Imperial College London

A Coupled Discrete Adjoint Method for Optimal Design with Dynamic Non-Linear Fluid Structure Interactions

Charanya Venkatesan-Crome

Imperial College London
Department of Aeronautics
London, SW7 2AZ
United Kingdom

This thesis is submitted for the degree of Doctor of Philosophy of Imperial College London

May 2021

Declaration

I hereby declare that this thesis is solely the product of my own work. No parts of this work have been submitted for another degree. References to other people's research and results have been fully acknowledged.

The copyright of this thesis rests with the author. The contents are licensed under a Creative Commons Attribution-Non Commercial-No Derivatives 4.0 International Licence (CC BY-NC-ND). Under this licence, you may copy and redistribute the material in any medium or format on the condition that; you credit the author, do not use it for commercial purposes and do not distribute modified versions of the work. When reusing or sharing this work, ensure that the licence terms are clear to others.

Charanya Venkatesan-Crome

To Thatha, for encouraging a pursuit of knowledge,

and for all those pursuing the same...

Acknowledgements

First and foremost, I would like to sincerely thank my supervisor Professor Rafael Palacios for his guidance and ever positive attitude. Rafa, without your support and feedback, this would not have been possible. Thank you for believing in me.

Next I would like to thank Dr Marco Fossati from University of Strathclyde and Dr Matthew Santer from Imperial College London for being my examiners. Thank you for showing interest, taking time to review and discussing my work.

I would like to express my appreciation to UK Engineering and Physical Science Research Council for funding my PhD and giving me the opportunity to pursue this field of research.

I would also like to acknowledge the SU2 community for providing a platform that I've been able to use in my research. In particular, I would like to thank Thomas Economon for all the support and encouragement as well as the opportunities for collaborative work. I would also like to thank Professor Nicolas Gauger and Tobias Kattmann for enabling this collaboration.

Thank you to all of my aeroelasticity friends - Salvatore, Ruben, Alfonso, Alvaro, Arturo, Marc, Pedro, Norberto, Myriam, Mohsen, Stephanie, Nikos - ranging from those who were on this journey before me, to those who've been through it all together with me and to those who are continuing on this journey. Ruben and Pedro in particular, thank you for our invaluable discussions. Alfonso, thank you for saving all of my data when my computer failed a year ago! The academic family is a big part of the PhD experience, though we haven't been able to see each other in person in a long time, the Friday coffees have been our link this past year.

I cannot have done this without the full support of all of my family and friends. Andy, I truly appreciate everything you've done for me, I am incredibly lucky to have you in my life. Amma you are my rock, thank you for always being there for me. My dogs, Chewie and Sootie, for being my constant companions and adding fun to every day. Paatiamma, Ragsy, Revathy, Akka and Chithappa, you might have been further away but you have always been close at heart, thank you for making me who I am today. Thank you to all the rest of my family for your love and belief in me. A special thank you to the family I gained during this PhD, your support has been more than I could have ever imagined. I would also like to thank all my friends who have been there throughout, thank you for lightening up my life.

Abstract

Design optimization of aerodynamic structures need accurate characterization of the coupled fluid-structure interactions (FSI). Incorporating high-fidelity analysis methods in the multi-disciplinary design optimization (MDO) necessitates efficient sensitivity evaluation, which is particularly important for time-accurate problems. Adjoint methods are well established for sensitivity analysis when large number of design variables are needed. The use of discrete adjoint method through algorithmic differentiation enables the evaluation of sensitivities using an approximation of the Jacobian of the coupled problem, thus enabling this approach to be applied for multidisciplinary analysis.

This thesis presents a new discrete adjoint formulation suitable for fully coupled, non-linear, dynamic FSI problems. A partitioned approach is considered with finite volume for the fluid and finite elements for the solid domains. The solution includes the time-dependent adjoint variables that arise from the grid motion and chosen time integration methods for both the fluid and structural domains. Implemented as a generic multizone discrete adjoint solver for time-accurate analysis in the open-source multiphysics solver SU2, this provides a flexible framework for a wide range of applications. The partitioned FSI solver approach has been leveraged to extend the dynamic FSI capabilities to low speed flows through the introduction of a density-based unsteady incompressible flow solver.

The developed methodology and implementation are demonstrated using a range of numerical test cases. Optimal design for steady, coupled FSI problems are firstly presented before moving to the building blocks of dynamic coupled problems using single domain analysis, for both structural and fluid domains in turn. The new unsteady incompressible fluid solver, for both the primal and adjoint analysis, are verified against a range of well-known benchmark test cases, including problems with grid motion. Finally, applications of coupled dynamic problems are presented to verify both the unsteady incompressible solver for FSI as well as the successful verification of the discrete adjoint sensitivities for the transient response of a transonic compliant airfoil for a variety of both aerodynamic and structural objective functions.

Nomenclature

State variables:

$\mathbf{w}$	Flow conservative variables
$\tilde{\mathbf{w}}$	Flow primitive variables
$\mathbf{u}$	Structural displacements
$\mathbf{q}$	Extended structural variable
$\mathbf{z}$	Fluid mesh nodes
$\mathbf{y}$	Extended fluid mesh variable

Residual equations:

$\mathbf{F}(\cdot)$	Fluid domain
$\mathbf{S}(\cdot)$	Structural domain
$\mathbf{M}(\cdot)$	Fluid mesh domain
$\mathbf{G}(\cdot)$	Coupled FSI equations

Fixed-point equations:

$\mathcal{S}(\cdot)$	Structural domain
$\mathcal{F}(\cdot)$	Fluid domain
$\mathcal{M}(\cdot)$	Fluid mesh domain

Fluid domain:

$\mathbf{F^c_{ALE}}$	Convective flux
$\mathbf{F^v}$	Viscous flux
$\mathbf{Q}$	Generic source term
ρ_f	Fluid density
$\mathbf{v}$	Flow velocities
$\dot{\mathbf{z}}$	Grid velocity
e	Total energy per unit mass
$\bar{\bar{\boldsymbol{\tau}}}$	Viscous stress tensor
μ	Fluid viscosity
C_p	Specific heat
T	Temperature
κ	Thermal conductivity
p	Dynamic pressure
$\boldsymbol{\Gamma}$	Density-based preconditioner
τ	Pseudo-time

Structural domain:

$\boldsymbol{\sigma}$	Cauchy stress tensor
ρ_s	Structural density
$\delta\mathbf{W}$	Virtual work
$\mathbf{f}$	Actuating forces per unit volume
$\boldsymbol{\lambda}$	External tractions

$\boldsymbol{S}$	Piola-Kirchhoff stress tensor
$\mathbf{E}$	Lagrangian strain tensor
$\dot{\mathbf{E}}$	Strain rate tensor
Ψ	Stored elastic energy function
$\mathbf{T}$	Internal equivalent forces
$\mathbf{F}_b$	Body forces
$\mathbf{F}_\Gamma$	Surface forces
$\mathbf{M}_s$	Mass matrix
E	Young's modulus
ν	Poisson ratio

Mesh domain:

$\tilde{\mathbf{K}}_\mathbf{M}$	Fictitious stiffness matrix
$\tilde{\mathbf{f}}$	Fictitious force

Coupled problem:

Γ	Fluid-Structure Interface
$\boldsymbol{\lambda}_f$	Fluid tractions
$\boldsymbol{\lambda}_s$	Tractions on structure
$\mathbf{u}_\Gamma$	Boundary displacements
$\dot{\mathbf{u}}_\Gamma$	Boundary velocities

Adjoint problem:

J	Objective function
$\boldsymbol{\alpha}$	Design variables
L	Lagrangian
$\overline{\mathbf{w}}$	Fluid adjoint variables
$\overline{\mathbf{u}}$	Structural adjoint variables
$\overline{\mathbf{z}}$	Mesh adjoint variables

Contents

Acknowledgements **v**

Abstract **vii**

1 Motivation **1**

1.1 Literature Review 3

1.2 Objectives 13

1.3 Publications 15

1.4 Thesis Layout 15

2 A High-Fidelity Dynamic Model for Fluid-Structure Interactions **17**

2.1 Fluid Domain 18

2.1.1 Unsteady Incompressible Solver 20

2.2 Structural Domain 26

2.3 Coupled Fluid-Structure Interaction 30

2.4 Software Architecture 32

3 Coupled Adjoint Methods for Gradient Based Optimization 36

3.1 Sensitivity Analysis 37

3.2 Discrete Adjoint Method for Steady Fluid-Structure Interactions 42

3.3 Discrete Adjoint Method for Time-Accurate Fluid-Structure Interactions 44

3.3.1 Implementation 48

4 Optimal Design of Coupled Problems in Equilibrium 52

4.1 Aeroelastic Stiffness Optimization 53

4.2 Aeroelastic Shape Optimization 57

4.2.1 Optimal Design with Small Structural Displacements 59

4.2.2 Optimal Design with Large Structural Displacements 62

5 Optimal Design of Structural Dynamic Problems 66

5.1 Optimal Stiffness of Elastic Bar under Axial Load 67

5.1.1 Elastic Bar under Static Axial Load 68

5.1.2 Elastic Bar under Sinusoidal Axial Load 68

5.2 Shape Optimization for Resonant Behaviour of an Airfoil Structure 72

6 Optimal Design of Dynamic Incompressible Flow Problems 78

6.1 Primal Unsteady Incompressible Solver 79

6.1.1 Lid-Driven Cavity 79

6.1.2 Laminar Flow Past a Circular Cylinder 82

6.1.3 Pitching NACA 0015 Airfoil 83

6.2 Discrete Adjoint Unsteady Incompressible Solver 86

6.2.1 Rotating Airfoil in Inviscid Flow 86

6.2.2 Laminar Flow Past a Circular Cylinder 89

6.2.3 Plunging NACA 0002 Airfoil 90

7 Coupled Analysis of Dynamic FSI Problems **95**

7.1 Verification of Dynamic Coupled FSI for Incompressible Flow 96

7.2 Limit Cycle Oscillations of Transonic Compliant Airfoil 100

7.3 Discrete Adjoint Sensitivities for Transient Response of Transonic Compliant Airfoil 103

7.4 Gust-Load Simulation with Fully Coupled FSI 106

8 Conclusions **113**

8.1 Summary of Thesis Achievements 113

8.2 Main Research Outcomes 118

8.3 Recommendations for Future Work 120

Bibliography **122**

List of Tables

4.1 Boundary conditions and properties. 59

5.1 Natural frequency and time period of first five natural modes. 69

6.1 Summarized results for the circular cylinder at Re=100. 84

6.2 Linear fit coefficients for unsteady lift. 86

List of Figures

2.1 Description of classes in SU2_CFD for Fluid-Structure Interaction problems. . . 33

4.1 Schematic for compliant NACA 0012 airfoil with 20 design variable segments for variable stiffness. 54

4.2 Gradient verification for discrete adjoint sensitivities with stiffness design variables using finite differences. 55

4.3 Optimization history for compliant NACA 0012 airfoil with design variables of stiffness distribution. 56

4.4 Comparison of C_P distribution and airfoil surface for fluid only, baseline compliant airfoil with uniform stiffness and optimized compliant airfoil with variable stiffness. 56

4.5 Comparison of C_P distribution, and airfoil surface and variation in airfoil stiffness for design iterations 9 and 10. 57

4.6 Schematic of NACA 0012 airfoil for optimization with FFD box. 58

4.7 Verification of discrete adjoint shape derivatives using finite differences, $E =$ 70 MPa. 59

4.8 Optimization history of NACA 0012 airfoil for efficiency, C_D and C_L. 60

4.9 C_P distribution and airfoil shape results from fluid only and stiff FSI optimization. 61

4.10 C_P distribution and airfoil shape results from flexible FSI optimization. 62

4.11 New baseline airfoil through deformed FFD box to produce large displacement due to aerodynamic load. 62

4.12 Comparison of non-linear and linear structural solver with baseline airfoil. 63

4.13 Comparison of discrete adjoint AD shape derivatives of non-linear and linear structural solver on the baseline airfoil. 64

4.14 Optimization history for the new baseline airfoil with large displacements and the optimized designs using both linear and non-linear structural solvers. 65

5.1 Static analysis results for objective function, sensitivity using discrete adjoint. . 68

5.2 Relative error of sensitivities for direct differentiation and central differences ($h = 5$ Pa) with discrete adjoint sensitivities as reference. 69

5.3 Variation in time-averaged magnitude of tip displacement under sinusoidal load of $1 - 15$ Hz is shown in (a) and (b) shows the time history results of absolute tip displacement for loading at $2, 4, 12$ Hz. 70

5.4 Dynamic analysis results with ω_1 frequency load: objective function, sensitivity using discrete adjoint reverse mode AD. 71

5.5 Relative error of sensitivities from direct differentiation forward mode AD and central differences ($h = 5$ kPa) using reference from discrete adjoint sensitivities. 72

5.6 Schematic of FFD around NACA 0012 airfoil. 73

5.7 Variation of time-averaged tip displacement using four grid refinement levels. . . 73

5.8 Frequency sweep of the airfoil with $E = 10$ and 8 MPa. 74

5.9 Gradient verification for the airfoil under sinusoidal load using discrete adjoint and finite differences with step size $10^{-3}c$. 75

5.10 Optimization history and design comparison for NACA 0012 airfoil under sinusoidal loading with objective function calculated using final two periods. 76

5.11 Frequency sweep of the NACA 0012 airfoil at $E = 10$ and 8 MPa and optimized airfoil at $E = 8$ MPa. 76

5.12 Optimization history and design comparison for NACA 0012 airfoil under sinusoidal loading with objective function calculated using final two and all six periods. 77

6.1 Normalized velocity magnitude contours of flow in the lid driven cavity for Re = $1000, 5000$. 80

6.2 Lid-driven cavity velocity distribution at center line for Re = $1000, 5000$. 81

6.3 Normalized velocity magnitude contours of flow past the circular cylinder in one period of oscillations. 83

6.4 Variation of mean lift and drag for coarse, medium, fine and finer grid refinement levels for pitching NACA 0015 airfoil ($k = 0.179$, Ma = 0.12). 84

6.5 Pitching NACA 0015 airfoil with reduced frequency, $k = 0.179$ for angle of attack $8 \pm 8^{\circ}$. 85

6.6 Schematic of the rotating frame NACA 0015 airfoil and the associated efficiency gradients calculated from discrete adjoint, direct differentiation and finite differences. 88

6.7 Optimization history of efficiency maximization with constrained airfoil area and the comparison of airfoil profiles for NACA 0015 with intermediate and final optimized shape. 88

6.8 Gradient verification for an objective function of time averaged drag using a Free-Form Deformation box. 89

6.9 Drag during sinusoidal plunging motion for $k = 1$ and $h_{max}/c = 0.01$. 91

6.10 Gradient verification for an objective function of time averaged drag using design variables of FFD control points. 92

6.11 Optimization history of time-averaged drag coefficient minimization and comparison of airfoil profile for NACA 0002 and optimized profile for dynamic and static problems. 92

6.12 Comparison of the time history for C_D during the plunging motion with amplitude of $0.01c$ and $k = 1$ for NACA 0002, dynamic and static optimized airfoil. . 94

6.13 Pressure coefficient distribution for NACA 0002 and optimized profile at $\phi = 0°$, $90°$. 94

7.1 Schematic for FSI Benchmark (Not to scale). 97

7.2 Time history for vertical tip displacement of elastic beam behind a rigid square cylinder and frequency results using FFT analysis, with coarse mesh results on top and fine mesh results at the bottom. 98

7.3 Discrete adjoint gradient verification against finite differences (h=$10^{-5}E_{BL}$) for objective functions of time-averaged tip displacement (7.3(a)) and drag (7.3(b)). 99

7.4 Time history results for 1.942GPa showing trailing edge displacement, lift and drag. 100

7.5 Airfoil deflection and flow field solution at $t = 0.02, 0.04, 0.06$s for flexible airfoil with $E = 70$MPa. 101

7.6 Time history and frequency results for $E = 70$MPa including trailing edge displacement, lift and drag. 102

7.7 Time history and frequency results for $E = 35$MPa including trailing edge displacement, lift and drag. 102

7.8 Schematic of NACA 0012 airfoil with segmented design variables $\alpha_i, i = 0, ..., 19$. 103

7.9 Time history results for the primal transient response of the NACA 0012 airfoil. 104

7.10 Discrete adjoint gradient validation against finite differences (h=$10^{-5}E_{BL}$) for objective functions of time-averaged drag (7.10(a)), lift (7.10(b)), efficiency (7.10(c)) and tip displacement (7.10(d)). 104

7.11 Grid and time convergence for lift and tip displacement on four refinement levels 108

7.12 C_L and C_D on a rigid airfoil for varying gusts of length H from $2.5c$ to $25c$. . . . 108

7.13 Coupled response of C_D, C_L and $|u|/c$ for varying gusts of length H from $2.5c$ to $25c$. 109

7.14 Discrete adjoint gradient verification against finite differences (h=$10^{-5}E_{BL}$) for objective functions of time-averaged lift (7.14(a)) and tip displacement (7.14(b)). 110

7.15 Optimization history for lift minimization of compliant NACA 0012 airfoil under gust load $H = 5c$ with design variables of stiffness distribution. 111

7.16 Optimization history for compliant NACA 0012 airfoil to minimize tip displacement under gust load $H = 5c$ with design variables of stiffness distribution. . . . 112

Chapter 1

Motivation

Design of aerodynamic structures have increasingly needed to consider coupled fluid-structure interaction effects for optimal performance. Efficiency demands on civil aviation, high-altitude pesudo-satellites and wind turbines for green energy have led to designs with high structural flexibility that deforms under the operational load thus necessitating considerations of fluid-structural interactions.

One of the most influential factors for fuel economy in civil aviation is aircraft mass. Reduction in mass has been achieved through advancements in composite material technology to produce components which weigh less whist still fulfilling the strength requirements. However the increased flexibility in these designs have often led to larger deformations in operation than previous designs.

In parallel, solar-powered aircraft have grown in interest and importance coinciding with the drive for sustainable, environmentally friendly renewable energy alternatives [1]. Typically, designs for solar aircraft have high aspect ratio profiles to maximize the wing span to accommodate solar panels and also minimize induced-drag thus improving the efficiency of energy used for flight. The high-aspect ratio profiles seen in solar aircraft and wind turbines, introduce higher levels of flexibility which results in deformation that no longer satisfy the small structural deformation assumption in the linear modelling techniques used for conventional aircraft or wing design.

The large deformations require non-linear structural analysis together with fully coupled analysis of the aerodynamic and structural interactions. High fidelity, accurate simulation models are essential in order to incorporate influences of coupled response in both analysis and design for steady state considerations but also are particularly important when designing for dynamic systems and responses.

The design and manufacture of aircraft is expensive with long lead times, thus places a large importance on simulation. Simulation provides valuable insight typically at a fraction of the cost compared to production of multiple prototypes. In particular, for the context of air vehicles, simulation is the only viable strategy at the early design phase.

High fidelity computational fluid dynamics tools are now routinely used for aerodynamic analysis. Through advancements in sensitivity analysis and optimization, these tools are increasingly applied to aircraft design. Structural consideration in design typically comes in the secondary stage with design optimization using sequential approach which leads to sub-optimal designs [2]. Clearly, designs that are created with the full understanding of the coupled response, such that it can capitalize on the inherent characteristics of the coupled system and their responses, can be expected to achieve better performance.

The design process can consider steady analysis for equilibrium conditions such as cruise or unsteady analysis for periodic or transient behaviour. Periodic analysis can be performed either through time-accurate simulations or through harmonic analysis which might be a more efficient method for purely periodic behaviour of interest, however when transient responses are also of interest, including responses to external stimulus, the time-accurate simulations become necessary. Accurate capturing of the transient response are very important for analysis but also for design in order to ensure stability of the optimized design or to maximize the advantages from the dynamic response for improved behaviour. Especially when considering non-linear dynamic responses, high fidelity and still computationally efficient simulation models become essential [3, 4].

As a result, optimal design problems can take many forms. For steady responses, optimization of shape of the wetted area or material property distribution inside the structure are examples

which can be part of the design cycle to lead to improved performance for chosen objectives. These can also be used for time-accurate optimization for passive control of the dynamic response such that the initial jig design is optimized to lead to overall improved coupled response over the chosen time. The design variables for the time-accurate optimization can also be chosen to vary in time for optimization of variables such as trajectory or external actuation to achieve longer flying range, or improved transient characteristics such as for example gust-load alleviation. Objectives for solar aircraft are primarily driven by efficiency to be able to travel as high as possible, as far as possible or as long as possible with the finite amount of energy. Therefore, trajectory optimization or gust-load alleviation either through passive or active control will play a key role in enabling the aircraft to fly in a path that maximizes the sun's rays on the aircraft or to minimize external disturbances from the environment [5].

1.1 Literature Review

In order to accurately capture the highly coupled aeroelastic responses as part of design optimization, as discussed in the introduction, the past and present developments in relevant areas are initially explored. Firstly, the development of the tools for analysis of such problems and the identification of the key features required to ensure accurate capture of the coupled fluid-structure interaction characteristics are reviewed. Methods of design optimization for multidisciplinary problems are then discussed in the context of aeroelastic application which is then explored further for dynamic problems. Finally, methods for obtaining accurate sensitivities of coupled, physical systems with sufficient accuracy to allow gradient based optimization will be discussed.

Coupled Fluid-Structure Interaction Analysis

Methods for coupled fluid-structure interaction typically take either monolithic or partitioned approaches. Monolithic methods solve the fluid and structural governing equations at once together with the chosen interface coupling [6, 7, 8]. This allows for a Newton linearization

of the fully coupled system of equations which can be solved using implicit solution schemes. Some limitations on the governing equations that can be modelled for the fluid and structural domains exist since they need to be solved together. However, this approach is well suited to coupled problems in which the fluid density is of similar value to the structural density, thus it is extensively used in bimechanical applications [9, 10, 11]. The implicit solution approach for these problems has an advantage of good convergence stability, thus enabling the use of larger time steps for dynamic problems [12]. With the use of tailored preconditioning for the iterative solutions of the large linear systems, the monolithic solvers can be applied for large scale problems and to improve the efficiency of weakly coupled FSI problems [13]

Partitioned methods, on the other hand, have independent solvers for the fluid and solid domains with a common interface to enforce the coupling required for fluid-structure interactions [14, 15, 16]. This approach offers flexibility for the types of physics that can be included in each domain as well as the ability to solve non-matching meshes for the two domains provided suitable interpolation methods are added at the interface. The convergence of the coupled solution using this approach is limited by the outer partitioning that controls the coupled solution. Typical coupling approaches include block Jacobi or block Gauss-Seidel. By definition, Jacobi methods use the solutions obtained from the previous iteration until the full solution for the next coupled iteration is found, whereas Gauss-Seidel methods use the current solutions as soon as they are available. Block Gauss-Seidel (BGS) approaches have commonly been used for coupled aeroelastic problems [16, 17, 18] using the partitioned approach which iterates the first solver whilst keeping the second fixed, then transfers the solution from the first solver to the second solver which is then iterated. The coupling of the solution at the interface is enforced and this process of the outer iteration is repeated until convergence of the coupled solution.

With the partitioned approach, strongly coupled problems with high fluid to structure density ratio have been known to converge slowly or even diverge [18]. The monolithic approach has been considered as the preferred method for very strongly coupled FSI in the past due to the better stability of the 'all-together' approach, which is intrinsically suited for such problems. The numerical difficulties are attributed to the effect of 'added mass', where there is additional inertia in the system arising from the accelerating solid requiring to move the volume of fluid.

This occurs when the fluid density is close to the structural density, however, the numerical instabilities from 'added mass' have been shown to be present challenges in both monolithic and partitioned solvers as well as for strongly or weakly coupled problems [19]. For the partitioned solver, the BGS convergence can be accelerated using relaxation approaches such as Aitken's relaxation [20, 21]. In addition to this, Interface-Newton techniques have been developed which reformulate the coupled problem as a discretized equation on the fluid-structure interface. Degroote has proposed a method for Interface-Quasi-Newton with inverse Jacobian from a least-squares model (IQN-ILS) [18] which has been shown to be even more efficient than the monolithic method for a series of incompressible fluid benchmark cases with FSI.

A coupled three-field formulation has been implemented by Sanchez et al. [22] in SU2 with BGS for the outer coupling. SU2 [23] is a high-fidelity computational design and analysis package suitable for solving and optimizing multiphysics problems. It is an open-source software, originally developed in Stanford University, and is well supported by numerous academic institutions including Imperial College London. The software is developed in a modular format and is actively developed; hence it provides a robust platform with diverse applications [24]. In addition to the fluid and structural domains, the mesh solution has been designed to be a separate third domain as originally suggested by [14] in order to robustly facilitate large structural deformations. In this partitioned approach, the continuity of fluid tractions and structural displacements are enforced. An IQN-ILS interface has been added by Gomes [25] for improved convergence.

The partitioned approach enables coupling across different types of fluid and structural solutions. The primary fluid solver in SU2 is a compressible RANS solver, however this has been expanded to include steady incompressible solver suitable for low speed flows including large density variations due to temperature [26], non-equilibrium models for hypersonic flows [27] as well as higher fidelity methods such as hybrid RANS-LES [28]. For the structural solver, both linear elastic and hyperelastic neo-Hookean material models have been implemented [22].

Multidisciplinary Design Optimization

Optimization of aeroelastic problems is a form of multidisciplinary design optimization (MDO). MDO focuses not only on the behaviour of individual disciplines but also on the interaction between them, which in this case is between the structural and fluid components. This has been an area of interest for aero-structural problems for many decades with works by Schmit [29] and Haftka [30, 31] as initial pioneers. Traditionally the design process was sequential with aerodynamic optimization followed by structural design and then control system design. This design process is repeated iteratively until the converged optimum result is found. With this approach either convergence for the optimum design would take a long time or otherwise a sub-optimum design is outputted as the final achievable result. However, in multidisciplinary optimisation paradigm, the sequential design process is replaced by a coupled approach that includes cross-dependencies in the search for the optimal design. Martins et al. [32] have presented an excellent review of the main MDO architectures, which has been followed on more recently by Hwang et al. [33] with a review of current state-of-the-art approaches in practical applications along with recommendations with particular focus on large-scale MDO.

Management of the coupling of the MDO system is one of the major challenges and the different approaches can be grouped into four main categories [32]. The All At Once (AAO) [34] approach is the most generic form of describing the MDO, however is the most expensive since all the coupling variables, state variables, consistency constraints and residuals of the governing equations are explicitly defined and included in the optimization. Introducing a single group of coupling variables in order to remove the consistency constraints yields the Simultaneous Analysis and Design (SAND) [35] approach whilst eliminating the discipline-analysis constraints yields the Individual Discipline Feasible (IDF) [34] approach. Both of these approaches are cheaper than AAO with IDF being even smaller than SAND since the coupling variables are treated through implicit functions. An alternative architecture is the Multidisciplinary Feasible (MDF), in which both the analysis and consistency constraints are removed from problem [34]. In this approach, the optimizer controls the objective function, design variables and design constraints whilst the system returns the solution and designs that have satisfied all the do-

mains and coupling criteria for the MDO. As a result, the cost of each iteration will be higher than IDF since the full multidisciplinary analysis has to be performed for each design iteration; however, this is the most robust approach for large scale problems [33].

Furthermore, the choice of a MDF approach separates the system of governing equations, which can be solved using the partitioned approach as discussed above, and the optimizer, which can be selected independently from a variety of available algorithms, typically separated into either zero order (gradient-free) or first order (gradient based) methods. Zero-order optimization is driven purely by the calculation of the objective function in each iteration. Three very common examples for this type of optimization are discussed next. Hooke and Jeeves [36] proposed a form of pattern search algorithm that directs local search through the domain, whilst keeping track of the direction of travel rather than restarting from the beginning for each new point. The Nelder-Mead simplex algorithm [37] is another form of direct search method, which instead is simplex based with generation of a triangle in 2D or tetrahedron in 3D in the design space. The function value at the corner points are aimed to be reduced in each iteration in order to move closer to the optimum. Specifically, the Nelder-Mead algorithm allows the shape as well as the size of the working simplex to change with each iteration, which makes it more efficient than other simplex approaches. Finally, the genetic algorithms, originally introduced by Holland [38] and later reviewed by Rios and Sahinidis in [39], are inspired from natural selection and survival of the fittest. In this approach, the state has an associate gene domain with genetic modifiers that represent and modify the fitness level such that the points that are the strongest can pass the gene forward to survive the longest, which will lead to the optimum for the objective function. These gradient free methods have been developed and prevalent for decades and are still widely used today [40, 41, 42] due to the suitability for application in non-smooth, discontinuous problems.

First-order optimization is driven using the direction dictated by the sensitivity of the coupled system using gradients for output of interest from the governing system with respect to chosen design parameters. Due to the computational cost for each iteration in an MDO framework, particularly for the MDF approach, gradient-based optimization methods become the only feasible option for very large coupled systems and large design space, as described by Martins

et al. [43]. The required gradient can be found in a number of ways which will be discussed in the next section and the most appropriate method should keep accuracy high and computational cost low.

A well established optimization algorithm (employed with the SU2 suite) is Sequential Least Squares Quadratic Programming (SLSQP) [44] with applications in aerodynamic optimization [45], aeroacoustic optimization [46] and fluid-structure interaction optimization [47, 25]. Therefore, continued use of SLSQP for small/medium scale problems should be sufficient. However, to move to large scale problems, interior point methods such as IPOPT [48], would be recommended. One of the major advantages of this method is that it has low sensitivity to the number of constraints [49]. In an interior point method, the inequality constraints are replaced by the addition of a penalizing term on the cost function (known as barrier function), which tends to infinity as the point approaches the bounds of the feasible region [33]. If continued use of active-set methods are preferred in which search directions are created using solutions to a series of quadratic sub-problems which have linearized inequality constraints, then an alternative to SLSQP is SNOPT [50] which has been used extensively in large scale MDOs [33, 51, 52].

The importance of coupled optimization particularly for highly coupled, dynamic systems are discussed by Allison [53] where the lack of the modelling of dynamic behaviour in MDO problems is highlighted since rather than using full non-linear dynamics of a system, either a simplified linear system or steady-state behaviour were typically utilized. The work by Haghighat et al. [2] also emphasises the importance of introducing aeroelasticity in the design process as early as possible in aeroservoelastic modelling applications.

The parallels between MDO and numerical strategies were also highlighted for optimal control, with co-design of dynamic systems capturing the dynamics of the system to a greater level used for such optimal control-systems whilst still lacking for optimal physical systems. For example of MDO application to aircraft dynamics, Klesh et al. [5] found optimal trajectories for perpetual flight of solar aircraft based on simplified flight-dynamics. However, for models with higher fidelity Maraniello [54] found that the number of design parameters could be controlled by separating the state and control variables through the single shooting or control

vector parameterization method. In this method, the coefficients of the control input parameterization alone are treated as design parameters at each iteration. The state of the system is evaluated separately through aeroelastic analysis which reduces the design space and makes the optimization problem computationally tractable.

Since the modelling of fully coupled dynamics is computationally expensive for the primal computation, this naturally becomes even more difficult for optimization. Fully non-linear elastic structural model have previously been used by Buoso for analysing aeromechanical performance of dynamically actuated membrane wings made of dielectric elastomers [55]. This methodology has extremely high computational cost and quickly becomes impractical. Maraniello et al. have used reduction techniques [56] to model the aeroelastic characteristic using a geometrically exact beam model to capture the structural dynamics and unsteady vortex lattice method for the fluid dynamics in a software called SHARPy (Simulation of high aspect ratio planes in Python) with application in co-design of very flexible beams using finite differences and a single-shooting approach. Though the cost associated with the model has been reduced compared to the fully non-linear model, due to the large number of simulations required by the single-shooting approach, high computing power is still necessary but the results have been shown to lead to a robust solution. Open loop responses have also been used for optimization of flexible flight dynamics in SHARPy with a successful optimization for catapult assisted take off manoeuvres [57].

Evaluation of System Sensitivities

The field of sensitivity analysis has been extensively studied due to its influence in gradient-based optimization, uncertainty quantification and model development. The evaluation of the required derivatives can be performed in a number of ways, with finite differences arguably the simplest. The mathematical derivation is based on the Taylor series expansions and due to its simplicity, it is widely used in engineering applications as the default method in gradient-based optimization despite the inherent cancellation and truncation errors [50, 58]. The cost of computing the complete sensitivity of the system is proportional to the number of input

variables as each one has to be perturbed in order to evaluate its sensitivity [59].

An alternative method through complex variables was originally developed by Lyness in 1967 [60], to compute n-th derivatives of real, analytic functions. This method was more widely adopted after Squire and Trapp presented a simple formula to estimate the first derivative [61]. The cost associated with this method is the same as the finite differences however the error is reduced as there is no cancellation error.

Through application of the chain rule, the derivatives can also be computed using a direct or adjoint method. The direct method solves a linear system for each input, therefore the cost of this method is the same as the finite differences or complex method. However, the advantage is that for each input the equations to be evaluated is a linear system regardless of whether the original governing equations are linear or non-linear [43]. The adjoint method is an alternative to the direct method to calculate the total derivative which is based on the number of output quantities of the system. Therefore, for systems with many input variables, the cost of the computation of the derivatives would be cheaper for the adjoint method than any others discussed.

Given the advantages and challenges of both methods, the adjoint method has been the preferred choice for many complex, engineering problems. Among the many contributions, Jameson has applied the adjoint method for optimal aerodynamic shape design [62], Maute et al. [63, 64, 65] have applied it for multidisciplinary sensitivity analysis and Martins et al. [66, 67] have applied it for aero-structural optimizations.

The difficulty with both the direct and adjoint method is that the exact analytic Jacobian of the system is required, which becomes impractical for very complex systems. The use of approximate Jacobian, which is typically computed for the Newton iterations of the primal problem, have been used for non-linear structures by Lund et al. [68] but only for academic proof of concept scenarios, since the problems are solved in monolithic form which cannot be scaled for realistic large-scale problems [17].

Algorithmic differentiation (AD) is an automated, systematic application of the chain rule to

obtain the Jacobian of the system without evaluating it analytically. Numerical methods to compute the sensitivity can introduce truncation error to the result however AD applies the chain rule of differentiation to every operation in the program such that for each intermediate variable, a variation due to one input variable is carried through the code and does not incur those truncation errors [69, 70].

In terms of existing AD tools, there are many options available in most programming languages that would be of interest, including Fortran, C/C++ and Matlab. The implementation of these AD tools can generally use source code transformation and/or operator overloading [47]. Operator overloading modifies or creates new semantics and data types which enable the program to store the value of a variable and its derivatives. Source code transformation creates a modified code that is based on the original code but contains the additional code necessary for computation of the derivatives [71]. AD has been used for physical application in a range of fields including race car performance [72], sonic boom reduction [73] and thermodynamic wavy film flows [74], among others.

As an introduction to the AD techniques, tools in Matlab offer a familiar platform, for the author as well as possibly many readers, to consider the advantages and disadvantages of the types of methods that could be employed since Matlab is used extensively in both academic and industrial settings. Coleman created one of the first AD methodologies in Matlab called ADMAT which was based on operator overloading with various AD algorithms [75]. This was later improved through an additional package ADMIT-1 which interfaces with ADMAT to obtain sparse Jacobian using bicoloring approach by combining row and column views to efficiently extract the non-zeros values [76]. ADMIT-1 also computes the Hessian through both direct and substitution methods as initiated by Coleman and Cai [77]. Forth improved the efficiency and in particular the scalability of the AD method in Matlab when compared to ADMAT which led to the tool called MAD [78]. MAD is another operator overloaded tool in Matlab with the improvement in efficiency due to the support of Matlab sparse data-type. This allows sparse derivatives to be held and propagated in the algorithm which is particularly effective when the sparsity cannot be specified beforehand.

Improving the work of Forth, Kharche et al. [79], proposed MSAD which is a development on MAD to include source code transformation for better efficiency which was seen to be much faster than MAD for small scale problems but time taken increased with the problem size to be similar for large problems.

Moving away from pure operator overloading methods, Patterson et al. [80], created ADiGator, which is a hybrid tool that uses its own Matlab object class called CADA (Computation of Analytic Derivatives Automatically). The employed methodology evaluates the function of interest using the CADA input object and the source code transformation is also performed by the CADA object which produces the file with the necessary code to compute the required derivatives. However, the main limitation of this code at present is that it is only available for the forward mode.

Bischof et al. [81], proposed an alternative hybrid tool called ADiMat using a combination of source transformation and operator overloading. Single directional derivatives are computed by the modified source code whereas for multidirectional derivatives either the Matlab overloading or Matlab-Mex is utilised. Despite improvements in the AD methodology in Matlab, the limitation for coupled FSI application comes due to the level of model fidelity that is possible in this environment.

For AD tools with established application in multidisciplinary software, ADFlow [82] is an option which uses source code transformation. This method has been presented as a desired option because with careful implementation [83] ADFlow has shown this approach to minimize memory usage and offer high computational speed. However additional changes are necessary to the original code in this approach. Code Differentiation Pack (CoDiPack) [84] on the other hand is a C++ library for AD based on operator overloading. This approach requires fewer changes to the original primal code, but can have lower efficiency since the method stores both the primal value and it's derivative for all variables. Improvements for computational speed have been addressed through use of Expression Templates, which has been demonstrated as a successful approach for overloaded AD tools in C++ having been applied originally FAD [85]. This method allows the evaluation of the expressions without the need for temporary values.

CoDiPack has been interfaced with SU2 to enable forward and reverse mode AD for direct and adjoint solutions. Albring et al. [86, 45] have developed and applied a consistent discrete adjoint solver for aerodynamic design framework. For this purpose, fixed-point iterators have been used to define the governing system with linearization of fixed-point iterations instead of the governing equations to enable the convergence properties of the primal solver to be inherited by the direct and adjoint solvers [86]. This methodology has been extended to static coupled FSI by Sanchez in [47] which has later been adopted for a generalized multizone discrete adjoint approach [87]. For dynamic problems, the memory requirements grow due to the need to store the primal solutions at each time step for the adjoint calculations. As the problem size and the duration of the time-accurate simulations increases, the memory requirements can become very high. To avoid prohibitively large requirements, methods such as checkpointing [88, 89] have been developed which can reduce the number of solutions that are stored. Only the primal solution data at known checkpoints are required because during the adjoint evaluation, the necessary primal solutions can be evaluated as and when required from the nearest stored checkpoint solutions.

1.2 Objectives

Review of the existing literature has highlighted that efficient methods for high fidelity MDO is still an open challenge for both steady and dynamic problems. Though frameworks towards aeroelastic optimizations have been investigated, the existing focus has been biased towards structural objectives and design variables with few existing for non-linear coupled solutions with high fidelity on both structural and fluid solutions. This drives the focus of this research towards efficient high-fidelity methods for fully non-linear, coupled FSI methodologies for aerodynamic performance and aerodynamic shape optimization.

In particular, dynamic problems are of interest with an aim to produce the first non-linear, dynamic FSI optimization at high fidelity. The importance of avoiding simplifications when considering dynamic MDO has been discussed. However, a framework that can support dynamic

MDO for high-fidelity analysis, with efficient sensitivity calculations for robust optimization, of various types of coupled problems, remains an open problem. Presentation of the theoretical and practical challenges for such a framework along with the pathway for their solutions will form part of the contribution from this research.

The existing examples in literature rarely had flexibility to model various flow regimes. The monolithic approaches for FSI typically restrict the variety of problems that can be explored. The partitioned FSI framework in SU2 for both the primal and adjoint solvers on the other hand, provides an ideal platform to consider a wider range of FSI applications. The extension of the current compressible FSI solver to also include incompressible flows will help facilitate this, which requires the development of a new unsteady incompressible solver within the SU2 suite. The addition of the incompressible solver within the same framework will readily allow incorporation of the unsteady incompressible physics for the discrete adjoint evaluations as well. This will also enable further expansion to other unsteady multi-physics problems such as conjugate heat transfer (CHT), which utlilizes the incompressible solver.

Adopting a generalized multiphysics approach for the unsteady discrete adjoint implementation will add significant value to the contribution from this work since the same implmentation developed for unsteady FSI can be employed by other unsteady multiphysics problems as well. A generalized framework that supports scaling up of the number of multiphysics domains will even allow problems such as unsteady FSI with CHT. The necessary characteristics of such a framework and their implications in terms of the individual domain solvers as well as the interface for coupling will also need to be explored further for the extension from steady to unsteady coupled analysis for both the primal and adjoint solutions.

The main objective is to explore the suitability of a generalized partitioned solver approach for evaluation of discrete adjoint sensitivities for unsteady fluid-structure interactions. To achieve this, this work will seek answers to:

1. What are the necessary changes for accurate capture of the unsteady coupled interactions that are not included in the steady discrete adjoint formulation?

2. What are the development and computational costs in the extension of the steady coupled discrete adjoint solver for dynamic problems?

3. What are the key challenges in the addition of an unsteady incompressible fluid solver in SU2 and what are the benefits of this for coupled optimal design?

1.3 Publications

Based on the work discussed in this thesis, the following papers have been published:

[90] Venkatesan-Crome, C., Sanchez, R., Palacios, R. (2018), Aerodynamic optimization using FSI coupled adjoints in SU2, *6th European Conference on Computational Mechanics (ECCM 6)*, Glasgow, UK, 11-15 June, 2018.

[91] Venkatesan-Crome, C., Gomes, P.C., Palacios, R. (2019), Optimal Compliant Airfoils Using Fully Non-Linear FSI Models, *AIAA Scitech 2019 Forum, (AIAA 2019-1216)*, San Diego, CA, USA, 7-11 January, 2019.

[92] Venkatesan-Crome, C., Kattmann, T., Palacios, R., Sanchez, R., Gauger, N., Economon, T. (2019), Discrete Adjoint for Unsteady Incompressible Flows Using a Density-based Formulation, *AIAA Aviation 2019 Forum (AIAA 2019-3669)*, Dallas, TX, USA, 17-21 June, 2019

[93] Venkatesan-Crome, C., Palacios, R. (2020), A Discrete Adjoint Solver for Time-Domain Fluid-Structure Interaction Problems with Large Deformations, *AIAA Scitech 2020 Forum, (AIAA 2020-0406)*, Orlando, FL, USA, 6-10 January, 2020.

1.4 Thesis Layout

This thesis is structured to firstly present methodology for high fidelity coupled FSI simulations including the incompressible flow solver adapted for unsteady simulations in Chapter 2. The discrete adjoint method to obtain efficient sensitivities for steady optimal design followed by the proposed formulation to obtain accurate sensitivities for unsteady coupled FSI problems is presented in Chapter 3. The methodology has been applied for aeroelastic shape optimization

for improved aerodynamic performance which is presented and discussed in Chapter 4. In order to build towards dynamic coupled FSI problems, initially optimal design using structural properties and shape parameters for dynamic structural problems are presented in Chapter 5. Chapter 6 presents the verification and validation of the new incompressible flow solver and concludes with optimal design of a plunging airfoil. The primal and adjoint results for dynamic coupled FSI problems are finally presented in Chapter 7 using the new framework for efficient sensitivities and optimal design implemented in SU2. The thesis concludes with Chapter 8 where a summary of the key findings, main contributions are presented along with proposals for future work to continue the development of optimal design and control of dynamic FSI problems for physical applications.

Chapter 2

A High-Fidelity Dynamic Model for Fluid-Structure Interactions

This chapter presents the technical formulation of a coupled fluid-structure interaction solver suitable for time-accurate problems with large deformations. The chosen environment for the implementation is SU2 [94], an open-source multiphysics solver. A three-field partitioned solver was originally implemented by Sanchez [22] using the compressible fluid solver, a new structural solver and a pseudo-linear elastic mesh solver. This has been used as the foundation for this work.

The fluid domain is firstly introduced with a brief description of the baseline compressible, RANS solver. An unsteady, incompressible solver using a density-based approach has been designed and implemented, the details of which are included in this chapter. The flow is solved using the Arbitrary Lagrangian-Eulerian formulation in order to support incompressible, coupled FSI problems. The structural solver is then described here with a full dynamic derivation and summary of the implementation.

The details of the boundary conditions at the interface are discussed, before presenting the final governing equations along with the coupling strategies. Finally, the architecture of the implementation is described.

2.1 Fluid Domain

There are multiple solvers for the modular SU2 framework. The primary solver is for compressible, laminar flow governed by the Navier-Stokes equations [94, 24], given in the Arbitrary Lagrangian-Eulerian formulation as

$$\frac{\partial \mathbf{w}}{\partial t} + \nabla \cdot \mathbf{F}^c_{ALE} - \nabla \cdot \mathbf{F}^v - \mathbf{Q} = \mathbf{0}, \; t > 0 \tag{2.1}$$

where $\mathbf{F}^c_{ALE}$ and $\mathbf{F}^v$ are the convective and viscous fluxes respectively and $\mathbf{Q}$ is a generic source term,

$$\mathbf{w} = \{\rho_f, \rho_f \mathbf{v}, \rho_f e\}^\mathsf{T}, \quad \mathbf{F}^c_{ALE} = \begin{Bmatrix} \rho_f(\mathbf{v} - \dot{\mathbf{z}}) \\ \rho_f \mathbf{v} \otimes (\mathbf{v} - \dot{\mathbf{z}}) + \bar{\bar{I}} p \\ \rho_f e(\mathbf{v} - \dot{\mathbf{z}}) + p\mathbf{v} \end{Bmatrix}, \quad \mathbf{F}^v = \begin{Bmatrix} \mathbf{0} \\ \bar{\bar{\boldsymbol{\tau}}} \\ \bar{\bar{\boldsymbol{\tau}}} \cdot \mathbf{v} + \mu C_p \nabla T \end{Bmatrix} \tag{2.2}$$

where ρ_f is the fluid density, $\mathbf{v} = \{u, v, w\}^\mathsf{T} \in \mathbb{R}^3$ is the vector of flow velocities in a Cartesian system of reference, $\dot{\mathbf{z}} = \{\dot{z}_u, \dot{z}_v, \dot{z}_w\}^\mathsf{T}$ is local inertial velocity due to mesh deformation, e is the total energy per unit mass and p is the static pressure, $\bar{\bar{\boldsymbol{\tau}}}$ is the viscous stress tensor, μ is the fluid viscosity, C_p is the specific heat and T is the temperature.

Boundary conditions such as flow tangency wall, no-slip adiabatic and isothermal wall can be enforced together with far-field boundary for external flows as well as with inlet and outlet boundary conditions for internal flows.

For spatial integration of the governing equations, finite-volume method based on dual-grid, with control volumes centred on grid vertices is used [24]. The convective and viscous fluxes are evaluated at the midpoint of control volume edges. The semi-discrete form of the governing equations for a control volume, Ω_i, are thus given by

$$\int_{\Omega_i(t)} \frac{\partial \mathbf{w}_i}{\partial t} \, d\Omega + \sum_{j \in \mathcal{N}(i)} (\tilde{\mathbf{F}}^c_{ALE_{ij}} - \tilde{\mathbf{F}}^v_{ij}) \Delta \tilde{S}_{ij} - \tilde{\mathbf{Q}}_i \, |\Omega_i| = 0, \tag{2.3}$$

where the projected numerical approximation for the convective and viscous fluxes flux are given

by $\tilde{\mathbf{F}}^c_{ALE}$ and $\tilde{\mathbf{F}}^v$ along all edges neighbouring to node i as defined by $\mathcal{N}(i)$, with area of the face given by $\tilde{S}_{ij}$ and the numerical approximations for the source term given by $\tilde{\mathbf{Q}}_i$. The convective fluxes can be discretized using either an upwind flux difference splitting Roe scheme [95] or a centered Jameson, Schmidt, and Turkel (JST) scheme [96]. The centered scheme is second order and second order is also achieved for the upwind scheme using a MUSCL approach [97]. The viscous fluxes are computed using a corrected average-gradient method [98] with the gradients calculated at all vertices using either Green-Gauss or Weighted Least-Squares approach [99]. Source terms are approximated at each vertex using piecewise constant reconstruction within each of the dual control volumes. Turbulence model terms are computed similarly to the mean flow, with convective terms through a scalar upwind scheme, viscous terms through a corrected average-gradient method and turbulent source terms using piecewise constant reconstruction. The terms from the spatial discretization are summarized as R_i thus simplifying (2.3) to

$$\int_{\Omega_i(t)} \frac{\partial \mathbf{w}_i}{\partial t}\, d\Omega + R_i(\mathbf{w}) = 0, \tag{2.4}$$

Dual-time stepping strategy is employed for time accurate solution of the fluid domain which introduces a time derivative with respect to pseudo-time τ as

$$\int_{\Omega_i(t)} \frac{\partial \mathbf{w}_i}{\partial \tau}\, d\Omega + \int_{\Omega_i(t)} \frac{\partial \mathbf{w}_i}{\partial t}\, d\Omega + R_i(\mathbf{w}) = 0, \tag{2.5}$$

such that upon convergence the pseudo-time derivative tends to zero and solution to the governing equations is found. The implementation transforms the unsteady problem into a series of steady problems at each physical time step. Using an implicit backward-Euler approximation for the pseudo-time derivative, the solution in the pseudo-time iterations, in p, are given by

$$\mathbf{w}^{p+1} = \mathbf{w}^p - \left(\frac{|\Omega_i|}{\Delta \tau_i} \delta_{ij} + \frac{\partial R_i}{\partial \mathbf{w}^p} \right)^{-1} R_i(\mathbf{w}^p). \tag{2.6}$$

The dual time stepping approach then solves the problem defining the fluid domain as

$$\mathbf{F}(\mathbf{w}, \mathbf{z}, \dot{\mathbf{z}}) = \int_{\Omega_i(t)} \frac{\partial \mathbf{w}_i}{\partial \tau}\, d\Omega + R^\star_i(\mathbf{w}) = 0, \tag{2.7}$$

with physical time iterations n to give

$$R_i^\star(\mathbf{w}) = \frac{3}{2\Delta t}\mathbf{w}_i + \frac{1}{|\Omega_i|^{n+1}}\left(R_i(\mathbf{w}) - \frac{2}{\Delta t}|\Omega_i|^n\mathbf{w}_i^n + \frac{1}{2\Delta t}|\Omega_i|^{n-1}\mathbf{w}_i^{n-1}\right) \tag{2.8}$$

given by second order accuracy with backward differences in time for Δt as the physical time step and $\mathbf{w} = \mathbf{w}^{n+1}$ once the steady problem is converged. Therefore the solution from the fluid domain can be written using a fixed point iterator $\mathcal{F}$ as

$$\mathbf{w} = \mathcal{F}(\mathbf{w}, \mathbf{z}, \dot{\mathbf{z}}), \tag{2.9}$$

where the fluid domain in this case was the compressible solver but can be replaced by other solvers of interest.

The grid velocity dependency on the fluid domain can arise from boundary motion due to coupled interfaces or through prescribed motion. As part of this work, deforming dynamic meshes, described in [100], with pitching and plunging motion have been implemented for analysis and design. In addition to this, a rotating reference frame, which solves the flow field by transforming the governing equations to a non-inertial frame of reference that rotates around an object, enables verification of dynamic grid problems in a steady environment. The benefits of the simplification to a steady problem will be used for preliminary verification of new unsteady solvers. In addition to these, wind gust has been applied and validated in SU2 by Padrón [101], which had considered fluid only problems. The wind gust is applied using the grid velocities such that the desired gust is expressed as negative grid velocity to the baseline grid velocity field following the proposed method by Bartels [102]. This will be of interest further in this work when wind gust will be extended to coupled problems.

2.1.1 Unsteady Incompressible Solver

As part of the contributions from this work, an unsteady incompressible solver in Arbitrary Lagrangian-Eulerian formulation has been developed and implemented [92]. The addition of unsteady incompressible solver will augment the range of validity of the solvers for analysis

of low-speed air vehicle across their flight envelope within the single code base. The new solver is suitable for constant density flow as well as flows with large density variations due to temperature. This generalized framework uses low-Mach approximations from asymptotic analysis using a simplified form of the energy equations [103].

The governing equations in domain $\Omega \subset \mathbb{R}^3$ have the same form as (2.1),

$$\frac{\partial \mathbf{w}}{\partial t} + \nabla \cdot \mathbf{F}^c_{ALE}(\mathbf{w}) - \nabla \cdot \mathbf{F}^v(\mathbf{w}, \nabla \mathbf{w}) - \mathbf{Q} = 0 \tag{2.10}$$

where the incompressible conservative variables are now used, given by $\mathbf{w} = \{\rho_f, \rho_f \mathbf{v}, \rho_f c_p T\}^\mathsf{T}$.

The convective and viscous fluxes can be written as

$$\mathbf{F}^c_{ALE} = \left\{ \begin{array}{c} \rho_f(\mathbf{v} - \dot{\mathbf{z}}) \\ \rho_f \mathbf{v} \otimes (\mathbf{v} - \dot{\mathbf{z}}) + \bar{\bar{I}} p \\ \rho_f C_p T(\mathbf{v} - \dot{\mathbf{z}}) \end{array} \right\}, \quad \mathbf{F}^v = \left\{ \begin{array}{c} \cdot \\ \bar{\bar{\tau}} \\ \kappa \nabla T \end{array} \right\}, \tag{2.11}$$

where dynamic pressure is given by p while the thermodynamic pressure, p_o, is assumed constant throughout the domain and in time. The decoupled pressure-density system of equations are closed using an equation of state for density as a function of temperature alone. Assuming an ideal gas with gas constant R, density is given by $\rho = p_o/RT$. The energy equations are also included in Eqn. (2.11) where C_p is the specific heat at constant pressure, T is the temperature, κ is the thermal conductivity.

Viscosity is given by μ and $\bar{\bar{\tau}}$ is the viscous stress tensor. For laminar flows, viscosity is purely given by the dynamic viscosity, μ_{dyn}, which can be constant or assumed to vary with temperature according to Sutherland's law [104]. For turbulent flows, Reynolds-averaged Navier-Stokes (RANS) equations are solved with either the Shear Stress Transport (SST) model of Menter [105] or the Spalart-Allmaras (S-A) [106] model.

A density-based solution method

The approach of Weiss and Smith [107] is applied to develop a time-accurate density-based strategy for the low-Mach equations in Eqn. (2.10). In this work, only incompressible flows are considered rather than a unified approach for compressible and incompressible flows. Similar to the density-based approach for steady incompressible flow given by Economon [26], the derivation for unsteady incompressible flow begins with the assumption that equation of state for density is a function of both pressure and temperature, $\rho_f = \rho_f(p, T)$.

For time-accurate solutions, a pseudo-time derivative (in τ) is introduced to the governing equations to give

$$\frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{w}}{\partial \tau} + \nabla \cdot \mathbf{F}^c_{ALE}(\mathbf{w}) - \nabla \cdot \mathbf{F}^v(\mathbf{w}, \nabla \mathbf{w}) - \mathbf{Q} = 0. \tag{2.12}$$

Primitive variables, $\tilde{\mathbf{w}} = \{p, \mathbf{v}, T\}^\mathsf{T}$, will be used as the working variables for the incompressible solver. However, the governing equations remain in conservative form, and the time derivative with respect to the physical time will be retained in terms of conservative variables, $\mathbf{w}$, due to the dual-time formulation. Therefore, the governing equations can be written as

$$\frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{w}}{\partial \tilde{\mathbf{w}}}\frac{\partial \tilde{\mathbf{w}}}{\partial \tau} + \nabla \cdot \mathbf{F}^c_{ALE}(\mathbf{w}(\tilde{\mathbf{w}})) - \nabla \cdot \mathbf{F}^v(\mathbf{w}(\tilde{\mathbf{w}}), \nabla \mathbf{w}(\tilde{\mathbf{w}})) - \mathbf{Q} = 0, \tag{2.13}$$

where we have introduced the Jacobian

$$\frac{\partial \mathbf{w}}{\partial \tilde{\mathbf{w}}} = \begin{bmatrix} \rho_p & 0 & 0 & 0 & \rho_T \\ \rho_p u & \rho & 0 & 0 & \rho_T u \\ \rho_p v & 0 & \rho & 0 & \rho_T v \\ \rho_p w & 0 & 0 & \rho & \rho_T w \\ \rho_p c_p T & 0 & 0 & 0 & \rho_T c_p T + \rho c_p \end{bmatrix} \tag{2.14}$$

of the transformation between the conservative variable vector and the primitive vector. The partial derivatives of the equation of state with respect to pressure and temperature are given by $\rho_p = \partial \rho_f / \partial p$ and $\rho_T = \partial \rho_f / \partial T$, respectively. The equations are converted to non-conservative

form through pre-multiplication of an additional transformation matrix K, given by

$$K = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -(u - \dot{z}_u) & 1 & 0 & 0 & 0 \\ -(v - \dot{z}_v) & 0 & 1 & 0 & 0 \\ -(w - \dot{z}_w) & 0 & 0 & 1 & 0 \\ -c_p T & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{2.15}$$

to give

$$K\frac{\partial \mathbf{w}}{\partial t} + K\frac{\partial \mathbf{w}}{\partial \tilde{\mathbf{w}}}\frac{\partial \tilde{\mathbf{w}}}{\partial \tau} + K\,\nabla \cdot \mathbf{F}^c_{ALE} - K\,\nabla \cdot \mathbf{F}^v - K\mathbf{Q} = 0. \tag{2.16}$$

Under convergence the pseudo-time derivative tends to zero, therefore matrix

$$K\frac{\partial \mathbf{w}}{\partial \tilde{\mathbf{w}}} = \begin{bmatrix} \rho_p & 0 & 0 & 0 & \rho_T \\ \rho_p \dot{z}_u & \rho_f & 0 & 0 & \rho_T \dot{z}_u \\ \rho_p \dot{z}_v & 0 & \rho_f & 0 & \rho_T \dot{z}_v \\ \rho_p \dot{z}_w & 0 & 0 & \rho_f & \rho_T \dot{z}_w \\ 0 & 0 & 0 & 0 & \rho_f c_p \end{bmatrix}, \tag{2.17}$$

can be manipulated without affecting the solution. For an incompressible flow, $\rho_p = 0$. To help with numerical convergence, this term is replaced by a value inversely proportional to the magnitude of the velocity in the domain. In particular, the first column of $K\frac{\partial \mathbf{w}}{\partial \tilde{\mathbf{w}}}$ is replaced by $\beta_f^{-1}(1, \dot{z}_u, \dot{z}_v, \dot{z}_w, 0)$. This controls the magnitudes of eigenvalues of the system, where β_f is a factor related to the maximum velocity in the domain. The definition of β_f draws parallels with the artificial compressibility method, and a problem independent scaling definition used by Riedel [108] is adopted here as

$$\beta_f^2 = \epsilon^2(\mathbf{v} \cdot \mathbf{v})_{max}, \tag{2.18}$$

where ϵ is a user-provided value (default value of 2.0 in our implementation) and $(\mathbf{v} \cdot \mathbf{v})_{max}$ is the maximum squared velocity in the domain, which is updated with each time step for an

adaptive β_f value. Pre-multiplying the governing equations by K^{-1} leads to the final form of the density-based approach in conservative form, such that the incompressible governing equations are recovered when the pseudo-time derivative term is relaxed to zero:

$$\frac{\partial \mathbf{w}}{\partial t} + \mathbf{\Gamma}\frac{\partial \tilde{\mathbf{w}}}{\partial \tau} + \nabla \cdot \mathbf{F}^c_{ALE} - \nabla \cdot \mathbf{F}^v - \mathbf{Q} = 0, \tag{2.19}$$

where

$$\mathbf{\Gamma} = \begin{bmatrix} \frac{1}{\beta_f^2} & 0 & 0 & 0 & \rho_T \\ \frac{u}{\beta_f^2} & \rho_f & 0 & 0 & \rho_T u \\ \frac{v}{\beta_f^2} & 0 & \rho_f & 0 & \rho_T v \\ \frac{w}{\beta_f^2} & 0 & 0 & \rho_f & \rho_T w \\ \frac{c_p T}{\beta_f^2} & 0 & 0 & 0 & \rho_f c_p + \rho_T c_p T \end{bmatrix}. \tag{2.20}$$

The system including matrix $\mathbf{\Gamma}$ achieves pressure-velocity coupling and enables a suitable solution approach for incompressible problems. Matrix $\mathbf{\Gamma}$ treats all equations simultaneously with a pressure time derivative term, rather than only the density equation treatment seen in the classic artificial compressibility approach. Modified approaches have been used for incompressible flows with grid movement [109], in which the preconditioning matrix explicitly depends on the grid velocities. However, this is avoided here. The approach presented in this work for unsteady flows with grid motion uses $\mathbf{\Gamma}$, defined in Eqn. (2.20), which remains identical to the steady incompressible case. Thus, this approach results in a generic formulation for steady and unsteady flows, with and without grid motion, which has the added benefit of simplifying the implementation.

Spatial and Time Integration

Spatial integration methods similar to the compressible solver are inherited from the steady incompressible solver implementation[26] and a brief overview is presented here. The spatial discretization is performed using a finite volume method and the semi-discretized, integral form of the governing equations is obtained by integrating Eqn. (2.19) over the control volume Ω_i

and applying the divergence theorem to give

$$\int_{\Omega_i} \frac{\partial \mathbf{w}_i}{\partial t}\, d\Omega + \int_{\Omega_i} \mathbf{\Gamma}_i \frac{\partial \tilde{\mathbf{w}}}{\partial \tau}\, d\Omega + \sum_{j \in \mathcal{N}(i)} (\tilde{\mathbf{F}}^c_{ALE_{ij}} - \tilde{\mathbf{F}}^v_{ij}) \Delta S_{ij} - \tilde{\mathbf{Q}}_i \, |\Omega_i| = 0, \tag{2.21}$$

where $\tilde{\mathbf{F}}^c_{ALE_{ij}}$ and $\tilde{\mathbf{F}}^v_{ij}$ are the numerical approximations of the convective and viscous fluxes projected along an edge, respectively. The convective fluxes can be discretized using either an upwind flux difference splitting scheme (FDS) or a centered scheme based on a modified version of the Jameson, Schmidt, and Turkel (JST) scheme [96].

The viscous fluxes are computed as mentioned previously using a corrected average-gradient method [98] and source terms are approximated using piece-wise constant reconstruction at each vertex within the dual control volumes.

For time-accurate simulations, a dual-time stepping is once again used. The simulation is marched in physical time with an internal pseudo-time marching at each physical time-step. Upon convergence of the inner iterations, the preconditioned pseudo time derivative vanishes, but the physical time derivative remains. The coupled system of ordinary differential equations represented by Eqn. (2.21), can be rewritten as

$$\frac{d}{dt}\left(\mathbf{w}_i |\Omega_i|\right) + \frac{d}{d\tau}\left(\mathbf{\Gamma}_i \tilde{\mathbf{w}}_i |\Omega_i|\right) + R_i(\tilde{\mathbf{w}}^n) = 0, \tag{2.22}$$

where $R_i(\tilde{\mathbf{w}}^n)$ is the numerical residual that represents the integrated sum of all spatial terms for the control volume surrounding vertex i with cell volume $|\Omega_i|$. The superscript n indicates the current physical time step. For dynamic grids, the mesh coordinates and grid velocities are updated before each physical time step and Geometric Conservation Law (GCL) [110] has been implemented.

The physical time-dependent term is evaluated through an implicit backward differences scheme that can be either first or second order accurate similarly to the compressible fluid solver. The system in Eqn. (2.22) can be rewritten by adding the discrete spatial residual $R_i(\tilde{\mathbf{w}}^n)$ and the

discrete physical time derivative as

$$\frac{d}{d\tau}\left(\mathbf{\Gamma}_i \tilde{\mathbf{w}}_i |\Omega_i|\right) + R_i^\star(\tilde{\mathbf{w}}) = 0. \tag{2.23}$$

The conservative variables can be straightforwardly recovered from the primitive variables $\mathbf{w}^n(\tilde{\mathbf{w}}^n)$, such that $R_i^\star$ remains dependent only on $\tilde{\mathbf{w}}^n$. During the pseudo-time iterations, the known conservative variables of the past physical time steps, $\mathbf{w}^{n-1}$ and $\mathbf{w}^{n-2}$, are held constant. As $\tau \to \infty$ the pseudo time derivative vanishes and the solution for $\mathbf{w}^n$ and $\tilde{\mathbf{w}}^n$ are found.

2.2 Structural Domain

The baseline non-linear structural solver in SU2 has been implemented by Sanchez [22], following the finite deformation framework of Bonet and Wood [111]. An overview is provided here with the inclusion of inertial effects for context in the development of the dynamic fluid-structure interaction solver presented in this work.

The pointwise equilibrium condition for the linear momentum of the structural domain, Ω_s, is defined by

$$\nabla \cdot \boldsymbol{\sigma} + \mathbf{f} - \rho_s \ddot{\mathbf{u}} = \mathbf{0}, \tag{2.24}$$

where $\boldsymbol{\sigma}$ is the Cauchy stress tensor, $\mathbf{f}$ is the actuating forces per unit volume acting on the structure, ρ_s is the structural density and $\ddot{\mathbf{u}}$ is acceleration.

The weak form of the equilibrium is found through the principle of virtual work [111]. For an arbitrary virtual displacement, $\delta\mathbf{u}$, this is given by

$$\delta\mathbf{W} = \int_{\Omega_s} (\nabla \cdot \boldsymbol{\sigma} + \mathbf{f} - \rho_s \ddot{\mathbf{u}}) \cdot \delta\mathbf{u} \, \mathrm{d}\Omega_s = \mathbf{0}. \tag{2.25}$$

Additionally, external tractions per unit area, $\boldsymbol{\lambda}$, on the surface, Γ_s, are given in the definition

of the Cauchy stress tensor as

$$\boldsymbol{\lambda} = \boldsymbol{\sigma}\mathbf{n}. \tag{2.26}$$

Rearranging from (2.25) and combining with (2.26) leads to the expression for virtual work as

$$\delta\mathbf{W} = \underbrace{\int_{\Omega_s} \boldsymbol{\sigma} : \nabla \cdot \delta\mathbf{u} \, \mathrm{d}\Omega_s}_{\text{Internal}} - \underbrace{\left(\int_{\Omega_s} \mathbf{f} \cdot \delta\mathbf{u} \, \mathrm{d}\Omega_s + \int_{\Gamma_s} \boldsymbol{\lambda} \cdot \delta\mathbf{u} \, \mathrm{d}\Gamma_s \right)}_{\text{External}} + \underbrace{\int_{\Omega_s} \rho_s \ddot{\mathbf{u}} \cdot \delta\mathbf{u} \, \mathrm{d}\Omega_s}_{\text{Inertial}} = \mathbf{0}, \tag{2.27}$$

where the internal, external and inertial components of the virtual work are highlighted. The internal work is calculated with respect to the reference configuration unlike external and inertial contributions which are with respect to the current configuration. The dependency on the spatial quantities for the internal work can be removed by rewriting the internal virtual work, the first term in (2.27), as

$$\delta\mathbf{W}_{int} = \int_{\Omega} \boldsymbol{S} : \delta\dot{\mathbf{E}} \, \mathrm{d}\Omega, \tag{2.28}$$

where $\boldsymbol{S}$ is the second Piola-Kirchhoff stress tensor, which is the work conjugate to the material strain rate tensor, $\dot{\mathbf{E}}$, which in turn is the derivative with respect to time of the Lagrangian strain tensor, $\mathbf{E}$.

Boundary conditions are implemented to constrain displacements or impose external load on the structure. In particular, the clamp boundary condition enforces zero displacement velocity and acceleration on required boundaries. Prescribed displacement on a boundary can also be enforced. External forces can be applied as normal or directional pressure load along a boundary with constant, ramp or sinusoidal loading for time-domain problems as well as constant body forces.

A variety of material models could be employed in the structural solver due to a modular approach similar to the fluid solver. A linear elastic material is modelled for problems with small deformations using the classical small strain linear elasticity equations defined by the Young modulus and Poisson ratio. For non-linear problems of interest, isotropic hyperelastic material model are typically used and the compressible neo-Hookean model described by Bonet [111]

has been implemented by Sanchez [22]. For a brief description, hyperelastic material, which cannot use linear elasticity equations, are now described through stresses derived from a stored elastic energy function, Ψ, given by

$$\boldsymbol{S} = \frac{\delta \Psi}{\delta \dot{\mathbf{E}}}. \tag{2.29}$$

For the neo-Hookean material model, the stored elastic strain energy [111] is defined by

$$\Psi = \frac{\mu}{2}(I_{\mathrm{C}} - 3) - \mu \ln J + \frac{\lambda}{2}(\ln J)^2 \tag{2.30}$$

where I_{C} is the trace of the right Cauchy deformation tensor, J is the determinant of the deformation gradient and the Lamé coefficients are given by

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)} \quad \text{and} \quad \mu = \frac{E}{2(1+\nu)}, \tag{2.31}$$

where E is the Young's modulus and ν is the Poisson ratio of the material. For completeness, the Cauchy stress can be obtained in terms of the left Cauchy-Green tensor, $\mathbf{b}$, as

$$\boldsymbol{\sigma} = \frac{\mu}{J}(\mathbf{b} - \mathbf{I}) + \frac{\lambda}{J}(\ln J)\mathbf{I} \tag{2.32}$$

The equilibrium condition defined by (2.27) is discretized in space using the finite element method with isoparamateric elements to lead to the discrete structural problem

$$\mathbf{S}(\mathbf{u}) = \mathbf{T}(\mathbf{u}) - \mathbf{F}_b - \mathbf{F}_\Gamma + \mathbf{M}_s\ddot{\mathbf{u}} = \mathbf{0}, \tag{2.33}$$

where $\mathbf{T}(\mathbf{u})$ are the internal equivalent forces, $\mathbf{F}_b$ are the body forces, $\mathbf{F}_\Gamma(\mathbf{u}, \boldsymbol{\lambda}(\mathbf{w}, \mathbf{z}))$ are the surface forces acting over the boundary Γ and $\mathbf{M}_s$ is the mass matrix, with element properties defined using linear shape functions.

The structural problem is then linearized in the direction of an increment of displacement for

all contributions to the equilibrium [22], given by

$$\frac{\partial \mathbf{S}(\mathbf{u})}{\partial \mathbf{u}} \Delta \mathbf{u} = -\mathbf{S}(\mathbf{u}). \tag{2.34}$$

The solution about the equilibrium is converged through Newton-Raphson iterations until $\mathbf{S}(\mathbf{u}) \approx 0$ for non-linear problems, which for a current time, t, is given by

$$\mathbf{K}_i^t \Delta \mathbf{u}_{i+1}^t = -\mathbf{S}(\mathbf{u}_i^t), \tag{2.35}$$

$$\mathbf{u}_{i+1} = \mathbf{u}_i + \Delta \mathbf{u}_{i+1} \tag{2.36}$$

where the tangent matrix, $\mathbf{K}_i = [\mathbf{S}(\mathbf{u})/\partial \mathbf{u}]_i$, is assembled by the components of constitutive, initial stress and external loads. Therefore, the structural is found using a fixed-point iterator, $\mathcal{S}$, given by

$$\mathbf{u}_{i+1}^t = \mathcal{S}(\mathbf{u}_i^t) = \mathbf{u}_i^t - \left[\mathbf{K}_i^t\right]^{-1} \mathbf{S}(\mathbf{u}_i^t) \tag{2.37}$$

Time-accurate solution is found by marching in time using Newmark time integration [112] to obtain the expression for acceleration at current time, t, given by

$$\ddot{\mathbf{u}}^t = \frac{1}{\beta_s \Delta t^2}(\mathbf{u}^t - \mathbf{u}^{t-1}) - \frac{1}{\beta_s \Delta t}\dot{\mathbf{u}}^{t-1} - \frac{1 - 2\beta_s}{2\beta_s}\ddot{\mathbf{u}}^{t-1} \tag{2.38}$$

which is based on the expressions for displacement and velocity given by

$$\begin{aligned} \mathbf{u}^t &= \mathbf{u}^{t-1} + \dot{\mathbf{u}}^{t-1}\Delta t + [(\frac{1}{2} - \beta_s)\ddot{\mathbf{u}}^{t-1} + \beta_s \ddot{\mathbf{u}}]^t \Delta t^2 \\ \dot{\mathbf{u}}^t &= \dot{\mathbf{u}}^{t-1} + [(1-\gamma)\ddot{\mathbf{u}}^{t-1} + \gamma \ddot{\mathbf{u}}^t]\Delta t, \end{aligned} \tag{2.39}$$

where constants γ and β_s can be chosen for numerical stability or to introduce numerical damping as given by

$$\gamma = \frac{1}{2} + \delta \ , \ \beta_s = \frac{1}{4}\left(\frac{1}{2} + \gamma\right)^2. \tag{2.40}$$

2.3 Coupled Fluid-Structure Interaction

The fluid-structure interaction problem is coupled by enforcing continuity of displacements and velocities over the common interface and the equilibrium conditions for the fluid and structural tractions. For the continuity of displacement in steady coupled FSI, the structural displacements at the fluid-structure interface nodes were transferred to the interface node of the fluid which were then propagated to the rest of the fluid domain using a pseudo-linear-elastic approach [47] as given by

$$\mathbf{M}(\mathbf{u}, \mathbf{z}) = \tilde{\mathbf{K}}_{\mathbf{M}}\mathbf{z} - \tilde{\mathbf{f}}(\mathbf{u}), \tag{2.41}$$

where $\tilde{\mathbf{K}}_{\mathbf{M}}$ is the fictitious stiffness matrix of the mesh and $\tilde{\mathbf{f}}$ is the fictitious force to enforce matching boundary displacements.

For dynamic problems, the continuity of velocities was then implicitly enforced when the grid velocities on the fluid domain were evaluated by finite differences. However, this introduces time dependency on previous meshes further complicating the desired discrete adjoint method. Therefore, the continuity of velocities are also explicitly enforced at the interface, at every physical time step, similarly to the displacement in which the fictitious force now also enforces matching velocities at the boundary. This can be written as

$$\mathbf{y} = \begin{bmatrix} \mathbf{z} \\ \dot{\mathbf{z}} \end{bmatrix} = \mathcal{M}\left(\begin{bmatrix} \mathbf{u}_\Gamma \\ \dot{\mathbf{u}}_\Gamma \end{bmatrix}\right) = \begin{bmatrix} \tilde{\mathbf{K}}_{\mathbf{M}}^{-1}\tilde{\mathbf{f}}(\mathbf{u}_\Gamma) \\ \tilde{\mathbf{K}}_{\mathbf{M}}^{-1}\tilde{\mathbf{f}}(\dot{\mathbf{u}}_\Gamma) \end{bmatrix} \tag{2.42}$$

where the dependency on the left and right hand sides are explicit due to the linearity in the pseudo-linear elastic approach. The mesh solution for both displacement and velocity are found by setting a fictitious stiffness for each element in the fluid domain. This can be chosen based on an inverse volume approach for each element or based on the inverse distance to the closest wall for each element. The stiffness matrix based on the constitutive matrix alone is used for this linear solver approximation. The boundary condition of clamped boundary at all non-moving walls are enforced followed by the displacement, $\mathbf{u}_\Gamma$, and velocity , $\dot{\mathbf{u}}_\Gamma$, boundary condition in

turn for the deforming walls. The solution for the linear system then leads to the propagated solution for the mesh displacement and grid velocity for the fluid domain.

The equilibrium of tractions are given by

$$\boldsymbol{\lambda}_f + \boldsymbol{\lambda}_s = \mathbf{0}, \tag{2.43}$$

where the tractions on the fluid, $\boldsymbol{\lambda}_f$, and tractions on the structure, $\boldsymbol{\lambda}_s$, are defined on the surface boundaries, Γ, of the respective domains by

$$\boldsymbol{\lambda}_f = -p\mathbf{n}_f + \bar{\bar{\boldsymbol{\tau}}}\mathbf{n}_f \tag{2.44}$$

$$\boldsymbol{\lambda}_s = \boldsymbol{\sigma}_s\mathbf{n}_s. \tag{2.45}$$

As a result, the coupled interface can be described such that the displacement at the interface is a function of the fluid tractions, $\mathbf{u}_\Gamma = \mathbf{S}(\boldsymbol{\lambda})$ and similarly, the tractions at the interface are a function of the structural displacement at the interface $\boldsymbol{\lambda}_f = \mathbf{F}(\mathbf{u}_\Gamma)$. Together, this can be represented as a fixed-point iteration for the interface given by

$$\mathbf{u}_\Gamma = \mathbf{S}(\mathbf{F}(\mathbf{u}_\Gamma)). \tag{2.46}$$

Note that if non-matching meshes are used for the structural and fluid domain, as is usually the case from independent grid convergence studies, the displacements, velocities and tractions on the interface need to be interpolated. This can be done using a radial basis function approach to generate the required interpolation matrices[25].

The sequential application of the interface coupling conditions led to the choice of strongly coupled Block Gauss-Seidel [17] solution approach. The final governing equations for the dynamic

coupled fluid-structure interaction are thus given by

$$\mathbf{G} = \begin{cases} \mathbf{F} & = \dot{\mathbf{w}} + \nabla \cdot \tilde{\mathbf{F}}^{c}_{ALE} - \nabla \cdot \tilde{\mathbf{F}}^{v} - \tilde{\mathbf{Q}} & = \mathbf{0} \\ \mathbf{S} & = \mathbf{T} - \mathbf{F}_b - \mathbf{F}_\Gamma + \mathbf{M}_s \ddot{\mathbf{u}} & = \mathbf{0} \\ \mathbf{M} & = [\tilde{\mathbf{K}}_\mathbf{M}\mathbf{z} - \tilde{\mathbf{f}}(\mathbf{u}) \quad \tilde{\mathbf{K}}_\mathbf{M}\dot{\mathbf{z}} - \tilde{\mathbf{f}}(\dot{\mathbf{u}})]^\mathsf{T} & = \mathbf{0}. \end{cases} \tag{2.47}$$

In each BGS iteration, the governing equations for the fluid and structural domains are solved sequentially together with the mesh solution as per the data transferred at the interface and the convergence is assumed when the residual of displacements at the interface are within tolerance, $\mathscr{R}_\mathbf{u} = \|\mathbf{u}_\Gamma^{n+1} - \mathbf{u}_\Gamma^{n}\| < \epsilon$. This is repeated for each physical time in order to obtain time-accurate results.

To prevent the appearance of numerical divergence issues, Aitken relaxation parameters are incorporated to the formulation [21, 22] in order to scale the structural solution during the sub-iterations using either a fixed or dynamic parameter. In addition to this, an incremental loading can also be used which transfers the loads to the structural domain over a number of incremental sub-iterations as required for each BGS iteration to prevent overshoots in the solution process that can arise due to large variations in the external and internal forces balance. As an alternative to the relaxation strategies, a quasi-Newton IQN-ILS method [18] has been implemented by Gomes in SU2 [25] which has been shown to increase the robustness of the coupled solver without relaxation as well as improve convergence of the coupled solution.

2.4 Software Architecture

The code structure in SU2 for single domain and coupled problems will be briefly discussed in this section along with contributions from this work. Further information can be found in [23] for the original implementation of SU2, while [22] discusses the changes introduced for the initial implementation of FSI simulations. SU2 is an object-oriented, C++ code which facilitates the modular structure to handle problems of various types of physics which can also be coupled for multi-physics problems. Given the open-source environment which is continually

developed, the structure continues to evolve but the fundamentals of the current status relevant for the presented primal solutions are described in this section.

The main component is `SU2_CFD` which solves the primal single or multiphysics problems. An illustration of the code structure used for FSI problems is shown in Figure 2.1. The `CConfig` class is first instantiated to determine the type of physics to be solved in order to instantiate the appropriate driver class. The single or multi-zone driver then drives the simulation by instantiating `CConfig` again to determine and store all relevant user defined parameters. `CIteration` is instantiated for each type of physics problem to be solved such that the problem is fully conditioned and solved at each time-step. As part of this step, additional classes down the hierarchy are used for the spatial and time integration as governed by the solver and the domain properties held within the variables. `CInterface` class is instantiated by the driver for multi-physics problems and a further class called `CInterpolator` is instantiated if interpolation is needed at the interface for the multi-physics problem. The final solution is output by the driver through the instantiation of `COutput`.

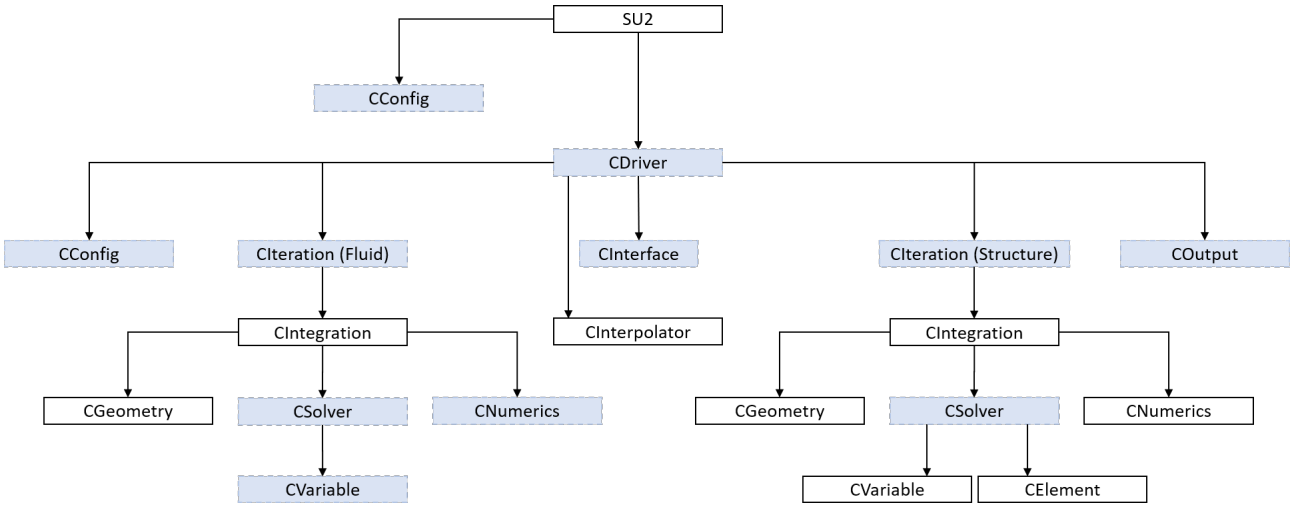


Figure 2.1: Description of classes in SU2_CFD for Fluid-Structure Interaction problems.

The description for the key classes in SU2 are provided below followed by the changes implemented for the primal solvers.

`CDriver` - Drives the calculation by instantiating the various sub-classes as required for each type of physics simulation

`CConfig` - Stores the user defined options from the config file

`COutput` - Compiles the required information and writes the output files

`CIteration` - Initiated for each physics zone to be simulated to manage the solution for one time-step

`CIntegration` - Solves the particular governing equations for the chosen physics using connections to subclasses `CGeometry`, `CSolver` and `CNumerics`

`CSolver` - Defines the solution procedure with physics specific solution through child classes

`CVariable` - Stores the computed solution in every node of the grid

`CElement` - Instantiated for structural problems to assemble the tangent matrix and impose boundary conditions

`CNumerics` - Performs the discretized numerical calculations for the governing equations with child classes to execute the user-prescribed methods

`CGeometry` - Reads the mesh and stores the computational grids

`CInterface` - Facilitates transfer of information from each physics zone to the other at the interface

`CInterpolator` - Defines interpolation routines to accommodate non-matching meshes at the interface

Other modules such as `SU2_DOT` (projection of shape gradients from grid nodes to chosen design parameters), `SU2_DEF` (deformation of meshes) and `SU2_GEO` (geometric evaluations), among others, lead to the full functionality of SU2.

Multizone mesh deformation using prescribed shape parameters has been achieved through consistent execution of the deformation for multiple zones in `SU2_DEF` as well as the addition of the deformed multizone mesh output in `COutput`.

Deforming, dynamic grid problems have been enabled using boundary displacements governed by prescribed pitching and plunging motions which were added to the mesh solution in `CMeshSolver`.For the implementation of the ALE formulation of the incompressible solver, grid velocity corrections were included in `CIncEulerSolver` and `CNumerics` for the convective flux calculation. Grid velocity corrected residuals are computed using modified upwind

FDS scheme (`CUpwFDSInc_Flow`) and centered JST (`CCentJSTInc_Flow`). A new child class, `CSourceIncRotatingFrame`, has also been created in `CNumerics` with `CDriver` updated for incompressible rotating frame problems.

The modified coupling of the fluid-structure interface to enforce velocity boundary conditions are implemented by adapting `CDisplacementInterface` under `CInterface` in order to include velocity for time-domain problems. Additional methods are introduced in `CMeshSolver` in order to propagate the velocity to the rest of the fluid domain together with the displacement.

Chapter 3

Coupled Adjoint Methods for Gradient Based Optimization

The progression from analysis to design typically requires efficient calculation of sensitivities, derivatives of the cost function with respect to the design parameters, for gradient-based optimization. This chapter initially discusses the different approaches for sensitivity analysis, including their main advantages and disadvantages. The sensitivity analysis of dynamic problems especially requires efficient methods due to the inherent additional cost for time-accurate simulations. As a result, the discrete adjoint method is the chosen approach to evaluate sensitivities for the individual domain problems as well as the coupled fluid-structure interactions presented in Chapter 2.

The discrete adjoint formulation for steady coupled problems is firstly presented, which introduces the template that will be extended for dynamic problems. As part of the solution process, algorithmic differentiation is used through an interface with CoDiPack [84]. This provides a robust implementation, which can be easily extended for new physics and also crucially removes the need to evaluate the analytical Jacobian of the governing equations.

The governing equations for each domain are considered and the dependencies are extended for time-accurate problems to accurately capture the coupled, dynamic characteristics of the system using the dynamic discrete adjoints. Implementation of the unsteady discrete adjoint

sensitivities in the SU2 framework is presented using the extension of the steady, generic multizone discrete adjoint driver [87]. The approach is also suitable for unsteady sensitivities for other coupled problems including conjugate heat transfer of low speed flows.

In addition to the discussion of the optimal design using the discrete adjoint sensitivities for both steady and unsteady problems that will be presented, the efficient calculation of the sensitivities for the large number of design variables have classically also been applied to optimal control of dynamic systems. Though optimal control problems are beyond the scope of this research, the methodology presented can be readily adopted by introducing the design variables as a function of time in order to consider active control of dynamic coupled FSI problems [54]. Together with the existing optimal design framework for optimization of structural or shape design, this would lead the pathway towards co-design of aeroservoelastic systems in a high fidelity environment.

3.1 Sensitivity Analysis

To briefly describe the common methods available in the literature for gradient calculation, consider a scalar function, $f(\xi, \eta(\xi))$, with state variable, η, and design variable, ξ. The derivative for this function can be defined by

$$\frac{\mathrm{d}f}{\mathrm{d}\xi} = \lim_{\Delta\xi \to 0} \frac{f(\xi + \Delta\xi) - f(\xi)}{\Delta\xi}. \tag{3.1}$$

For simple functions, the analytic gradient can be found explicitly, however for most physical problems alternative methods are needed such as finite differences, complex-step, direct or adjoint methods.

Finite Differences

Finite differences are widely used due to its simplicity and flexibility which allows this method to be used for a wide range of applications. This method is derived from the definition of the derivative and introduces a small perturbation, $\Delta\xi$, for each design variable which can be to

achieve forward, backward or central differences, respectively

$$\frac{\mathrm{d}f}{\mathrm{d}\xi} = \frac{f(\xi + \Delta\xi) - f(\xi)}{\Delta\xi} + \mathcal{O}(\Delta\xi) \tag{3.2a}$$

$$\frac{\mathrm{d}f}{\mathrm{d}\xi} = \frac{f(\xi) - f(\xi - \Delta\xi)}{\Delta\xi} + \mathcal{O}(\Delta\xi) \tag{3.2b}$$

$$\frac{\mathrm{d}f}{\mathrm{d}\xi} = \frac{f(\xi + \Delta\xi) - f(\xi - \Delta\xi)}{2\Delta\xi} + \mathcal{O}[(\Delta\xi)^2] \tag{3.2c}$$

depending on the needs of the problem. In these equations, the forward and backward differences are first order schemes in which the computational cost for the calculation of the gradient scales with the number of design variables of the problem, n. The central differences (3.2c) is a second-order scheme which scales with $2n$. All three methods have cancellation errors if $\Delta\xi$ is small and truncation errors when $\Delta\xi$ is not small enough, therefore even for the optimum value of $\Delta\xi$, there exists inherent inaccuracy in the result [70]. The order of accuracy of the derivative can be increased by evaluating additional terms in the Taylor series expansion, however these add to the cost of the calculation per design variable.

Complex-Step Method

Complex-step is similar to finite differences with the main difference being that the perturbation applied is in the complex domain [60].

$$\frac{\mathrm{d}f}{\mathrm{d}\xi} = \frac{\mathrm{Im}[f(\xi + i\Delta\xi)]}{\Delta\xi} + \mathcal{O}(\Delta\xi^2) \tag{3.3}$$

This method removes the cancellation error previously present in the finite differences, therefore is more accurate and independent of the chosen perturbation step but it still has truncation errors. However, if very small values of the perturbation are chosen, these truncation errors can be minimised. The computational cost for the gradient is the same as finite differences. Given the greater accuracy this would appear to be a more preferable choice, however, the implementation of this method is more challenging as the governing physical problem needs to be modified to a complex problem.

Direct and Adjoint methods

For an objective function, f, defined by

$$f = f(\xi, \eta(\xi)) \tag{3.4}$$

where ξ represents the design variables and η represents the state variables, the total sensitivity of the objective function to the design variables can be written using of the chain rule as

$$\frac{\mathrm{d}f}{\mathrm{d}\xi} = \frac{\partial f}{\partial \xi} + \frac{\partial f}{\partial \eta}\frac{\mathrm{d}\eta}{\mathrm{d}\xi}. \tag{3.5}$$

The partial derivatives in (3.5) can be calculated more easily than the total derivative [67], since the partial derivative evaluation varies only the denominator whilst keeping everything else constant during the reevaluation of the function in the numerator. This is particularly relevant for multidisciplinary problems since the total derivative will require the solution for the whole multidisciplinary problem.

In order to find the best way of evaluating the total derivative of the state variables with respect to the design variables, the governing equations for the system of interest can be written in residual form with respect to these two variables as

$$R(\xi, \eta(\xi)) = 0. \tag{3.6}$$

The governing equations must be satisfied for any choice of design variables, which leads to the total derivative of the residuals with respect to the design variables being zero, that is,

$$\frac{\mathrm{d}R}{\mathrm{d}\xi} = \frac{\partial R}{\partial \xi} + \frac{\partial R}{\partial \eta}\frac{\mathrm{d}\eta}{\mathrm{d}\xi} = 0. \tag{3.7}$$

Using (3.7), the total derivative of the state variables with respect to the design variables can be found as

$$\frac{\mathrm{d}\eta}{\mathrm{d}\xi} = -\left[\frac{\partial R}{\partial \eta}\right]^{-1}\frac{\partial R}{\partial \xi}. \tag{3.8}$$

Therefore, by substituting (3.8) in (3.5), the total derivative of the objective function can be found as

$$\frac{\mathrm{d}f}{\mathrm{d}\xi} = \frac{\partial f}{\partial \xi} - \underbrace{\frac{\partial f}{\partial \eta}}_{\overline{\eta}^{\mathsf{T}}} \overbrace{\left[\frac{\partial R}{\partial \eta}\right]^{-1}}^{d\eta/d\xi} \frac{\partial R}{\partial \xi}. \tag{3.9}$$

This is the direct method. The cost of the gradient calculation scales with the number of design variables using this method because in (3.8) there is one linear system to be solved per design variable.

Alternatively, from (3.9) an adjoint variable, $\overline{\eta}$, can be defined instead which is reiterated as

$$\overline{\eta}^{\mathsf{T}} = -\frac{\partial f}{\partial \eta}\left[\frac{\partial R}{\partial \eta}\right]^{-1}. \tag{3.10}$$

The adjoint variable by contrast to the direct method has a dependency on the number of objective functions and thus the cost is independent to the number of design variables.

Therefore the best choice from the two methods depends on the application of interest. For a large number of objective functions and smaller number of design variables, the direct method is the computationally efficient method. Conversely, the adjoint method is a highly efficient technique to calculate the sensitivity to a fewer number of objective functions but with respect to an arbitrary number of design variables. Both methods however require the exact calculation of the Jacobian matrix, $\partial R/\partial \eta$, in the form of the residual with respect to the state variables, which is non-trivial and in some cases might not be explicitly available.

Algorithmic differentiation (AD) is a programmatic method to evaluate sensitivities by sequentially applying the chain rule for each mathematical operation in the solution process [70]. An iterative approach exists for the adjoint system similar to the primal solver that allows the use of the same approximate Jacobian [113]. This links the convergence behaviour of the two problems such that under convergence, if the primal solution is contractive, then the adjoint solution will also be contractive. In order to apply the direct and adjoint methods to the calculation of gradients in a physical problem, the calculation can be recast using AD.

Similar to the direct and adjoint methods, AD can be performed in two modes, namely forward mode and reverse mode. For the forward mode, the information is propagated through the code such that the operations and derivatives for all outputs are calculated with respect to one input. This method is similar to the complex-step method as illustrated by [69] with the fundamental difference being that the complex-step has perturbations in the complex domain whereas in the forward mode the perturbation is the gradient. As a consequence, the methodology has to be run for each input variable to calculate the complete Jacobian, $\partial f/\partial \eta$. The reverse mode propagates the adjoint variable such that the calculation has to be run for each objective function. Therefore, similar to the direct and adjoint methods, the associated cost for the forward and reverse mode of AD scales with number of design variables or objective functions respectively.

The primal solution path from input to the final calculation of the output is broken down into a sequence of elementary mathematical operations and recorded. AD recursively applies the chain rule for each of these operations, in either a forward mode which is the same direction as the primal solution or in the reverse mode. Since many physical applications have a smaller number of output variables of interest compared to the input variables, this method is often more computationally efficient [43], [70]. The open-source C++ library Code-Differentiation Package (CoDiPack) [84] has been interfaced with SU2 for algorithmic differentiation. The Jacobians are thus computed iteratively through linearization of the fixed-point operators that define the governing system. Prior to this work, this has been successfully applied for discrete adjoints via AD for fluid problems [86, 46] and steady fluid-structure interaction problems [47].

Local preaccumulation methods are applied in the implementation where possible which computes and stores the local derivatives of enclosed functions of code instead of storing the contributions from each individual statement within that function. This provides a significant reduction in the memory requirements, particularly for functions which are executed for every element in the mesh. The problem size, in time-domain, is also affected by the necessity to store the full solution path. The full solution vector of the primal solver is stored at each timestep and loaded when necessary during the adjoint evaluation. This can become very expensive particularly for time-domain problems involving long series as the path for each time-step will need

to be recorded. Alternative approaches being developed which could be applied in the future include checkpointing techniques [88, 89] where the data is stored only for certain checkpoint states. During the adjoint sweep, the necessary states for each adjoint iteration are recomputed starting from the nearest checkpoint.

3.2 Discrete Adjoint Method for Steady Fluid-Structure Interactions

For completeness, the formulation for the steady coupled discrete adjoints are briefly presented below, following [47]. For a given objective function, J, the baseline optimization problem can be defined as

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & J(\mathbf{w}, \mathbf{u}, \mathbf{z}, \boldsymbol{\alpha}) \\ \text{subject to} \quad & \mathcal{F}(\mathbf{w}, \mathbf{z}, \boldsymbol{\alpha}) - \mathbf{w} = \mathbf{0}, \\ & \mathcal{S}(\mathbf{w}, \mathbf{u}, \mathbf{z}, \boldsymbol{\alpha}) - \mathbf{u} = \mathbf{0}, \\ & \mathcal{M}(\mathbf{u}, \boldsymbol{\alpha}) - \mathbf{z} = \mathbf{0}. \end{aligned} \tag{3.11}$$

where $\boldsymbol{\alpha}$ is the vector of design variables and the constraint is the satisfaction of the governing equations for the fluid, structural and mesh solutions given by $\mathcal{F}$, $\mathcal{S}$ and $\mathcal{M}$ respectively. The governing equations from Chapter 2 (2.47) are now written in the form of fixed-point operators in order to set up an iterative solution process for each zone and the mesh deformation. This approach lends itself to Algorithmic Differentiation and the linearization of the fixed-point iterators is performed using CoDiPack [84].

The Lagrangian of the coupled problem (3.11) can be written, with adjoint variables $\overline{\mathbf{w}}$, $\overline{\mathbf{u}}$ and $\overline{\mathbf{z}}$, as

$$\begin{aligned} L(\boldsymbol{\alpha}, \mathbf{w}, \mathbf{u}, \mathbf{z}, \overline{\mathbf{w}}, \overline{\mathbf{u}}, \overline{\mathbf{z}}) = J &+ \overline{\mathbf{w}}^{\mathsf{T}} \left[\mathcal{F}\left(\mathbf{w}, \mathbf{z}, \boldsymbol{\alpha}\right) - \mathbf{w}\right] \\ &+ \overline{\mathbf{u}}^{\mathsf{T}} \left[\mathcal{S}\left(\mathbf{w}, \mathbf{u}, \mathbf{z}, \boldsymbol{\alpha}\right) - \mathbf{u}\right] \\ &+ \overline{\mathbf{z}}^{\mathsf{T}} \left[\mathcal{M}\left(\mathbf{u}, \boldsymbol{\alpha}\right) - \mathbf{z}\right]. \end{aligned} \tag{3.12}$$

Differentiating with respect to the design variables and rearranging leads to

$$\begin{aligned}\frac{\mathrm{d}L}{\mathrm{d}\boldsymbol{\alpha}} = \frac{\mathrm{d}J}{\mathrm{d}\boldsymbol{\alpha}} =& \frac{\partial J}{\partial \boldsymbol{\alpha}} + \overline{\mathbf{w}}^{\mathsf{T}}\frac{\partial \mathcal{F}}{\partial \boldsymbol{\alpha}} + \overline{\mathbf{u}}^{\mathsf{T}}\frac{\partial \mathcal{S}}{\partial \boldsymbol{\alpha}} + \overline{\mathbf{z}}^{\mathsf{T}}\frac{\partial \mathcal{M}}{\partial \boldsymbol{\alpha}} \\ &+ \left[\frac{\partial J}{\partial \mathbf{w}} + \overline{\mathbf{w}}^{\mathsf{T}}\frac{\partial \mathcal{F}}{\partial \mathbf{w}} - \overline{\mathbf{w}}^{\mathsf{T}} + \overline{\mathbf{u}}^{\mathsf{T}}\frac{\partial \mathcal{S}}{\partial \mathbf{w}}\right]\frac{\mathrm{d}\mathbf{w}}{\mathrm{d}\boldsymbol{\alpha}} \\ &+ \left[\frac{\partial J}{\partial \mathbf{u}} + \overline{\mathbf{u}}^{\mathsf{T}}\frac{\partial \mathcal{S}}{\partial \mathbf{u}} - \overline{\mathbf{u}}^{\mathsf{T}} + \overline{\mathbf{z}}^{\mathsf{T}}\frac{\partial \mathcal{M}}{\partial \mathbf{u}}\right]\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}\boldsymbol{\alpha}} \\ &+ \left[\frac{\partial J}{\partial \mathbf{z}} + \overline{\mathbf{w}}^{\mathsf{T}}\frac{\partial \mathcal{F}}{\partial \mathbf{z}} + \overline{\mathbf{u}}^{\mathsf{T}}\frac{\partial \mathcal{S}}{\partial \mathbf{z}} - \overline{\mathbf{z}}^{\mathsf{T}}\right]\frac{\mathrm{d}\mathbf{z}}{\mathrm{d}\boldsymbol{\alpha}}. \end{aligned} \tag{3.13}$$

Defining the adjoint variables such that the terms in brackets vanishes, results in a simpler expression for the gradient of J and a set of equations for the adjoint variables. Notations are simplified such that the subscript on the operator indicates the variable with respect to which the derivative is taken. The gradient is thus defined by

$$\mathrm{d}_{\boldsymbol{\alpha}} J = J_{\boldsymbol{\alpha}} + \overline{\mathbf{w}}^{\mathsf{T}}\mathcal{F}_{\boldsymbol{\alpha}} + \overline{\mathbf{u}}^{\mathsf{T}}\mathcal{S}_{\boldsymbol{\alpha}} + \overline{\mathbf{z}}^{\mathsf{T}}\mathcal{M}_{\boldsymbol{\alpha}} \tag{3.14}$$

with the adjoint equations in matrix form highlighting the pure field derivatives (diagonal) and cross-term derivatives (off-diagonal) shown by

$$\begin{pmatrix}\overline{\mathbf{w}}\\ \overline{\mathbf{u}}\\ \overline{\mathbf{z}}\end{pmatrix} = \begin{pmatrix}J_{\mathbf{w}}^T\\ J_{\mathbf{u}}^T\\ J_{\mathbf{z}}^T\end{pmatrix} + \begin{bmatrix}\mathcal{F}_{\mathbf{w}} & \mathbf{0} & \mathcal{F}_{\mathbf{z}}\\ \mathcal{S}_{\mathbf{w}} & \mathcal{S}_{\mathbf{u}} & \mathcal{S}_{\mathbf{z}}\\ \mathbf{0} & \mathcal{M}_{\mathbf{u}} & \mathbf{0}\end{bmatrix}^T \begin{pmatrix}\overline{\mathbf{w}}\\ \overline{\mathbf{u}}\\ \overline{\mathbf{z}}\end{pmatrix}. \tag{3.15}$$

This coupled discrete adjoint method was initially verified with structural objectives and structural design variables in [47], and has since been extended to aerodynamic problems with shape design for this work [91]. For optimal design, the objective function can be based on structural or aerodynamic domains with typical examples such as structural displacement or aerodynamic efficiency. Typically, the optimal distribution of material properties, shape parameters or both act as suitable design variables with respect to which the sensitivities can be evaluated and updated in an optimization process.

For shape design, geometric sensitivities need to be evaluated on each zone with respect to

an objective function from either the fluid or structural zones. The sensitivities of the mesh nodes could be used directly to gain an insight into the geometric sensitivities of the problem, however this is unlikely to perform well to provide smooth, continuous profiles if directly used for shape design. Shape parameterization is achieved typically with either Hicks-Henne bumps in 2D [114] or a free-form deformation (FFD) approach with Bezier solid in 2D or 3D [115]. This parameterizes the relationship between the shape variables and the surface shape that is analyzed. This process also requires a method for mesh deformation [116] to transfer the deformation of the design surface to the interior nodes. The geometric sensitivities on the design surfaces are projected to the shape parameters by differentiating the parameterization step of the chosen form of shape variable. The sum of the shape parameter sensitivities from the two zones leads to the final sensitivity of the coupled problem.

3.3 Discrete Adjoint Method for Time-Accurate Fluid-Structure Interactions

A similar approach to that of the previous section can be derived for time-accurate coupled adjoint problem through the use of dynamic governing equations. For the fluid domain, the primal solver employs a second-order backwards dual-time stepping strategy, therefore the fluid solution, $\mathbf{w}^n$ at a given time step, n, has dependency on $\mathbf{w}^n$, $\mathbf{w}^{n-1}$ and $\mathbf{w}^{n-2}$.

For a steady problem, the fluid solution has dependency only on the position of the mesh nodes. However, for a dynamic problem the mesh velocity also influences the fluid solution due to the grid velocity throughout the domain caused by the rate of change of the structural deformation in the coupled domain. Previously, finite differences approach has been suggested for the velocity calculation [117], which again requires second order backward approximation thus introducing dependency on $\mathbf{z}^n$, $\mathbf{z}^{n-1}$ and $\mathbf{z}^{n-2}$. However, this can be simplified by including the grid velocity at the interface as suggested in the primal problem (2.42). The mesh variable thus contains both the grid position and grid velocity which is reiterated as $\mathbf{y} = [\mathbf{z}, \dot{\mathbf{z}}]$. Therefore for the mesh dependency, fluid problem at time n, now depends only on $\mathbf{y}^n$. The dependencies

which need to be accounted for in the dynamic fluid solution can thus be written for the fixed-point operator of the fluid domain by

$$\mathcal{F}(\mathbf{w}^n, \mathbf{w}^{n-1}, \mathbf{w}^{n-2}, \mathbf{y}^n, \boldsymbol{\alpha}) - \mathbf{w}^n = \mathbf{0}, \quad n = 1, ..., N. \tag{3.16}$$

For the extension of the structural domain to dynamic problems, the use of Newmark time integration introduces dependency on the current and previous time step solutions. However, this includes not only the displacement, but also the velocity and acceleration of the structural problem. Therefore, for the dynamic problem a new structural state variable is chosen to be $\mathbf{q}^\mathsf{T} = [\mathbf{u}^\mathsf{T}, \dot{\mathbf{u}}^\mathsf{T}, \ddot{\mathbf{u}}^\mathsf{T}]$ which now includes velocity and acceleration and the time dependency for the solution at time step n, can thus be given by $\mathbf{q}^n$ and $\mathbf{q}^{n-1}$. At the fluid-structure interface, the equilibrium of tractions for the current time step is transferred to the structure which brings in the dependency on the fluid solution and interface nodes at the current time step. Fluid tractions arising from the time history of fluid loads on the structure are not explored here. Therefore, the fixed-point operator for the structural domain can then be written by

$$\mathcal{S}(\mathbf{w}^n, \mathbf{q}^n, \mathbf{q}^{n-1}, \mathbf{y}^n, \boldsymbol{\alpha}) - \mathbf{q}^n = \mathbf{0}, \quad n = 1, ..., N \tag{3.17}$$

Note that, to simplify the algebra, we use the same symbol for the fluid and structure operators $\mathcal{F}$, $\mathcal{S}$ for steady and time-domain problems. The mesh operator transfers the displacement and grid velocity only at the current time step, therefore this can simply be written by

$$\mathcal{M}(\mathbf{q}^n, \boldsymbol{\alpha}) - \mathbf{y}^n = \mathbf{0}, \quad n = 1, ..., N \tag{3.18}$$

Combining the three fixed point iterations for the governing equations, an unsteady coupled

optimization problem can be written as

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & \sum_{n=1}^{N} J^n(\mathbf{q}^n, \mathbf{w}^n, \mathbf{y}^n, \boldsymbol{\alpha}) \\ \text{subject to} \quad & \mathcal{F}(\mathbf{w}^n, \mathbf{w}^{n-1}, \mathbf{w}^{n-2}, \mathbf{y}^n, \boldsymbol{\alpha}) - \mathbf{w}^n = \mathbf{0}, \qquad n = 1, ..., N, \\ & \mathcal{S}(\mathbf{w}^n, \mathbf{q}^n, \mathbf{q}^{n-1}, \mathbf{y}^n, \boldsymbol{\alpha}) - \mathbf{q}^n = \mathbf{0}, \qquad n = 1, ..., N, \\ & \mathcal{M}(\mathbf{q}^n, \boldsymbol{\alpha}) - \mathbf{y}^n = \mathbf{0}, \qquad n = 1, ..., N, \end{aligned} \tag{3.19}$$

where the objective function is averaged over all the time steps and the satisfaction of the governing equations at all time steps act as the constraint. The unsteady problem is started from initial conditions at $n = 1$, where the previous time solutions, $n-1$ and $n-2$, are assumed to be as per the initial conditions defined for the problem. As the problem is marched forward in time, solutions which have been found for the previous time steps then occupy the previous time solutions when evaluating the solution at the current time. The Lagrangian of the coupled problem with adjoint variables $\overline{\mathbf{w}}$, $\overline{\mathbf{q}}$ and $\overline{\mathbf{y}}$ is given by

$$\begin{aligned} L = \sum_{n=1}^{N} J^n(\mathbf{w}^n, \mathbf{q}^n, \mathbf{y}^n, \boldsymbol{\alpha}) &+ \sum_{n=1}^{N} [\overline{\mathbf{w}}^n]^\mathsf{T} \left[\mathcal{F}\left(\mathbf{w}^n, \mathbf{w}^{n-1}, \mathbf{w}^{n-2}, \mathbf{y}^n, \boldsymbol{\alpha}\right) - \mathbf{w}^n\right] \\ &+ \sum_{n=1}^{N} [\overline{\mathbf{q}}^n]^\mathsf{T} \left[\mathcal{S}\left(\mathbf{w}^n, \mathbf{q}^n, \mathbf{q}^{n-1}, \mathbf{y}^n, \boldsymbol{\alpha}\right) - \mathbf{q}^n\right] \\ &+ \sum_{n=1}^{N} [\overline{\mathbf{y}}^n]^\mathsf{T} \left[\mathcal{M}\left(\mathbf{q}^n, \boldsymbol{\alpha}\right) - \mathbf{y}^n\right] \end{aligned} \tag{3.20}$$

Differentiating with respect to the parameters and grouping the derivatives for each state variable with respect to the design variables leads to

$$\begin{aligned} \mathrm{d}_{\boldsymbol{\alpha}} L = \mathrm{d}_{\boldsymbol{\alpha}} J = &\sum_{n=1}^{N} \left[J^n_{\boldsymbol{\alpha}} + [\overline{\mathbf{w}}^n]^\mathsf{T} \mathcal{F}^n_{\boldsymbol{\alpha}} + [\overline{\mathbf{q}}^n]^\mathsf{T} \mathcal{S}^n_{\boldsymbol{\alpha}} + [\overline{\mathbf{y}}^n]^\mathsf{T} \mathcal{M}^n_{\boldsymbol{\alpha}} \right] \\ &+ \sum_{n=1}^{N} \left[J^n_{\mathbf{w}^n} + [\overline{\mathbf{w}}^n]^\mathsf{T} \mathcal{F}^n_{\mathbf{w}^n} + \left[\overline{\mathbf{w}}^{n+1}\right]^\mathsf{T} \mathcal{F}^{n+1}_{\mathbf{w}^n} + \left[\overline{\mathbf{w}}^{n+2}\right]^\mathsf{T} \mathcal{F}^{n+2}_{\mathbf{w}^n} - [\overline{\mathbf{w}}^n]^\mathsf{T} + [\overline{\mathbf{q}}^n]^\mathsf{T} \mathcal{S}^n_{\mathbf{w}^n} \right] \mathrm{d}_{\boldsymbol{\alpha}} \mathbf{w}^n \\ &+ \sum_{n=1}^{N} \left[J^n_{\mathbf{q}^n} + [\overline{\mathbf{q}}^n]^\mathsf{T} \mathcal{S}^n_{\mathbf{q}^n} + \left[\overline{\mathbf{q}}^{n+1}\right]^\mathsf{T} \mathcal{S}^{n+1}_{\mathbf{q}^n} - [\overline{\mathbf{q}}^n]^\mathsf{T} + [\overline{\mathbf{y}}^n]^\mathsf{T} \mathcal{M}^n_{\mathbf{q}^n} \right] \mathrm{d}_{\boldsymbol{\alpha}} \mathbf{q}^n \\ &+ \sum_{n=1}^{N} \left[J^n_{\mathbf{y}^n} + [\overline{\mathbf{w}}^n]^\mathsf{T} \mathcal{F}^n_{\mathbf{y}^n} + [\overline{\mathbf{q}}^n]^\mathsf{T} \mathcal{S}^n_{\mathbf{y}^n} - [\overline{\mathbf{y}}^n]^\mathsf{T} \right] \mathrm{d}_{\boldsymbol{\alpha}} \mathbf{y}^n. \end{aligned} \tag{3.21}$$

Defining the adjoint variables such that the terms in brackets vanishes, results in a simpler expression for the gradient of J and a set of equations for the adjoint variables. The gradient is thus defined by

$$\mathrm{d}_{\boldsymbol{\alpha}} J = \sum_{n=1}^{N} \left[J^n_{\boldsymbol{\alpha}} + \left[\overline{\mathbf{w}}^n\right]^{\mathsf{T}} \mathcal{F}^n_{\boldsymbol{\alpha}} + \left[\overline{\mathbf{q}}^n\right]^{\mathsf{T}} \mathcal{S}^n_{\boldsymbol{\alpha}} + \left[\overline{\mathbf{y}}^n\right]^{\mathsf{T}} \mathcal{M}^n_{\boldsymbol{\alpha}} \right] \tag{3.22}$$

with the adjoint equations

$$\begin{aligned}
\left[\overline{\mathbf{w}}^n\right]^{\mathsf{T}} &= J^n_{\mathbf{w}^n} + \left[\overline{\mathbf{w}}^n\right]^{\mathsf{T}} \mathcal{F}^n_{\mathbf{w}^n} + \left[\overline{\mathbf{w}}^{n+1}\right]^{\mathsf{T}} \mathcal{F}^{n+1}_{\mathbf{w}^n} + \left[\overline{\mathbf{w}}^{n+2}\right]^{\mathsf{T}} \mathcal{F}^{n+2}_{\mathbf{w}^n} + \left[\overline{\mathbf{q}}^n\right]^{\mathsf{T}} \mathcal{S}^n_{\mathbf{w}^n}, \qquad & n = N, ..., 1, \\
\left[\overline{\mathbf{q}}^n\right]^{\mathsf{T}} &= J^n_{\mathbf{q}^n} + \left[\overline{\mathbf{q}}^n\right]^{\mathsf{T}} \mathcal{S}^n_{\mathbf{q}^n} + \left[\overline{\mathbf{q}}^{n+1}\right]^{\mathsf{T}} \mathcal{S}^{n+1}_{\mathbf{q}^n} + \left[\overline{\mathbf{y}}^n\right]^{\mathsf{T}} \mathcal{M}^n_{\mathbf{q}^n}, \qquad & n = N, ..., 1, \\
\left[\overline{\mathbf{y}}^n\right]^{\mathsf{T}} &= J^n_{\mathbf{y}^n} + \left[\overline{\mathbf{w}}^n\right]^{\mathsf{T}} \mathcal{F}^n_{\mathbf{y}^n} + \left[\overline{\mathbf{q}}^n\right]^{\mathsf{T}} \mathcal{S}^n_{\mathbf{y}^n}, \qquad & n = N, ..., 1.
\end{aligned} \tag{3.23}$$

As the adjoint problem is marched in time beginning from $n = N$, initially contributions from $n + 1$ and $n + 2$ for the adjoint variables do not exist. As the adjoint problem is marched in reverse from $n = N, ..., 1$, the adjoint solutions evaluated at the appropriate time steps, $n + 1$ and $n + 2$, are stored and passed as dual-time derivative contributions for time step n.

The time-accurate discrete adjoint can still be solved iteratively similar to the primal problem. Each adjoint variable is solved while holding the rest constant and then iterated until the coupled solution for each time step is found which is then marched in time. Crucially, for the adjoint method, this march in time is performed in reverse such that the adjoint problem begins with the end of the primal solution which in then marched in reverse time until the beginning.

The dynamic discrete adjoint equations from (3.23) can also be written in matrix form for current time step, n, as

$$\begin{pmatrix} \overline{\mathbf{w}}^n \\ \overline{\mathbf{q}}^n \\ \overline{\mathbf{y}}^n \end{pmatrix} = \begin{pmatrix} \left[J^n_{\mathbf{w}^n}\right]^{\mathsf{T}} \\ \left[J^n_{\mathbf{q}^n}\right]^{\mathsf{T}} \\ \left[J^n_{\mathbf{y}^n}\right]^{\mathsf{T}} \end{pmatrix} + \begin{bmatrix} \mathcal{F}^n_{\mathbf{w}^n} & \mathbf{0} & \mathcal{F}^n_{\mathbf{y}^n} \\ \mathcal{S}^n_{\mathbf{w}^n} & \mathcal{S}^n_{\mathbf{q}^n} & \mathcal{S}^n_{\mathbf{y}^n} \\ \mathbf{0} & \mathcal{M}^n_{\mathbf{q}^n} & \mathbf{0} \end{bmatrix}^{\mathsf{T}} \begin{pmatrix} \overline{\mathbf{w}}^n \\ \overline{\mathbf{q}}^n \\ \overline{\mathbf{y}}^n \end{pmatrix} + \begin{bmatrix} \mathcal{F}^{n+1}_{\mathbf{w}^n} & \mathbf{0} & \mathbf{0} \\ \mathcal{F}^{n+2}_{\mathbf{w}^n} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathcal{S}^{n+1}_{\mathbf{q}^n} & \mathbf{0} \end{bmatrix}^{\mathsf{T}} \begin{pmatrix} \overline{\mathbf{w}}^{n+1} \\ \overline{\mathbf{w}}^{n+2} \\ \overline{\mathbf{q}}^{n+1} \end{pmatrix}, \tag{3.24}$$

which is in a similar form to the steady equations but now includes the time source terms given

by the addition of the final matrix of Jacobian transposed products with the time dependent adjoint variables.

Windowed weighting techniques can be adopted for dynamic optimization. Time-averaged objective functions have uniform weighting defined in J^n for all time steps in the design period. Alternative high-order windowing approaches such as Hann-square, in which the period boundaries are weighted lower than the middle of the periods, have been shown to improve convergence characteristics of optimization designs [118]. Given the computational cost associated with dynamic coupled simulations, reducing the number of design iterations in an optimal design process will be of significant benefit.

3.3.1 Implementation

The dynamic discrete adjoints for the coupled fluid-structure interaction problems presented above are implemented in SU2 using the generic multizone discrete adjoint driver which was originally introduced for steady problems [87]. In order to highlight the modifications made for the extension to dynamic adjoints, the adjoint solution for a coupled FSI problem at each time step is given by $\overline{\mathbf{x}}^n = \overline{\mathbf{w}}^n, \overline{\mathbf{w}}^{n+1}, \overline{\mathbf{w}}^{n+2}, \overline{\mathbf{q}}^n, \overline{\mathbf{q}}^{n+1}, \overline{\mathbf{y}}^n$.

The solution path suitable for dynamic discrete adjoints of multizone problems, which now loops for all the time steps of the dynamic problem in reverse, is summarized below including the modifications from the steady implementation in red.

Before expanding on the functions given by the algorithm above, firstly it needs to be emphasized that this solution approach is meant to be generic and suitable for any multiphysics problems. Therefore, the method needs to be independent of the type of zones, number of zones and the order of zones. In order to facilitate this, the adjoints for the coupled problems are evaluated in multiple stages. Using the matrix formulation given in (3.24), the contributions from the objective, the diagonals in the transposed Jacobian matrix product (pure field derivatives) and the off-diagonals of this matrix (coupled cross-term derivatives) are evaluated separately. Therefore, in addition to the adjoint solutions each zone also carries an `External`

Algorithm 1: Time-accurate multizone discrete adjoint driver.

```
Data: Files for problem config, mesh, primal solution of coupled problem
for n ← N to 0 do
    for j ← 1 to r do
        for i ← 0 to NSol do
            SetExternalZero();
        end
    end
    for j ← 1 to r do
        Preprocess();
    end
    ComputeAdjointsObjectiveFunction();
    for j ← 1 to r do
        AddSolutionToExternal(k, Iterate(k, bDiagonal_False))
    end
    for i ← 1 to OuterIter do
        for j ← 1 to r do
            for l ← 0 to InnerIter do
                x̄^n[j] = Sum(x̄^n[j], GetExternal(j));
                ComputeAdjoints(j, x̄^n[j]);
                x̄^n[j] = Iterate(j, bDiagonal_True);
            end
            for k ← 1 to r do
                if k ≠ j then
                    x̄^n[k] = Iterate(k, bDiagonal_False);
                    UpdateCrossTerm(j, k);
                end
            end
        end
    end
    for j ← 1 to r do
        UpdateCumulativeSensitivity();
    end
end
```

solution which handles the contributions from objective function and cross-term derivatives. The time dependent terms in the final matrix product are evaluated together with the diagonal, pure field derivatives. A brief description of the operation in Algorithm 1 is included next.

`SetExternalZero` - Resets the `External` solution containing the contributions from objective and cross-term derivatives to zero. This is applied for all solvers used by all zones at the beginning of each time step.

`Preprocess` - Existing method for steady adjoints modified for the needs of dynamic adjoints. Method to load the primal solution from each zone for the current time step and their time dependent solutions. For dynamic problems, the time source terms for the current time steps are also processed by the solvers at this stage.

`ComputeAdjointsObjectiveFunction` - Initializes the adjoint value of the objective function. This seeding can be varied for each time step. As part of this subroutine, the derivatives of the objective function with respect to the state variables are evaluated for all zones.

`ComputeAdjoints` - The adjoint values for the design variables of the current zone are initialized with the saved adjoint solutions for that zone and then evaluates the transposed Jacobian matrix product

`Iterate` - Extracts the current derivative solutions for the objective function or the transposed Jacobian matrix product for the zone currently iterated. A flag has been added to allow the time dependent solutions to be updated only for the diagonal terms of the transposed Jacobian matrix product (pure field derivatives). For example in the fluid domain, if $\overline{\mathbf{w}}^{n+1}$ and $\overline{\mathbf{w}}^{n+2}$ are updated during an iteration other than the diagonal term products, then incorrectly the time dependent solutions will be updated and the cross-term derivatives at the previous times will be used instead in the dual time derivative.

`GetExternal` - Retrieves the `External` solution. When called in the inner loop iterations for each zone, the contributions from the objective function and the cross-term derivatives are included in the summation for the current adjoint solution.

`AddSolutionToExternal` - Adds the current solution to the `External` container

`UpdateCrossTerm` - Once the cross-term adjoints are evaluated, the difference between the current cross-term derivatives and previous cross-term derivatives are updated to the `External` container

`UpdateCumulativeSensitivity` - Sensitivities are cumulatively added at each time step to provide the final sensitivities for each zone. This cumulative addition of sensitivities will lead to the time-averaged sensitivity for the objective function (3.22), as per the applied seeding in the initialization, when the sensitivities from each zone are finally added together.

One of the challenges for the implementation of this problem, is that the adjoint problem, as

mentioned previously, needs to be run in reverse which requires setting up all the details for the problem exactly from the last solution in the primal problem. For problem with dynamic mesh as is the case for FSI, the pre-processing requires not only all the state variables to be read exactly from the solutions but also the geometric mesh and mesh volumes for all the dependent time steps for each time step that is evaluated. In addition to this, the dynamic problem introduces additional variables to be recorded and evaluated, which needs to be well converged for accurate sensitivities.

The implementation for the dynamic multizone problem has also extended the indexed tape for the time dependent variables. The indexed tape approach was introduced [87] to remove the need for `ComputeAdjoints` to go through the whole recording of the primal solution for all solvers and for all evaluations to be performed in one. For cases where only part of the primal recorded path is relevant, `ComputeAdjoints` can evaluate the derivative only for this part of the tape making the computation more efficient which is particularly significant for dynamic problems. In order to accommodate this, indices are introduced when the variables are registered in the tape which now includes the time dependent variables $\mathbf{w}^{n-1}$, $\mathbf{w}^{n-2}$, $\dot{\mathbf{u}}^{n}$, $\dot{\mathbf{u}}^{n-1}$, $\ddot{\mathbf{u}}^{n}$ and $\ddot{\mathbf{u}}^{n-1}$. The indices are also saved during the evaluation in order to define specific regions such as the transfer for the interface coupling or the objective function derivative calculations such that the contributions from these to the adjoint variable solution can be called forth when needed.

This implementation is exercised in the results chapters to verify the building blocks and the final framework for dynamic FSI discrete adjoint sensitivities. Firstly the steady FSI sensitivities are verified in Chapter 4 for an aerodynamic objective function with stiffness design variable before moving to shape optimization. Chapter 5 then moves to time-accurate problems beginning with the verification of shape sensitivities for dynamic structural problem with application in optimal design. The unsteady fluid discrete adjoint formulation is then verified in Chapter 6 for the incompressible flow solver, including discrete adjoint sensitivities for problems with grid motion. Finally, Chapter 7 exercises the implementation for the complete dynamic FSI discrete adjoint sensitivities.

Chapter 4

Optimal Design of Coupled Problems in Equilibrium

The methodology proposed in Chapters 2 and 3 for the primal and adjoint solvers have been implemented in SU2 and the verification of the coupled discrete adjoints for static coupled fluid-structure problems are presented in this chapter. The compressible fluid solver with Arbitrary Lagrangian-Eulerian (ALE) formulation and the non-linear structural solver are used for the optimization problems in this chapter. Non-linear FEM advantages over a linear solution are explored for problems with large deformations. The coupled fluid-structure interactions are solved using Block Gauss-Seidel iterations with coupling on structural boundary displacement and fluid tractions.

The coupled sensitivities are evaluated using the discrete adjoint method with algorithmic differentiation via CoDiPack [84]. A large number of design variables are chosen for the problems presented in this chapter, which are then solved efficiently using the discrete adjoints. Two types of designs variables are explored, variable stiffness for optimal design of structural properties and Free-Form Deformation (FFD) control points for optimal shape design.

This chapter firstly presents the results from optimal design of transonic compliant airfoil using variable stiffness as design variables for improved characteristics. The optimized variable stiffness are chosen such that the compliant airfoil under the transonic flow loads led to an

optimized wet shape. Secondly, the shape is optimized for an airfoil with uniform stiffness using FFD control points as design variables. The unloaded shape is optimized such that under aerodynamic load, the structural deformation leads to an optimized shape. The derived discrete adjoint methodology and implementation have been kept generic. This facilitates inclusion of different levels of physical complexity as required in both the structural and fluid domains. To demonstrate the functionality of the developed framework, these proof of concept test cases are restricted to inviscid problems without aerodynamic constraints. Additional constraints can be included in these problems, however they will require additional adjoint calculations for each constraint which will add to the computational cost. It must be remarked that if the effects of turbulence through RANS simulations are included together with lift or pitching moment targets, the optimal designs will likely be very different. Nonetheless, the current emphasis is on the demonstration of the optimization framework for steady, coupled FSI problems which will be presented for stiffness and shape design variables.

4.1 Aeroelastic Stiffness Optimization

The optimal design of structural properties for aerodynamic objective function is performed through choice of stiffness as design variables. Though structural properties in airfoil design are not typically varied through the use of stiffness, this can be used as a proxy for more complicated internal structures. Also, recent work in the use of metamaterials, to allow micro-architectures which have significantly different properties to the baseline material, has been shown to be an effective approach in tailored design of lightweight structures [119].

A compliant NACA 0012 airfoil in transonic, inviscid 2-D flow is chosen with 10216 elements fluid elements and 550 structural elements. The transonic problem is defined by Mach Number, Ma = 0.8, angle of attack, $\alpha = 1.0$, and free-stream pressure, $P_\infty = 101325.0$ Pa. The boundary conditions of Euler wall are used on the airfoil surface and far-field boundary condition at 20 chord length. The airfoil structure has baseline stiffness, $E_{ref} = 10$ GPa, density, $\rho = 2700\ kg/m^3$, Poisson ratio, $\nu = 0.35$, and solved using neo-Hookean material model. This

simple problem is addressed first since the inviscid fluid with small structural displacement provides an ideal starting point to verify the discrete adjoint methodology for static coupled design. The structure is supported with an inner clamp near the leading edge as highlighted in Figure 4.1. The clamp is rigid while the rest of airfoil structure has variable stiffness. The airfoil stiffness is defined by 20 design variable segments, with 10 segments on each of the lower and upper halves of the airfoil as shown in Figure 4.1. Each segment has a width of 10% chord length. The design variable are multipliers to increase or decrease the stiffness of each design variable segment. Initially, this is given by $\alpha_i = 1, \forall\ i$.

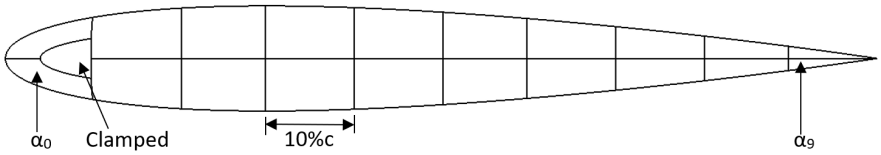


Figure 4.1: Schematic for compliant NACA 0012 airfoil with 20 design variable segments for variable stiffness.

An aeroelastic optimization is set up to minimize drag by varying the structural stiffness in the different segments. A lower bound is enforced for the stiffness multiplier to maintain feasible design. The resulting optimization is given by

$$\begin{aligned} \min \quad & J(\alpha_i) = C_D \\ \text{s.t.} \quad & \alpha_i \geq 10^{-4}. \end{aligned} \tag{4.1}$$

Firstly, the gradients are verified using first order forward finite differences with step size $10^{-5}E_{ref}$. The discrete adjoint results in Figure 4.2 show very high sensitivity of the leading edge design variables on the airfoil drag and minimal influence of the airfoil stiffness after the half-chord point. Comparison against finite differences (FD) confirm excellent agreement with discrete adjoint (DA) for all design variables, with less than 1% relative error for the variables with highest sensitivities.

The discrete adjoint sensitivities are used to drive a gradient based optimization to minimize drag without any constraint on lift. The drag observed in this problem is purely wave drag since viscous effects are not included. The optimization uses the sequential least squares quadratic

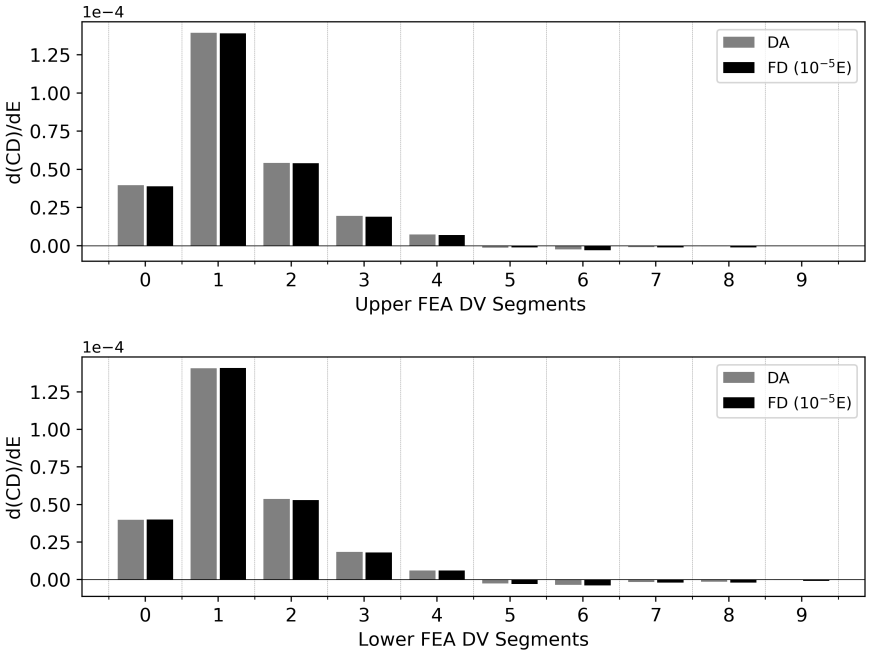


Figure 4.2: Gradient verification for discrete adjoint sensitivities with stiffness design variables using finite differences.

programming (SLSQP) method implemented in SciPy [120] and the converged optimization history is presented in Figure 4.3. The baseline airfoil with uniform stiffness of 10 GPa results in a very stiff airfoil with less than 1% tip displacement. As a result, the original compliant airfoil performs very closely to a fluid only problem, which can be seen in the pressure distribution and in the surface profile under load shown in Figure 4.4. The baseline airfoil has two shocks, a stronger shock on the upper surface at 60% chord and a weaker second shock on the lower surface near 40% chord. The optimization has reduced the drag by 47.0%. The change in stiffness design variables from the initial uniform stiffness distribution are also shown which indicates an increase in stiffness towards the trailing edge and decrease in stiffness towards the leading edge.

The optimized airfoil with variable stiffness has reduced drag by reducing the strength of the shock on the upper surface and moving it towards the leading edge. This is achieved by the airfoil deflection upwards at the trailing edge. As a result, the shock on the lower surface is strengthened in comparison to the original design and has moved further aft. The optimized design has both shocks of similar strength and is positioned half way along the airfoil, this symmetrical response indicates that the airfoil effectively is at zero angle of attack. Since lift

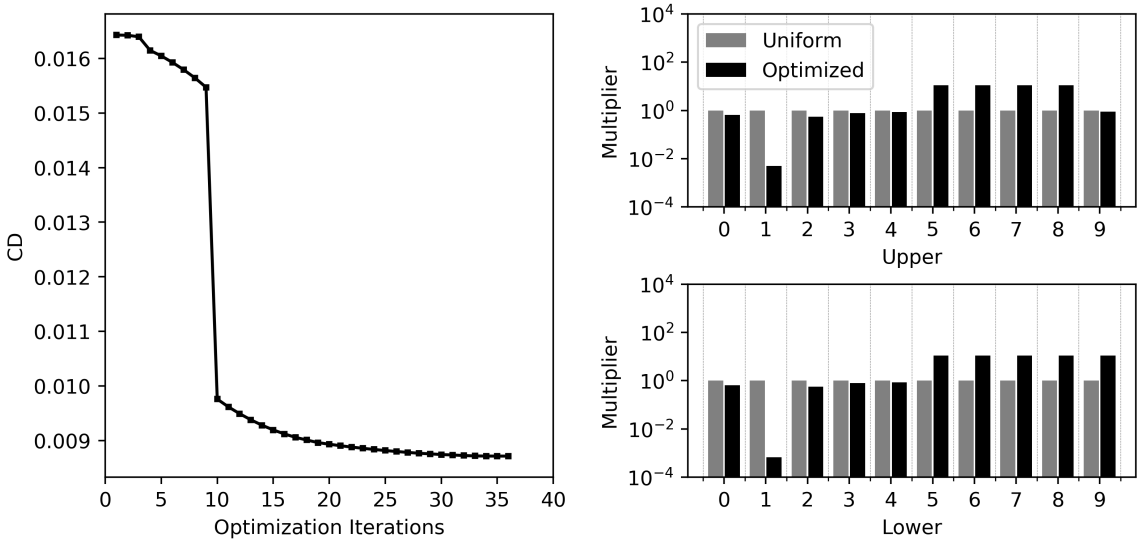


Figure 4.3: Optimization history for compliant NACA 0012 airfoil with design variables of stiffness distribution.

is not constrained, the optimization has driven the airfoil to reduce lift in order to weaken the shock, which reduces the drag objective function.

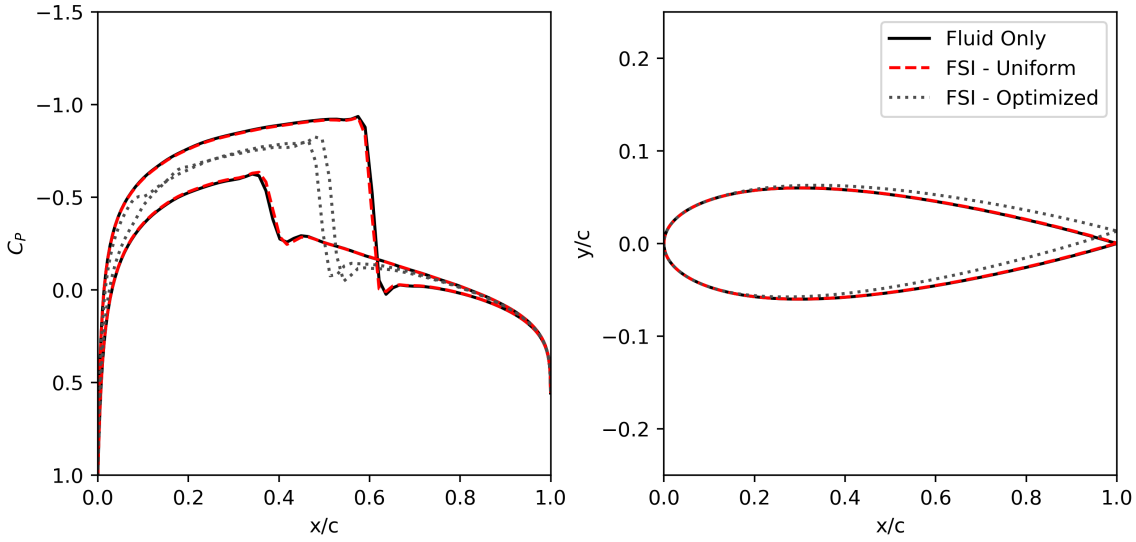


Figure 4.4: Comparison of C_P distribution and airfoil surface for fluid only, baseline compliant airfoil with uniform stiffness and optimized compliant airfoil with variable stiffness.

The biggest improvement in drag is seen between designs 9 and 10. This is explored further and the comparison of the two designs are shown in Figure 4.5. Design 9 has reduced the stiffness at the leading edge compared to the initial uniform distribution, however, the biggest change in design 10 is the greater reduction of the stiffness of segments between 10% chord and 20%

chord on both the upper and lower halves. This is located directly aft from the inner clamp allowing the airfoil to deflect upwards which causes the bigger movement in shock locations and strength as can be seen in the pressure profile comparison.

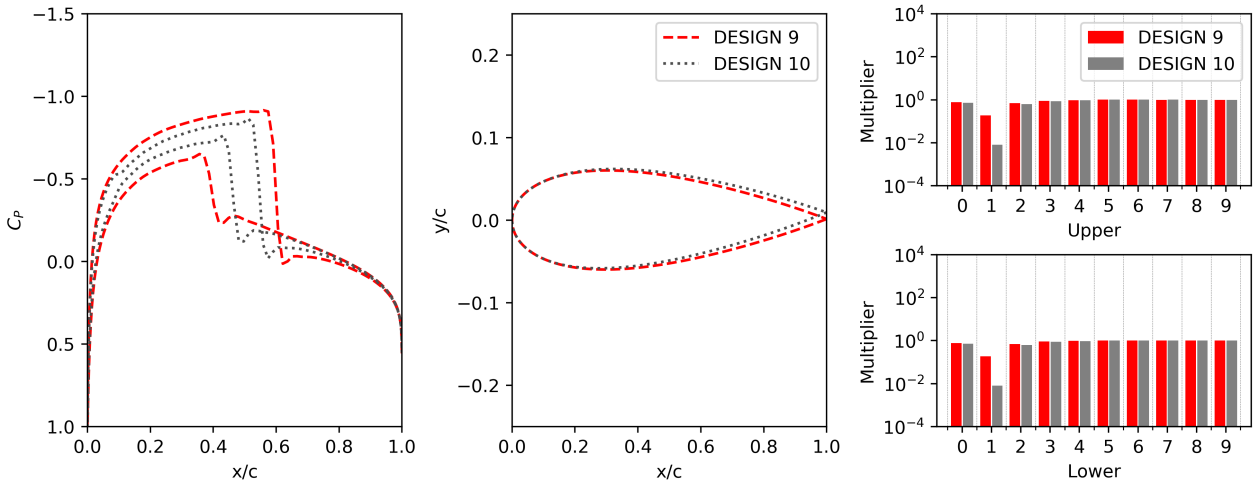


Figure 4.5: Comparison of C_P distribution, and airfoil surface and variation in airfoil stiffness for design iterations 9 and 10.

Further design iterations progressively decrease the stiffness directly behind the clamp further whilst increasing stiffness at the trailing edge leading to final stiffness distribution. The stiffness is increased at the trailing edge in order to control the tip displacement with increased flexibility behind the clamp. Though the upward deflection of the airfoil at the shock location has helped to reduce the strength of the shock on the upper surface and in turn the drag on the airfoil, increased tip deflection would lead to further increased shock strength on the lower surface thus negating the drag loss. As a result, the optimum stiffness distribution has found a compromise of similar shock strengths on the upper and lower surfaces, by reducing the angle of attack of the deformed airfoil under aerodynamic load to zero, to provide the drag improvement from the baseline airfoil.

4.2 Aeroelastic Shape Optimization

A compliant NACA 0012 airfoil is now used to demonstrate aeroelastic shape optimization in which the outer shape of the undeformed airfoil is modified such that under operational

conditions, the coupled response results in a deformed shape which is optimal. Uniform stiffness is defined in the airfoil, therefore a clamp to constrain the structural problem is now simplified to a vertical clamp at 20% chord. Increased flexibility will be introduced to enable bigger displacements. The fluid domain is discretized with 10216 elements and the structural mesh with 1468 elements. In this section, problems with small and large displacements will be considered for both analysis and shape optimization. As a result, a refined structural mesh, particularly towards the trailing edge, which match the surface nodes of the fluid mesh is used. The schematic of the problem is shown in Figure 4.6, including the definition of the FFD box for shape parameterization. FFD box with 18 control points, α_i, (0 to 8 on the lower surface and 9 to 17 on the upper surface, left to right) will be used as design variables, restricted to vertical movement, for the shape optimization.

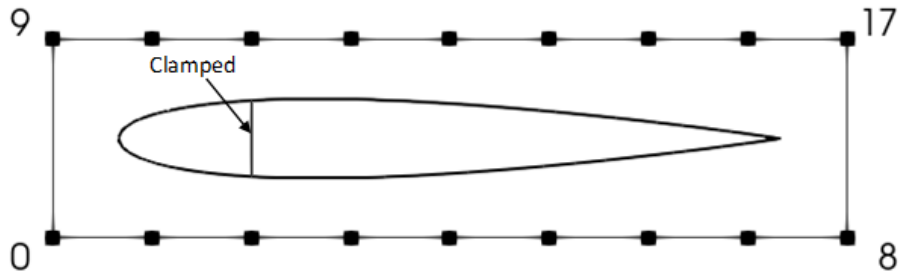


Figure 4.6: Schematic of NACA 0012 airfoil for optimization with FFD box.

The airfoil and simulation properties are described in Table 4.1. Euler equations are used again for the fluid and hyper-elastic neo-Hookean material model for the structure. Aerodynamic efficiency is maximized under a minimum area constraint equal to the starting airfoil area,

$$\begin{aligned} \max \quad & J(\alpha_i) = C_L/C_D \\ \text{s.t.} \quad & A_i = A_0. \end{aligned} \tag{4.2}$$

The optimization problem was driven again using the sequential least squares quadratic programming (SLSQP) method implemented in SciPy [120].

The FSI shape derivatives are firstly evaluated using discrete adjoint for the objective function of aerodynamic efficiency and compared against finite differences with step size of 10^{-5}c. Figure 4.7 presents the gradient comparison for each design variable which confirms excellent agreement

Table 4.1: Boundary conditions and properties.

Chord	1 m	Angle of attack	1.25 deg
Far field radius	20.0 m	Free-stream Mach number	0.8
Elasticity modulus	70 MPa, 7 TPa	Free-stream pressure	101325 Pa
Poison ratio	0.35	Free-stream temperature	273.15 K

with less than 0.1% maximum relative error. The same trends in sensitivities are seen for both the lower and upper surfaces with peak magnitudes at control points 1 and 6 for the lower surface and control points 9 and 15 for the upper surface.

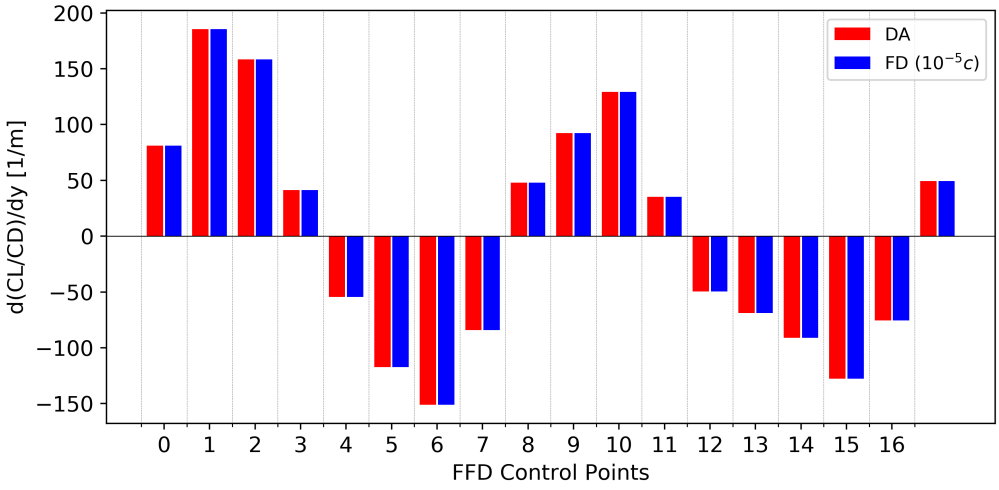


Figure 4.7: Verification of discrete adjoint shape derivatives using finite differences, $E = 70$ MPa.

4.2.1 Optimal Design with Small Structural Displacements

Fluid-only shape optimization was performed and compared with coupled FSI optimizations on two airfoils, namely a stiff and flexible airfoil. The stiff airfoil test case has $E = 7$ TPa to give negligible trailing edge displacement, therefore this can be expected to perform similarly to the fluid-only problem. The flexible airfoil has stiffness of $E = 70$ MPa which results in initial trailing-edge displacement of 1.5%c. The coupled simulation with increased flexibility will investigate the effect on the resulting shape from optimization when the structure is allowed to deform under the aerodynamic loads.

Figure 4.8 shows the optimization history for all three problems which confirms that initially

the fluid and stiff airfoil optimization paths are extremely similar whilst the flexible airfoil takes a different path. In all cases, the optimizer increases efficiency initially with increase in lift and drag. However without a lift constraint and viscous drag, the problem is predominantly driven by wave drag minimization. The optimizer reduces the drag to very low levels for all three optimizations, thus resulting in very high efficiency designs for this problem.

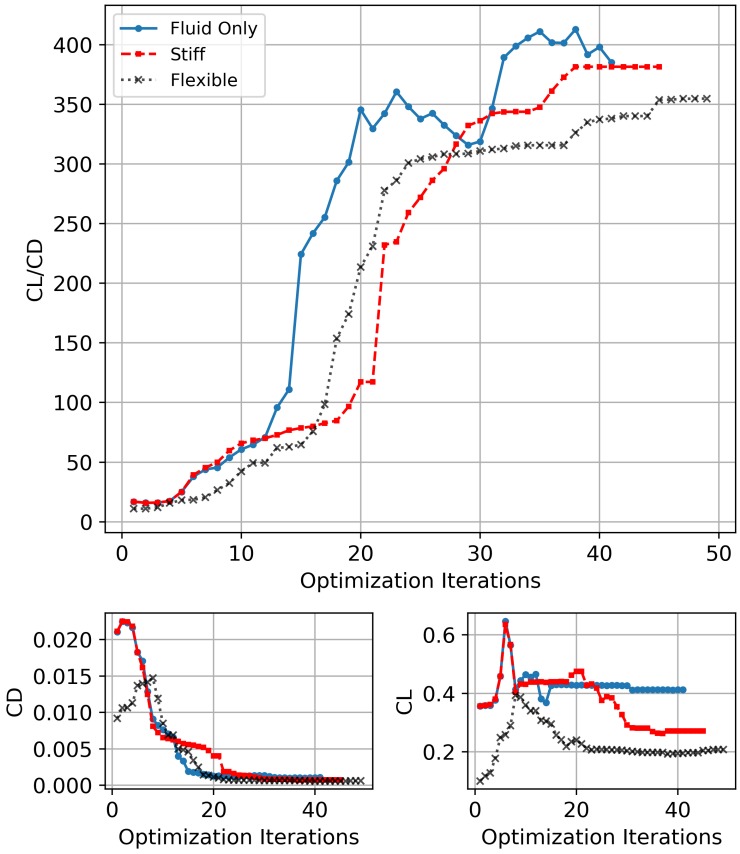


Figure 4.8: Optimization history of NACA 0012 airfoil for efficiency, C_D and C_L.

Comparing the fluid only and stiff airfoil optimization in more detail, for the first five design iterations, the two are in excellent agreement. In order to find the sixth design iteration, the optimizer for both fluid and stiff airfoil needed to perform two line searches as the first one was not deemed an improvement. The line searches were performed independently, and resulting designs for the fluid and stiff airfoil problems had lesser agreement. The resulting small differences in the designs from the two problems can be seen in the optimization history. Between design iterations 12 and 13, three line searches were required and the two optimizations can be seen to diverge from this point. Nonetheless, both converge towards similar optimized profiles overall. Due to the large design space for the optimization without the aerodynamic

constraint, the fluid optimization in particular can be seen to have had difficulty with continual improvement of efficiency with each design iteration, in particular during the second half of the optimization when the drag values are already very low.

Figure 4.9 compares the NACA 0012 profile and pressure coefficient distribution with resulting shape from the final design for both the fluid only and stiff airfoil. Despite the different optimization histories, both cases show reduced shocks on the upper and lower surfaces with a flattened profile on the upper surface and cambered aft section which is characteristic of transonic airfoils. Small variations on the airfoil surface between the designs lead to slightly different pressure profiles, however the overall reduction in drag through weakened shocks remains common to both.

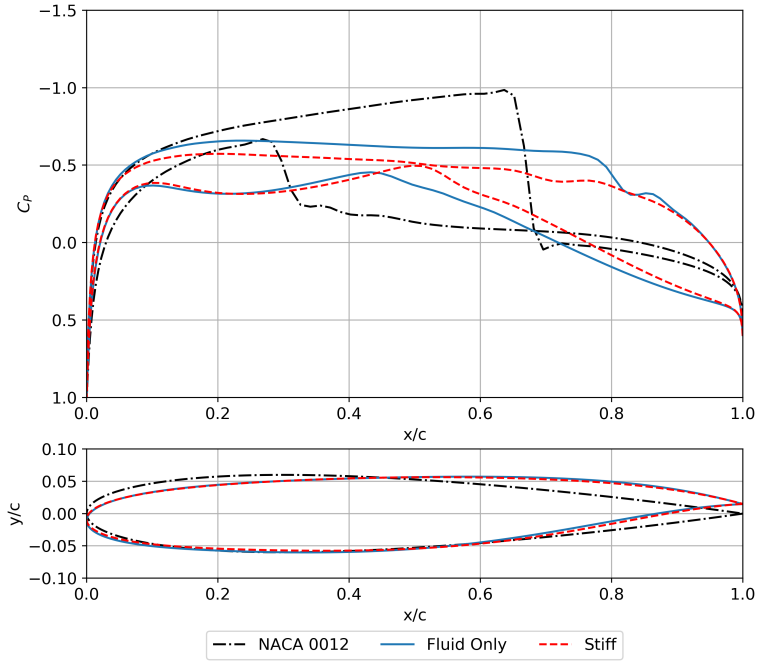


Figure 4.9: C_P distribution and airfoil shape results from fluid only and stiff FSI optimization.

For the flexible airfoil, the optimizer pitches up the undeformed airfoil which is shown in Figure 4.10. Under aerodynamic load, the trailing edge of the airfoil bends up resulting in the grey shape seen by the flow which successfully reduces shock on both the upper and lower surfaces.

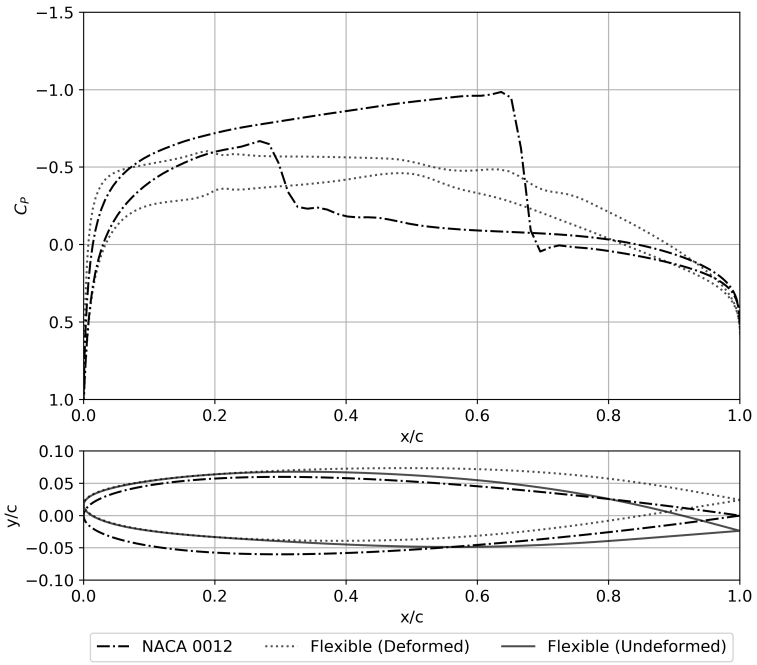


Figure 4.10: C_P distribution and airfoil shape results from flexible FSI optimization.

4.2.2 Optimal Design with Large Structural Displacements

In order to consider problems with larger displacements, the flexible NACA 0012 airfoil with 70 MPa was initially deformed using the FFD box as shown in Figure 4.11. Control points 7 and 8 on the lower surface were moved, by -0.664c and -0.964c respectively, to obtain a new baseline shape for this section with 18% tip vertical coordinate. This design now represents the undeformed shape shown in Figure 4.11 (right) in grey and the deformed shape under aerodynamic load in white.

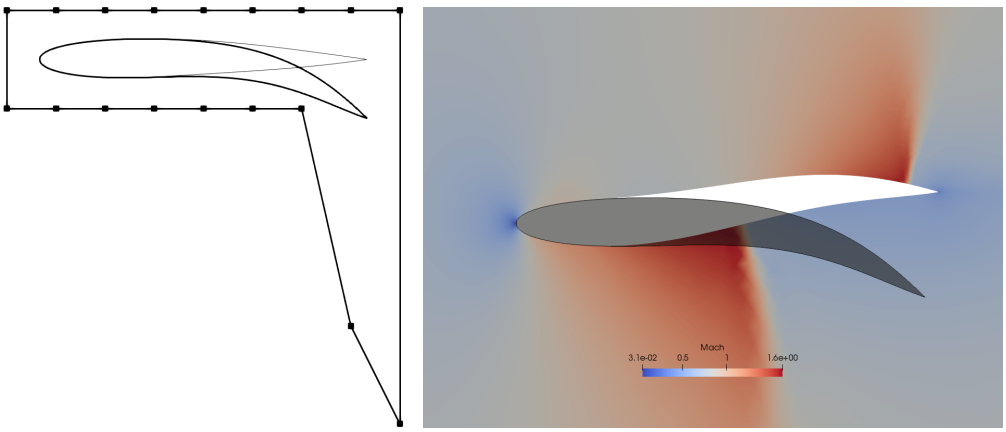


Figure 4.11: New baseline airfoil through deformed FFD box to produce large displacement due to aerodynamic load.

The airfoil was simulated with neo-Hookean hyper-elastic material model for the non-linear structural solver and with linear elastic model for the linear solver suited for small deformations. The result for the baseline airfoil is shown in Figure 4.12 which highlights the significantly different pressure profiles from the undeformed baseline shape to the deformed shape from both linear and non-linear solutions. The fluid problem has only one shock on the upper surface whereas the deformed shape of the compliant airfoil leads to strong shocks on both the upper and lower surfaces which are captured by both the FSI solutions presented in Figure 4.12. Comparing the two structural solvers, it can be seen that they both show similar deformation profiles but the linear solver over-predicts the deformed chord length of the airfoil.

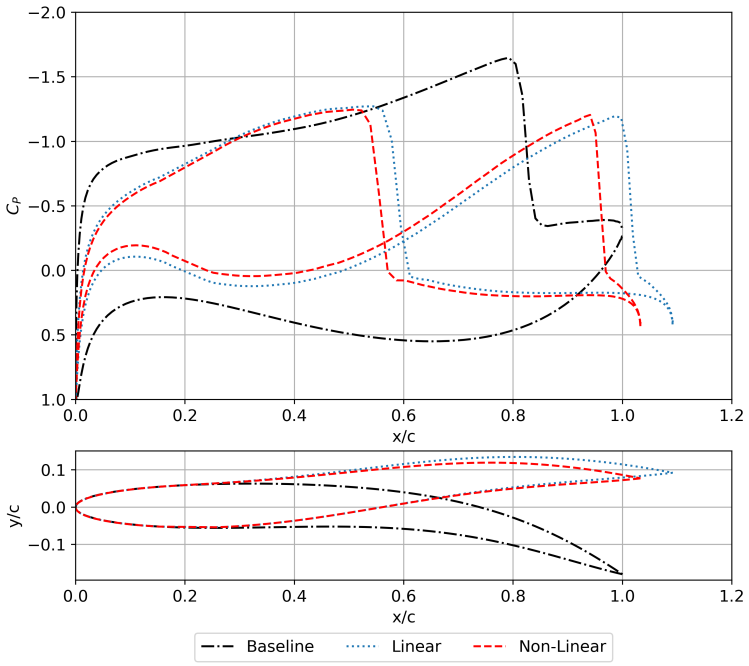


Figure 4.12: Comparison of non-linear and linear structural solver with baseline airfoil.

The coupled FSI shape derivatives from both structural solvers for the baseline airfoil are presented in Figure 4.13. Across all derivatives, it can be seen that both linear and non-linear solutions lead to gradients with similar trends but differing magnitudes, with the linear solver under-predicting the gradient magnitude.

An optimization to maximize aerodynamic efficiency under minimum area constraint is set up for the new baseline airfoil. The resulting optimization history is illustrated in Figure 4.14 for the first 20 iterations which indicates that both linear and non-linear simulations have similar

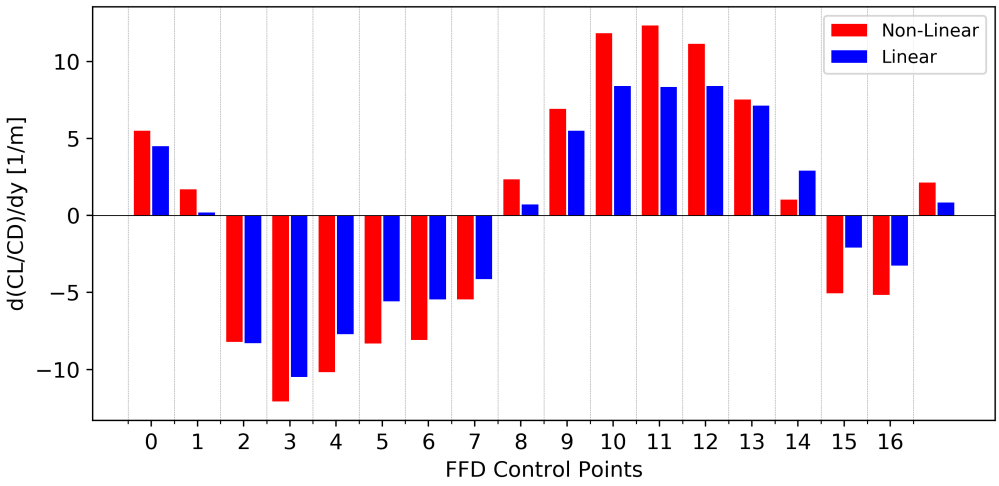


Figure 4.13: Comparison of discrete adjoint AD shape derivatives of non-linear and linear structural solver on the baseline airfoil.

trends with improvements to aerodynamic efficiency in similar iterations.

As a result, the optimal undeformed structural shape of the airfoils shown in Figure 4.14 agree very closely between the linear and non-linear solvers despite the differences in sensitivity seen previously. Note however that the linear elastic model predicts again a longer chord after deformation and 11% lower optimum efficiency compared to the non-linear analysis. This suggests that a linear structural solver can be used to reduce the cost of the optimization, however non-linear optimization and design verification will be needed to refine the optimum found with a linear solver.

Overall, the results from this chapter have verified the discrete adjoint sensitivities and demonstrated their use in optimal design of coupled FSI problems. The practical verification of this new methodology has been the primary focus with simplified problem definition, nonetheless this has been expanded further to coupled FSI problems with RANS in collaboration with Gomes [91]. Additional physical constraints, from either the structural or aerodynamic domain, can be included for designs with practical application as well as the extension to 3-D problems.

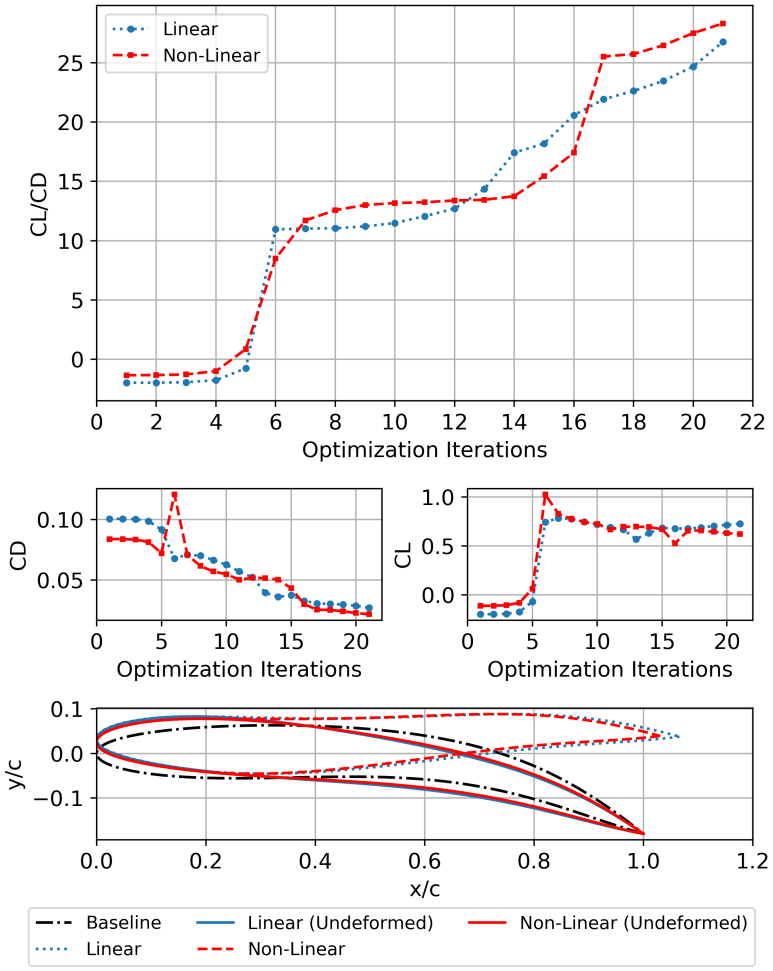


Figure 4.14: Optimization history for the new baseline airfoil with large displacements and the optimized designs using both linear and non-linear structural solvers.

Chapter 5

Optimal Design of Structural Dynamic Problems

Applying the structural formulation and methods of evaluating the system sensitivity discussed in the previous chapters, this chapter discusses numerical investigations for structural dynamic problems. Initially, stiffness is used as the design variable for the discrete adjoint verification. The introduction of shape variables for structural problems is then verified and applied for dynamic structural shape optimization. The secondary motivation for this work is to establish structural dynamic characteristics under vacuum conditions as well as to understand the compromises and structural properties exploited for optimal design when only structural implications are included. When coupled simulations are then considered, this could lead to a better understanding of the changes due to the coupled response.

The chosen objective function for the demonstration of the dynamic structural optimization in this chapter are time-averaged magnitude of displacement on a reference node. This is defined by

$$J(\mathbf{u}(E,t)) = \frac{k}{N}\sum_{i=1}^{N}\sqrt{(\mathbf{u}_i - \mathbf{u}_{ref})^{\mathsf{T}}(\mathbf{u}_i - \mathbf{u}_{ref})} \tag{5.1}$$

where $\mathbf{u}_i$ is the displacement at time step, i, $\mathbf{u}_{ref}$ is the reference displacement, k is the scalar penalty factor and N is the number of time steps. Note that for static problems, the objective

is simplified from Equation 5.1 by removing the averaged summation over time. The sign of k can be chosen to switch between minimization and maximization problems such that positive value will lead to minimization of the objective function whilst negative value will lead to maximization problems. Minimization problems can be used to reduce overall structural displacements in the solution or to allow the structure to deform a desired amount to match a reference profile. Conversely, the maximization problem can be used to identify and design structural properties to manipulate the natural frequency at which peak displacements are found.

The chapter has been structured to begin with a canonical problem of a single-clamped bar under axial load, with known analytic solutions and optimum design properties, for preliminary verification of the structural solver and discrete adjoint framework. The structural characteristics are explored and their identification through sensitivity analysis is evaluated. The adjoint solver is interfaced with CoDiPack [84] for algorithmic differentiation. Next, a compliant airfoil under vertical loading is introduced and the structural characteristics are evaluated. The geometric sensitivities are verified through design variables of shape parameterization ahead of structural shape optimization. This forms a foundation for the compliant airfoil characteristics and optimal design from pure structural analysis ahead of coupled analysis.

5.1 Optimal Stiffness of Elastic Bar under Axial Load

The verification is performed using a single-clamped bar under axial loading. The bar has length $L = 1$ m, thickness $t = 0.1$ m and 20 quadrilateral elements. The bar has density $\rho = 312.5$ kg/ m^3, and it is subject to an external load $P = 100N$ at its free end. The bar has uniform Young's modulus, $E = 80$ kPa, and this is the chosen design parameter in the range of $6 - 100$ kPa. Zero Poisson ratio, $\nu = 0.0$, is used for these simulations, which therefore assumes that there is no transverse strain but only longitudinal strain present in the bar, in order to allow comparisons against analytical solutions.

5.1.1 Elastic Bar under Static Axial Load

Firstly, the bar is tested under static loading condition. The present results using 20 elements agreed very well with the analytic objective function using the expression for tip displacement as $u = PL/Et$ with 0.6% error at $E = 80$ kPa.

The design space is evaluated to compute the objective function and sensitivity with Young's modulus. This is a trivial problem with monotonic dependency of the objective function with Young's modulus as shown by the inversely proportional curve in Figure 5.1. The gradient is calculated using discrete adjoint, which is also shown in Figure 5.1 which has been scaled by the stiffness to obtain comparable order of magnitude in the sensitivities and objective function.

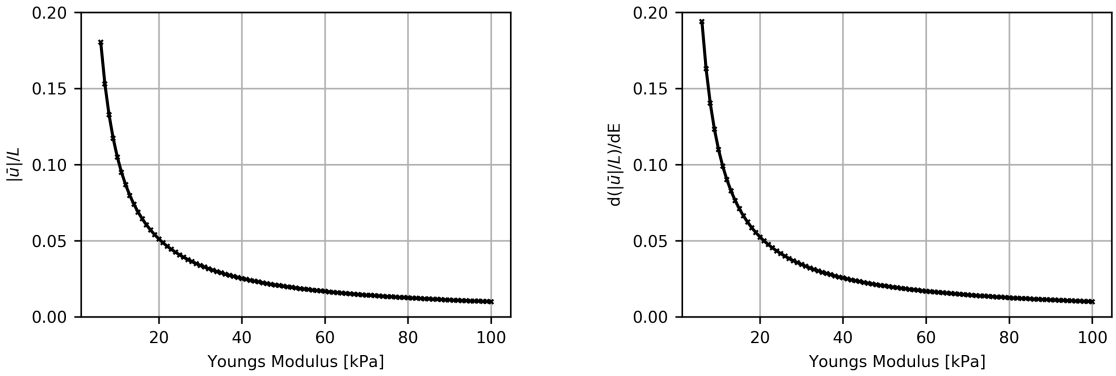


Figure 5.1: Static analysis results for objective function, sensitivity using discrete adjoint.

The adjoint sensitivities are verified using direct differentiation and second order central differences using step size of 5 Pa. Figure 5.2 shows the error in sensitivities compared to that calculated from the discrete adjoint method. The adjoint sensitivities are verified using both alternative sensitivities calculations with the error in central differences is seen to be of the order of 10^{-8}, whereas, the direct differentiation agrees to machine precision with order of 10^{-14}.

5.1.2 Elastic Bar under Sinusoidal Axial Load

Moving to dynamic analysis, the identification of the natural frequencies of the uniform elastic bar is considered. Since the bar has uniform stiffness and constant thickness along the length,

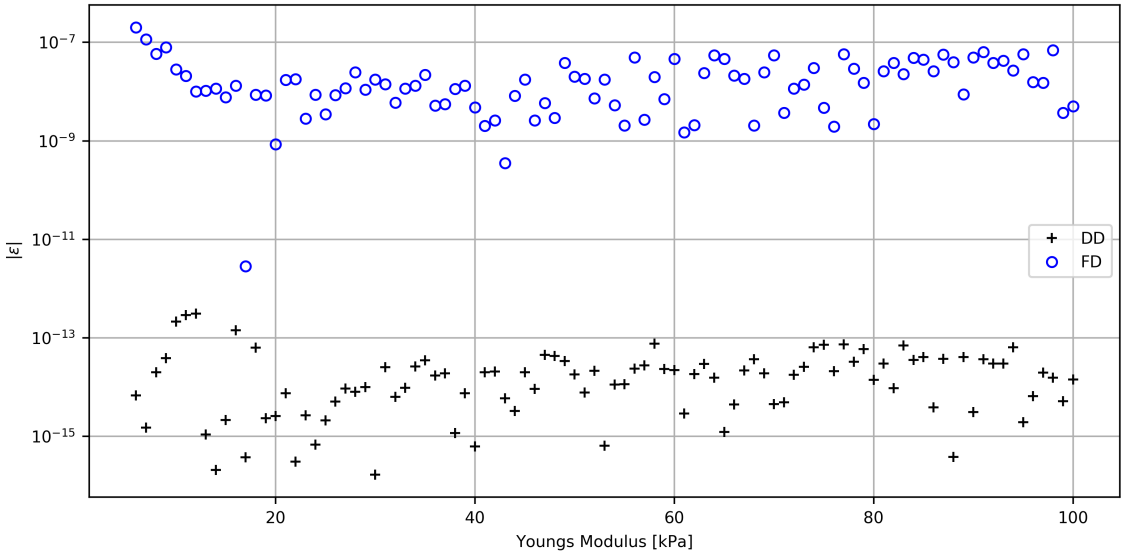


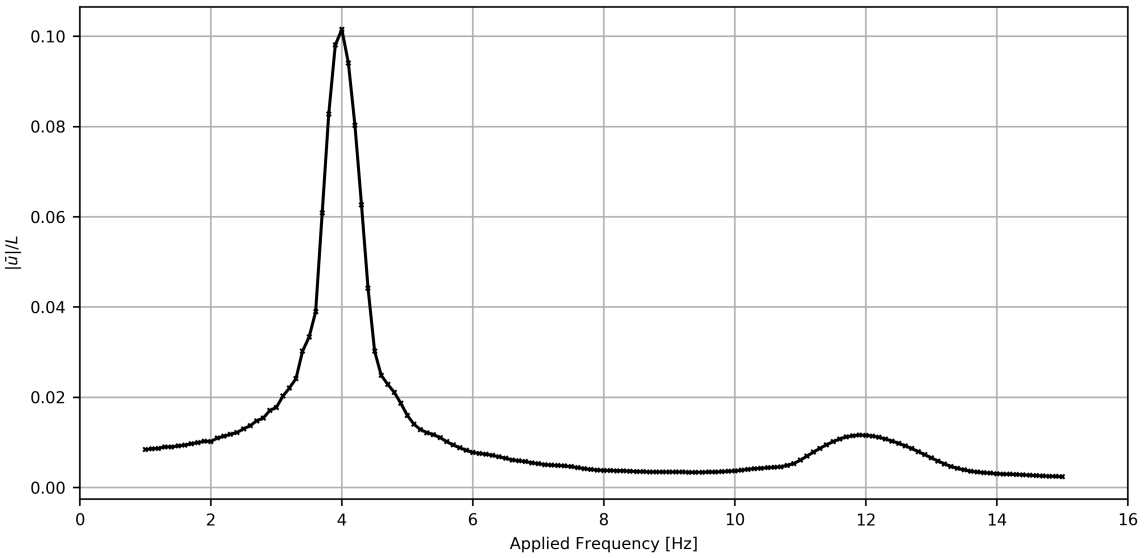
Figure 5.2: Relative error of sensitivities for direct differentiation and central differences ($h = 5$ Pa) with discrete adjoint sensitivities as reference.

the natural frequency can be analytically evaluated. The first five natural frequencies of the uniform bar described in the introduction of Section 5 are thus given in Table 5.1.

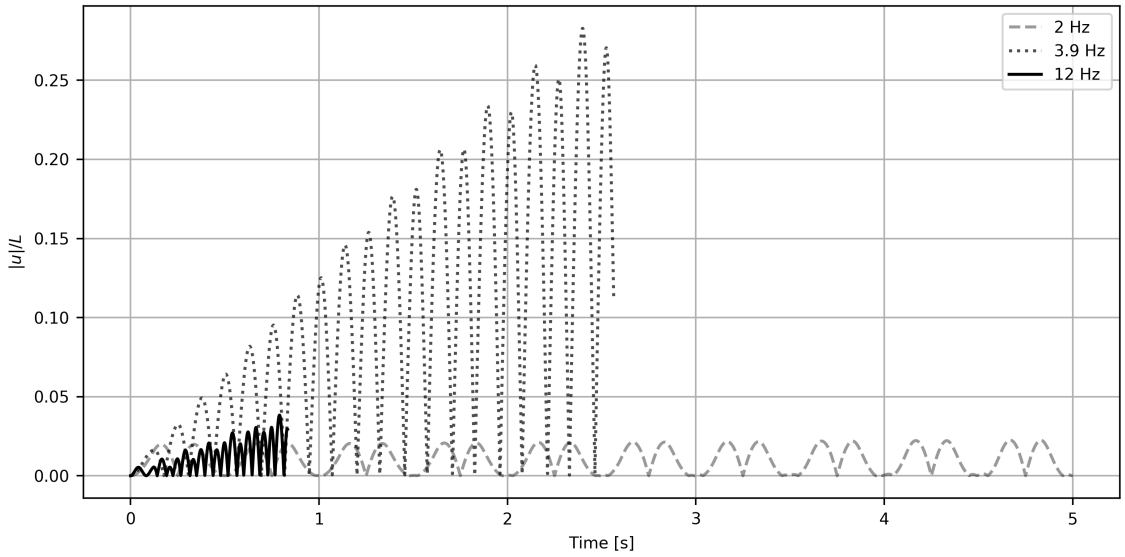
	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5
Natural Frequency (Hz)	4.0	12.0	20.0	28.0	36.0

Table 5.1: Natural frequency and time period of first five natural modes.

The load on the bar is now applied with a sinusoidal function with amplitude 100N. The dynamic simulations are run for ten periods with time step of 0.0005 s. The Newmark constants from Equation (2.40) have been chosen as $\gamma = 0.52$ and $\beta = 0.2601$ for 2% numerical damping. The frequency of the applied load is varied and the time-averaged magnitude of tip displacement over the ten periods are recorded to identify which excitation frequency triggers the largest response in the structure. The first two modes are of interest, therefore the applied frequency range of $1 - 15$ Hz is chosen and the results are plotted in Figure 5.3(a). The peaks in Figure 5.3(a) confirms the identification of the first and second structural natural frequencies at 4 and 12 Hz. The time history of the absolute tip displacement plotted in Figure 5.3(b) highlights the stable oscillations for loading at 2 Hz, whereas the growing oscillations at the first natural frequency of 4 Hz. Though the amplitude of the second natural frequency at 12 Hz is smaller in comparison, the characteristic growing amplitudes of the second mode are clearly seen.



(a)



(b)

Figure 5.3: Variation in time-averaged magnitude of tip displacement under sinusoidal load of $1 - 15$ Hz is shown in (a) and (b) shows the time history results of absolute tip displacement for loading at $2, 4, 12$ Hz.

Using the evaluated structural properties, the single-clamped bar is subjected to a sinusoidal load, P, such that

$$P = 100\sin(\omega_1 t), \tag{5.2}$$

where $\omega_1 = 8\pi$ rad/s, which corresponds to the first natural frequency of the bar. The range of Young's modulus considered is $6-100$ MPa. The chosen objective function is the time-averaged absolute tip displacement (Equation 5.1). Figure 5.4 shows the variation in objective function in the explored range of stiffness and a peak can be seen at $E = 80$ kPa as expected. A second peak is seen at $E = 8$ kPa. For the bar with $E = 8$ kPa, the second natural frequency is 3.79 Hz and given the excitation from the loading frequency of 4 Hz, the influence from the second mode is observed. The sensitivities calculated from the discrete adjoint method are also given in Figure 5.4 which identifies the natural frequencies with zero sensitivities indicating the peaks of the objective function at 80 kPa and between $8-9$ kPa, given the step size of stiffness that has been investigated. The sensitivities have once again been scaled by the stiffness for comparable order of magnitude in the sensitivities to the objective function. Note that local minima and maxima are also found in between the two natural frequencies as a result of noise/numerical excitations which are also seen in the objective functions. Therefore, if an optimization is to be set up to identify the optimum stiffness for a given mode, this will need to be started from an appropriate baseline stiffness and the sensitivities scaled accordingly in order to find the true maximum.

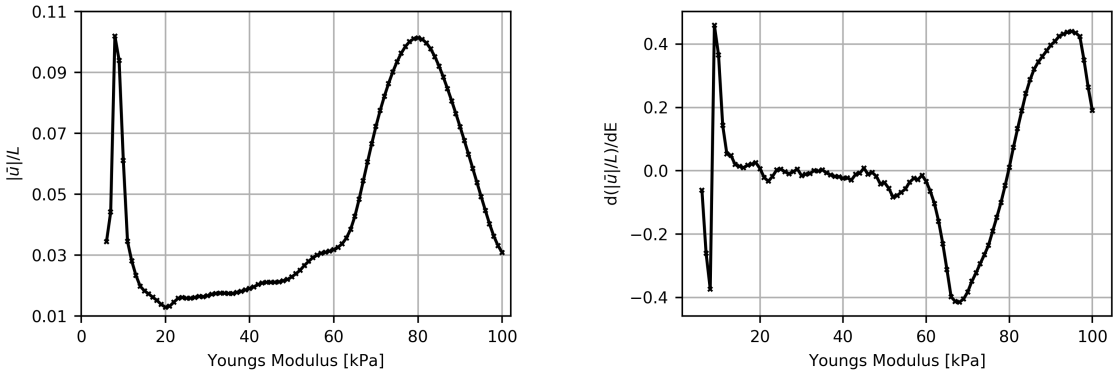


Figure 5.4: Dynamic analysis results with ω_1 frequency load: objective function, sensitivity using discrete adjoint reverse mode AD.

Figure 5.5 shows the spread in error taken as the difference between the adjoint sensitivities and the alternative methods of central differences with step size of 5 Pa and direct differentiation. The error against central differences is mostly in the order of 10^{-4} and comparing the discrete adjoint sensitivities to that from direct differentiation, the error for most stiffness design variables are in the order of 10^{-11} to 10^{-12}. Both display errors of magnitude significantly lower than the objective function thus confirming excellent agreement for the adjoint sensitivities.

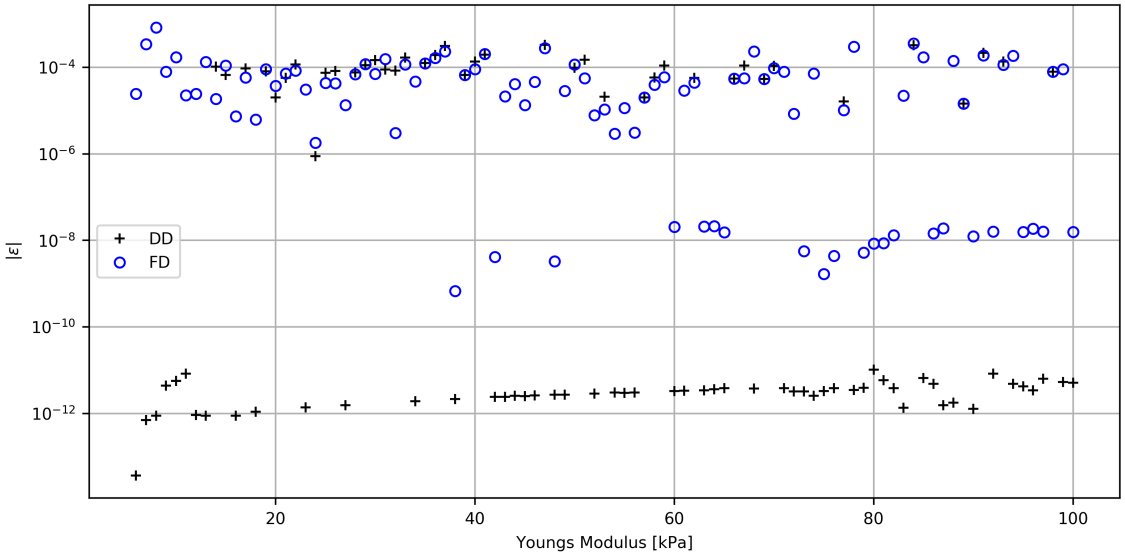


Figure 5.5: Relative error of sensitivities from direct differentiation forward mode AD and central differences ($h = 5$ kPa) using reference from discrete adjoint sensitivities.

5.2 Shape Optimization for Resonant Behaviour of an Airfoil Structure

A NACA 0012 airfoil in vacuum under vertical sinusoidal load is chosen as the second test case. A sinusoidal pressure load is applied vertically on 5% chord from the trailing edge with an amplitude of 1 kPa and varying frequencies. The airfoil has uniform stiffness with baseline Young's modulus $E = 10$ MPa, Poisson ratio, $\nu = 0.35$, and material density $\rho = 300$ kg/ m^3. An internal clamp is used near the leading edge of the airfoil, between $5 - 15\%c$, is shown in Figure 5.6 and a shape parameterization for the airfoil is performed using a Free-Form Deformation (FFD) box with 20 control points, α_i. The outer airfoil profile is allowed to be

deformed using the FFD control points, whilst holding the inner clamp fixed at the initial shape.

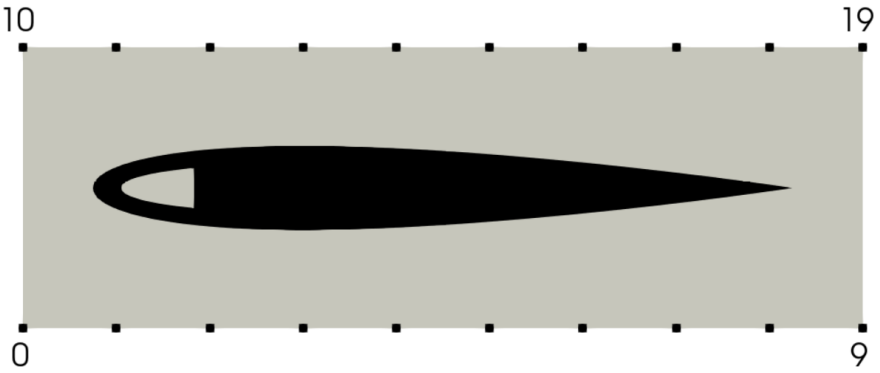


Figure 5.6: Schematic of FFD around NACA 0012 airfoil.

All simulations are run for six periods with time step of 1 ms and 10% numerical damping. The objective function of time-averaged magnitude of tip displacement, over the final two periods, is chosen using zero tip displacement as the reference at the trailing edge of the airfoil.

In order to identify the required grid size, a brief grid convergence was performed with Coarse (506 elements), Medium (912 elements), Fine (1968 elements) and Finer (3872 elements) grid levels. The time-averaged tip displacement of the airfoil under sinusoidal loading at 1 Hz for each grid are presented in Figure 5.7 as percentage error from the Finer grid level. The time averaged time displacement for the Fine grid is within 2.2% of the Finer grid and is chosen for the following simulations.

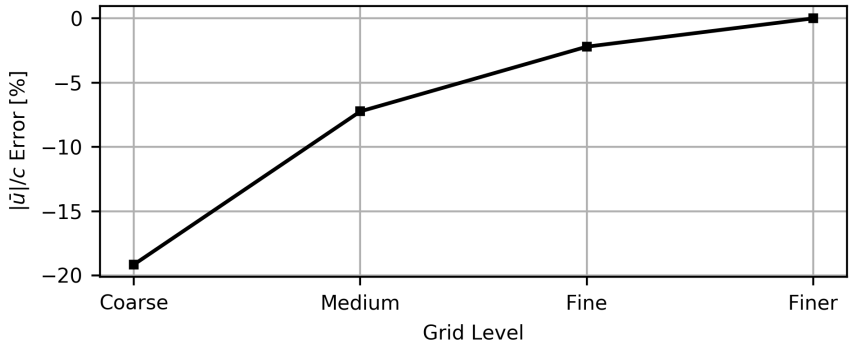


Figure 5.7: Variation of time-averaged tip displacement using four grid refinement levels.

The sinusoidal load on the airfoil is varied for a range of frequencies, $1 - 10$ Hz, in order to identify the natural frequency of the airfoil. Figure 5.8 shows the result of the time-averaged

magnitude of tip displacement for the range of applied frequencies which identifies a natural frequency peak at 7.3 Hz. If the stiffness of the airfoil is reduced to $E = 8$ MPa, the frequency sweep indicates a greater peak tip displacement but now with a lower natural frequency of 6.5 Hz. The time history results for the absolute tip displacement for both airfoils are also given in Figure 5.8 at both natural frequencies which confirms that the airfoil with $E = 10$ MPa has growing amplitudes of oscillations at 7.3 Hz whilst the airfoil with $E = 8$ MPa has growing amplitudes of oscillations at 6.5 Hz.

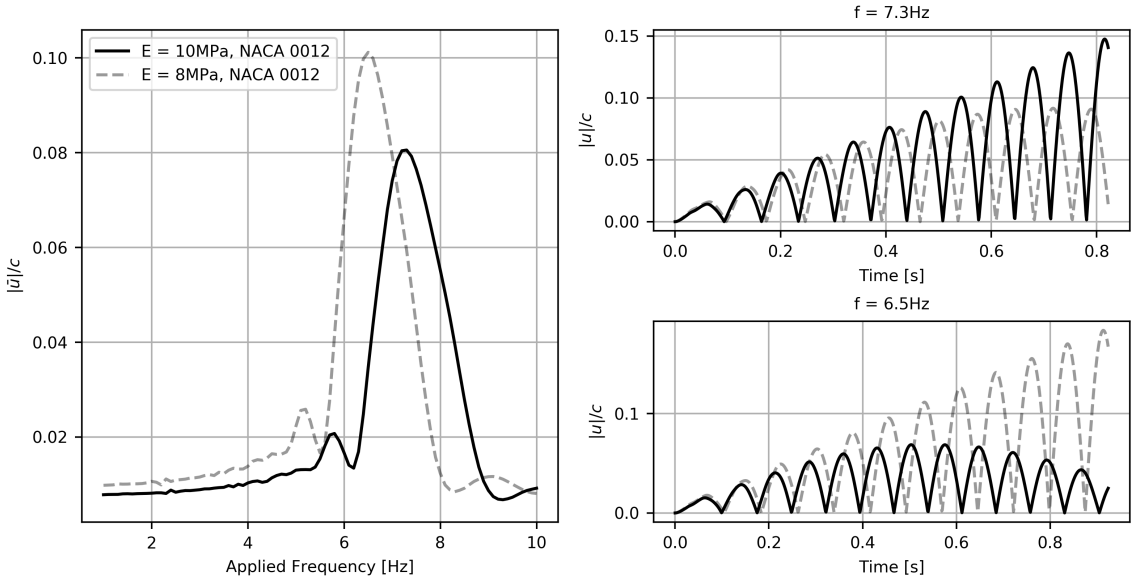


Figure 5.8: Frequency sweep of the airfoil with $E = 10$ and 8 MPa.

A constrained optimization problem is set up to maximize time-averaged magnitude of tip displacement over the final two periods, for the airfoil with the reduced stiffness of $E = 8$ MPa under sinusoidal loading at 7.3 Hz (the natural frequency for the stiffer airfoil) in order to explore whether the optimization can efficiently shift the natural frequency of the design to match the actuating frequency. A constraint of fixed airfoil area is enforced resulting in the following optimization,

$$\begin{aligned} \max \quad & J(\alpha_i) = \frac{1}{N}\sum_{i=1}^{N}\sqrt{(\mathbf{u}_i - \mathbf{u}_{ref})^{\mathsf{T}}(\mathbf{u}_i - \mathbf{u}_{ref})} \\ \text{s.t.} \quad & A_i = A_0. \end{aligned} \tag{5.3}$$

Firstly, the shape derivatives are evaluated using discrete adjoint method and finite differences

with step size of $10^{-3}c$. The results are shown in Figure 5.9 which shows sufficiently good agreement for optimization with average discrepancy of less than 2% across the design variables.

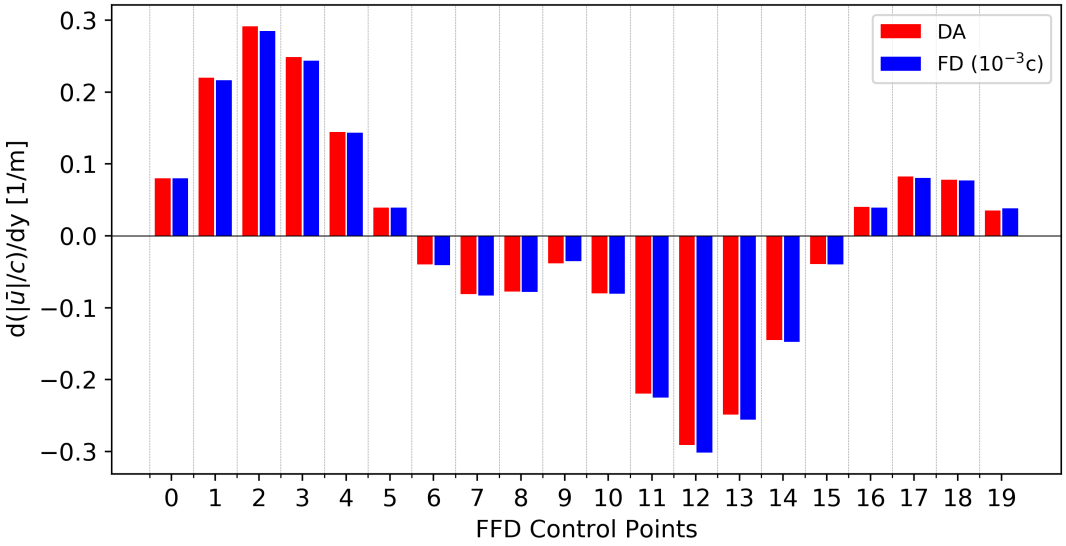


Figure 5.9: Gradient verification for the airfoil under sinusoidal load using discrete adjoint and finite differences with step size $10^{-3}c$.

The gradient-based optimization is driven by SciPy SLSQP optimizer[1] with gradients from the discrete adjoint analysis. The time-averaged objective function is evaluated for the final two periods in order to remove transient effects and the optimization history are presented in Figure 5.10 along with the optimal shape comparison. The growth in objective function can be clearly seen with a converged optimal design. The optimal shape has significantly reduced the thickness of the airfoil towards the trailing edge leaving only minimal thickness for a physical design. The airfoil area has been increased towards the leading edge and the optimizer satisfies the area constraints for all designs.

In order to increase the magnitude of tip displacement, the optimizer could have achieved this by either increasing the flexibility or through increasing the resonance characteristics. In this problem, the optimizer has found a design that captures both of these approaches. The increased flexibility at the trailing edge has led to increased tip displacement, but also a shape has been found which also capitalizes on the intrinsic dynamic structural properties to produce a peak displacement at the applied sinusoidal frequency. To highlight this, a frequency

[1] http://scipy.org

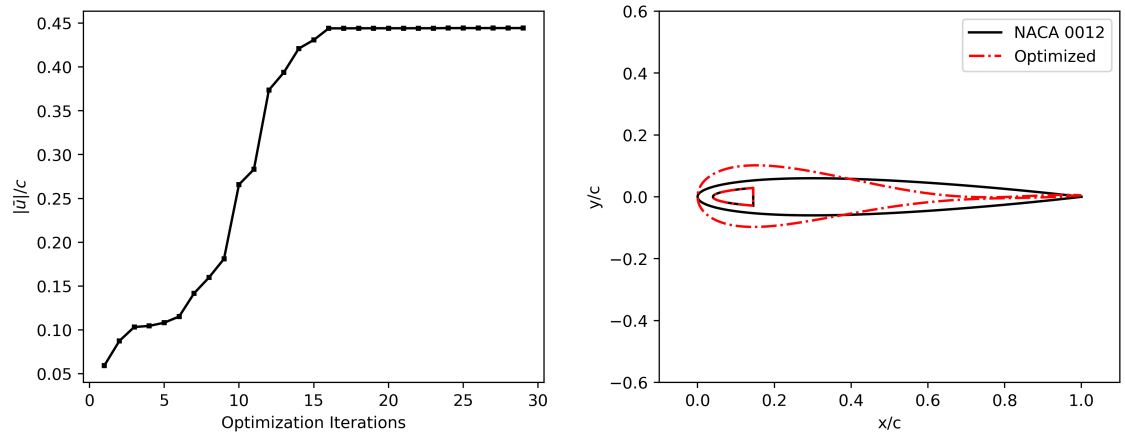


Figure 5.10: Optimization history and design comparison for NACA 0012 airfoil under sinusoidal loading with objective function calculated using final two periods.

sweep was run with the optimized shape in order to evaluate the change in dynamic structural characteristics which is shown in Figure 5.11. Despite the increase in trailing-edge flexibility, the natural frequency for the optimal design has been increased from 6.5 Hz to the applied loading frequency of 7.3 Hz. The time history of magnitude of tip displacement for the optimized shape with $E = 8$ MPa confirms that growing amplitudes are no longer seen at the original natural frequency of 6.5 Hz. The growing amplitudes for the optimized airfoil are now seen at 7.3 Hz, same as the stiffer NACA 0012, now with increased magnitudes of displacement.

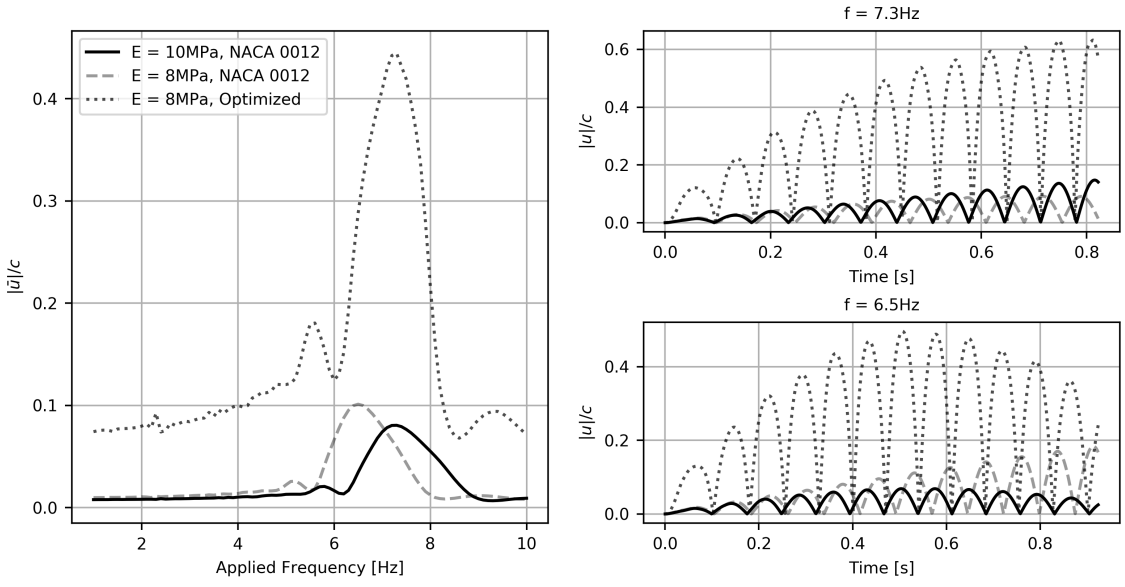


Figure 5.11: Frequency sweep of the NACA 0012 airfoil at $E = 10$ and 8 MPa and optimized airfoil at $E = 8$ MPa.

The optimization used an objective function of time-averaged tip displacement magnitude only

from the final two periods. In order to highlight the reason for this, a partial optimization history for the same problem but with the objective evaluated from all six periods are shown in Figure 5.12. Though both optimizations start similarly to each other, the influence from the transient behaviour is not controlled as well as that in the established periodic response leading to an optimization path with smaller objective function. The optimized profile also shows increased asymmetry as a result of needing to accommodate for the transient response.

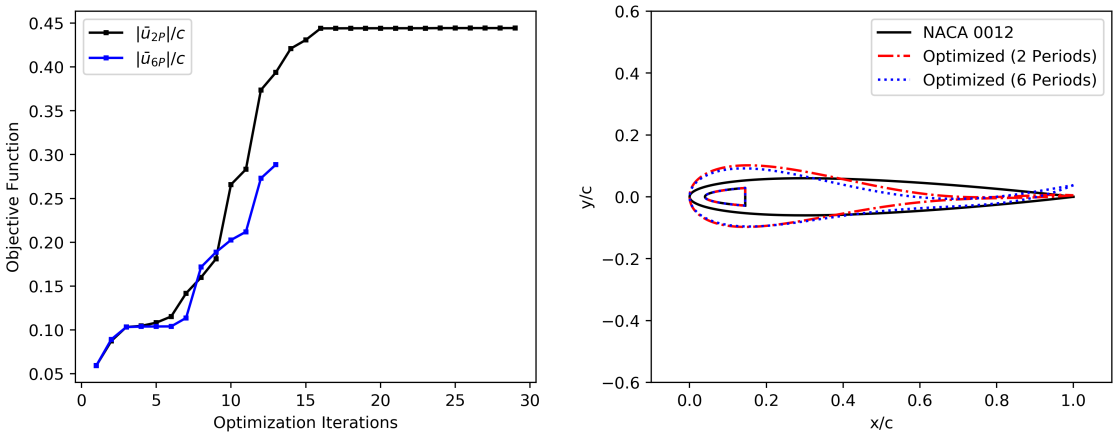


Figure 5.12: Optimization history and design comparison for NACA 0012 airfoil under sinusoidal loading with objective function calculated using final two and all six periods.

As it has been seen, the structural optimizer in SU2 identifies dynamic features of interest. This will be exploited on dynamic FSI problems in Chapter 7.

Chapter 6

Optimal Design of Dynamic Incompressible Flow Problems

This chapter focuses on the verification and validation of the new unsteady incompressible flow solver with final application for an optimal design to maximize unsteady characteristics. The use of the Arbitrary Lagrangian-Eulerian formulation of the flow solver enables problems with grid motion, and, therefore, the validation of this solver creates a pathway towards unsteady fluid-structure interaction problems.

The methodology for the unsteady primal and adjoint solvers presented in Section 2 and 3, respectively, have been implemented by the author in SU2. Implicit, 2nd order dual-time stepping method is used for all presented results. Viscous fluxes are calculated using a corrected average-gradient method. Convective fluxes can be calculated using second order numerical schemes with results to be presented using both JST and flux difference splitting scheme (FDS). Turbulent variables for the S-A and SST models are convected with a first-order scalar upwind method. Boundary conditions for the turbulence models are as given for the standard S-A and SST models on the NASA Turbulence Modeling Resource (TMR)[1].

Firstly, the primal validation is presented using three well-known benchmark cases from literature. The lid-driven cavity problem provides a preliminary validation of the incompressible

[1] https://turbmodels.larc.nasa.gov/

ALE using moving boundary condition whilst maintaining the simplicity of the steady solver. The unsteady flow solver is then validated using a laminar circular cylinder. The final validation for the primal solver introduces grid motion using a pitching NACA 0015 airfoil in turbulent flow.

Secondly, the discrete adjoint validation is presented through three test cases including the application in optimal design. The rotating airfoil in inviscid flow is a steady problem with a change in reference frame which uses grid velocity but enables the verification of the adjoint implementation using the steady solver. The verified gradients will be used for shape optimization of the rotating airfoil in order to demonstrate the optimal design framework. The verification of the discrete adjoint gradients are then presented for the unsteady circular cylinder using gradients from direct differentiation and finite differences as references. The final problem will be the gradient verification and optimal design of a plunging NACA 0002 airfoil in turbulent flow.

6.1 Primal Unsteady Incompressible Solver

Three canonical flow problems are considered in this section.

6.1.1 Lid-Driven Cavity

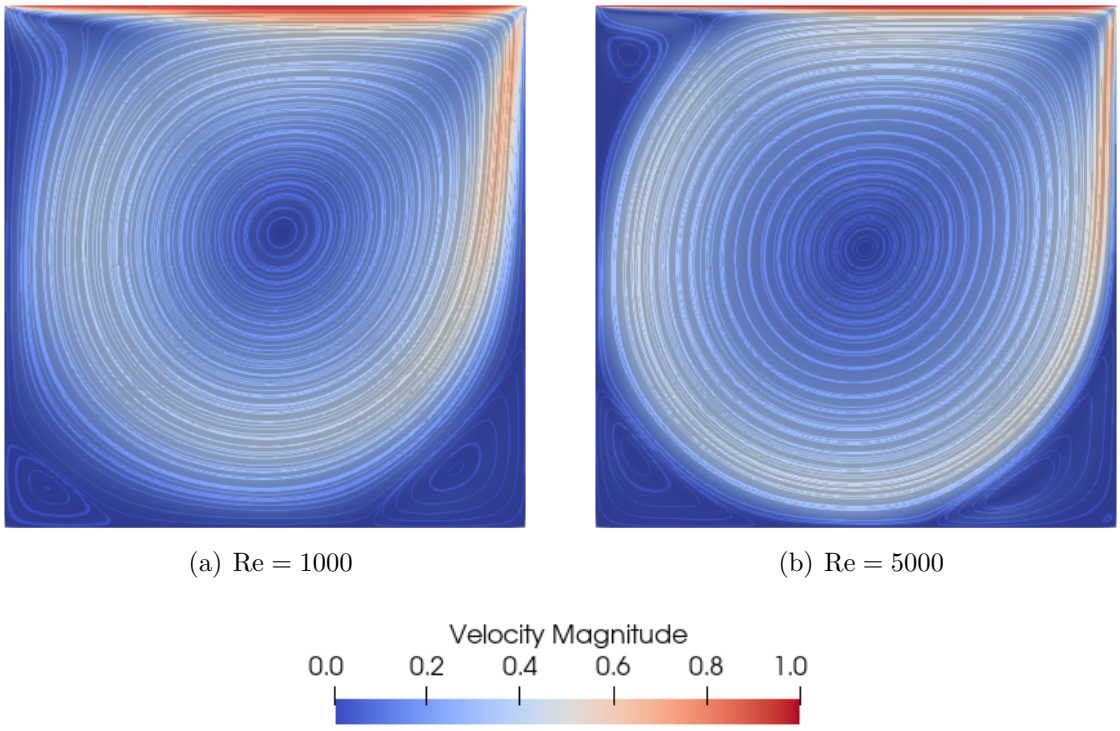
A lid-driven cavity flow is chosen as the preliminary verification of the incompressible solver. This is a well studied and documented benchmark for computational fluid dynamics with the advantage of simple geometry whilst retaining sufficient intricacies to evaluate the accurate capture of flow physics [121, 122].

For a unit square, $1D$, cavity with four solid walls, a moving wall boundary condition at the upper wall is enforced with a translation velocity equivalent to 1Hz with amplitude $1D$ to simulate the lid-driven cavity flow. The steady, laminar flow solution using the incompressible

solver is obtained to initially verify the introduction of grid velocity at the moving wall boundary condition.

A uniform grid with 600×600 elements is used with the solution found using second order flux difference splitting scheme (FDS). The maximum absolute residuals of the governing equations are converged to 10^{-10}. The lid-driven cavity flow is simulated at Re = 1000 and 5000 with comparison against reference results calculated by Erturk [122] and Ghia [121].

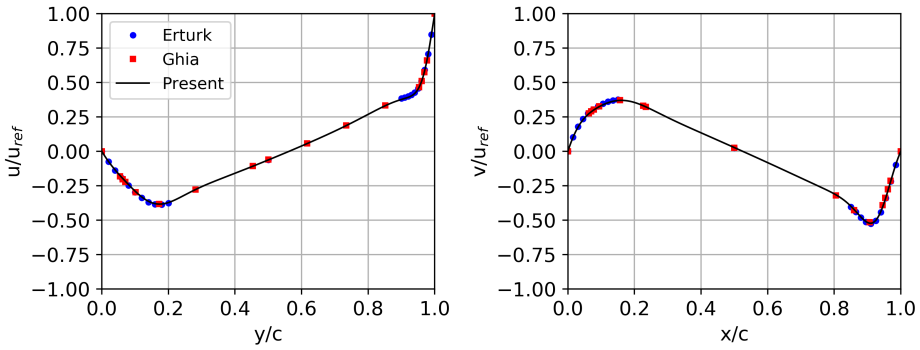
The moving boundary at the upper wall introduces a vortical flow with a linear change in velocity at the centre of the cavity and counter rotating vortices at the corners of the cavity. The accurate capture of the corner vortices is often used as a measure for the accuracy of the flow solution and thus is used to verify the present results. Firstly, the flow field results are shown in Figure 6.1 in which the primary vortical flow solution in the cavity can be seen. The secondary vortices at the bottom two corners can be clearly seen for both Reynolds numbers simulated whilst the top left corner of the cavity can also be seen to develop the secondary vortices at Re = 5000, as expected [121].



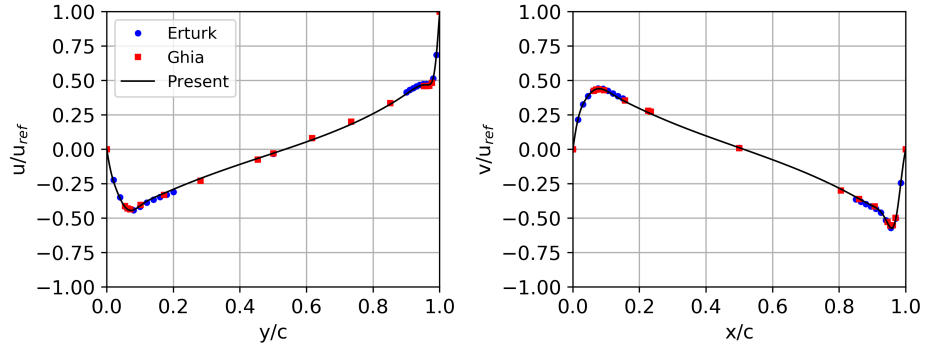
(a) Re = 1000 (b) Re = 5000

Figure 6.1: Normalized velocity magnitude contours of flow in the lid driven cavity for Re = 1000, 5000.

Centre-line slices of the cavity with vertical and horizontal velocities are used for comparison with the reference results. Figure 6.2 shows the comparison of the present results against both references [122, 121], with the horizontal velocity profiles along a vertical slice through the geometric centre on the left and the vertical velocity profiles along the horizontal slice on the right. The present results are in excellent agreement with both references at Re = 1000 as seen in Figure 6.2(a). Linear change in velocity seen near the center of the cavity in both the horizontal and vertical velocities. The velocities return to zero at the three fixed walls and the horizontal velocity at the upper wall is as enforced by the translation boundary condition. For Re = 5000, Figure 6.2(b) shows good agreement of the results with the references and captures the extended linear region of the velocities and the sharper peaks near the walls. Increasing the Reynolds number to 10000 begins to introduce small differences which can be expected since the addition of turbulence modelling and grid refinements become necessary.



(a) Re=1000



(b) Re=5000

Figure 6.2: Lid-driven cavity velocity distribution at center line for Re = 1000, 5000.

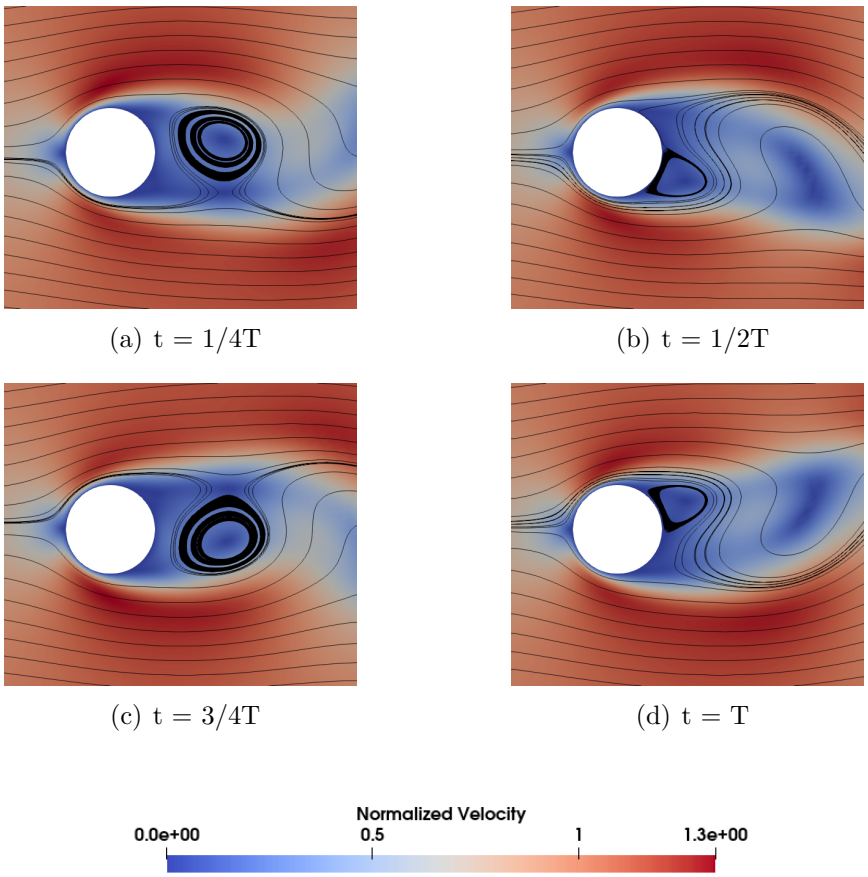
6.1.2 Laminar Flow Past a Circular Cylinder

This verification and validation exercise uses the primal solution for unsteady, incompressible, laminar flow past a circular cylinder at Re = 100. At this Reynolds number, the flow is expected to display unsteady flow characteristics with periodic vortex shedding.

A circular structured grid of radius $20D$ is used with reduced time-step, $v_\infty \Delta t/D = 0.051$. The grid has a total of 22200 elements which is known to be an appropriate size based on grids used by the benchmark references. For spatial discretization, both the second order centred method based on JST scheme [96] and upwind flux difference splitting (FDS) schemes [26] are used to confirm the robustness of the incompressible solver. The solution is converged in each time-step to satisfy a relative change of five orders of magnitude in the governing residuals.

The flow has been allowed to develop from initial conditions, with perturbations originating from the intrinsic flow solver and numerical scheme. This was to ensure that the periodic solution can be obtained without intervention. As a result, there is a long transient response with increasing unsteadiness until the formation of the Kármán vortex paths with the alternating vortices which are convected and diffused away from the cylinder. To reduce the transient duration, a perturbation can be introduced to artificially induce the quicker formation of the vortex shedding behaviour but this was not deemed necessary here. Snapshots of the periodic flow solution is shown using velocity field normalized by the free-stream velocity in Fig. 6.3, where the evolution of the vortices in one time period, $T = 1/f$, can be observed with oscillating shedding of the vortices.

The converged periodic response from the present results and reference results are summarized in Table. 6.1. The Strouhal number, $\mathrm{St} = fD/v_\infty$, is used to represent the vortex shedding characteristics. Experimental results from Roshko [123] and Tritton [124] confirm excellent agreement in St for the present results and help to verify the unsteady incompressible solver. Four computational results are also included in Table. 6.1 with further data for the mean drag along with variation in drag and lift. The present results using FDS slightly under-predicts the mean drag and variation in lift compared to the references however, overall the references verifies



(a) t = 1/4T (b) t = 1/2T

(c) t = 3/4T (d) t = T

Figure 6.3: Normalized velocity magnitude contours of flow past the circular cylinder in one period of oscillations.

the present results with very good agreement using both FDS and JST numerical methods.

6.1.3 Pitching NACA 0015 Airfoil

This test case aims to verify the unsteady incompressible solver implementation for problems with moving grids using the ALE formulation. A pitching NACA 0015 airfoil in equivalent Ma = 0.12 flow at Re = 1.5×10^6, with Spalart-Allmaras [106] turbulence model, is the chosen problem. The results from the new unsteady, incompressible implementation are compared against the solution from existing compressible solver in SU2 and experimental results by Angell [129].

Sinusoidal pitching motion was enforced for an angle of attack at 8° with an amplitude of 8°, reduced time-step, $2v_\infty\Delta t/c = 1.4$ and reduced frequency, $k = \pi fc/v_\infty = 0.179$. The choice of reduced frequency was sufficiently high to delay separation effects for the range of angle of

Table 6.1: Summarized results for the circular cylinder at Re=100.

	C_D	C_L	St
Roshko, 1954 [123]	-	-	0.167
Tritton, 1959 [124]	1.26	-	0.164
Braza et al., 1986 [125]	1.36 ± 0.015	± 0.25	0.16
Calhoun, 2002 [126]	1.35 ± 0.014	± 0.300	0.175
Russell and Wang, 2003 [127]	1.38 ± 0.007	± 0.322	0.169
Choi et al., 2007 [128]	1.34 ± 0.011	± 0.315	0.164
Present work (FDS)	1.33 ± 0.008	± 0.309	0.166
Present work (JST)	1.35 ± 0.009	± 0.316	0.167

attacks investigated here. All simulations were run, using second order JST scheme, for six periods. The time averaged solution over the final two periods are presented in the results below to exclude transient behaviour since only the converged periodic response is of interest.

A grid convergence study was performed on a series of O-grids with 100, 200, 300 and 400 airfoil surface elements with the far-field location at 50 chord lengths. The wall spacing was chosen to satisfy y^+ of less than 1, leading to grids of 14600 (coarse), 29200 (medium), 43800 (fine) and 58400 (finer) elements. The error in mean lift and drag coefficients are presented in Figure 6.4 using the solution from the finer mesh as reference. The error progression shows that lift is well converged for all grids above the coarse level. A clear convergence from coarse to finer results with the variation between fine and finer grids within 1% can also be seen for the mean drag coefficient. Therefore, the fine grid was chosen for the validation.

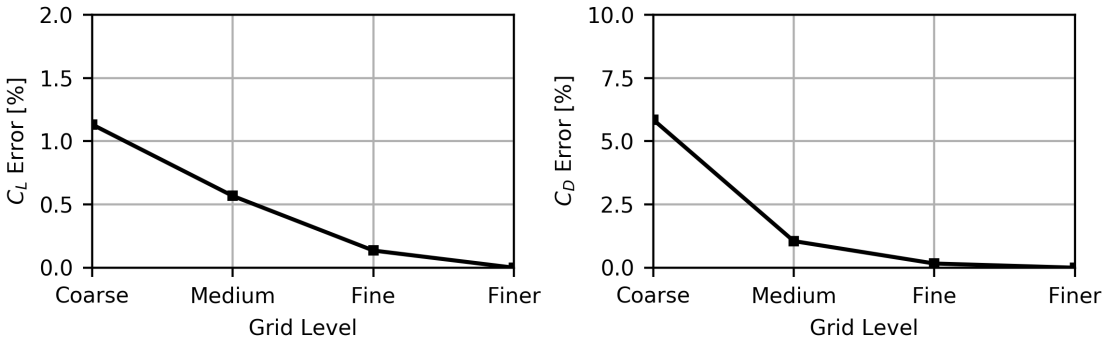


Figure 6.4: Variation of mean lift and drag for coarse, medium, fine and finer grid refinement levels for pitching NACA 0015 airfoil ($k = 0.179$, Ma $= 0.12$).

The chosen operating range of $8 \pm 8°$gives a linear lift-curve slope for high pitching frequencies

even at high angles of attack when non-linear response is observed with steady analysis. Results from the final two averaged periods of pitching motion are plotted in Figure 6.5 along with the experimental reference results [129].

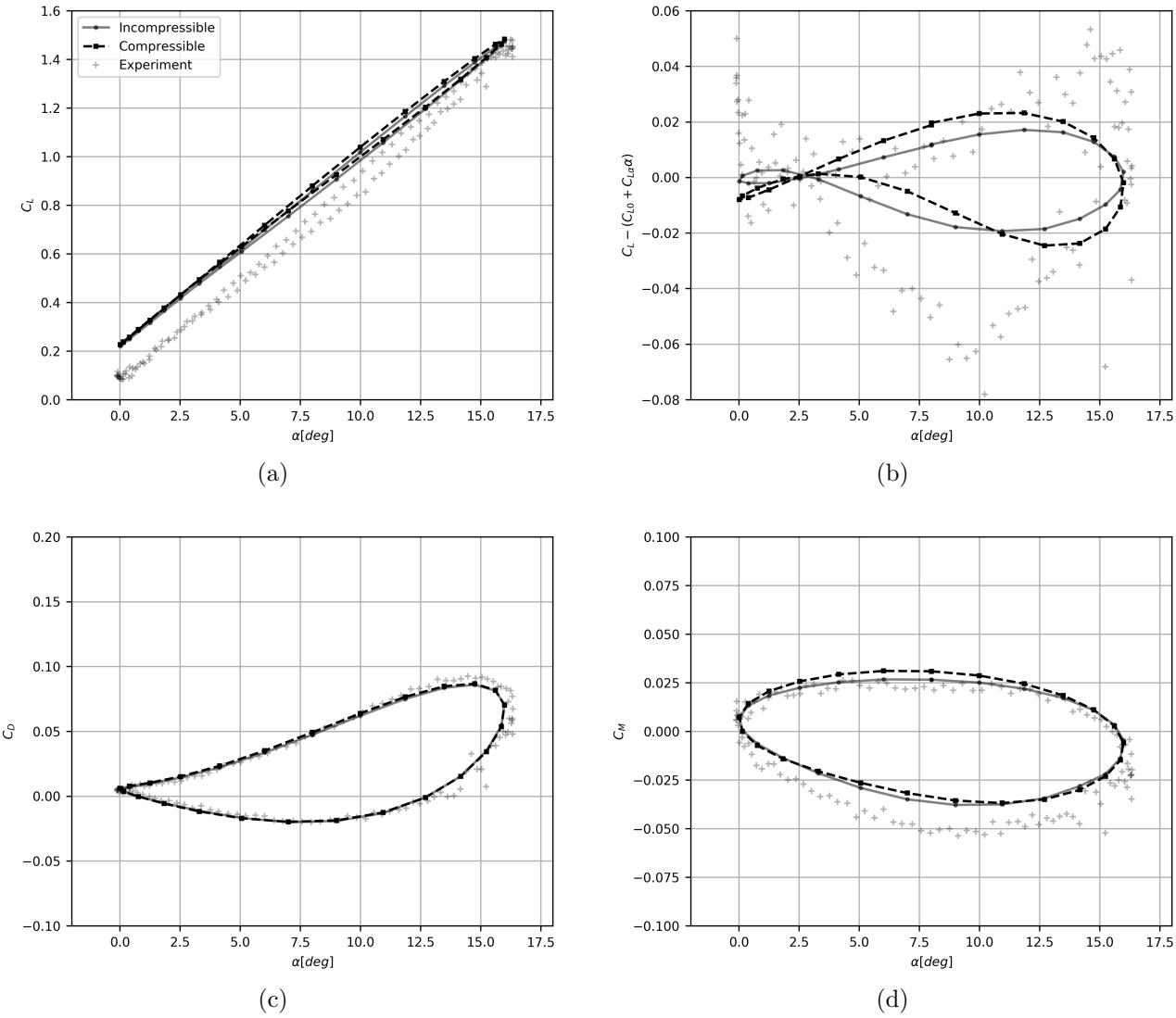


Figure 6.5: Pitching NACA 0015 airfoil with reduced frequency, $k = 0.179$ for angle of attack $8 \pm 8°$.

The computed lift curve slope agrees very well with the experiment at small angles however for larger angles hysteresis in experimental results skews the slope although we note the very small amplitude of those changes. The unsteady response in lift pivots around the steady lift coefficient at the mid angle of attack (8°) with a shallower gradient than that expected purely from steady analysis. For comparison of the agreement in the unsteady response, a linear fit was applied for each of the results, as given in Table 6.2. The time history response was subtracted

from the linear fit of the respective results in Figure 6.1.3, which removes the steady offset at 8°, between the experimental and computational results.

Table 6.2: Linear fit coefficients for unsteady lift.

	C_{L0}	$C_{L\alpha}$
Angell [129]	0.073	0.084
Compressible	0.234	0.078
Incompressible	0.221	0.078

The lift-curve slope, $C_{L\alpha}$, in Table 6.2 indicates that the new incompressible solver agrees with the compressible solver and they are both in reasonable agreement with the experimental results. The magnitude of the spread of time history results from the linear fit is small and comparable for all three results. The time history results also confirm good agreement in the tear drop profile for the unsteady drag and although the minimum moment is underestimated against the experimental reference, the elliptic profile for moment coefficient is well captured.

6.2 Discrete Adjoint Unsteady Incompressible Solver

Verification cases for the discrete adjoint sensitivities for incompressible flow problems are presented in this section. Firstly, for a rotating frame problem with the inclusion of grid velocities the sensitivities are verified and the application in shape optimization is also presented. Next, the sensitivities are verified for an unsteady laminar flow problem, before finally verifying the sensitivities for an unsteady, turbulent problem and present the optimal design results from a gradient-based shape optimization.

6.2.1 Rotating Airfoil in Inviscid Flow

The first case to verify the discrete adjoint implementation for the unsteady incompressible solver is a NACA 0015 airfoil rotating in still, inviscid flow. The free-stream velocity is zero and the airfoil in the x-y plane is rotated about the z-axis with constant velocity as seen in

Figure 6.6. The flow field is solved by transforming the governing equations to the non-inertial frame of reference which rotates with the airfoil. As a result, this can now be solved as a steady problem ($\mathrm{d}\mathbf{w}/\mathrm{d}t = 0$) with the addition of the source term,

$$\mathbf{Q} = \begin{Bmatrix} \mathbf{0} \\ -\rho(\boldsymbol{\omega} \times \mathbf{v}) \\ \mathbf{0} \end{Bmatrix}, \tag{6.1}$$

where $\boldsymbol{\omega}$ is the steady angular velocity of the airfoil. The grid velocity in Equation (2.1) is finally calculated as $\dot{\mathbf{z}}_\Omega = \boldsymbol{\omega} \times \mathbf{r}$ where $\mathbf{r}$ is the vector defining the points in the flow domain from a chosen origin of motion. Though, this is now solved as a steady problem, the inclusion of grid velocities enables the adjoint verification and optimal design using the Arbitrary Lagrangian-Eulerian implementation without the cost of the time-accurate simulation.

The rotational radius and velocity were chosen such that the airfoil motion is equivalent to Mach 0.1. An unstructured mesh with 7496 elements was used with 100 airfoil surface elements. The rotating frame solution is found using first order flux difference splitting (FDS) scheme with maximum absolute residuals of the governing equations that were less than 10^{-10}. Force coefficients are computed using free-stream density and free-stream linear velocity as a result of the rotating frame.

For the discrete adjoint sensitivities, efficiency was chosen as the objective function. Free-form deformation (FFD) box is used to parameterize the airfoil shape with 20 design variables. FFD control points 0 to 9 are defined along the lower surface whilst 10 to 18 are used for the upper surface. Sensitivities calculated using discrete adjoint (DA) method, direct differentiation (DD) using forward mode AD and finite differences (FD) with step size of $10^{-5}c$ are presented in Figure 6.6 which shows excellent agreement for all three methods.

To demonstrate the use of the adjoint sensitivities for design, a shape optimization is performed to maximize efficiency of the airfoil whilst constraining the airfoil area to be greater than or equal to the original airfoil area.

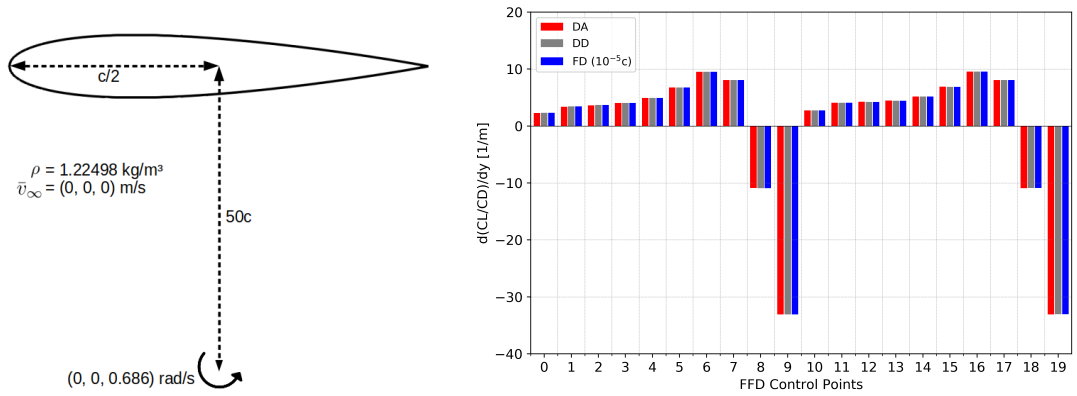


Figure 6.6: Schematic of the rotating frame NACA 0015 airfoil and the associated efficiency gradients calculated from discrete adjoint, direct differentiation and finite differences.

$$\begin{aligned} \max \quad & J(\alpha_i) = C_L/C_D \\ \text{s.t.} \quad & A_i \geq A_0. \end{aligned} \tag{6.2}$$

Figure 6.7 shows the optimization history with the evolution of the airfoil efficiency as well as the satisfaction of the airfoil area constraint.

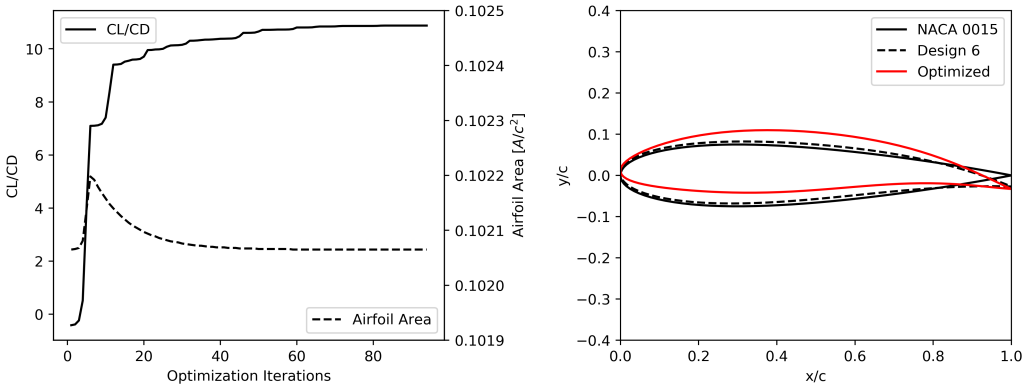


Figure 6.7: Optimization history of efficiency maximization with constrained airfoil area and the comparison of airfoil profiles for NACA 0015 with intermediate and final optimized shape.

Comparing the airfoil profiles highlight that efficiency is increased initially with the lowering of trailing edge, as seen in design 6, with subsequent gains in efficiency as a result of increased camber to yield the optimized profile.

6.2.2 Laminar Flow Past a Circular Cylinder

The second adjoint verification uses the laminar flow past a circular cylinder previously presented in 6.1.2. The time-averaged drag over one shedding period is chosen as the objective function. The simulation is restarted from the established periodic flow solution, thus removing transient characteristic in the gradient evaluations. The primal problem is run for one shedding period with the solution converged in each time-step to satisfy a relative change of 10^{-5} in drag. The adjoint problem is then run in reverse with the adjoint continuity equation converged to a relative change of four orders of magnitude in each time step.

A Free-Form Deformation (FFD) box was set up around the circular cylinder with 10 control points as shown in Figure 6.8. The geometric sensitivities from the adjoint solution are projected to the FFD control points. The sensitivities from the discrete adjoint (DA) method are plotted in Figure 6.8 together with the sensitivities calculated using direct differentiation (DD) and finite differences (FD) with step size of $10^{-6}c$ which shows excellent agreement for all sensitivities.

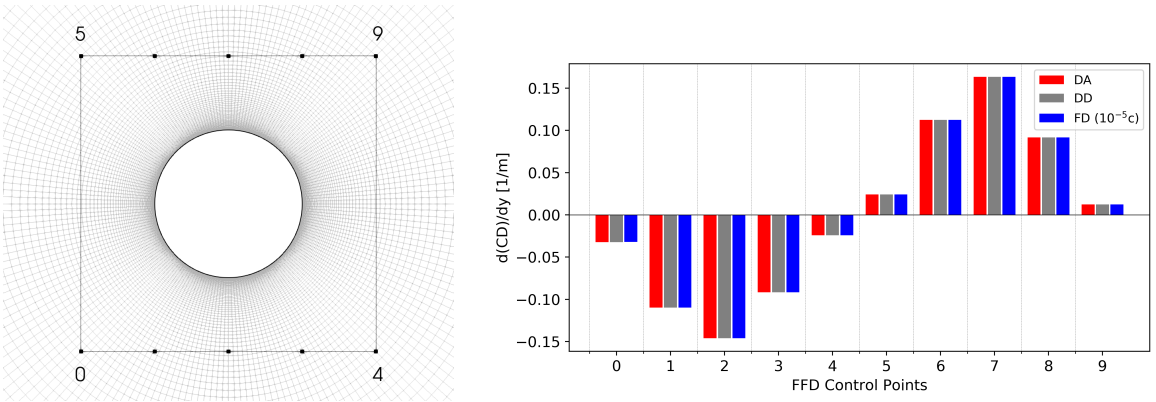


Figure 6.8: Gradient verification for an objective function of time averaged drag using a Free-Form Deformation box.

If an optimal design is set up for this problem to minimize time-averaged drag, the sensitivities of Figure 6.8 suggest reducing the frontal projected area by reducing the thickness progressively with the maximum reduction at the centre of the cylinder. However, it must be noted that the restarted solution approach will not be suitable for optimal design since the change in shape during design iterations is likely to significantly change the fundamental flow solution and, in particular, the St of the unsteady wake. As a result, if the flow is restarted and interpolated for

the new shape, the simulation is likely to need a large number of time steps before converged periodic behaviour is obtained.

6.2.3 Plunging NACA 0002 Airfoil

The final case is the optimal shape design of plunging NACA 0002 airfoil with harmonic plunge dynamics. A very thin airfoil with 2% thickness was chosen such that results can be compared with the thin plate results typically used in the literature. The airfoil is in equivalent Ma = 0.12 flow at Re = 1.5×10^6, with Spalart-Allmaras [106] turbulence model. An O-grid is used with 43800 elements including 300 airfoil surface elements which has previously been shown to be adequate from the grid convergence study for dynamic problems. It is well known that the plunging of an airfoil can generate thrust forces [130], and the goal of the optimization is to identify an airfoil shape which magnifies that process for a given amplitude and frequency of oscillations.

A sinusoidal plunging motion was enforced with reduced frequency $k = 1$ and amplitude $h_{max} = 0.01c$. The problem was simulated for two periods using 25 time steps per period, in which the transient behaviour was settled within the first period, therefore the time-averaged coefficients were calculated from the second period. The flow solution was found using second order JST scheme with maximum relative residuals for pressure being reduced by three orders of magnitude for each time-step. Firstly, the primal solution is compared against drag calculated using Garrick's potential-flow analytical solution [130] for the time-dependent induced-drag as a function of the non-dimensional plunging amplitude and frequency. The phases of one period of the plunging motion are described starting with $\phi = 0^\circ$ at the bottom of the plunge motion. Garrick's solution is offset using the delta to the viscous simulation results at $\phi = 0^\circ$, and the result is shown in Figure 6.9. The present work captures the expected behaviour and the variation in drag during the plunging motion. A difference in minimum value of the dynamic response can be seen which could be attributed to the slightly smaller viscous amplitudes in the present results.

Ahead of the optimization, the discrete adjoint sensitivities are compared against direct differ-

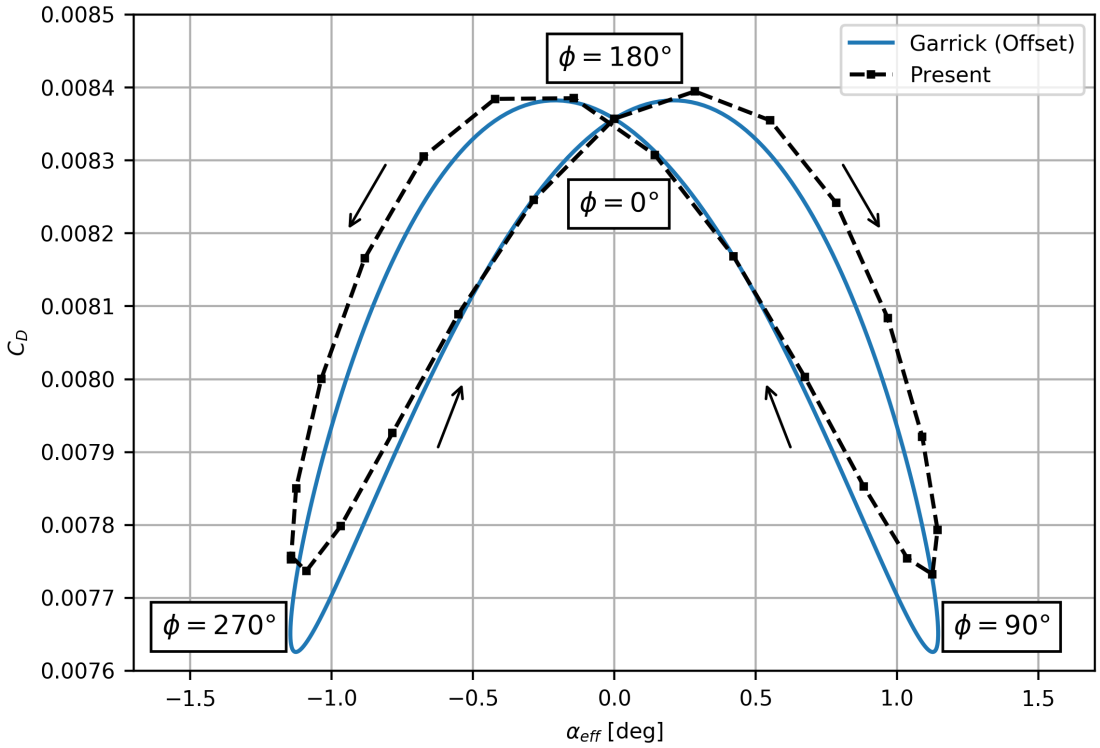


Figure 6.9: Drag during sinusoidal plunging motion for $k = 1$ and $h_{max}/c = 0.01$.

entiation using forward mode AD and finite differences with time-averaged drag as the chosen objective function. For finite differences, step size of $10^{-6}c$ was used. A free-form deformation box was set up around the airfoil with 20 FFD control points, α_i, restricted to movement in the vertical direction, resulting in design variables 0 to 9 along the lower surface from the leading edge to the trailing edge and 10 to 19 along the upper surface in the same direction. The height of the box was chosen to be $0.04c$ centered on the airfoil chord line. Compared to the circular cylinder problem previously presented, there is a significant reduction in magnitude of drag and drag sensitivity for this thin airfoil problem. Figure 6.10 shows sufficiently good agreement in sensitivities for optimal design with higher sensitivities seen near the leading and trailing edges of the airfoil.

A gradient-based shape optimization problem is finally set up as above to minimize the time-averaged drag of the airfoil whilst constraining the airfoil area to be the original airfoil area. The FFD control points are bounded to prevent intersection of the upper and lower surfaces in order to maintain physical designs.

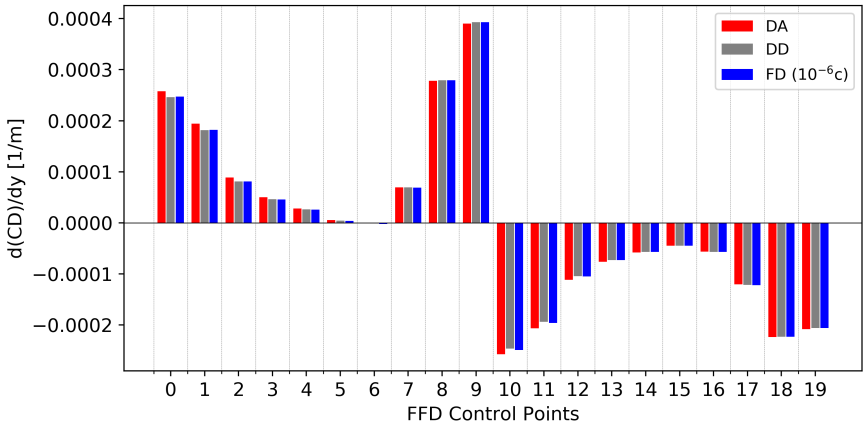


Figure 6.10: Gradient verification for an objective function of time averaged drag using design variables of FFD control points.

$$\begin{aligned} \min \quad & J(\alpha_i) = \frac{1}{N}\sum_{i=1}^{N} C_{D_i} \\ \text{s.t.} \quad & A_i = A_0 \\ & \alpha_{lower} - \alpha_{upper} < 0.04c. \end{aligned} \tag{6.3}$$

The optimization is driven by SciPy SLSQP optimizer[2] with gradients from the discrete adjoint analysis. Figure 6.11 shows the optimization history for the plunging airfoil with the time-averaged drag. The geometric constraint of fixed airfoil area was satisfied for all designs. The optimized profile has moved the peak thickness further aft with a reduction in thickness at the leading and trailing edges.

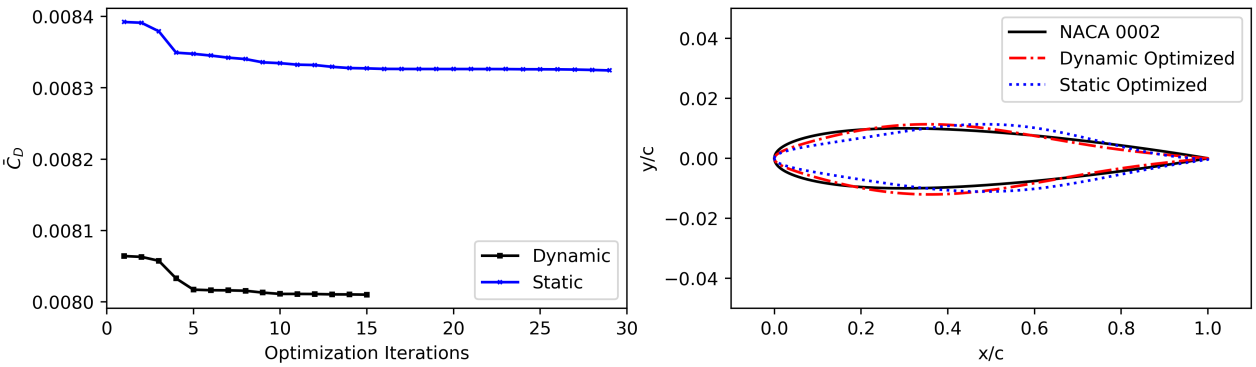


Figure 6.11: Optimization history of time-averaged drag coefficient minimization and comparison of airfoil profile for NACA 0002 and optimized profile for dynamic and static problems.

[2]http://scipy.org

A static drag minimization with the fixed area geometric constraint was also run for comparison against the dynamic results. Figure 6.11 includes the static drag minimization history and the associated optimized profile. The optimal design from the static problem increases the thickness of the airfoil and shifts the position of maximum thickness further aft compared to the unsteady optimized profile. The static problem operates at the symmetric condition with zero angle of attack, however, effective angle of attack is induced in the dynamic problem due to the plunging motion. As a result, though the sharpening of the leading edge is still the positive direction for drag minimization due to a lowered suction peak, the dynamic optimization has moved the maximum thickness location not as aft as the static optimization and the design is not as extreme in the diamond profile to maintain a slightly larger leading edge radius. Note finally that the area constraint means that there are only minimal drag reductions.

In order to highlight the importance of including the unsteady characteristics in the optimization process, the time history of the plunging motion is compared for the optimized designs from the dynamic and static drag minimization problems in Figure 6.12. The optimized profile is no longer symmetric even though the baseline angle of attack is zero, because the optimizer finds a solution which exploits the drag reduction in one half of the plunging motion. This can be seen in the dynamic optimized time history for drag in Figure 6.12. The dynamic optimization has reduced the peak drag seen at $\phi = 90^\circ$ whilst the drag at $\phi = 270^\circ$ remained similar. Though the optimal design from static optimization reduced both the steady drag and the maximum drag during the plunging motion compared to the reference NACA 0002 airfoil, this profile does not capture the improved drag at $\phi = 90^\circ$ seen by the dynamic optimized design.

Finally, Figure 6.13 shows the pressure coefficient distribution at $\phi = 0^\circ$ and 90°. The magnitude of the suction peak is significantly higher for both the original and optimized profiles at $\phi = 90^\circ$ compared to $\phi = 0^\circ$. The biggest contribution to reduce drag is in the increase of the peak suction magnitude at the leading edge where the resolved forces in the normal direction to the surface act almost exclusively along the drag axis. The optimization has clearly increased the peak suction at the leading edge compared to the NACA 0002 at $\phi = 90^\circ$. The movement of the peak thickness location towards the trailing edge has also resulted in a second delayed

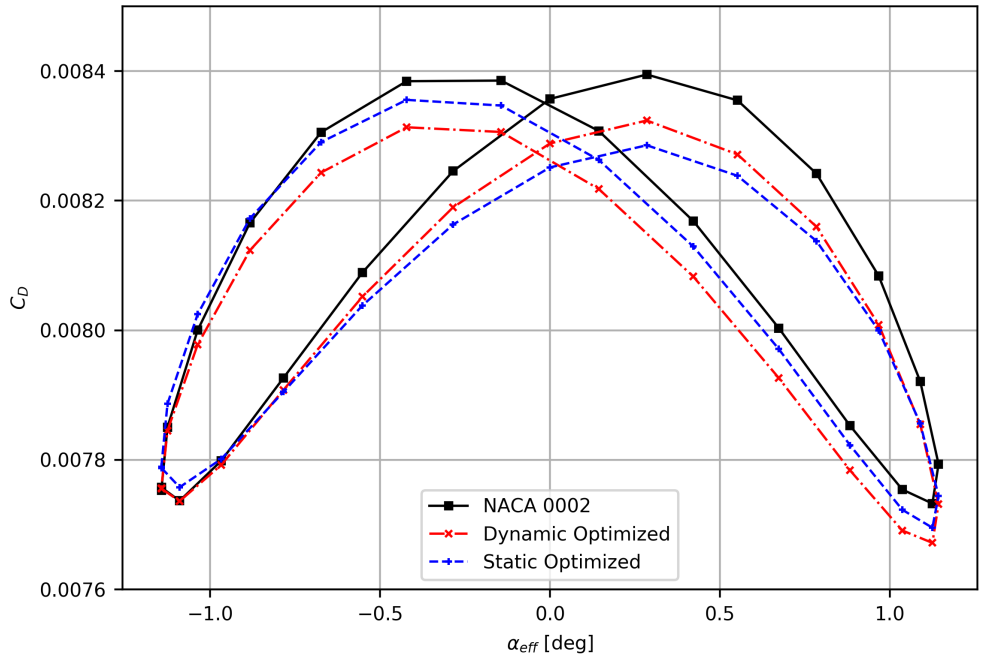


Figure 6.12: Comparison of the time history for C_D during the plunging motion with amplitude of $0.01c$ and $k = 1$ for NACA 0002, dynamic and static optimized airfoil.

increase in suction force which can be seen throughout the plunge motion.

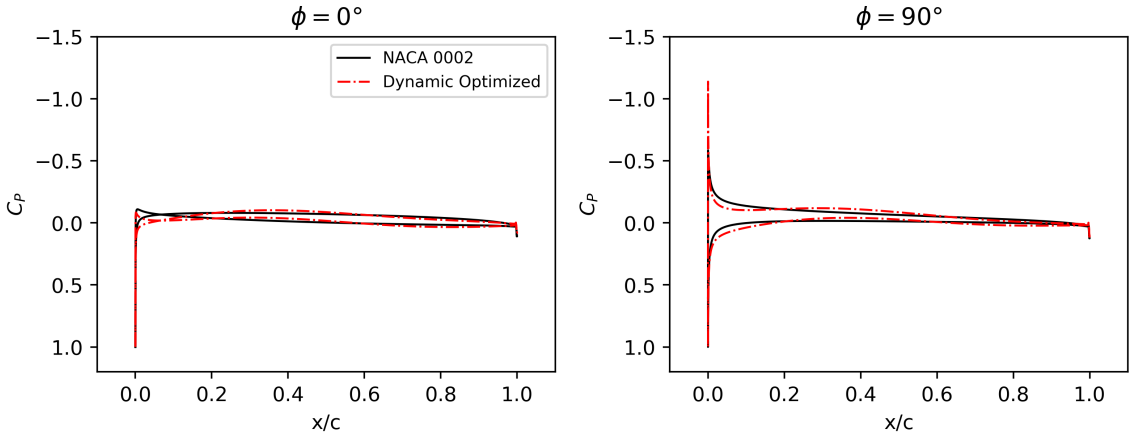


Figure 6.13: Pressure coefficient distribution for NACA 0002 and optimized profile at $\phi = 0°$, 90°.

Chapter 7

Coupled Analysis of Dynamic FSI Problems

Numerical results from the proposed coupled fluid-structure interaction methods for direct and sensitivity analysis described in Chapters 2 and 3 are presented in this chapter. This builds on the previous two chapters, where only one discipline was considered. Existing applications for coupled FSI problems using the methodology and framework in SU2 has been predominantly focused on steady problems for analysis and design. For dynamic coupled simulations, the primal solver was previously validated for compressible flows using the well-established benchmark of an elastic cantilever beam in vortical flow behind a square cylinder [22]. This chapter aims to expand and highlight the dynamic coupled problems that can be solved using the presented methodology.

Firstly, in this chapter the unsteady incompressible solver of Chapter 6 is first used for coupled dynamic FSI application, thus extending the flow regimes that can be used for coupled fluid-structure interaction problems within the single existing framework presented in SU2. The unsteady incompressible solver has been thoroughly exercised and validated for fluid only applications, including enforced grid motions. The sensitivities for the coupled problem are evaluated using the developed and implemented unsteady multizone discrete adjoint methodology and verified against finite differences.

After the initial investigation for low speed flows, aeroelastic performance in transonic flows is considered in Sections 7.2 to 7.4. Using a compliant airfoil, the dynamic aeroelastic response is evaluated for increasing structural flexibility to analyse limit cycle oscillations induced by the coupled movement of shocks and structural airfoil deformation. The discrete adjoint methodology is applied to characterize the transient response of an airfoil in transonic flow. The sensitivities from the discrete adjoint analysis are verified against finite difference gradients for multiple objective functions covering both fluid and structural objectives.

The final section of this chapter introduces gust load simulations with fully coupled fluid structure interaction analysis and design. The response to (1-cos) gusts for fluid-only problems as well as fully coupled FSI problems are presented for varying gust lengths. Both aerodynamic and structural objective functions are considered and the discrete adjoint sensitivities are verified against finite differences to characterize the design variables for coupled response to gust loads. The sensitivities are carried forward to optimal design of stiffness distribution to highlight the type of problems that the developed methodology has enabled.

7.1 Verification of Dynamic Coupled FSI for Incompressible Flow

A vortex induced vibration problem was chosen for the verification of the dynamic fluid-structure interaction primal and discrete adjoint solver for incompressible flows. The unsteady incompressible solver has been verified and validated for fluid only problems in Chapter 6, which is extended to coupled dynamic fluid-structure problems in this section.

Rigid square cylinder with a thin, long, elastic beam is a standard benchmark case originally proposed by Wall et al. [131]. It was first used in SU2 by Sanchez [22] for the compressible flow validation. The flow field is simulated in laminar flow with Re = 332 and viscosity of 1.82×10^{-5} kg/m^3. The flow solution is found using the incompressible solver with second-order upwind FDS scheme for spatial integration and second-order dual-time stepping, as described

in Equation (2.8), for time integration with time step of 0.005 s and 0.0075 s. The structural properties for this problem is defined with E = 2.5×10^5 Pa, structural density of 100 kg/m^3 and Poisson ratio of 0.35. Two mesh refinements levels were used for the fluid zone, in which the coarse mesh has 30496 fluid elements and the fine mesh has 47352 fluid elements. In both meshes, there are 1128 structural elements. The schematic for this problem is given in Figure 7.1, which also includes the inlet, outlet and slip wall boundary conditions.

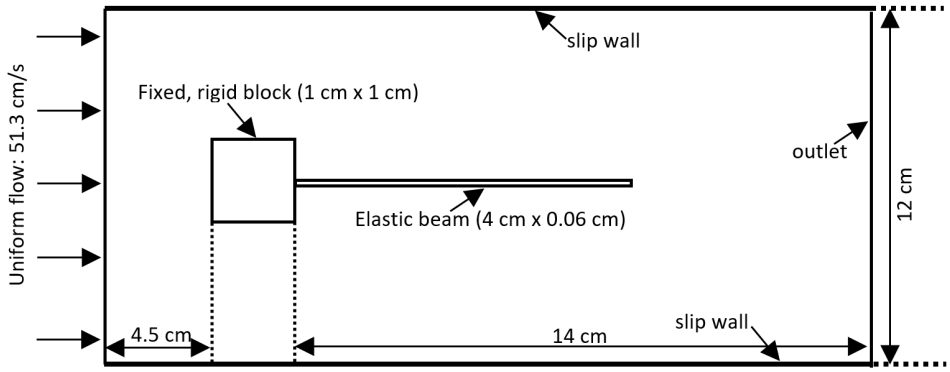


Figure 7.1: Schematic for FSI Benchmark (Not to scale).

The shedding of vortices from the square cylinder generates regions of low pressure which interacts with the elastic beam. The alternating vortex street produced by the square cylinder leads to periodic displacement of the cantilever beam. The results for the unsteady coupled problem from the initial transient response to the established periodic regime are shown in Figure 7.2.

Firstly the results using the coarse mesh indicate a transient response dominated by the fluid domain resulting in oscillations with small tip displacement which builds toward the coupled response with peak tip displacement of 1.0 cm. Using Fast-Fourier Transform (FFT) analysis to highlight peak frequencies in the simulation, the dominant frequency is identified at 3.3 Hz, with a second peak at 6 Hz resulting from the fluid response. When the finer time discretization of 0.005 s is used, the coupled response takes longer to build, leading to the peak displacement at 0.99 cm and dominant frequency of 3.2 cm. However, if the simulations are run for longer and the coupled response continued to develop for this finer time discretization, it can be expected that the results will agree exactly with the coarse time discretization.

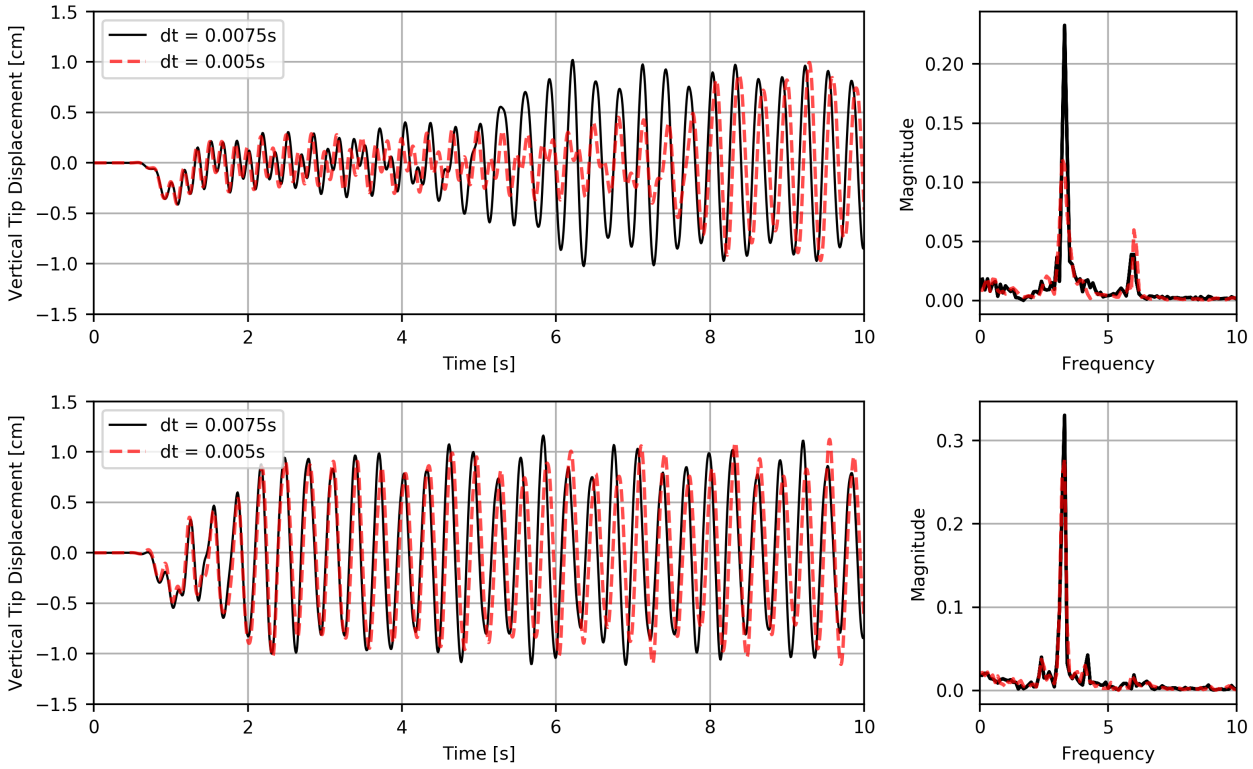


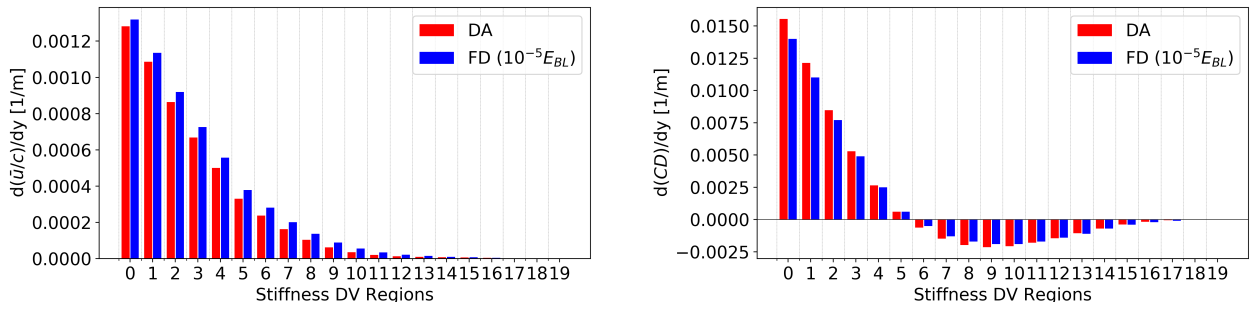
Figure 7.2: Time history for vertical tip displacement of elastic beam behind a rigid square cylinder and frequency results using FFT analysis, with coarse mesh results on top and fine mesh results at the bottom.

Moving to the fine mesh, the coupled response can be seen to be built more quickly for both time discretizations. The maximum tip displacement agrees well with the literature [131, 132, 22], which indicates tip displacement in the range of $1.0 - 1.3$ cm. The coarse mesh gives peak displacement on the lower end of the range and the fine mesh leads to the tip displacements in the middle of the range seen in literature. The FFT analysis highlight that the dominant frequency with the fine mesh is once again at 3.3 Hz and the fluid response frequency at 6 Hz is less prominent. The Strouhal number is used to characterize the frequency response. It is calculated here as $\mathrm{St} = 0.064$. The Strouhal number is also in excellent agreement with the reference results [131, 132, 22] which are in the range of $0.06 - 0.065$.

High frequency modulations can be seen in the coupled responses at all grid and time discretizations. Previously during the verification of the FSI solver using the compressible fluid solver [22], this high frequency response was seen only with the finer mesh. However, with the incompressible solver, the definition of the vortices formed and shed during the coupled re-

sponse is captured more accurately, with less dissipation, thus enabling the use of coarser grids to characterise the coupled dynamic response. This will provide a significant improvement on computational cost for time-domain FSI problems of low speed flows.

For the Fine mesh, with time step of $0.0075s$, the underlying sensitivities have been explored for the development of the periodic response in the first three seconds using stiffness distribution as a design variable. The elastic beam was separated into 20 variable stiffness regions, each with the same thickness as the beam and length of 0.05c. The design variable, α_i, acts as a multiplier of the initial uniform stiffness of the elastic beam, with α_0 at the root of the beam next to the square cylinder and α_{20} at the tip of elastic beam. The discrete adjoint sensitivities for this dynamic incompressible FSI problem are evaluated using the methodology presented in this work and verified against finite differences for time-averaged tip displacement and drag in Figure 7.3. The peak sensitivity for tip displacement is observed at the root for the elastic beam and the discrete adjoint sensitivity agrees to within 2.8% against finite differences. The consistently smaller magnitude of all sensitivities introduces higher percentage error, however good agreement in the trends of all design variables for both objective functions are found.



(a) (b)

Figure 7.3: Discrete adjoint gradient verification against finite differences (h=$10^{-5}E_{BL}$) for objective functions of time-averaged tip displacement (7.3(a)) and drag (7.3(b)).

Stiffening of the elastic beam close to the clamp will reduce the displacement which propagates further towards the tip. This is confirmed by the calculated sensitivities which emphasizes the peak sensitivities near the root. The drag sensitivities highlight peak sensitivity near the root similar to the displacement since larger tip displacements introduce larger drag values in turn. The flow solution indicates that the periodic coupled response is instigated by the formation of a low pressure region near the middle of the beam which coincides with the location of the

reduction in stiffness recommended by the drag sensitivities.

7.2 Limit Cycle Oscillations of Transonic Compliant Airfoil

Next, time accurate fluid structure interaction is simulated for inviscid flow around NACA 0012 airfoil at Ma = 0.8 and P_∞ = 101325.0 at angle of attack of 8° with time step of 2.3555ms. The fluid domain is discretized with 10216 elements and the structural mesh with 1468 elements. The structure has isotropic material properties, with material density of 2700kg/m^3, Poisson ratio $\nu = 0.35$, varying Young's modulus and it is clamped at 20% chord and under load prescribed only by the fluid tractions at the interface. Therefore, the structural deflection and movement is only a function of the coupled response of the system.

Starting with a configuration with less than 1% tip displacement using Young's modulus of 1.942GPa, the time history is shown in Figure 7.4. The time history illustrates the initial high rate of displacement from the starting NACA 0012 profile towards the steady state displacement with small amplitude, high frequency oscillations which are quickly damped.

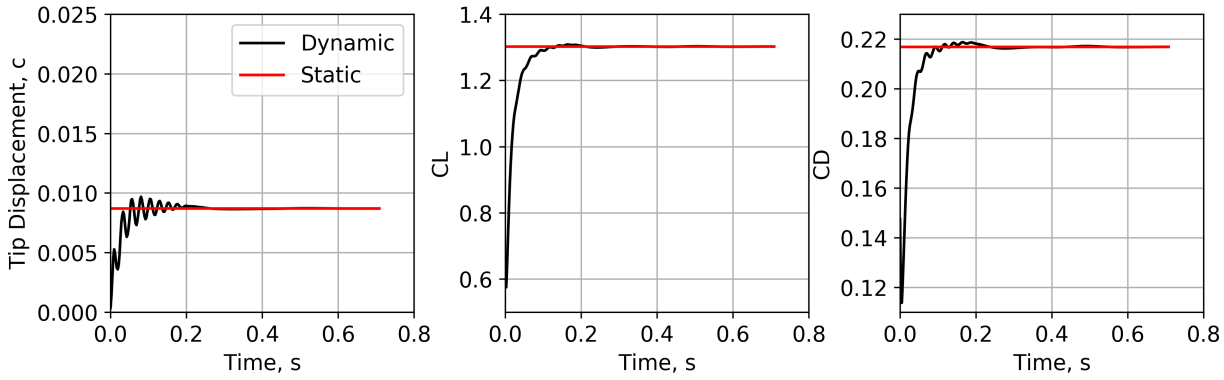
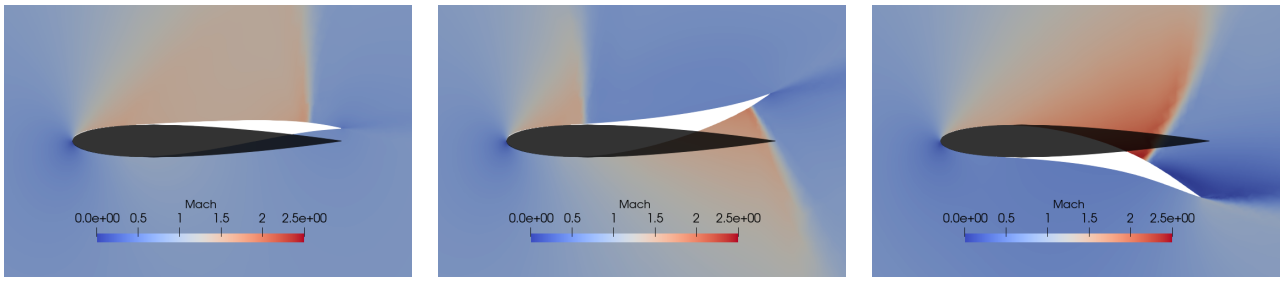


Figure 7.4: Time history results for 1.942GPa showing trailing edge displacement, lift and drag.

Increasing the flexibility of the airfoil by using E = 70MPa results in the pressure field around the airfoil which initially due to the low pressure behind the shock on the upper surface leads to an upward displacement of the trailing edge as shown in Figure 7.5(a). The resulting shape

of the airfoil in turn modifies the pressure field and thus the applied aerodynamic load on the structure leading to further deflection (Figure 7.5(b)). This strengthens the shock on the lower surface which leads to a downward trailing edge displacement shown in Figure 7.5(c). The interaction between the shock motion and the airfoil deformation introduces an oscillatory motion.



(a) $t = 0.02$s (b) $t = 0.04$s (c) $t = 0.06$s

Figure 7.5: Airfoil deflection and flow field solution at $t = 0.02, 0.04, 0.06$s for flexible airfoil with $E = 70$MPa.

With this stiffness, with the reduced level of damping, the airfoil displays flutter which grows into the limit cycle oscillations shown in Figure 7.6. The tip displacements are greatly increased to 38% chord which necessitates the non-linear structural solver used in this methodology. The tip displacement and in turn the lift oscillates at a frequency of 11.3Hz whilst drag oscillates as expected at twice that frequency.

Varying the airfoil stiffness varies the natural bending frequencies of the airfoil structure whilst the effect of the coupled dynamics also modifies the associated damping imposed on the oscillations for that configuration. Therefore, continuing to increase the flexibility by halving the stiffness to $E = 35$MPa increases the tip displacement, to 48% chord. The time history and FFT results shown in Figure 7.7, illustrate the increase in tip displacement. However the system still displays limit cycle oscillations in its coupled response, and the frequency of the first mode is reduced as expected with the decrease in structural stiffness. As the tip displacement amplitudes are increased, a more complex aerodynamic environment is created which is shown by the increased noise at the peaks of the aerodynamic response. Viscous effects would also be of high relevance here, therefore this limits further analysis using the inviscid problem definition.

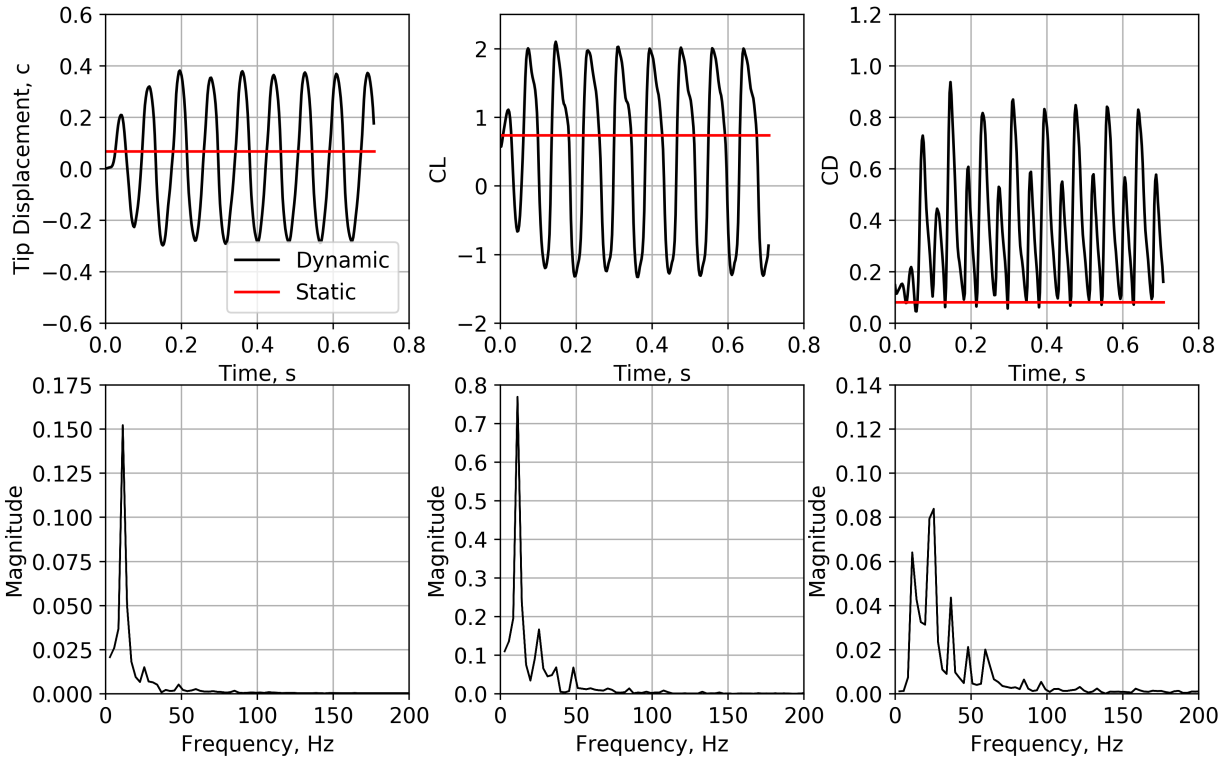


Figure 7.6: Time history and frequency results for $E = 70$MPa including trailing edge displacement, lift and drag.

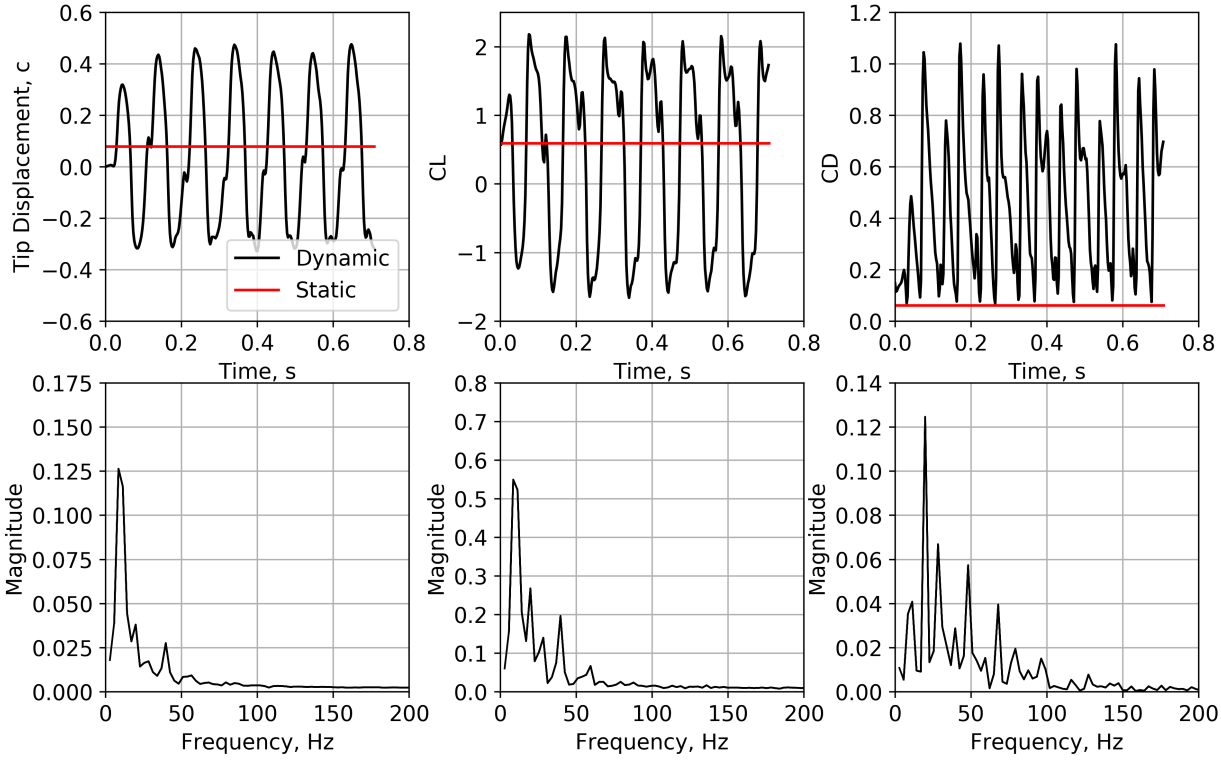


Figure 7.7: Time history and frequency results for $E = 35$MPa including trailing edge displacement, lift and drag.

7.3 Discrete Adjoint Sensitivities for Transient Response of Transonic Compliant Airfoil

The discrete adjoint sensitivities of dynamic coupled FSI problems using the formulation of Section 3.3 will be verified using a compliant NACA 0012 airfoil at Ma = 0.8. For this verification, a simple problem, whilst still exercising the non-linear solver, is chosen by considering the transient response of a NACA 0012 airfoil. The transient response, for the initial 0.1s, as the coupled problem develops towards the limit cycle oscillations previously discussed in Section 7.2 is evaluated. For the sensitivity analysis, only the initial transient response is considered for convenience and the problem is redefined for peak tip displacement in the order of 5%c. The coupled problem is defined with airfoil stiffness, E_{BL}, and angle of attack are defined as 20 MPa and 1°, respectively. The airfoil is separated into 20 design variable segments with stiffness multiplier of each segment as design variables, as previously in Section 4.1, with 10 segments on the lower and upper halves, as shown in Figure 7.8. The fluid and structural grids from Section 4.1 with 10216 fluid elements and 550 structural elements are employed again with a rigid clamp in the leading edge.

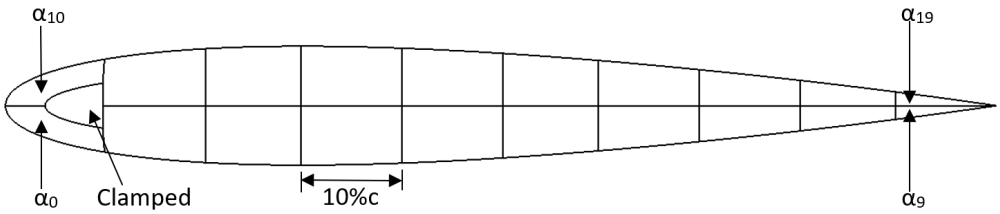


Figure 7.8: Schematic of NACA 0012 airfoil with segmented design variables $\alpha_i, i = 0, ..., 19$.

The time history of drag, lift and aerodynamic efficiency for this problem are presented in Figure 7.9 along with magnitude of the tip displacement. For the gradient verification, the objective functions are the time averaged value of each of these four parameters, chosen in turn, in order to consider both aerodynamic and structural objective functions. The discrete adjoint (DA) sensitivities are compared against gradients calculated using finite differences (FD) with step size of 200Pa, which equates to $10^{-5}E_{BL}$. The gradients for all four objective functions with respect to the stiffness design variables are presented in Figure 7.10.

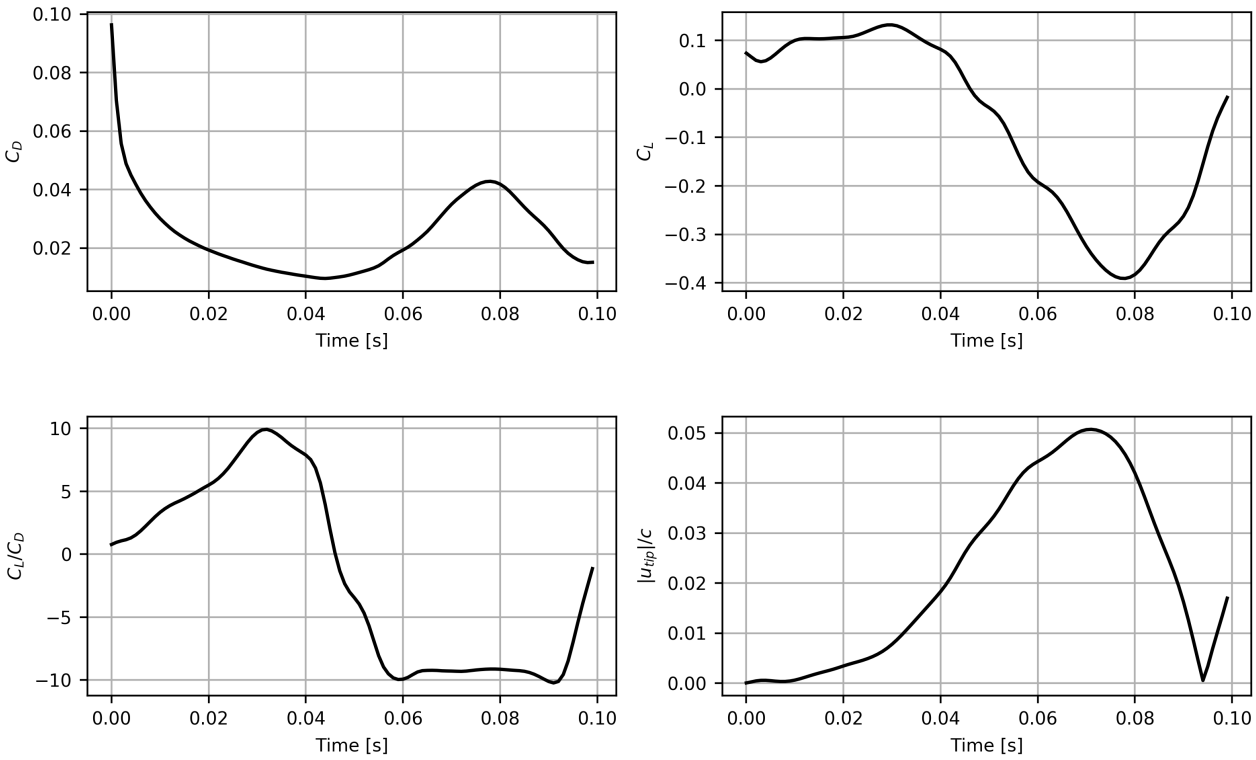
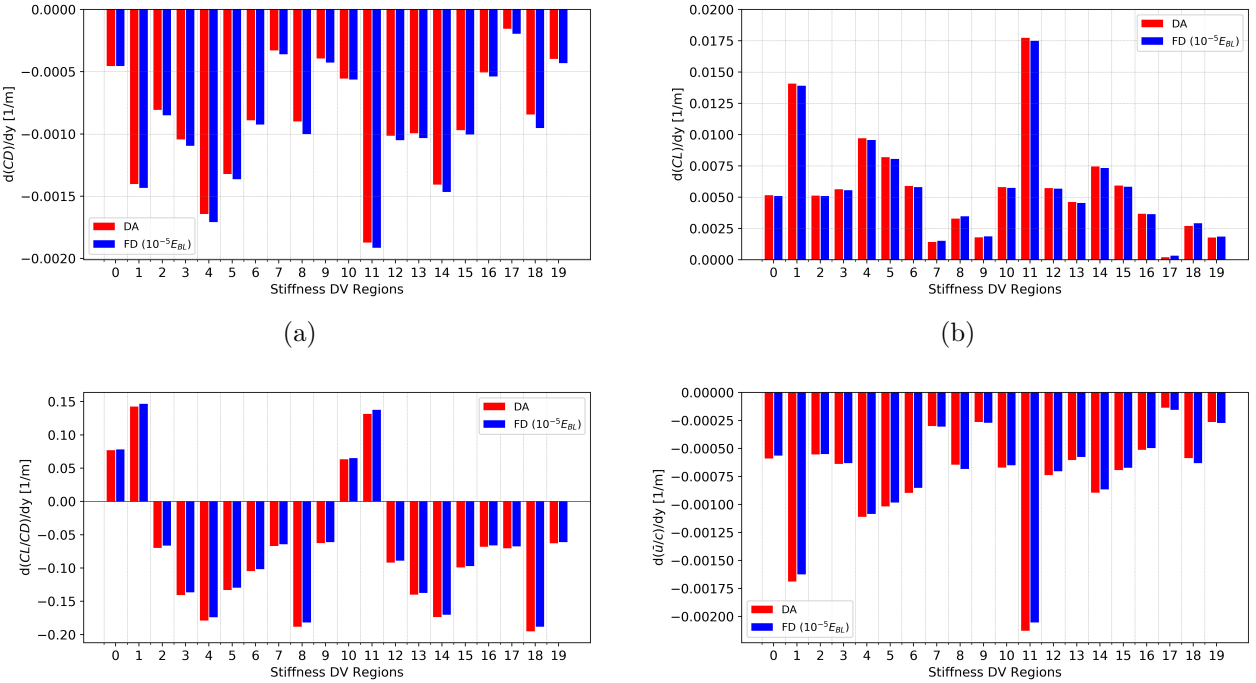


Figure 7.9: Time history results for the primal transient response of the NACA 0012 airfoil.



(a) (b) (c) (d)

Figure 7.10: Discrete adjoint gradient validation against finite differences (h=$10^{-5}E_{BL}$) for objective functions of time-averaged drag (7.10(a)), lift (7.10(b)), efficiency (7.10(c)) and tip displacement (7.10(d)).

Beginning with the time-averaged aerodynamic objective functions, the sensitivities for the lower and upper surfaces, segments 0 to 9 and 10 to 19 respectively, show similar trends however the magnitudes are not symmetric due to the non-zero angle of attack of the aeroelastic problem. To consider a structural objective function, the tip displacement was chosen and the discrete adjoint analysis highlights peak sensitivity at the segment behind the clamped boundary condition which can be expected since the root deformation at the clamp will have a large compounded impact on the deformation further aft.

The comparison of the sensitivities, for all objective functions, show that the discrete adjoint sensitivities agree well with that from finite differences, with less than 5% error on average over all the design variables. Biggest variations are seen near the trailing edge for the aerodynamic objective functions whilst the tip displacement has a larger variation near the clamp.

The dynamic coupled problem has various dependencies arising from the grid motion and fluid tractions at the surface at each time step which need to be differentiated in addition to the evaluations within the fluid and structural domains. One of the changes introduced was the explicit enforcement of the continuity of boundary grid velocities at the interface in addition to the boundary displacements, both of which are now propagated to the rest of the fluid domain using the linear-elastic approach. The calculation of the grid velocities through backward differences using the grid locations at current and previous time steps in the dynamic FSI problem have previously been shown to be satisfactory for the verification of the primal coupled response [22]. However, for the discrete adjoint sensitivities, this method led to an average error of 26% across the four objective functions considered. Whereas, by transferring the structural velocities at the interface to the fluid grid nodes, this dependency and the resulting coupled response are better captured in the discrete adjoint evaluation leading to the improved agreement of sensitivities.

The sensitivities are also dependent on good convergence of the coupled adjoint system which in turn relies on good convergence of the primal problem at every time step. Small errors at one time can be exaggerated in the time-accurate simulations, since the adjoint variables continue to influence the solutions at the following time steps through the time-dependent source terms.

Even though computing accurate sensitivities through discrete adjoint method for dynamic FSI is challenging, the results in Figure 7.10 have verified this method for objective functions across both structural and fluid domains. Therefore, the discrete adjoint sensitivities can be used for optimal design of dynamic coupled problems. The advantage of reduced cost from the discrete adjoint method will significantly improve the viability of the use of these high-fidelity methods for gradient based optimization.

7.4 Gust-Load Simulation with Fully Coupled FSI

Finally, the coupled FSI response to gust load is considered as a representative problem for typical application of the new methods developed in this thesis. In the literature, gust load simulation and analysis is typically performed using lower fidelity methods [133]. A reduced order model of the structural problem is typically used with focus on control surface adjustments for their contributions to gust load alleviation. This is a well studied problem for both analysis [134] as well as design of gust load alleviation controllers for flexible aircraft [135, 136]. However, these existing methods do not support fully coupled analysis and design of structural and shape properties. In SU2, gust load simulations were previously performed to provide a higher fidelity alternative to vortex-lattice methods for improved accuracy of the effect of gust on maximum forces in the existing multidisciplinary frameworks for gust analysis [101, 102]. With the developments made to the structural and coupled FSI solver for dynamic analysis and design in SU2, aeroelastic gust load problems can now be considered entirely with high-fidelity methods.

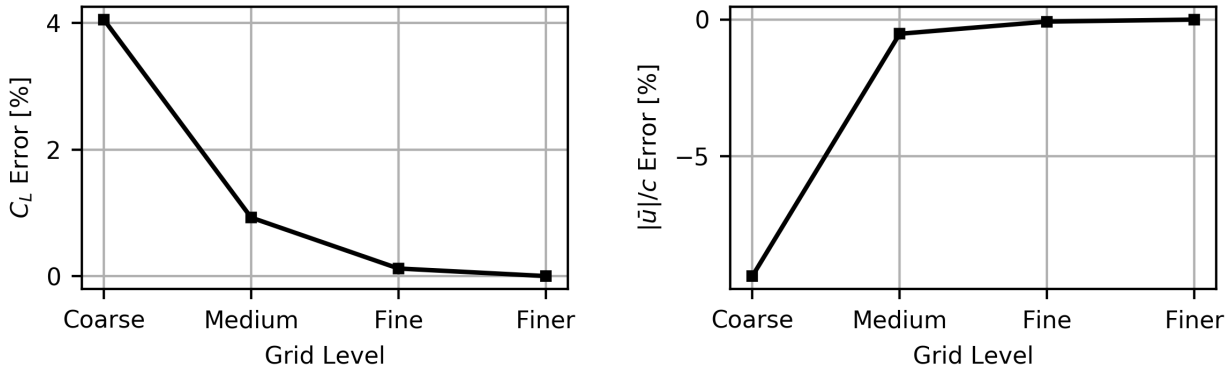
The gust load is simulated by specifying grid velocities at the fluid nodes, as required by the choice of gust and varies in time to simulate the movement of gust. For the coupled FSI problem with deforming grids, the grid velocities as dictated by the coupled solution are initially calculated. The grid velocity contribution from the gust is then imposed on the grid velocity field in an additive process.

To demonstrate the fully coupled gust load simulations, a NACA 0012 airfoil under varying

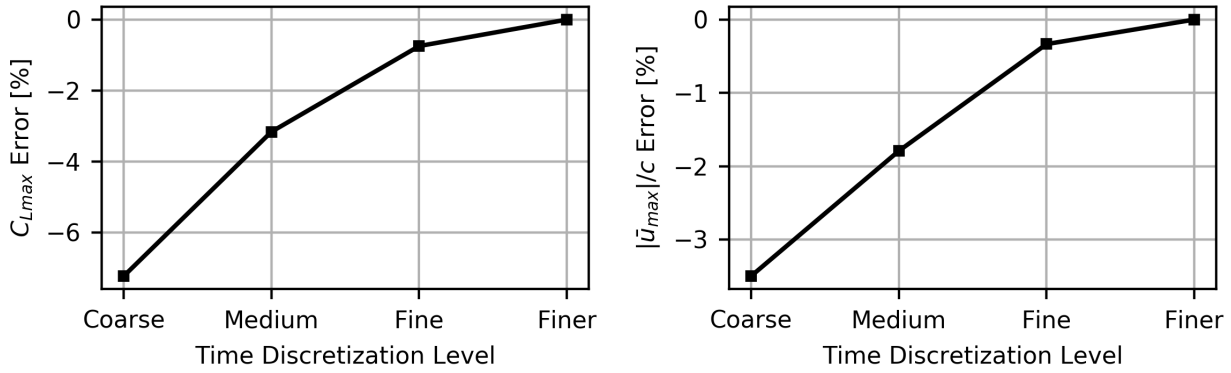
gust load simulations is presented. The problem is defined for the airfoil at Ma = 0.3, angle of attack of 5°. The gusts were defined with amplitude of 30m/s and gust lengths, H, ranging from $2.5c$ to $25c$. The gust length, H, is defined such that the wave length of the "1-cos" gust is $L = 2H$. The airfoil has uniform structural stiffness of 10 MPa and density of 300 kg/m^3. The airfoil has an internal clamp and segmented design variable regions for variable stiffness distribution as described previously in Figure 7.8. A second order dual-time stepping strategy is employed for the fluid domain and Newmark time integration for the structural domain with Newmark constants chosen to provide 2% numerical damping. A grid and time convergence study was firstly performed with steady lift and tip displacement considered for the grid level and time-averaged lift and tip displacement for the time discretization under $H = 5c$ gust loading. The dynamic gust load problems are restarted from the steady solution to reduce the computational cost in the solution prior to gust, since the transient behaviour prior to gust is not considered here. The delayed start of gust is applied such that the initial iterations of the simulation are used to confirm the stable starting point for subsequent dynamics.

The grid size for both the fluid and structural domains were increased together from 4122 fluid elements, 504 structural elements up to 28912 fluid elements and 3514 structural elements. Figure 7.11(a) confirms that the Fine grid level with 13350 fluid elements and 1790 structural elements agree to within 1% of both lift and tip displacement. For the time convergence, the reduced time steps were halved in each time discretization with the Fine refinement using reduced time step, $\Delta s = v_\infty \Delta t/(c/2) = 0.398$, identified from Figure 7.11(b) as a suitable value with less than 1% error from the further halved reduced time step.

Firstly, the gust load simulations for the fluid problem are considered on a rigid airfoil. The gust length is varied from $2.5c$ to $25c$ and the results for the variation in C_D and C_L are plotted in Figure 7.12 which display the expected response from literature [102]. The drag from the gust response is higher for short gust length but stabilizes to the same level for gusts above $H = 10c$. The lift increases in magnitude with the increasing gust length. The difference in time for the peak lift compared to the peak drag is known as the aerodynamic lag. The response to the shorter gusts are more affected by this lag, thus results in lower peak lift despite the same gust amplitude for all simulations.



(a)



(b)

Figure 7.11: Grid and time convergence for lift and tip displacement on four refinement levels

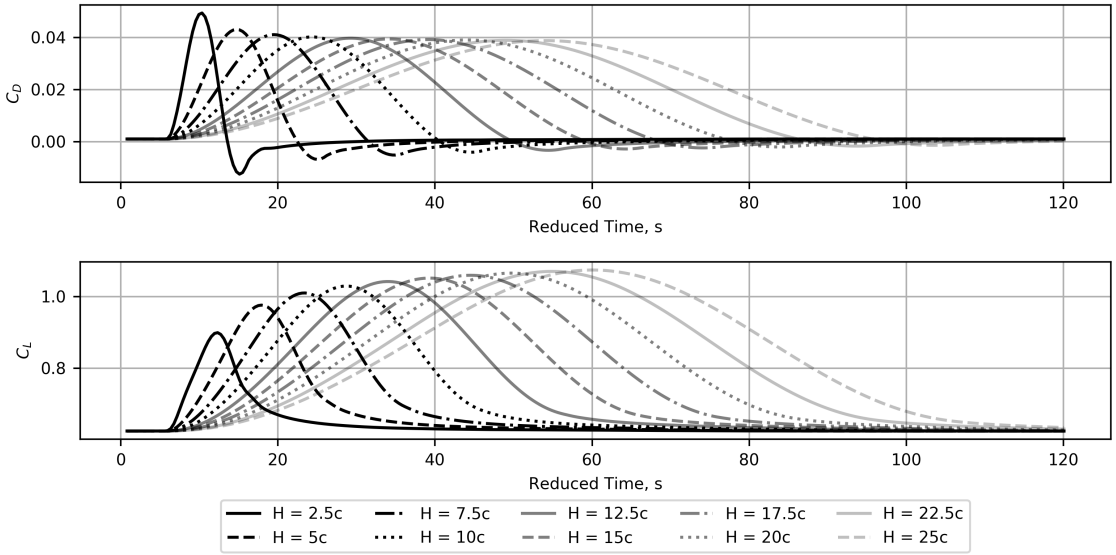


Figure 7.12: C_L and C_D on a rigid airfoil for varying gusts of length H from $2.5c$ to $25c$.

Moving to coupled simulations under gust loads, Figure 7.13 shows that the peak drag is higher for shorter gust lengths, which reduces as the gust length increases similar to that seen for the

fluid only problem, but at a slightly slower rate. The steady lift for the coupled problem is lower compared to the fluid only solution due to the upward displacement of the trailing edge hence the two profiles have different starting points in the time history. As the lift begins to build due to the gust load, the structure deforms initially with a small upward tip displacement which is followed by a larger downward displacement and a sinusoidal response is triggered. As a result, the lift profile for the coupled problem is significantly different to rigid airfoil results presented previously from the fluid only simulations in Figure 7.12. The tip displacement history highlights that though the shorter gusts trigger a larger structural response, the displacements are well damped by the coupled aerodynamic damping.

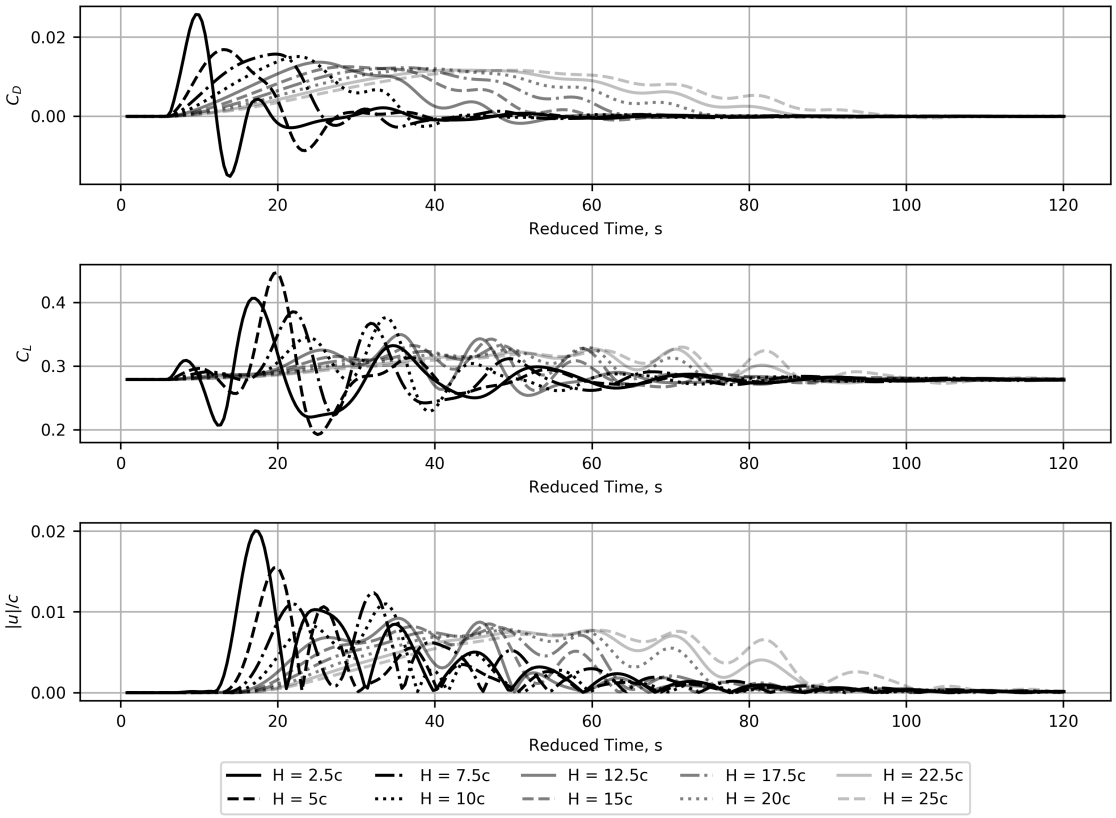
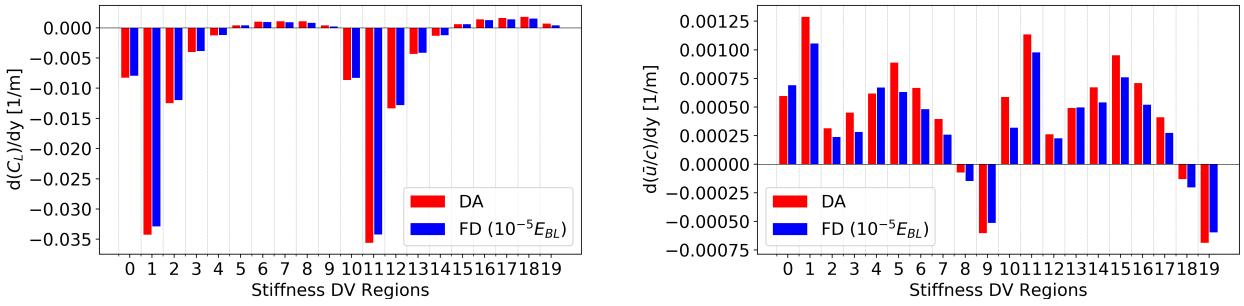


Figure 7.13: Coupled response of C_D, C_L and $|u|/c$ for varying gusts of length H from $2.5c$ to $25c$.

The gust load with $H = 5c$ had the second highest tip displacement and largest range in lift observed in the study and was chosen for sensitivity verification. Since the transient response to gust was of most interest, the problem was limited to 40s reduced time. Using 20 design variables as previously described in Figure 7.8 with α_0 at the lower leading edge and α_{19} at

the upper trailing edge, sensitivities to time-averaged lift and tip displacement was evaluated using both the newly implemented discrete adjoint method as well as the traditional finite differences. Figure 7.14 shows that the discrete adjoint sensitivities are in very good agreement against finite differences, with 4% error in peak lift sensitivity. The trends for both lift and tip displacement are well captured, with the magnitudes in better agreement for lift which has sensitivity values of an order of magnitudes higher than that for tip displacement. Peak values in sensitivity is clearly seen for lift at α_1 and α_{11} which is adjacent to the clamp on the lower and upper sections of the airfoil. For tip displacement, though the peak sensitivities are still identified as α_1 and α_{11} in Figure 7.14(b), there is larger sensitivity towards the trailing edge.



(a) (b)

Figure 7.14: Discrete adjoint gradient verification against finite differences (h=$10^{-5}E_{BL}$) for objective functions of time-averaged lift (7.14(a)) and tip displacement (7.14(b)).

The application of the discrete adjoint sensitivities for optimal design is explored by considering an aeroelastic optimization to minimize lift whilst constrained to be greater than or equal to the lift prior to gust in the initial design. A lower bound is enforced for the stiffness multiplier design variable to maintain feasible designs.

$$\begin{aligned} \min \quad & J(\alpha_i) = \frac{1}{N}\sum_{i=1}^{N} C_{L_i} \\ \text{s.t.} \quad & \bar{C}_{L_i} \geq C_{L_0} \\ & \alpha_i \geq 0.01. \end{aligned} \tag{7.1}$$

For each design iteration, the steady solution prior to gust is evaluated and the gust load solution restarted from the corresponding solution. The optimization is driven once again by sequential least squares quadratic programming (SLSQP) method implemented in SciPy [120].

The sensitivities are provided by the discrete adjoint evaluations and the optimization history is presented in Figure 7.15. As seen from the sensitivity analysis, the optimization has reduced the stiffness adjacent to the internal clamp. This introduces a tip displacement in the optimization designs compared to the initial design prior before a gust load is introduced. The coupled response from the greater tip displacement is a reduction in lift which can be seen as an offset in lift at the beginning of the lift time history of the optimized design. Though the minimum lift constraint was set to prevent the optimization from simply creating a downward shift of the lift profile, the efficient reduction in time-averaged lift was still predominantly achieved by the optimization in this way. The peak lift was similar to the initial design, resulting in a larger range in lift for the optimized design.

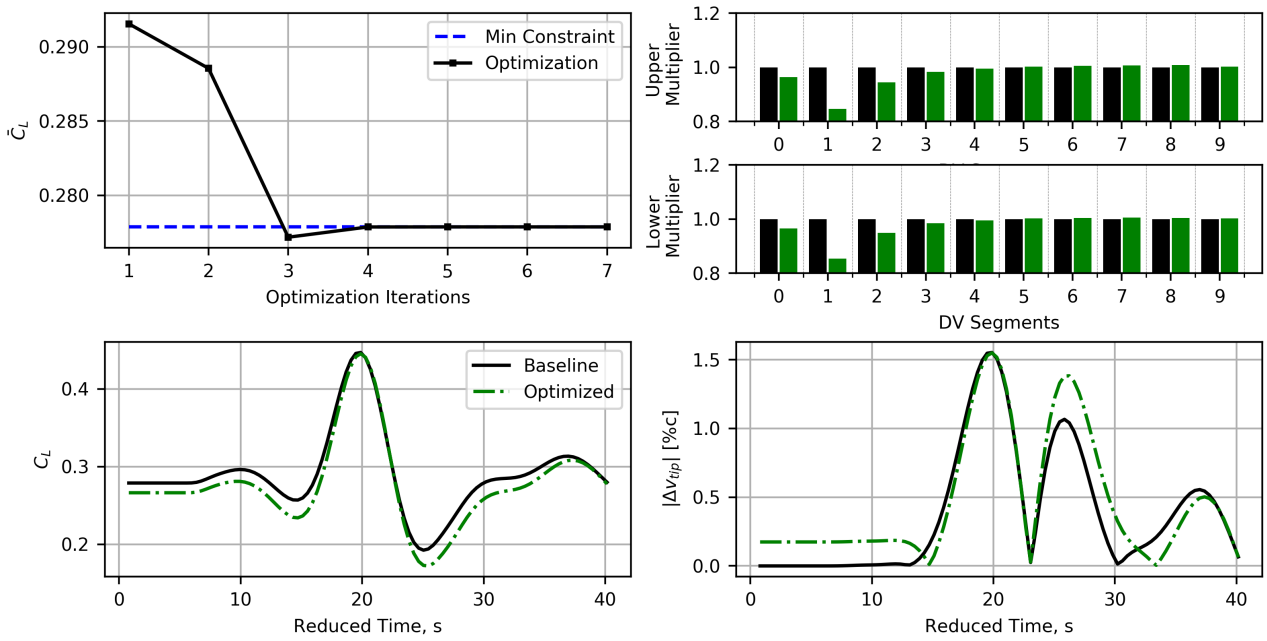


Figure 7.15: Optimization history for lift minimization of compliant NACA 0012 airfoil under gust load $H = 5c$ with design variables of stiffness distribution.

Therefore, the problem was recast as a minimization of magnitude of time-averaged tip displacement. The reference displacement is the steady tip displacement of the initial design prior to gust. The minimum lift constraint was removed and the optimization was subject only to the lower bound for the stiffness design variable to maintain feasible designs.

$$\begin{aligned} \min \quad & J(\alpha_i) = \frac{1}{N}\sum_{i=1}^{N}\sqrt{(\mathbf{u}_i - \mathbf{u}_{ref})^{\mathsf{T}}(\mathbf{u}_i - \mathbf{u}_{ref})} \\ & \alpha_i \geq 0.01. \end{aligned} \tag{7.2}$$

The optimization history for the tip displacement is shown in Figure 7.16. The sensitivities drive towards reducing the stiffness at the trailing edge whilst stiffening the airfoil near the clamp as well as at mid-chord. This reduces the tip displacement prior to gust which is still identified as an increase in magnitude of tip displacement compared to the initial design for the optimization to minimize, however a reduction in the displacement as a result of the gust load can clearly be seen. The peak displacement and peak lift are both reduced as well as the range in lift in the time-history response of the optimized design.

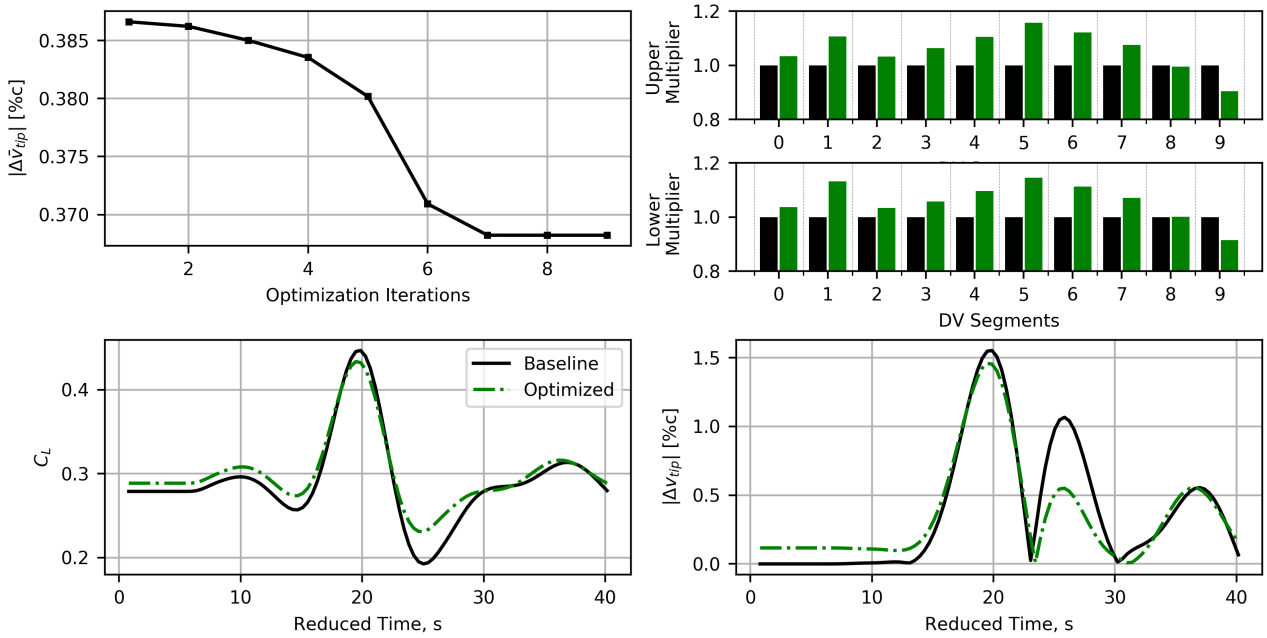


Figure 7.16: Optimization history for compliant NACA 0012 airfoil to minimize tip displacement under gust load $H = 5c$ with design variables of stiffness distribution.

Chapter 8

Conclusions

This chapter presents some concluding remarks from the research presented in this thesis. The choice of research topic and relevance for application is initially discussed before moving to the theoretical formulation that forms the backbone of the methodology developed in this work. The key results that demonstrate functionality for the new dynamic fluid structure interaction capabilities are summarized from the individual building blocks to the final method. The key research contributions are emphasized and connected back to the research objectives laid out for this work. Finally, suggestions for future work are presented that would build on the present work to improve efficiency and suitability for varied, realistic applications and continue to further dynamic optimization capabilities beyond current possibilities.

8.1 Summary of Thesis Achievements

This thesis has presented an efficient methodology for coupled high-fidelity FSI analysis and design of steady and dynamic problems. The partitioned FSI solver adopted for this work enables different solvers to be used for each domain, thus creating a flexible framework which will be suitable for a variety of problems. A new unsteady incompressible solver has been developed, implemented and verified for individual domain problems as well as fully coupled FSI analysis. The subsequent optimal design framework hinges on evaluation of sensitivities using a

discrete adjoint methodology implemented through reverse mode of Algorithmic Differentiation. Building on the foundation established for coupled design on stationary conditions [47], the steady analysis has been extended to shape optimization and a new dynamic discrete adjoint framework for fully coupled FSI problems has been formulated and demonstrated. Finally, an open-source implementation offers this new capability to the aerospace community to perform dynamic coupled optimization for a large variety of problems that can easily be extended further by incorporating additional physics in the flexible partitioned environment.

Firstly, the motivations for the optimal design framework suitable for fully coupled, high-fidelity, dynamic FSI was presented in Chapter 1. Literature review of the current state of the art methods for coupled FSI analysis as well as design with focus on dynamic analysis have been presented, including the known challenges, assumptions and simplifications typically used. The adjoint methodology has been well established as a highly desirable approach due to the high accuracy and low cost associated for large number of design variables as opposed to other alternatives. Beginning with applications in aerodynamic optimization, this method has also been shown to be suitable for multidisciplinary optimization. However, a lack of fully coupled, non-linear dynamic optimization methods was highlighted due to the complexity of the primal problem as well as feasibility of design with full dynamic analysis for all domains.

Chapter 2 has presented the coupled FSI solver adopted in this work. A partitioned FSI framework has been chosen for increased flexibility of the physics that can be incorporated for the coupled analysis. Therefore, governing equations for dynamic coupled FSI have been presented using a partitioned FSI solver approach. The individual domains for the fluid, structural and mesh solvers has been presented. This included the new unsteady incompressible fluid solver and velocity interface coupling. The unsteady incompressible fluid solver has been developed in ALE formulation to support dynamic mesh problems. The implementation details for the spatial integration, as well as the time integration using dual-time stepping strategy with preconditioned pseudo-time derivative developed from the density-based approach, have been presented. For dynamic coupled simulation, the continuity of velocities at the interface has been explicitly enforced and the changes to the mesh solver for the propagation of velocities, in addition to the displacements, from the boundary to the fluid nodes have also been

presented. The main novel contributions from this work in Chapter 2 are therefore on the new unsteady incompressible solver for the fluid domain and the dynamic FSI coupling at the interface, including the mesh domain with velocity interpolation.

Moving to sensitivity analysis, Chapter 3 briefly presented the typical options available for the evaluation of sensitivities for a given system of equations and design variables. The adjoint method has been highlighted as the most efficient for large number of design variables, and the choice of design variables for optimal design has also been discussed, including the new definitions for shape optimization for aeroelastic optimal design. The discrete adjoint formulation has relied on the use of reverse mode Algorithmic Differentiation with the governing equations written in the form of fixed-point iterators. This method has removed the need for the evaluation of the exact analytic Jacobian and allowed further developments in the primal solution as well as the incorporation of new governing equations for the individual domains to be differentiated more easily in this framework. The formulation for dynamic discrete adjoint sensitivities has been presented including the time-dependent adjoint variables that arise from the chosen time integration methods of the primal problem. The implementation details in SU2 have also been provided, which involved the extension of the generic multizone discrete adjoint solver in SU2 for time-accurate analysis. This retains the generality of the methodology, which is a core philosophy of this work, such that the resulting framework is usable for a wide variety of applications. As a result, the main contribution from this chapter lies in the new formulation for dynamic discrete adjoint sensitivities for fully coupled FSI problems which have been implemented as a generic multizone driver in SU2 to provide a flexible framework for a wide range of applications.

The numerical results from the application of the created framework have been presented in the following chapters beginning with stiffness and shape optimization for steady FSI problems in Chapter 4. The discrete adjoint sensitivities have been firstly verified for both stiffness and shape design variables. The optimal design framework has then been demonstrated for aeroelastic optimization with aerodynamic objective function using both design variables. To move towards dynamic coupled application, firstly the individual domains have been considered. Chapter 5 has presented sensitivity analysis results for an elastic bar under axial sinusoidal

load to verify discrete adjoint sensitivities against direct differentiation using forward mode Algorithmic Differentiation as well as finite differences. Secondly, in this chapter using an airfoil under sinusoidal load, the natural frequency of the airfoil has been evaluated for varying stiffness values. A shape optimization has then been performed to maximize the intrinsic structural properties to maximize tip displacement under user prescribed loading. The main achievements arising from these are in the demonstration of the introduced methodology for shape optimization for steady FSI problems and dynamic structural problems, together with the verification of discrete adjoint sensitivities.

Next, Chapter 6 has presented the verification and validation of the unsteady incompressible solver using benchmark cases with increasing complexity. A lid-driven cavity has been used for preliminary verification of the ALE formulation using a moving boundary condition on a steady problem. An unsteady, fixed grid benchmark has then been chosen, namely the laminar flow past a circular cylinder. This is a well-established benchmark and the new solver has been shown to accurately capture both the lift and drag properties as well as the frequency response of the problem. Finally, turbulence and moving grid have been added to the complexity of the problem and the solver has been successfully validated for a pitching NACA 0015 airfoil benchmark against experimental results. The discrete adjoint sensitivities for the unsteady, incompressible solver have been verified for a number of chosen cases, again with varying complexities. Firstly, a rotating frame has been chosen for evaluation of the sensitivities in a steady solution but with grid velocity and applied to optimal design. The discrete adjoint sensitivities have then been successfully verified for an unsteady circular cylinder before demonstrating this methodology for optimal design of a plunging airfoil in turbulent flow that maximizes the unsteady characteristics of the problem for drag minimization. The main achievements are therefore in the verification and validation of the new unsteady incompressible fluid solver together with discrete adjoint sensitivities that support grid motion which have been applied for dynamic optimal design.

Finally, Chapter 7 has presented the primal and discrete adjoint results for fully coupled dynamic FSI problems. While there has been considerable work in the literature on dynamic FSI simulations, the intrinsic complexity of unsteady coupled adjoints, makes this implementation one of the very few available for that problem. The unsteady incompressible solver has been

verified for application in dynamic FSI application using the low speed standard FSI benchmark of elastic cantilever beam behind a square cylinder. The results have been shown to be in excellent agreement with literature and compared to the previous validation using the compressible solver, the incompressible solver has been shown to successfully identify high-frequency effects with coarser grid and time discretization which can significantly reduce the associated computational cost for the time-accurate coupled simulation. The discrete adjoint sensitivities for the stiffness distribution of the elastic beam have been shown to be in very good agreement against finite differences for objective functions of both tip displacement and drag. This chapter has also presented primal coupled results using the compressible fluid solver for transonic airfoil undergoing limit cycle oscillations induced by the movement of shocks on the compliant airfoil. The initial transient response for this problem has been chosen for the verification of the discrete adjoint sensitivities which have been shown to be in good agreement against finite differences for a variety of objective functions based on both the aerodynamic and structural performance metrics. To demonstrate the suitability of this framework for gust-load alleviation problems, gust load simulations for fluid-only as well as fully coupled FSI problems have been presented for varying gust lengths. Following the verification of the discrete adjoint sensitivities, the application of the developed framework for minimization of lift and tip displacement on a compliant airfoil subject to gust perturbation have been presented. The main achievements from this chapter are therefore in the optimal design using the verified dynamic sensitivities obtained from the novel dynamic, coupled FSI discrete adjoint formulation. Optimization under gust loads using fully coupled dynamic problems have never previously been attempted which have now been made possible using the developed method and implementation in this thesis. In addition to this, the FSI solver has been extended to incompressible flows and has demonstrated the suitability for capturing transient dynamics for optimal design enabling future coupling of further multiphysics problems such as conjugate heat transfer with FSI.

8.2 Main Research Outcomes

Review of the literature had highlighted that given the recent improvements in coupled FSI analysis and discrete adjoint methodology, an optimization framework for coupled, dynamic FSI problems using discrete adjoint sensitivities would be a highly desirable approach. Though coupled FSI analysis has been an area of much interest, flexible frameworks capable incorporating a variety of flow regimes for high-fidelity coupled FSI were still missing, especially for dynamic problems. Multidisciplinary optimizations have been established for coupled FSI problems, however fully non-linear, dynamic FSI optimization without either reduced order modelling or steady assumptions still remained an open problem. This led to the main research objective, as described in Chapter 1, to explore the suitability of a generalized partitioned solver approach for discrete adjoint sensitivities for dynamic FSI problems.

The partitioned approach initially proposed by Farhat [14] was adopted by Sanchez [22] in SU2 for the primal FSI solver which was used successfully for coupled steady discrete adjoint sensitivities [47]. For dynamic coupled FSI sensitivities, the partitioned approach enables the building blocks for the methodology to be constructed from individual domains. Therefore, the discrete adjoint sensitivities for the individual domains were firstly verified, including grid motion in the fluid domain. The partitioned approach for the discrete adjoint method, has lent itself for the coupled FSI problems with various flow and structural regimes. The solution process iterates each domain in turn to evaluate both the diagonal (pure domain) and off-diagonal (cross-domain) adjoint variable and for each time step the coupled solution is converged using outer iterations similar to the primal solution process. This enables the use of any established discrete adjoint solvers for each domain, thus providing a flexible framework for different governing equations. The generic implementation also enables additional objective functions and design variables from any domain. As the problem size and duration in time increases, the memory challenges will also increase with this framework, with potential solutions which could be found through the use of checkpointing schemes [89, 88].

The research questions formed around the thesis objective are re-iterated below together with the found solutions.

1. **What are the necessary changes for accurate capture of the unsteady coupled interactions that are not included in the steady discrete adjoint formulation?**
   The dual-time stepping strategy for the fluid domain and Newmark time integration for the structural domain have necessitated the addition of time source terms in order to capture the influence of previous time solutions at the current time step. In addition to this, the cross-term solutions from the fluid, structural and mesh domains now include the time-dependent solutions. The coupling boundary conditions had been modified to explicitly enforce continuity of velocities, which in turn has required the propagation of velocity from the interface to the rest of the fluid domain.

2. **What are the development and evaluation costs in the extension of the steady coupled discrete adjoint solver for dynamic problems?**
   The multizone discrete adjoint approach enables transition to time-domain through a process of extension from the steady discrete adjoint solver rather than complete reformulation of the implementation. However, the complexities arising from the additional number of variables and recordings necessary from each domain at each time-step including the accurate capture of the dynamic coupling across the fluid and structural domains makes this a challenging problem. The development relies on the verification of the unsteady discrete adjoint implementation of the individual domains. Unsteady discrete adjoints had been previously verified for the fluid solver for compressible flows [46] and structural solver [47] using stiffness design variables. This had been extended to discrete adjoint sensitivities for the incompressible flow solver by the addition of time-dependent adjoints as well as addition of shape design variables for the structural solver. To the best of the author's knowledge, this is the first discrete adjoint methodology for dynamic FSI problems which has been successfully verified. The developed approach is suitable for a variety of dynamic coupled problems and has been demonstrated for the first optimization under gust load.

3. **What are the key challenges in the addition of an unsteady incompressible fluid solver in SU2 and what are the benefits of this for coupled optimal design?**

The density based approach for the incompressible solver has been extended for unsteady problems with grid motion. Modified approaches have typically been required for unsteady flows with grid motion [109] if the preconditioning matrix explicitly depends on grid motion. However, the approach developed in this work leads to a final expression for the preconditioning matrix which is independent of the grid motion. This simplifies the implementation since the formulation remains mostly unchanged between steady and unsteady incompressible flows. The use of the appropriate solver that best represents the flow regime has been shown to enable the use of less expensive grid and time discretization in coupled FSI analysis. For coupled time-accurate simulations, this leads to a significant advantage in reduced computational cost for each primal evaluation, which will have a compounded benefit for optimization problems.

8.3 Recommendations for Future Work

An accurate, flexible and efficient framework has been developed for dynamic FSI analysis and optimization using discrete adjoint sensitivities. The implementation has been applied for a number of verification and prototype cases to demonstrate the functionality available with the new method. Therefore, the aims of the research have been met and the research questions answered as discussed above. To conclude the thesis, recommendations are provided for future work to further expand the capabilities of the developed framework, improve efficiency as well as suggestions for future applications.

Firstly, the generalized framework lends itself to further inclusion of wider range of physics that can be solved. The conjugate heat transfer (CHT) methods in SU2 [137] rely on the new incompressible solver. With the extension of the unsteady incompressible solver for primal and discrete adjoints, the extension to unsteady CHT discrete adjoint sensitivities would be a natural progression. The multizone discrete adjoint framework for steady flows [87] which has been extended for dynamic FSI has been kept generic and should be a suitable platform for unsteady CHT analysis as well. This could also be taken one step further for dynamic FSI

with CHT for low speed thermal simulations with flexible structures such as valves or blades in heated internal flows.

Improvements to the convergence characteristics of the unsteady incompressible solver have been made by modifying the numerical properties of the Jacobian matrix in the convective flux calculation as well as the Jacobian contributions to the dual-time residual calculation. Deeper studies into the convergence behaviour of the incompressible flow solver, including multi-grid approaches with dynamic meshes, would be beneficial to reduce computational cost as the problems are scaled up in complexity and size.

Moving to high-speed flows, the recent addition of the NonEquilibrium MOdels (NEMO) for Hypersonic Flows Using Mutation++ [27] enables consideration of high-Mach and high-enthalpy external aerothermodynamics. Adoption of this solver for coupled FSI would enable hypersonic control with compliant surfaces. When the NEMO solver is extended for discrete adjoint sensitivities, coupled multizone sensitivities can also be explored for hypersonic flows.

Linear elements have continued to have been used in this work for the finite-element discretization of the solid domain, but they are known to have problems with artificial stiffening leading to shear locking. Use of higher order elements will improve robustness of the structural solver. This would ideally be accompanied by higher order elements for the fluid, since both solvers read the same multizone grid files and share data structures.

As the objective was a proof of concept of the proposed formulation, the methodology developed in this work has been demonstrated for 2D problems. However, the method is generic and can be extended to 3D problems. Though the methodology can be transferred to 3D problems, the associated cost can reduce the viability for the application of this methodology for realistic problems. Preliminary demonstration of the use of this framework for steady 3D problems have been shown by Gomes [87]. Further refinements of the grid to reduce y^+ to be less than 1 will be needed for detailed flow field analysis, however this has significant increase in associated cost unless wall functions are added. Improvements to mesh deformation routines, which currently uses the linear solver approach, are also necessary to reduce the computational cost which increases with the size of the domain.

For dynamic FSI sensitivities, the use of stiffness design variables have been verified in this work. This could be extended to geometric sensitivities for use in dynamic FSI shape optimization. For highly refined meshes and long time simulations, the discrete adjoint simulation can lead to large memory storage requirements since the primal solution files for every time step need to be stored. Check-pointing [138] is a well-established method in discrete adjoint implementations to reduce the associated overhead cost of memory usage. Instead of the solution for every time step being stored, check-pointed solution files can be stored from the primal run, using which the remaining direct solutions are recomputed during the adjoint run. This increases the time taken for the discrete adjoint solution since additional primal evaluations are performed during the adjoint run, however reduces the number of solution files stored which can make dynamic problems with very large time period more tractable.

The combined performance of shape and topology optimization has been explored by Gomes [25] for steady problems. Modifying both fluid and structural properties will be highly relevant for application in time domain problems. This will allow improved shifts in inertial properties of the structure whilst continuing to optimize the wet shape under operational load for dynamic problems which should lead to better designs than from isolated optimal designs of these two problems.

The developed framework will be an ideal platform for gust-load alleviation optimization. Passive control through optimal designs which modify the structural properties to prevent an unstable response to a gust perturbation whilst maintaining aerodynamic performance can be explored. Additional constraints can be added to the methodology to consider maximum values during the time period as well as time-averaged solutions. The discrete adjoint methodology can be readily expanded to time-dependent design variables which can be used for optimal control, thus creating a pathway for active gust load alleviation optimization.

Bibliography

[1] G. Abbe and H. Smith, "Technological development trends in Solar-powered Aircraft Systems," *Renewable and Sustainable Energy Reviews*, vol. 60, pp. 770–783, 2016.

[2] S. Haghighat, J. Martins, and H. Liu, "Aeroservoelastic design optimization of a flexible wing," *Journal of Aircraft*, vol. 49, no. 2, pp. 432–443, 2012.

[3] T. E. Noll, J. M. Brown, M. E. Perez-davis, S. D. Ishmael, G. C. Tiffany, and M. Gaier, "Investigation of the Helios Prototype Aircraft Mishap," Tech. Rep. January, NASA, 2004.

[4] R. Palacios, J. Murua, and R. Cook, "Structural and aerodynamic models in nonlinear flight dynamics of very flexible aircraft," *Aiaa Journal - AIAA J*, vol. 48, pp. 2648–2659, 11 2010.

[5] A. T. Klesh and P. T. Kabamba, "Solar-Powered Aircraft: Energy-Optimal Path Planning and Perpetual Endurance," *Journal of Guidance, Control, and Dynamics*, vol. 32, no. 4, pp. 1320–1329, 2009.

[6] T. Dunne, "An eulerian approach to fluid - structure interaction and goal-oriented mesh adaptation," *International Journal for Numerical Methods in Fluids*, vol. 51, pp. 1017 – 1039, 07 2006.

[7] T. Richter, "A monolithic geometric multigrid solver for fluid-structure interactions in ale formulation," *International Journal for Numerical Methods in Engineering*, vol. 104, 05 2015.

[8] Y. Wang, P. Jimack, M. Walkley, and O. Pironneau, "An energy stable one-field monolithic arbitrary lagrangian–eulerian formulation for fluid–structure interaction," *Journal of Fluids and Structures*, vol. 98, p. 103117, 10 2020.

[9] U. Küttler, M. Gee, C. Förster, A. Comerford, and W. Wall, "Coupling strategies for biomedical fluid–structure interaction problems," *International Journal for Numerical Methods in Biomedical Engineering*, vol. 26, pp. 305 – 321, 03 2010.

[10] P. Ryzhakov, E. Soudah, and N. Dialami, "Computational modeling of the fluid flow and the flexible intimal flap in type b aortic dissection via a monolithic arbitrary lagrangian/eulerian fluid-structure interaction model," *International Journal for Numerical Methods in Biomedical Engineering*, vol. 35, 07 2019.

[11] R. A. Sauer and T. Luginsland, "A monolithic fluid–structure interaction formulation for solid and liquid membranes including free-surface contact," *Computer Methods in Applied Mechanics and Engineering*, vol. 341, pp. 1 – 31, 2018.

[12] T. Richter, "A fully eulerian formulation for fluid-structure interaction problems," *Journal of Computational Physics*, vol. 233, p. 227–240, 01 2013.

[13] M. Heil, A. Hazel, and J. Boyle, "Solvers for large-displacement fluid-structure interaction problems: Segregated versus monolithic approaches," *Computational Mechanics*, vol. 43, pp. 91–101, 12 2008.

[14] C. Farhat, M. Lesoinne, and N. Maman, "Mixed explicit/implicit time integration of coupled aeroelastic problems: Three-field formulation, geometric conservation and distributed solution," *International Journal for Numerical Methods in Fluids*, vol. 21, no. 10, pp. 807–835, 1995.

[15] J. Degroote, S. Annerel, and J. Vierendeels, "Stability analysis of gauss-seidel iterations in a partitioned simulation of fluid-structure interaction," *Computers & Structures - COMPUT STRUCT*, vol. 88, 03 2010.

[16] C. Kassiotis, A. Ibrahimbegovic, R. Niekamp, and H. Matthies, "Nonlinear fluid-structure interaction problem. part i: Implicit partitioned algorithm, nonlinear stability proof and validation examples," *Computational Mechanics*, vol. 47, pp. 305–323, 03 2011.

[17] M. Barcelos and K. Maute, "Aeroelastic design optimization for laminar and turbulent flows," *Computer Methods in Applied Mechanics and Engineering*, vol. 197, no. 19-20, pp. 1813–1832, 2008.

[18] J. Degroote, K.-J. Bathe, and J. Vierendeels, "Performance of a new partitioned procedure versus a monolithic procedure in fluid–structure interaction," *Computers & Structures*, vol. 87, pp. 793–801, 06 2009.

[19] S. Idelsohn, F. Del Pin, R. Rossi, and E. Oñate, "Fluid–structure interaction problems with strong added-mass effect," *International Journal for Numerical Methods in Engineering*, vol. 80, pp. 1261 – 1294, 07 2009.

[20] B. M. Irons and R. C. Tuck, "A version of the aitken accelerator for computer iteration," *International Journal for Numerical Methods in Engineering*, vol. 1, no. 3, pp. 275–277, 1969.

[21] U. Küttler and W. A. Wall, "Fixed-point fluid-structure interaction solvers with dynamic relaxation," *Computational Mechanics*, vol. 43, no. 1, pp. 61–72, 2008.

[22] R. Sanchez, R. Palacios, T. D. Economon, H. L. Kline, J. J. Alonso, and F. Palacios, "Towards a Fluid-Structure Interaction Solver for Problems with Large Deformations Within the Open-Source SU2 Suite," in *57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, (San Diego, CA), 2016.

[23] F. Palacios, M. R. Colonno, A. C. Aranake, A. Campos, S. R. Copeland, T. D. Economon, A. K. Lonkar, T. W. Lukaczyk, T. W. R. Taylor, and J. J. Alonso, "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*, January 2013.

[24] T. D. Economon, F. Palacios, S. R. Copeland, T. W. Lukaczyk, and J. J. Alonso, "SU2: An open-source suite for multiphysics simulation and design," *AIAA Journal*, vol. 54, no. 3, pp. 828–846, 2016.

[25] P. Gomes and R. Palacios, "Aerodynamic-driven topology optimization of compliant airfoils," *Structural and Multidisciplinary Optimization: computer-aided optimal design of stressed solids and multidisciplinary systems*, vol. 62, pp. 2117–2130, 2020.

[26] T. D. Economon, "Simulation and adjoint-based design for variable density incompressible flows with heat transfer," *AIAA Journal*, vol. 58, no. 2, pp. 757–769, 2020.

[27] C. Garbacz, M. Fossati, W. Maier, J. Alonso, J. Scoggins, T. Magin, and T. Economon, "Numerical study of shock interference patterns for gas flows with thermal nonequilibrium and finite-rate chemistry," in *AIAA Scitech 2020 Forum*, (Orlando, FL), January 2020.

[28] E. Molina, C. Spode, R. Silva, D. Manosalvas-Kjono, S. Nimmagadda, T. Economon, J. Alonso, and M. Righi, "Hybrid rans/les calculations in su2," in *23rd AIAA Computational Fluid Dynamics Conference*, (Denver, CO), June 2017.

[29] L. A. Schmit, "Structural synthesis - its genesis and development," *AIAA Journal*, vol. 19, no. 10, pp. 1249–1263, 1981.

[30] R. T. Haftka, J. H. Starnes, F. W. Barton, and S. C. Dixon, "Comparison of two types of structural optimization procedures for flutter requirements," *AIAA Journal*, vol. 13, no. 10, pp. 1333–1339, 1975.

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

[32] J. R. R. A. Martins and A. B. Lambe, "Multidisciplinary Design Optimization: A Survey of Architectures," *AIAA Journal*, vol. 51, no. 9, pp. 2049–2075, 2013.

[33] J. T. Hwang, A. V. Jain, and T. H. Ha, "Large-scale multidisciplinary design optimization—review and recommendations," in *AIAA Aviation 2019 Forum*, 2019.

[34] E. J. Cramer, J. E. Dennis, Jr., P. D. Frank, R. M. Lewis, and G. R. Shubin, "Problem formulation for multidisciplinary optimization," *SIAM Journal on Optimization*, vol. 4, no. 4, pp. 754–776, 1994.

[35] R. T. Haftka, "Simultaneous analysis and design," *AIAA Journal*, vol. 23, no. 7, pp. 1099–1103, 1985.

[36] R. Hooke and T. A. Jeeves, ""Direct Search" Solution of Numerical and Statistical Problems," *Journal of the ACM (JACM)*, vol. 8, no. 2, pp. 212–229, 1961.

[37] J. A. Nelder and R. Mead, "A Simplex Methd for Function Minimization," *The Computer Journal*, vol. 7, no. 4, pp. 308–313, 1965.

[38] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. MIT Press, 1992.

[39] L. M. Ruis and N. V. Sahinidis, "Derivative-free optimization : a review of algorithms and comparison of software implementations," *Journal of Global Optimization*, vol. 56, no. 3, pp. 1247–1293, 2013.

[40] A. Azad and J. Hansen, "Method of regular simplexes: A difference-assisted simplex-based search algorithm," in *12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, (Victoria, British Columbia), September 2008.

[41] X. Yao, S. K. Moon, and G. Bi, "Multidisciplinary design optimization to identify additive manufacturing resources in customized product development," *Journal of Computational Design and Engineering*, vol. 4, no. 2, pp. 131–142, 2017.

[42] T. Cocke, Z. Moscicki, and R. Agarwal, "Optimization of hydrofoils using a genetic algorithm," *Journal of Aircraft*, vol. 51, no. 1, pp. 78–89, 2014.

[43] J. R. R. A. Martins and J. T. Hwang, "Review and Unification of Methods for Computing Derivatives of Multidisciplinary Computational Models," *AIAA Journal*, vol. 51, no. 11, pp. 2582–2599, 2013.

[44] D. Kraft, "A software package for sequential quadratic programming," tech. rep., DLR German Aerospace Center - Institute for Flight Mechanics, 1988.

[45] T. Albring, M. Sagebaum, and N. Gauger, "Efficient aerodynamic design using the discrete adjoint method in SU2," *17th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, June 2016.

[46] B. Y. Zhou, T. A. Albring, N. R. Gauger, T. D. Economon, F. Palacios, and J. J. Alonso, "A Discrete Adjoint Framework for Unsteady Aerodynamic and Aeroacoustic Optimization," in *16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, (Dallas, TX), 2015.

[47] R. Sanchez, T. Albring, R. Palacios, N. R. Gauger, T. D. Economon, and J. J. Alonso, "Coupled adjoint-based sensitivities in large-displacement fluid-structure interaction using algorithmic differentiation," *International Journal for Numerical Methods in Engineering*, vol. 113, no. 7, pp. 1081–1107, 2018.

[48] A. Wächter and L. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical programming*, vol. 106, pp. 25–57, 03 2006.

[49] R. A. Bartlett, A. Wachter, and L. T. Biegler, "Active set vs. interior point strategies for model predictive control," in *Proceedings of the 2000 American Control Conference. ACC (IEEE Cat. No.00CH36334)*, vol. 6, pp. 4229–4233 vol.6, 2000.

[50] P. E. Gill, W. Murray, and M. A. Saunders, "SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization," *SIAM Journal on Optimization*, vol. 12, no. 4, pp. 979–1006, 2002.

[51] J. P. Jasa, C. A. Mader, and J. R. R. A. Martins, "Trajectory optimization of a supersonic aircraft with a thermal fuel management system," in *2018 Multidisciplinary Analysis and Optimization Conference*, (Atlanta, GA), June 2018.

[52] J. Yan, N. Li, T. Luo, M. T. Tolley, and J. T. Hwang, "Optimal control and design of an underactuated ball-pitching robotic arm using large-scale multidisciplinary optimization," in *AIAA Aviation 2019 Forum*, (Dallas, TX), June 2019.

[53] J. T. Allison and D. R. Herber, "Multidisciplinary Design Optimization of Dynamic Engineering Systems," *AIAA Journal*, vol. 52, no. 4, pp. 691–710, 2014.

[54] S. Maraniello and R. Palacios, "Optimal rolling maneuvers with very flexible wings," *AIAA Journal*, vol. 55, pp. 1–16, 08 2017.

[55] S. Buoso and R. Palacios, "Electro-aeromechanical modelling of actuated membrane wings," *Journal of Fluids and Structures*, vol. 58, pp. 188–202, 2015.

[56] S. Maraniello and R. Palacios, "Optimal vibration control and co-design of very flexible actuated structures," *Journal of Sound and Vibration*, vol. 377, pp. 1–21, 2016.

[57] A. del Carre and R. Palacios, "Simulation and optimization of takeoff maneuvers of very flexible aircraft," *Journal of Aircraft*, vol. 0, no. 0, pp. 1–14, 2020.

[58] R. E. Perez, P. W. Jansen, and J. R. Martins, "PyOpt: A Python-based object-oriented framework for nonlinear constrained optimization," *Structural and Multidisciplinary Optimization*, vol. 45, no. 1, pp. 101–118, 2012.

[59] G. J. Kennedy and J. R. R. A. Martins, "Parallel solution methods for aerostructural analysis and design optimization," in *Proceedings of the 13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference*, (Fort Worth, TX), September 2010.

[60] J. N. Lyness, "Numerical algorithms based on the theory of complex variable," *Proceedings - ACM National Meeting*, pp. 125–133, 1967.

[61] W. Squire and G. Trapp, "Using complex variables to estimate derivatives of real functions," *SIAM Review*, vol. 40, no. 1, pp. 110–112, 1998.

[62] A. Jameson, "Aerodynamic design via control theory," *Journal of Scientific Computing*, vol. 3, pp. 233–260, 1988.

[63] K. Maute, M. Nikbay, and C. Farhat, "Analytically based sensitivity analysis and optimization of nonlinear aeroelastic systems," in *8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, (Long Beach, CA), pp. 1–10, 2000.

[64] K. Maute, M. Nikbay, and C. Farhat, "Coupled Analytical Sensitivity Analysis and Optimization of Three-Dimensional Nonlinear Aeroelastic Systems," *AIAA Journal*, vol. 39, no. 11, pp. 2051–2061, 2001.

[65] K. Maute, M. Nikbay, and C. Farhat, "Sensitivity analysis and design optimization of three-dimensional non-linear aeroelastic systems by the adjoint method," *International Journal for Numerical Methods in Engineering*, vol. 56, no. 6, pp. 911–933, 2003.

[66] J. R. R. A. Martins, J. J. Alonso, and J. Reuther, "Aero-Structural Wing Design Optimization Using High-Fidelity Sensitivity Analysis," *In Proceedings-CEAS Conference on Multidisciplinary Aircraft Design Optimization*, pp. 211–226, 2001.

[67] J. R. R. A. Martins, J. J. Alonso, and J. J. Reuther, "Complete configuration aero-structural optimization using a coupled sensitivity analysis method," in *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, (Atlanta), 2002.

[68] E. Lund, H. Møller, and L. A. Jakobsen, "Shape design optimization of stationary fluid-structure interaction problems with large displacements and turbulence," *Structural and Multidisciplinary Optimization*, vol. 25, no. 5-6, pp. 383–392, 2003.

[69] J. R. R. A. Martins, P. Sturdza, and J. J. Alonso, "The Connection Between the Complex-Step Derivative Approximation and Algorithmic Differentiation," in *39th Aerospace Sciences Meeting/AIAA*, 2001.

[70] A. Griewank and A. Walther, *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*. Philadelphia: SIAM, second ed., 2008.

[71] R. V. Kharche, *MATLAB Automatic Differentiation using Source Transformation*. PhD thesis, Cranfield University, 2012.

[72] D. Casanova, R. S. Sharp, M. Final, B. Christianson, and P. Symonds, *Application of Automatic Differentiation to Race Car Performance Optimisation*, pp. 117–124. New York, NY: Springer New York, 2002.

[73] L. Hascoët, M. Vázquez, and A. Dervieux, "Automatic differentiation for optimum design, applied to sonic boom reduction," in *ICCSA 2003: Computational Science and Its Applications*, pp. 85–94, 2003.

[74] C. H. Bischof, H. M. Bücker, A. Rasch, and E. Slusanschi, "Evaluation of a Computer Model for Wavy Falling Films Using EFCOSS," in *ICCSA 2003: Computational Science and Its Applications*, vol. 2668, pp. 78–84, 2003.

[75] T. F. Coleman and A. Verma, "ADMAT: An automatic differentiation toolbox for MATLAB," in *In Proceedings of the SIAM Workshop on Object Oriented Methods for Inter-Operable Scientific and Engineering Computing*, (Philadelphia), 1998.

[76] T. F. Coleman and A. Verma, "ADMIT-1: Automatic Differentiation and MATLAB Interface Toolbox," *ACM Transactions on Mathematical Software*, vol. 26, no. 1, pp. 150–175, 2000.

[77] T. F. Coleman and J.-Y. Cai, "The cyclic coloring problem and estimation of sparse {H}essian matrices," *SIAM Journal on Algebraic and Discrete Methods*, vol. 7, no. 2, pp. 221–235, 1986.

[78] S. A. Forth, "An efficient overloaded implementation of forward mode automatic differentiation in MATLAB," *ACM Transactions on Mathematical Software*, vol. 32, no. 2, pp. 195–222, 2006.

[79] R. V. Kharche and S. A. Forth, "Source Transformation for MATLAB Automatic Differentiation," in *Computational Science – ICCS 2006: 6th International Conference, Reading, UK, May 28-31, 2006, Proceedings, Part IV*, (Berlin), pp. 558–565, 2006.

[80] M. A. Patterson, M. J. Weinstein, and A. V. Rao, "An efficient overloaded method for computing derivatives of mathematical functions in MATLAB," *ACM Transactions on Mathematical Software*, vol. 39, no. 3, pp. 17:1–17:36, 2013.

[81] C. H. Bischof, H. M. Bucker, B. Lang, A. Rasch, and A. Vehreschild, "Combining source transformation and operator overloading techniques to compute derivatives for MATLAB programs," *SCAM 2002 - Proceedings, 2nd IEEE International Workshop on Source Code Analysis and Manipulation*, pp. 65–72, 2002.

[82] C. A. Mader, G. K. W. Kenway, A. Yildirim, and J. R. R. A. Martins, "ADflow—an open-source computational fluid dynamics solver for aerodynamic and multidisciplinary optimization," *Journal of Aerospace Information Systems*, 2020.

[83] G. Kenway, C. Mader, P. He, and J. Martins, "Effective adjoint approaches for computational fluid dynamics," *Progress in Aerospace Sciences*, 06 2019.

[84] M. Sagebaum, T. Albring, and N. R. Gauger, "High-performance derivative computations using CoDiPack," *ACM Transactions on Mathematical Software (TOMS)*, vol. 45, December 2019.

[85] P. Aubert, N. D. Césaré, and O. Pironneau, "Automatic differentiation in c++ using expression templates and. application to a flow control problem," *Computing and Visualization in Science*, vol. 3, pp. 197–208, 2001.

[86] T. Albring, M. Sagebaum, and N. R. Gauger, "Development of a consistent discrete adjoint solver in an evolving aerodynamic design framework," *16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, June 2015.

[87] O. Burghardt, N. Gauger, P. Gomes, R. Palacios, T. Kattmann, and T. Economon, "Coupled discrete adjoints for multiphysics in su2," *AIAA Aviation Forum*, 06 2020.

[88] A. Griewank and A. Walther, "Algorithm 799: revolve: An implementation of checkpointing for the reverse or adjoint mode of computational differentiation," *ACM Transactions on Mathematical Software (TOMS)*, vol. 26, no. 1, pp. 19–45, 2000.

[89] Q. Wang, P. Moin, and G. Iaccarino, "Minimal repetition dynamic checkpointing algorithm for unsteady adjoint calculation," *SIAM Journal on Scientific Computing*, vol. 31, no. 4, pp. 2549–2567, 2009.

[90] C. Venkatesan-Crome, R. Sanchez, and R. Palacios, "Aerodynamic optimization using fsi coupled adjoints in su2," in *6th European Conference on Computational Mechanics (ECCM 6)*, (Glasgow, UK), 2018.

[91] C. Venkatesan-Crome, P. Gomes, and R. Palacios, "Optimal compliant airfoils using fully non-linear fsi models," in *AIAA Scitech 2019 Forum, AIAA SciTech Forum, (AIAA 2019-1216)*, (San Diego, CA), 2019.

[92] C. Venkatesan-Crome, R. Palacios, T. Kattmann, R. Sanchez, N. Gauger, and T. Economon, "Discrete adjoint for unsteady incompressible flows using a density-based formulation," in *AIAA Aviation Forum 2019*, (Dallas, TX), 06 2019.

[93] C. Venkatesan-Crome and R. Palacios, "A discrete adjoint solver for time-domain fluid-structure interaction problems with large deformations," *AIAA Scitech Forum*, January 2020.

[94] F. Palacios, T. D. Economon, A. C. Aranake, S. R. Copeland, A. K. Lonkar, T. W. Lukaczyk, D. E. Manosalvas, K. R. Naik, A. S. Padron, B. Tracey, A. Variyar, and J. J. Alonso, "Stanford University Unstructured (SU^2): Open-source analysis and design technology for turbulent flows," *52nd Aerospace Sciences Meeting*, 2014.

[95] P. L. Roe, "Approximate riemann solvers, parameter vectors, and difference schemes," *Journal of Computational Physics*, vol. 43, no. 2, pp. 357–372, 1981.

[96] A. Jameson, W. Schmidt, and E. Turkel, "Numerical solution of the Euler equations by finite volume methods using Runge-Kutta time stepping schemes," *14th Fluid and Plasma Dynamics Conference*, June 1981.

[97] B. van Leer, "Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov's method," *Journal of Computational Physics*, vol. 32, no. 1, pp. 101–136, 1979.

[98] J. M. Weiss, J. P. Maruszewski, and A. S. Wayne, "Implicit solution of the Navier-Stokes equation on unstructured meshes," *13th Computational Fluid Dynamics Conference*, June 1997.

[99] J. Blazek, *Computational Fluid Dynamics: Principles and Applications*. Oxford: Elsevier, 2005.

[100] T. D. Economon, F. Palacios, and J. J. Alonso, "Unsteady continuous adjoint approach for aerodynamic design on dynamic meshes," *AIAA Journal*, vol. 53, no. 9, pp. 2437–2453, 2015.

[101] A. S. Padron, J. J. Alonso, F. Palacios, M. F. Barone, and M. S. Eldred, "Multi-fidelity uncertainty quantification: application to a vertical axis wind turbine under an extreme gust," in *15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, (Atlanta, GA), p. 3013, June 2014.

[102] R. E. Bartels, "Development, verification and use of gust modeling in the nasa computational fluid dynamics code fun3d," tech. rep., NASA, Langley Research Center, 2012.

[103] B. Müller, "Low-mach-number asymptotics of the navier-stokes equations," *Journal of Engineering Mathematics*, vol. 34, no. 1, pp. 97–109, 1998.

[104] F. M. White, *Viscous Fluid Flow*. New York: McGraw Hill Inc., 1974.

[105] F. Menter, "Zonal two equation $k-\omega$, turbulence models for aerodynamic flows," *23rd Fluid Dynamics, Plasmadynamics, and Lasers Conference*, July 1993.

[106] P. Spalart and S. Allmaras, "A one-equation turbulence model for aerodynamic flows," *30th Aerospace Sciences Meeting and Exhibit*, January 1992.

[107] J. M. Weiss and W. A. Smith, "Preconditioning applied to variable and constant density flows," *AIAA Journal*, vol. 33, no. 11, pp. 2050–2057, 1995.

[108] U. Riedel, "A finite volume scheme on unstructured grids for stiff chemically reacting flows," *Combustion Science and Technology*, vol. 135, no. 1-6, pp. 99–116, 1998.

[109] V. Gleize and A. L. Pape, "Low mach number preconditioning for unsteady flow in general "ALE" formulation," *44th AIAA Aerospace Sciences Meeting and Exhibit*, January 2006.

[110] P. D. Thomas and C. K. Lombard, "Geometric conservation law and its application to flow computations on moving grids," *AIAA Journal*, vol. 17, no. 10, pp. 1030–1037, 1979.

[111] J. Bonet and R. D. Wood, *Nonlinear Continuum Mechanics for Finite Element Analysis*. Cambridge University Press, 2nd ed., 2008.

[112] N. M. Newmark, "A Method of Computation for Structural Dynamics," *Journal of the Engineering Mechanics Division*, vol. 85, pp. 67–94, 1959.

[113] V. M. Korivi, P. A. Newman, G. W. Hou, and H. E. Jones, "An incremental strategy for calculating consistent discrete CFD sensitivity derivatives," tech. rep., National Aeronautics and Space Administration, Langley Research Center, 1992.

[114] R. M. Hicks and P. A. Henne, "Wing design by numerical optimization," *Journal of Aircraft*, vol. 15, no. 7, pp. 407–412, 1978.

[115] J. A. Samareh, "A survey of shape parameterization techniques," Tech. Rep. NASA/CP-1999-209136, NASA, 1999.

[116] R. P. Dwight, "Robust mesh deformation using the linear elasticity equations," *Proceedings of the Fourth International Conference on Computational Fluid Dynamics (ICCFD)*, pp. 401–406, 2006.

[117] R. Sanchez, *A coupled adjoint method for optimal design in fluid-structure interaction problems with large displacements*. PhD thesis, Imperial College London, 2017.

[118] S. Schotthöfer, B. Y. Zhou, T. A. Albring, and N. R. Gauger, "Windowing regularization techniques for unsteady aerodynamic shape optimization," *AIAA Aviation Forum*, 06 2020.

[119] M. Nightingale, R. Hewson, and M. Santer, "Multiscale optimisation of resonant frequencies for lattice based additive manufactured structures," *Structural and Multidisciplinary Optimization: computer-aided optimal design of stressed solids and multidisciplinary systems*, 2020.

[120] E. Jones, T. Oliphant, P. Peterson, *et al.*, "SciPy: Open source scientific tools for Python," 2001–. [Online; accessed June 2017].

[121] U. Ghia, K. Ghia, and C. Shin, "High-re solutions for incompressible flow using the navier-stokes equations and a multigrid method," *Journal of Computational Physics*, vol. 48, no. 3, pp. 387 – 411, 1982.

[122] E. Erturk, T. Corke, and C. Gokcol, "Numerical solutions of 2-d steady incompressible driven cavity flow at high reynolds numbers," *International Journal for Numerical Methods in Fluids*, vol. 48, no. 07, pp. 747 – 774, 2005.

[123] A. Roshko, "On the development of turbulent wakes from vortex streets," tech. rep., National Advisory Committee for Aeronautics, 1954.

[124] D. J. Tritton, "Experiments on the flow past a circular cylinder at low reynolds numbers," *Journal of Fluid Mechanics*, vol. 6, no. 4, p. 547–567, 1959.

[125] M. Braza, P. Chassaing, and H. Ha Minh, "Numerical study and physical analysis of the pressure and velocity field in the near wake of a circular cylinder," *Journal of Fluid Mechanics*, vol. 165, no. 04, pp. 79–130, 1986.

[126] D. Calhoun, "A cartesian grid method for solving the two-dimensional streamfunction-vorticity equations in irregular regions," *Journal of Computational Physics*, vol. 176, no. 2, pp. 231–275, 2002.

[127] D. Russell and Z. J. Wang, "A cartesian grid method for modeling multiple moving objects in 2d incompressible viscous flow," *Journal of Computational Physics*, vol. 191, no. 1, pp. 177–205, 2003.

[128] J.-I. Choi, R. C. Oberoi, J. R. Edwards, and J. A. Rosati, "An immersed boundary method for complex incompressible flows," *Journal of Computational Physics*, vol. 224, no. 2, pp. 757–784, 2007.

[129] R. Angell, P. Musgrove, and R. Galbraith, "Collected data for tests on a NACA 0015 aerofoil," G.U. Aero Report 8803, University of Glasgow, Department of Aerospace Engineering, 1988.

[130] I. Garrick, "Propulsion of a flapping and oscillating airfoil," NACA Report 567, National Advisory Committee for Aeronautics, 1937.

[131] W. Wall and E. Ramm, "Fluid-structure interaction based upon a stabilized (ale) finite element method," *Computational Mechanics. New Trends and Applications*, 01 1998.

[132] Y. Bazilevs, V. Calo, T. J. R. Hughes, and Y. Zhang, "Isogeometric fluid-structure interaction: Theory, algorithms, and computations," *Computational Mechanics*, vol. 43, pp. 3–37, 12 2008.

[133] A. Hubner and L. Reimer, "Gust encounter simulations of a generic transport aircraft and analysis of load alleviation potentials by control surface deflections using a rans-cfd-based multidisciplinary simulation environment," in *AIAA Aviation 2019 Forum*, (Dallas, TX), June 2019.

[134] A. Hübner and L. Reimer, "Multidisciplinary simulation for gust load alleviation control surface analysis," in *New Results in Numerical and Experimental Fluid Mechanics XII*, (Cham), pp. 576–585, Springer International Publishing, 2020.

[135] S. Haghighat, H. Liu, and J. R. R. A. Martins, "Application of model predictive control to gust loads alleviation systems," in *AIAA Atmospheric Flight Mechanics Conference*, (Chicago, IL), August 2009.

[136] S. Haghighat, H. H. T. Liu, and J. R. R. A. Martins, "Model-predictive gust load alleviation controller for a highly flexible aircraft," *Journal of Guidance, Control, and Dynamics*, vol. 35, no. 6, pp. 1751–1766, 2012.

[137] O. Burghardt, N. R. Gauger, and T. D. Economon, "Coupled adjoints for conjugate heat transfer in variable density incompressible flows," in *AIAA Aviation 2019 Forum*, (Dallas, TX), June 2019.

[138] A. Carnarius, F. Thiele, E. Özkaya, A. Nemili, and N. R. Gauger, "Optimal control of unsteady flows using a discrete and a continuous adjoint approach," *IFIP Conference on System Modeling and Optimization*, pp. 318–327, 2011.