume consisted of quadrilateral elements conforming to the airfoil surfaces. The mesh contained 400 nodes along the wing surface while 420 nodes along the surface of flap. The mesh spacing in the surface normal direction at the aerofoil surface is $1 - 2 \times 10^{-6}c$, equivalent to a $y^+ < 1$. In terms of numerical scheme, the convective flux is calculated using the second-order JST scheme with the Venkatakrishnan limiter. The turbulence is modelled using SpalartAllmaras (S-A) model, and the convective flux within SA model is discretized using first-order scalar upwind method. For the purpose of this study, analysis was done using multi-grid approach with a fixed CFD number of 4.0, and frozen-turbulence for the adjoint calculations. The convergence of the residuals of the density and correspondent adjoint variable equations for the initial flap geometry is shown in figure 14.

The design velocities for the CAD parameters (figure 15) is calculated by perturbing each parameter following the methodology outlined in section II. The initial and optimized shapes of the flap and the corresponding pressure distribution on the surface is compared in figure 16, while the forces from each element are shown in table 1. The shape optimization of the flap leads to a reduction in the gap between the flap causing a larger area of stagnant flow on the trailing edge of the lower surface of the main element. The impact of the changes on the flap, increase the pressure differential between the surface of the main element, leading to an overall gain in L/D . During optimization, the overall L/D ratio increased from 79.88 to 81.36 (nearly 1.85%) as shown in figure 17. The pressure and Mach number contours for the initial and optimized flap geometry are shown in figure 18 and figure 19 respectively.

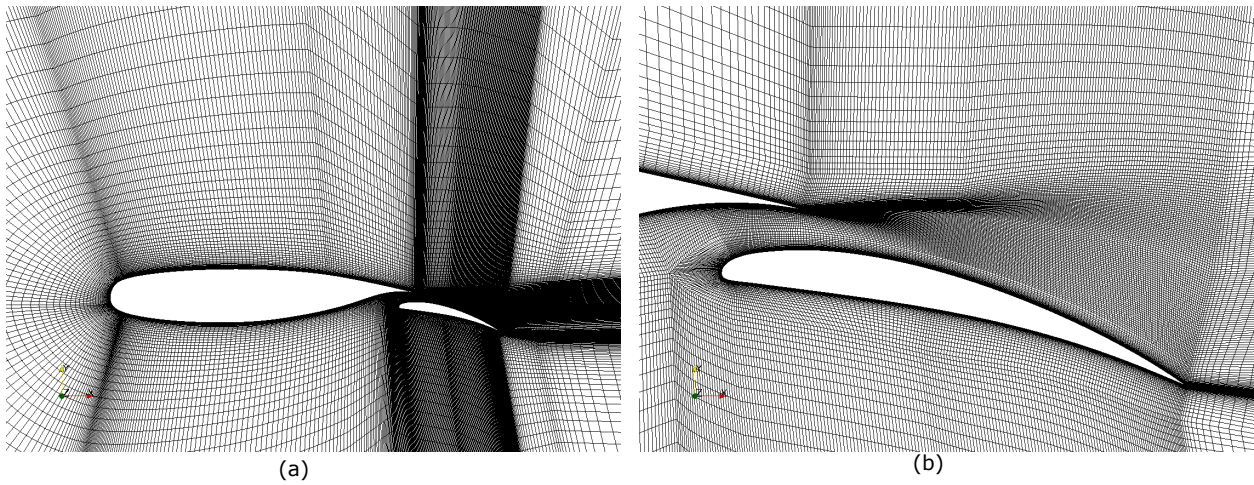


Figure 13: (a) Mesh for NLR 7301. (b) Mesh near the flap of NLR 7301

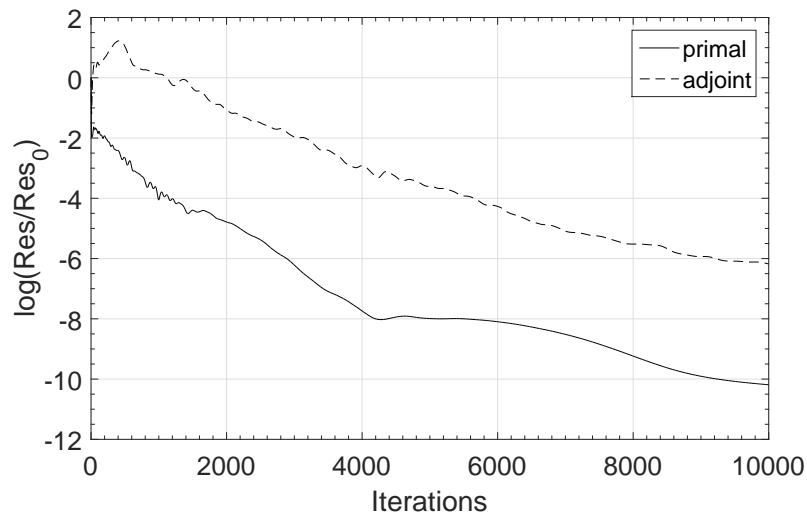


Figure 14: Residual convergence for initial NLR flap

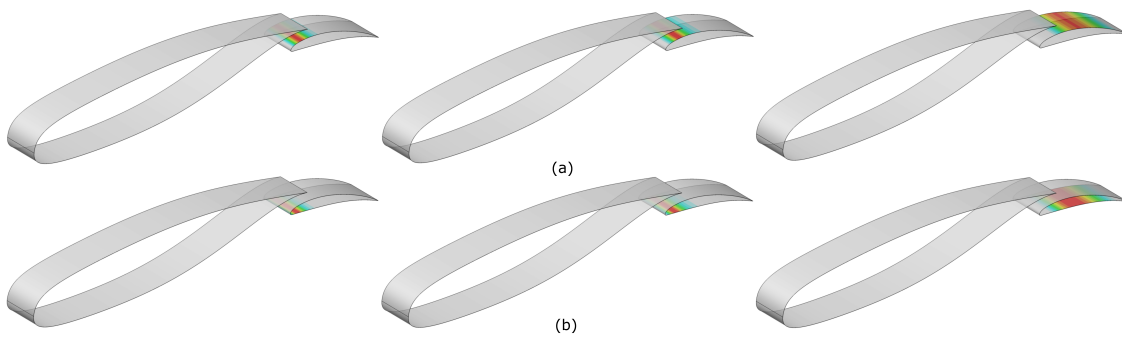


Figure 15: Design velocity contours for CAD parameters on flaps (a) Upper surface (b) Lower surface

V. Conclusion

A CAD based aerodynamic optimization framework is presented in this paper, which uses the parameters created in CAD modelling software as optimization variable. The performance gradients with respect to CAD parameters are calculated by linking adjoint functions with a design velocity approach, and have shown to match to a high degree of accuracy, with those calculated using finite difference approach. The primary benefit of this work arises from the fact that the model created using any commercial CAD modelling

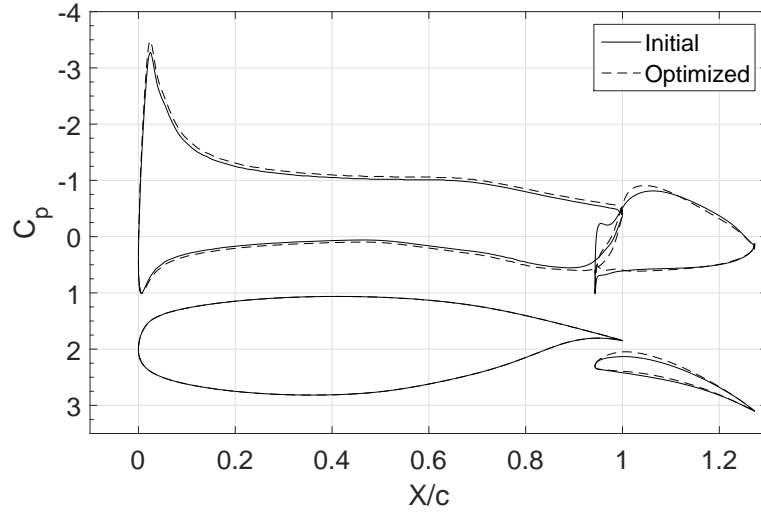


Figure 16: C_p and shape comparison of initial and optimized NLR flap

Table 1: Force breakdown for initial and optimized NLR geometry

	initial C_l	optimized C_l	initial C_d	optimized C_d
Upper Wing	0.8284	0.8785	-0.0508	-0.0581
Lower Wing	0.187	0.2268	0.0039	0.0020
Upper Flap	0.1186	0.1085	0.0293	0.0396
Lower Flap	0.1346	0.1326	0.0334	0.0330

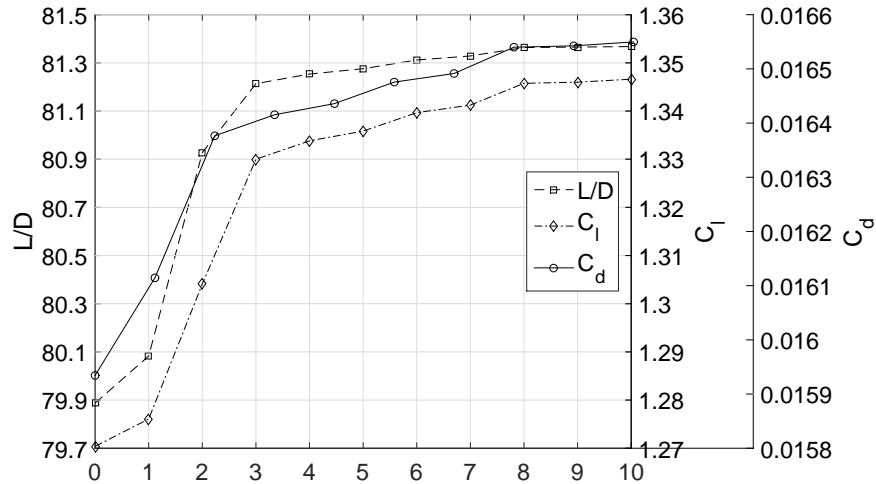


Figure 17: Function evaluations during optimization of NLR flap

package can be directly used for optimization, and the optimized geometry is always available in the native CAD format, which can be directly used for other downstream applications. In this paper, the adjoint capabilities an Open-Source CFD solver SU^2 is exploited, which can be easily extended to work with other CFD solvers including adjointFoam,¹⁹ HELYX,²⁰ DLR-TAU,²¹ HYDRA²² etc. The applicability of the developed framework is demonstrated using three different test cases of varied design complexities and for both inviscid and turbulent flows.

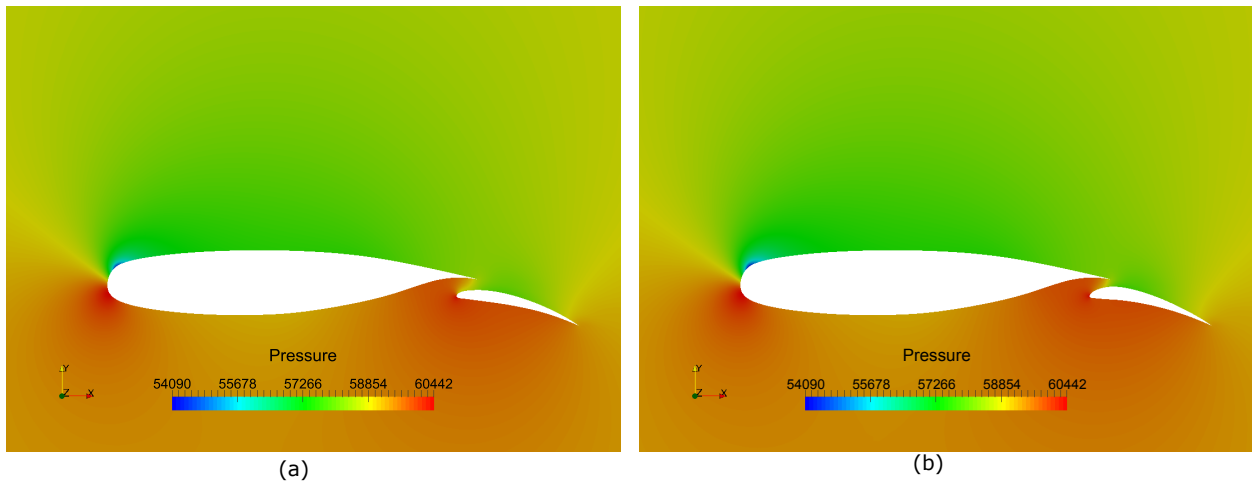


Figure 18: Pressure contours (a) Original aerofoil (b) Optimized aerofoil

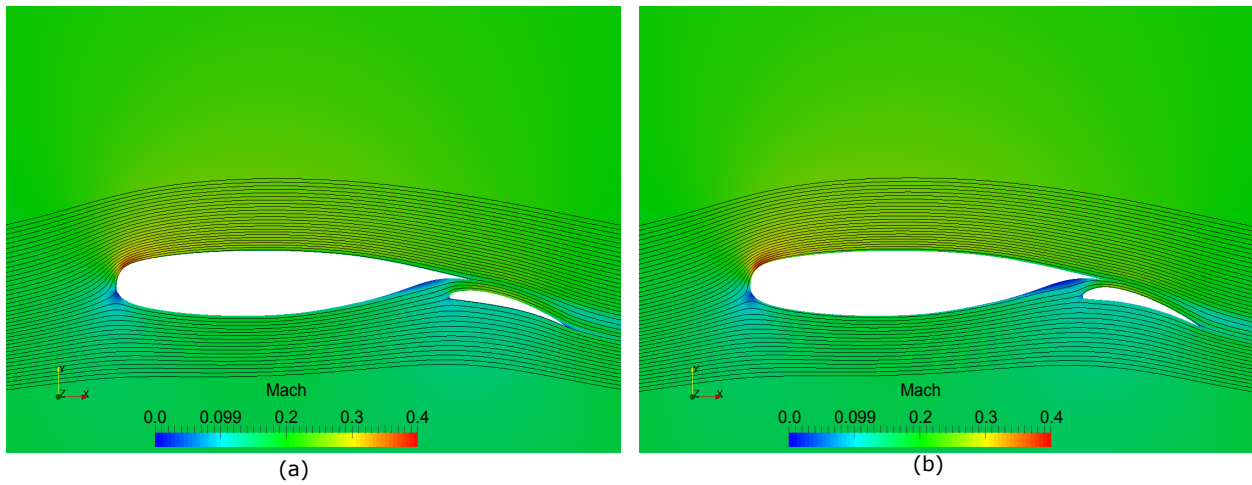


Figure 19: Mach number contours with streamlines (a) Original aerofoil (b) Optimized aerofoil

Acknowledgments

Author D. Agarwal is a PhD researcher within the IODA project (<http://ioda.sems.qmul.ac.uk>), funded by the European Union HORIZON 2020 Framework Programme for Research and Innovation under Grant Agreement No. 642959. Author P. Hewitt gratefully acknowledges the support from Engineering and Physical Sciences Research Council (EPSRC), UK.

References

- ¹Peter, J. E. and Dwight, R. P., “Numerical sensitivity analysis for aerodynamic optimization: A survey of approaches,” *Computers & Fluids*, Vol. 39, No. 3, 2010, pp. 373–391.
- ²Reuther, J., Alonso, J. J., Rimplinger, M. J., and Jameson, A., “Aerodynamic shape optimization of supersonic aircraft configurations via an adjoint formulation on distributed memory parallel computers,” *Computers and Fluids*, Vol. 28, 1999, pp. 675–700.
- ³Brezillon, J. and Gauger, N. R., “2D and 3D aerodynamic shape optimization using the adjoint approach,” *Aerospace Science and Technology*, Vol. 8, 2004, pp. 715–727.
- ⁴Castro, C., Lozano, C., Palacios, F., and Zuazua, E., “Systematic continuous adjoint approach to viscous aerodynamic design on unstructured grids,” *AIAA journal*, Vol. 45, No. 9, 2007, pp. 2125–2139.
- ⁵Mader, C. A., RA Martins, J., 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.
- ⁶Samareh, J. A., “Survey of shape parameterization techniques for high-fidelity multidisciplinary shape optimization,” *AIAA journal*, Vol. 39, No. 5, 2001, pp. 877–884.
- ⁷Martin, M. J., Andres, E., Widhalm, M., Bitrian, P., and Lozano, C., “Non-uniform rational B-splines-based aerodynamic

shape design optimization with the DLR TAU code,” *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, Vol. 226, No. 10, 2012, pp. 1225–1242.

⁸CATIAV5, <http://www.3ds.com/products-services/catia/>, Accessed 01/27/2017.

⁹SiemensNX, https://www.plm.automation.siemens.com/en_gb/products/nx/about-nx-software.shtml, Accessed 01/27/2017.

¹⁰Vasilopoulos, I., Agarwal, D., Meyer, M., Robinson, T. T., and Armstrong, C. G., “Linking parametric cad with adjoint surface sensitivities,” *ECCOMAS Congress 2016 - Proceedings of the 7th European Congress on Computational Methods in Applied Sciences and Engineering*, Vol. 2, 2016, pp. 3812–3827.

¹¹Hardee, E., Changb, K.-H., Tua, J., Choia, K. K., Grindeanua, I., and Yu, X., “A CAD-based design parameterization for shape optimization of elastic solids,” *Advances in Engineering Softwares*, Vol. 30, 1999, pp. 185–199.

¹²Kripac, J., “A mechanism for persistently naming topological entities in history based parametric solid models,” *3rd ACM symposium on Solid modelling and application*, 1995.

¹³Chang, K.-H., Choi, K. K., Tsai, C.-S., Chen, C.-J., Choi, B. S., and Yu, X., “Design sensitivity analysis and optimization tool (DSO) for shape design applications,” *Computing Systems in Engineering*, Vol. 6, No. 2, 1995, pp. 151–175.

¹⁴Agarwal, 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,” *submitted*, 2017.

¹⁵Geuzaine, C. and Remacle, J. F., “GMSH: a three dimensional finite element mesh generator with built-in pre- and post-processing facilities,” *International Journal of Numerical methods in Engineering*, Vol. 79, 2009, pp. 1309–1331.

¹⁶Palacios, F., Alonso, J., Duraisamy, K., Colonno, M., Hicken, J., Aranake, A., Campos, A., Copeland, S., Economon, T., Lonkar, A., et al., “Stanford University Unstructured (SU 2): an open-source integrated computational environment for multi-physics simulation and design,” *51st AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, 2013, p. 287.

¹⁷Economon, 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.

¹⁸Van den Berg, B. and Gooden, J., “Low-Speed Surface Pressure and Boundary Layer Measurement Data for the NLR 7301 Airfoil Section With Trailing Edge Flap,” *A Selection of Experimental Test Cases for the Validation of CFD Codes - AGARD AR-303*, Vol. 2, No. A-9, 1994.

¹⁹Othmer, C., “A continuous adjoint formulation for the computation of topological and surface sensitivities of ducted flows,” *International Journal for Numerical Methods in Fluids*, Vol. 58, No. 8, 2008, pp. 861–877.

²⁰“HELYX,” <http://engys.com/products/helyx>, Accessed 01/27/2017.

²¹Schwaborn, D., Gerhold, T., and Heinrich, R., “The DLR TAU-code: recent applications in research and industry,” *ECCOMAS CFD 2006: Proceedings of the European Conference on Computational Fluid Dynamics, Egmond aan Zee, The Netherlands, September 5-8, 2006*, Delft University of Technology; European Community on Computational Methods in Applied Sciences (ECCOMAS), 2006.

²²Lapworth, L., “Hydra CFD: A Framework for Collaborative CFD Development,” *International Conference on Scientific and Engineering Computation, Singapore, July 5-8, 2004*.