The Development and Evaluation of a Fully-Coupled Monolithic Approach to Aero-structural Analysis and Optimization

by

Neil J. McCormick

A thesis submitted in conformity with the requirements
for the degree of Master of Applied Science
Graduate Department of Aerospace Engineering
University of Toronto

Abstract

The Development and Evaluation of a Fully-Coupled Monolithic Approach to Aero-structural Analysis and Optimization

Neil J. McCormick
Master of Applied Science
Graduate Department of Aerospace Engineering
University of Toronto
2013

A monolithic approach to aero-structural analysis and optimization has been developed and implemented. In contrast to a partitioned approach which uses individual fluid and structural solvers to solve their respective systems separately, the monolithic approach solves a fully-coupled system simultaneously, enforcing solution compatibility across the sub-system interfaces at each iteration. In this work, a three-field formulation is used, consisting of fluid, structural, and fluid mesh-movement sub-systems. The performance of the monolithic approach is characterized using 1-D unsteady and 2-D steady analysis problems, and compared with a partitioned approach. Four steady model aero-structural optimization problems are also investigated. Gradients of the objective function are computed using the discrete-adjoint and flow-sensitivity (direct) methods. In each case, the monolithic approach is shown to be a promising option for efficient aero-structural analysis and optimization, though the implementation requires additional development of coupling sub-matrices when compared to a partitioned approach.

Acknowledgements

I would like to extend my sincere gratitude to my supervisor, Professor David Zingg. His flexibility, patience, and belief in my abilities allowed me to work through my research with independence and helped me to gain a deeper understanding of the topics at hand. Our discussions have always been enlightening and your guidance has helped to resolve many roadblocks along the way. I could not have asked for a better mentor.

I would also like to thank the vibrant UTIAS community and all those who have provided help and support along the way during my time in Toronto. Much credit for maintaining my sanity goes to Tom Reist for the many discussions over afternoon coffee runs, and the occasional couch to sleep on.

Thank you to my colleagues at Martec who have shown genuine interest in my progress, and in particular to Rob Ripley for being accommodating and supportive throughout the process of competing my research.

I must also thank my parents for all of their love and support throughout my academic endeavours, and Megan for providing the balance in my life. This work would not have been possible without you.

Finally, I gratefully acknowledge the financial support provided by the Natural Sciences and Engineering Research Council of Canada (NSERC), the Government of Ontario through the Ontario Graduate Scholarship (OGS), and the University of Toronto.

Neil J. McCormick

University of Toronto Institute for Aerospace Studies
28 September 2013

Contents

List of Tables

List of Figures

List of Symbols and Abbreviations

Acronyms & Abbreviations

ALE	Arbitrary Lagrangian Eulerian
BC	Boundary conditions
BFGS	Broyden-Fletcher-Goldfarb-Shanno
CFD	Computational Fluid Dynamics
CSM	Computational Structural Mechanics
CSR	Compressed Sparse Row
FD	Finite difference
FSI	Fluid-Structure Interaction
GMRES	Generalized Minimal Residual
ILUT	Incomplete LU factorization with dual truncation
KS	Kreisselmeier-Steinhauser
L-BFGS-B	Large-scale Bound-constrained BFGS
SNOPT	Sparse Non-linear OPTimizer
SQP	Sequential Quadratic Programming

Alphanumeric

A	Area
A	Aerodynamic (subscript)
$\mathbf{A}$	Fluid Jacobian
$\mathcal{A}, \mathcal{B}$	Inviscid convective flux Jacobians
B_{s}	Strain-displacement matrix
C_l	Lift coefficient
C_l^*	Target lift coefficient
D	Artificial dissipation term
$\mathcal{D}$	Discrete artificial dissipation vector

D_{s}	Constitutive law matrix
d	Vector of structural nodal displacements
d_{tip}	Structural y-displacement of tab tip (top) node
E_{s}	Young's modulus
E,F	Inviscid convective flux vectors
e	Elemental (structural) quantity (subscript)
$\mathcal{E}$,$\mathcal{F}$	Discrete inviscid flux vectors
$\mathcal{G}$	Objective gradient
g_i	Elemental stress constraint
$\mathcal{H}_p$	Inverse Hessian
interior	Interior nodes (superscript)
$\mathcal{J}$	Objective function
J	Metric Jacobian
K	Stiffness matrix
K'	Interior node rows of stiffness matrix
$\mathcal{K}$	KS function value
M	Consistent mass matrix
M	Fluid mesh-movement (subscript)
N	Number of nodes in the domain
n_Q	Number of discrete state variables in global system
n_X	Number of design variables
Q	Solution vector
$\mathcal{Q}$	Discrete solution vector
R	Residual of equations
$\mathcal{R}$	Discrete residual vector
S	Structural (subscript)
s_p	BFGS search vector
T	Structural tab tip deflection
$\mathbf{T}$	Diagonal matrix of inverse local time steps
U,V	Contravariant velocities
W	Structural weight
w	Interface velocity (1-D)
X	Design variable
z_m	Structural movement design variable

Greek

α	Angle of attack
β_p	BFGS step size
ϵ	Structural strain tensor
η	Transformed spatial coordinate
Γ	Fluid-structure interface (subscript)
γ	Ratio of specific heats
ν	Poisson's ratio
ω	Fluctuation frequency
ω_{n}	Lowest structural natural frequency
ϕ	Flow-sensitivity vector
$\phi(\beta)$	BFGS objective along search direction
ψ	Adjoint vector
ρ	Fluid density
ρ_{s}	Structural density
$\rho_{\mathcal{K}}$	Positive KS function parameter
σ	Structural stress tensor
σ_{Vi}	Elemental von Mises stress
σ_Y	Elemental yield stress
ξ	Transformed spatial coordinate

Modifiers

$\sim$	Coordinate-transformed quantity (1-D) (overscript)
$\wedge$	Coordinate-transformed quantity (2-D) (overscript)
$+$	Perturbed variable (superscript)
∞	Freestream quantity (subscript)
$*$	Target value (superscript)

Chapter 1

Introduction

1.1 Motivation

There exist two primary motivating factors for increasing fuel efficiency in commercial aviation. First, the rising cost of aviation fuel has significantly increased operating expenses for airlines. In fact, in the last ten years, fuel has surpassed labour as the most significant operational expense, at 32.3% of the total operating cost [1]. Second, there is an increasing demand to lessen the environmental impact of commercial aircraft, which is achievable through reduced emissions of various compounds including greenhouse gases, nitrogen oxides, and sulphur oxides.

In recent years, advances in the fields of computational fluid dynamics (CFD) and computational structural mechanics (CSM) have allowed for increased efficiency and capabilities in the analysis and optimization of aerodynamic and structural systems in aircraft design. Historically, the development of these fields has been largely separate, which necessitates the communication of information between CFD and CSM solvers when considering a system with interactions between the fluid and structural domains. These communications, however, can be cumbersome and do not inherently enforce solution compatibility between the two fields. As such, these disciplinary advances, when coupled with the objective of increasing fuel efficiency of commercial aircraft, motivate the development of a computational tool to consider aerodynamic and structural disciplines, and their respective interactions, simultaneously.

We use fluid-structure interaction (FSI) as the collective term that describes the mutual interaction of a fluid flow with a movable or elastic structure. FSI considerations are critical in the design and analysis of many physical and engineered systems. From

an aircraft design perspective, FSI can be useful to understand both steady phenomena, such as structural divergence, as well as unsteady phenomena, such as aero-elastic flutter or other flow-induced oscillations. In this work, the interactions of aerodynamic flows with aircraft components, and the resulting structural stresses and deformations, are of particular interest.

There are two primary methods used to solve FSI problems. The first is a traditional partitioned approach, in which the fluid and structural systems are solved sequentially using existing CFD or CSM solution schemes until convergence to the coupled fluid-structure solution is achieved. The second is a monolithic approach, in which the fluid and structural systems are expressed as a single, tightly-coupled system, which is solved simultaneously with a novel, integrated CFD/CSM solution technique. The development of a monolithic approach for FSI analysis and aero-structural optimization is a primary focus of this work.

By considering aerodynamic and structural systems concurrently (with either a partitioned or monolithic approach), we are able to understand the dependence between the two systems and the trade-offs that occur when optimizing one system separately from the other. For instance, a typical aircraft design objective may be to maximize range. Consider the Breguet range equation [22]:

$$R = \frac{V}{c}\frac{C_L}{C_D}\ln\frac{W_i}{W_f} \tag{1.1}$$

where V is the aircraft speed, c is the specific fuel consumption, C_L and C_D are lift and drag coefficients (aerodynamic variables), and W_i and W_f are initial and final weights (structural variables). To maximize the value of R, one may intuitively wish to maximize the lift to drag ratio, but the effect of this maximization on structural weights (and conversely, the effect of structural weights on lift and drag) is not known without considering a fully-coupled system. Aero-structural optimization allows the designer to handle this interdependence.

For an aero-structural optimization scheme to be efficient, it requires two effective components: a method to compute the gradients of the objective with respect to the system variables, and an efficient aero-structural solver to perform the FSI analysis. Aero-structural optimization is a useful tool with many practical applications. For instance, in the typical case, the design phase of new aircraft uses a cut-and-try approach in which a design is analyzed using experimental or computational techniques, and then modified based on the designer's input to obtain a desired performance objective. However, as

the number of design variables increases, the ability of the designer to make correct choices becomes limited. As such, an efficient and robust aero-structural optimization scheme would integrate well into the design process and could provide a relatively fast and inexpensive method to develop and evaluate unconventional configurations for the next generation of aircraft.

1.2 Background and literature review

The field of fluid-structure interaction, and more generally, multi-physics analysis, is not new, and as such much work has been completed using both partitioned and monolithic approaches. For instance, Reuther et al. [32] coupled a finite-volume fluid approach with a linear finite-element structural model and obtained solutions through pseudo-time marching (strongly-coupled partitioned approach). As well, Maute et al. [24] solved the Euler equations coupled to a linear finite-element structural model using an iterative, staggered approach.

In general, a monolithic approach is usually more robust than a partitioned approach, with improved time-accuracy and stability properties [5]. Robinson-Mosher et al. [34] used a monolithic FSI approach to solve steady and unsteady incompressible flow problems and found that problems are inherently more stable when either the fluid or structural systems dominate the interaction. Heil et al. [8] used preconditioning and Newton's method with a variety of two- and three-dimensional test cases to demonstrate that monolithic solvers are competitive (compared to partitioned solvers), even for problems with weak fluid-structure interaction, to which partitioned solvers are well-suited. Degroote et al. [4] compared a partitioned approach to a monolithic approach for solving model FSI problems with an incompressible fluid and various structural configurations, and found the monolithic approach to be between 0.5 to 4 times faster than the partitioned. The relative performance depends both on size of the problem as well as level of interaction between the fluid and structure. Barcelos et al. [2] developed a Schur-Newton-Krylov method of solving coupled fluid-structure interaction problems. The approach uses an approximate Newton method to solve the system of coupled equations based on a Schur complement formulation of the steady-state aeroelastic problem. A 3-D monolithic approach with algebraic multigrid was employed by Gee et al. [6], which coupled the fluid and structural fields on coarse approximations to the problem to yield significant performance benefits over an existing monolithic approach.

When we refer to aero-structural optimization, we desire a design that is optimal with respect to a specified objective function of both aerodynamic and structural variables. Traditionally, most high-fidelity optimization schemes are based on either pure aerodynamic or structural analysis. Aerodynamic shape optimization has been used in wing design by, for example, Reuther et al. [33], Leung et al. [18], and Hicken et al. [11], while studies in wing structural optimization have been performed by Liu et al. [20] and Kennedy et al. [13]. However, these approaches cannot capture the primary benefit of an aero-structural approach over a single-discipline approach to optimization: the inclusion of the interdependence between the fluid and structural systems, achieved by performing optimization on the fully-coupled system. Although sequential optimization of aerodynamic and structural systems may also be completed, where, aerodynamic and structural optima are found in succession, the result is not a true optimum of the coupled system [23]. As such, the use of high-fidelity aero-structural analysis in shape optimization is useful to obtain meaningful results. For example, the partitioned approach by Maute et al. [24] was used to perform aero-structural optimization on aircraft wings. Martins et al. [23] used the coupled-adjoint equations in the sensitivity analysis of the partitioned aero-structural optimization of a full supersonic business jet with many design variables. More recently, a monolithic aero-structural approach with a hybrid algebraic-linear elasticity mesh-movement scheme was used by Kenway et al. [14] to perform aero-structural optimization with coupled-adjoint sensitivity analysis of a 3-D wing-body-tail configuration. These examples demonstrate that much work has been completed in the area of aero-structural optimization, but there exists a gap and a need for efficient and robust high-fidelity aero-structural optimization schemes capable of handling many design variables. In particular, more work is required for monolithic methods to fully understand the trade-offs with partitioned approaches, and for monolithic methods to reach their full potential in terms of efficiency and robustness.

In general, optimization methods fall into two categories: zeroth-order methods, and gradient-based methods. Zeroth-order methods, such as grid searching, are based on the value of the objective function only. These methods systematically search the design space and evaluate the objective function at various points. As a result, zeroth-order methods become computationally expensive as the number of design variables increases. In contrast, gradient-based methods rely on the values of the objective function as well as the gradient of the objective function with respect to the design variables. In this case, the optimization scheme is able to move in the direction of the optimum, typically

resulting in faster convergence with fewer objective function evaluations. However, only convergence to a local optimum is guaranteed. The method and efficiency of calculating the gradient (sensitivity analysis) is extremely important to the overall efficiency of the optimization scheme. The total sensitivity can be expressed analytically in the form of either the discrete-adjoint equations or the flow-sensitivity (direct) equations. With the flow-sensitivity approach the cost scales with the number of design variables independent of the number of constraints, while with the discrete-adjoint approach the cost scales with the number of constraints, independent of the number of design variables. As such, the adjoint method is preferred if the number of design variables is larger than the number of constraints, which is typically the case in a practical aero-structural optimization problem. Additional information on sensitivity analysis is given in Section 3.1.2.

1.3 Thesis objectives

The objective of the current work is to develop a fully-coupled, monolithic approach to aero-structural analysis and optimization that can be used towards the solution of practical problems in aero-structural design. The development of the monolithic methodology shall be developed in a general manner, such that the procedure can be applied to other systems of equations for other applications. The monolithic approach developed in this work shall be evaluated for accuracy, efficiency, and applicability to practical problems. The monolithic approach will be closely compared to a traditional, partitioned (loosely-coupled) approach to analysis and optimization.

The thesis objectives will be achieved through the use of two model problem configurations – a 1-D unsteady FSI problem and a 2-D steady aero-structural problem with a rigid airfoil/flexible tab geometry. Each problem will be solved with partitioned and monolithic approaches, and optimization problems will be formulated for each. Objective function and constraint gradients will be evaluated by means of both an aero-structural adjoint approach and a direct (flow-sensitivity) approach.

In general, the current work will address the following questions:

- What are the advantages and disadvantages to a monolithic approach to aero-structural analysis and optimization?
- How does a monolithic approach compare to a partitioned approach in terms of performance?

- What recommendations can be made for the use of a monolithic approach to aero-structural analysis and optimization in solving practical 3-D problems of interest?

1.4 Thesis outline

Following this introduction, the thesis is divided into six further chapters. Chapter 2 introduces the general framework of the aero-structural analysis performed in this work, including descriptions of partitioned and monolithic approaches. Chapter 3 describes the framework of the aero-structural optimization in this work, with a particular focus on the development of the discrete-adjoint method for gradient calculations. In general, Chapters 2 and 3 are presented in an abstract setting, where possible, with some elements specific to this work identified as such. Chapter 4 outlines the one-dimensional unsteady FSI problem, with subsections on FSI analysis and optimization. Chapter 5 describes the two-dimensional steady aero-structural problem, developing the monolithic coupling scheme and setting up the analysis and optimization framework. Chapter 6 presents results from sample analysis and optimization problems described in Chapters 4 and 5. Finally, Chapter 7 contains the conclusions of the thesis and recommendations for future work.

Chapter 2

Aero-structural Analysis Framework

In general, there are two approaches to performing an aero-structural, or any multi-system, analysis. A partitioned approach uses existing solvers and methods for the individual systems being considered, while a monolithic approach uses a novel, system-specific solution method to solve the entire domain, including multiple sub-systems, simultaneously. These two approaches are described in detail herein.

2.1 Partitioned approach

A partitioned approach uses existing solution techniques to solve the problem sub-systems individually. In a traditional FSI partitioned approach, the fluid and structural domains are categorized as loosely-coupled. In this scheme, the solutions of the sub-systems are communicated to each other only after each time step or iteration, resulting in asynchronous imposition of boundary conditions across the fluid-structure interface. Such approaches, also called staggered coupling methods, are simple to implement and can yield reasonable results for some problems, though have also shown to be energy-increasing and numerically unstable for others [25]. Thus, this approach is appropriate only for problems with relatively weak fluid-structure interaction. A diagram of this approach is illustrated in Figure 2.1, where a and b subscripts represent two individual sub-systems, such as fluid and structure, R is the residual of the governing equations, and Q is the solution vector.

In an unsteady, time-accurate case, a strongly-coupled partitioned approach is achieved through the inclusion of several sub-iterations at each time step or full iteration. Fluid and structural solutions are communicated after each sub-iteration and allowed to reach

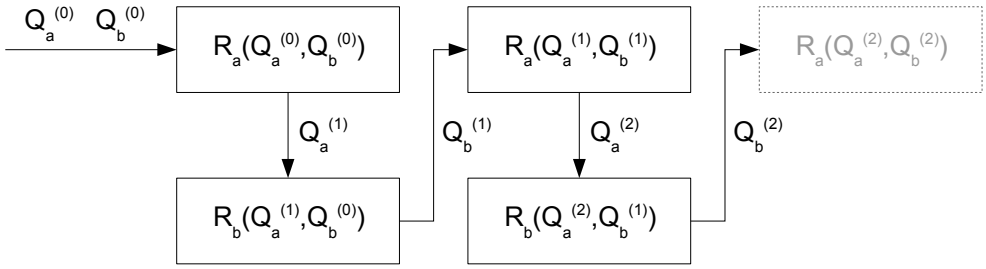

Figure 2.1: Loosely-coupled partitioned approach

a pseudo-steady state before advancing the global solution in time, enforcing solution compatibility across the fluid-structure interface at each iteration. However, each sub-iteration requires the solution of the fluid and structural equations, which can greatly increase computational cost for systems requiring many sub-iterations, such as those exhibiting strong fluid-structure interaction. A schematic of this method is shown in Figure 2.2, where dashed boundaries indicate regions in which sub-iterations between the two systems take place until some convergence criterion is met.

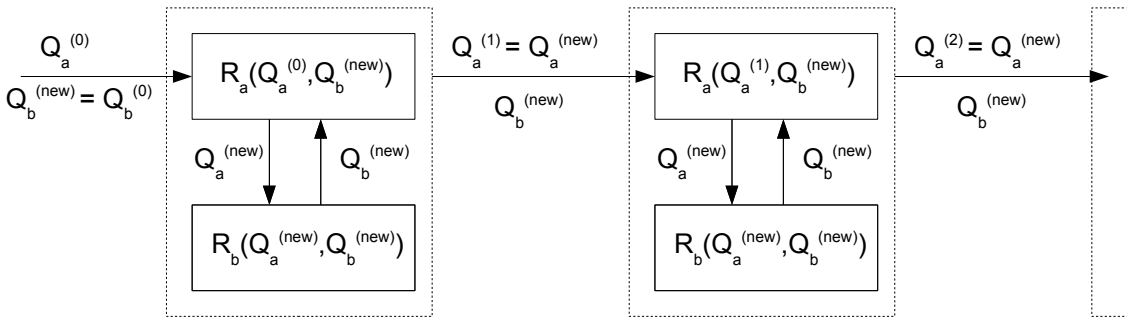

Figure 2.2: Strongly-coupled partitioned approach

2.2 Monolithic approach

In a monolithic approach, the governing equations of each individual sub-system are combined and expressed as a single system, which is then solved simultaneously. The result is an inherently strongly-coupled system which requires a single solution at each

time step or iteration to enforce solution compatibility across the interface. The coupling between the sub-systems must be developed by considering the interdependence between the individual components, and the system must be solved using a novel, integrated solution technique.

Consider a system of equations expressed in the following vector form

$$R(Q) = 0 \tag{2.1}$$

where R is the vector of residuals of each equation in the system, and Q is the vector of solution variables. Applying Newton's method, we select an initial guess, $Q^{(0)}$, which is unlikely to satisfy (2.1). In that case, we solve for a correction term, $\Delta Q^{(n)}$, where n is the iteration number, such that:

$$R(Q^{(n)} + \Delta Q^{(n)}) \approx 0 \tag{2.2}$$

Performing a Taylor series expansion of (2.2) about $Q^{(n)}$ yields

$$R(Q^{(n)} + \Delta Q^{(n)}) \approx R(Q^{(n)}) + \left.\frac{\partial R}{\partial Q}\right|_{(n)} \Delta Q^{(n)} \tag{2.3}$$

and setting the right-hand side to zero we obtain the following linear system

$$A^{(n)} \Delta Q^{(n)} = -R^{(n)} \tag{2.4}$$

where $R^{(n)} = R(Q^{(n)})$, and $A^{(n)}$ is the Jacobian matrix according to

$$A^{(n)} = \left.\frac{\partial R}{\partial Q}\right|_{(n)} \tag{2.5}$$

In this case we solve (2.4) for the solution update, ΔQ, and apply it to the current solution vector according to

$$Q^{(n+1)} = Q^{(n)} + \Delta Q^{(n)} \tag{2.6}$$

and repeat this iterative process until convergence is achieved.

Consider now a system of equations with multiple subsystems, such that

$$Q = \begin{bmatrix} Q_1 \\ Q_2 \\ \vdots \\ Q_N \end{bmatrix}, \qquad R(Q) = \begin{bmatrix} R_1(Q_1, Q_2, \cdots, Q_N) \\ R_2(Q_1, Q_2, \cdots, Q_N) \\ \vdots \\ R_4(Q_1, Q_2, \cdots, Q_N) \end{bmatrix} \tag{2.7}$$

We can now write Newton's method (2.4) for our system with multiple subsystems in the following form

$$\begin{bmatrix} \frac{\partial R_1}{\partial Q_1} & \frac{\partial R_1}{\partial Q_2} & \cdots & \frac{\partial R_1}{\partial Q_N} \\ \frac{\partial R_2}{\partial Q_1} & \frac{\partial R_2}{\partial Q_2} & \cdots & \frac{\partial R_2}{\partial Q_N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial R_N}{\partial Q_1} & \frac{\partial R_N}{\partial Q_2} & \cdots & \frac{\partial R_N}{\partial Q_N} \end{bmatrix}^{(n)} \begin{bmatrix} \Delta Q_1 \\ \Delta Q_2 \\ \vdots \\ \Delta Q_N \end{bmatrix}^{(n)} = - \begin{bmatrix} R_1(Q_1, Q_2, \cdots, Q_N) \\ R_2(Q_1, Q_2, \cdots, Q_N) \\ \vdots \\ R_N(Q_1, Q_2, \cdots, Q_N) \end{bmatrix}^{(n)} \tag{2.8}$$

where each term in the Jacobian matrix on the left-hand side represents a block sub-matrix. The diagonal blocks are the individual Jacobian matrices of each individual sub-system. The off-diagonal blocks are the monolithic coupling sub-matrices, which represent the dependence of the residuals of a set of equations for a given sub-system on the solution variables of another sub-system. For the case where the sub-system residuals do not depend explicitly on the independent variables of another sub-system, then the respective coupling Jacobian sub-matrix are zero. Therefore, it is possible that not all of the coupling sub-matrices appear in the monolithic Jacobian matrix, depending on the interdependence of the sub-systems in the problem.

Chapter 3

Aero-structural Optimization Framework

The goal of the optimization process is to find the minimum of an objective function, $\mathcal{J}$. The objective is a function of both the state variables of the system, $\mathcal{Q}$, and the design variables, X, which are modified as the optimization progresses. As well, the state variables depend implicitly on the design variables through the solution of the system, which can be expressed as:

$$\mathcal{J} = \mathcal{J}(\mathcal{Q}(X), X) \tag{3.1}$$

We also express the residuals of the governing equations of our system, equal to zero at the steady-state solution, in the following form:

$$\mathcal{R}(\mathcal{Q}(X), X) = 0 \tag{3.2}$$

This chapter describes the framework of the aero-structural optimization performed in this work and is divided into three distinct sections: gradient computation, optimization algorithms, and aero-structural constraints. The discussion of gradient calculation and optimization algorithms is general, while the description of aero-structural constraints pertains specifically to this work.

3.1 Gradient computation

3.1.1 Finite-difference gradient

The gradient of the objective function with respect to the design variables can be approximated using finite-differences. The vector of design variables, X, is expressed as a scalar quantity to distinguish between partial and total derivatives. For problems with multiple design variables, $\mathrm{d}\mathcal{J}/\mathrm{d}X$ is a row vector of length of the number of design variables (n_X). A second-order, centred-difference scheme can be used according to

$$\frac{\mathrm{d}\mathcal{J}}{\mathrm{d}X} \approx \frac{\mathcal{J}(\mathcal{Q}(X+\Delta X), X+\Delta X) - \mathcal{J}(\mathcal{Q}(X-\Delta X), X-\Delta X)}{2\Delta X} + O(\Delta X^2) \tag{3.3}$$

where ΔX is the finite-difference step size. It is important to note that this method requires two system solutions per design variable being considered. A first-order forward or backward scheme could also be used to compute the gradient, which would reduce the computational requirement to one system solve per design variable, but the cost remains prohibitive for any optimization problems of practical interest, which usually consist of a large number of design variables. In addition, finite-difference calculations are susceptible to truncation error for larger values of ΔX, and subtractive cancellation error for smaller values ΔX. As a result, the value of ΔX is difficult to select.

3.1.2 Discrete-adjoint and direct methods

It is desirable to compute the gradient such that the cost is not proportional to the number of design variables. Throughout this development, we express the vector of design variables, X, as a scalar quantity to clearly distinguish between partial and total derivatives. For problems with multiple design variables we note that $\mathrm{d}\mathcal{J}/\mathrm{d}X$ and $\partial\mathcal{J}/\partial X$ are row vectors of length of the number of design variables (n_X), $\partial\mathcal{J}/\partial\mathcal{Q}$ is a row vector of length of the number of state variables in the global system (n_Q), and $\mathrm{d}\mathcal{Q}/\mathrm{d}X$ is a matrix of n_Q rows and n_X columns.

From (3.1) the gradient of the objective function with respect to the design variable, X, can be expressed as:

$$\frac{\mathrm{d}\mathcal{J}}{\mathrm{d}X} = \frac{\partial\mathcal{J}}{\partial X} + \frac{\partial\mathcal{J}}{\partial\mathcal{Q}}\frac{\mathrm{d}\mathcal{Q}}{\mathrm{d}X} \tag{3.4}$$

As well, we can also differentiate the residuals of the governing equations (3.2) with respect to X using the chain rule:

$$\frac{\mathrm{d}\mathcal{R}}{\mathrm{d}X} = \frac{\partial\mathcal{R}}{\partial X} + \frac{\partial\mathcal{R}}{\partial\mathcal{Q}}\frac{\mathrm{d}\mathcal{Q}}{\mathrm{d}X} = 0 \tag{3.5}$$

Solving (3.5) for $\mathrm{d}\mathcal{Q}/\mathrm{d}X$ and substituting in (3.4), we obtain the following:

$$\frac{\mathrm{d}\mathcal{J}}{\mathrm{d}X} = \frac{\partial \mathcal{J}}{\partial X} - \frac{\partial \mathcal{J}}{\partial \mathcal{Q}} \left(\frac{\partial \mathcal{R}}{\partial \mathcal{Q}} \right)^{-1} \frac{\partial \mathcal{R}}{\partial X} \tag{3.6}$$

The expression for the gradient given in (3.6) can be solved by defining one of two intermediate problems. The first method, called the direct or flow-sensitivity method, requires the solution of the following intermediate problem

$$\left(\frac{\partial \mathcal{R}}{\partial \mathcal{Q}} \right) \phi = \frac{\partial \mathcal{R}}{\partial X} \tag{3.7}$$

for ϕ, and then evaluates the gradient by substituting ϕ back into (3.6), which yields:

$$\frac{\mathrm{d}\mathcal{J}}{\mathrm{d}X} = \frac{\partial \mathcal{J}}{\partial X} - \frac{\partial \mathcal{J}}{\partial \mathcal{Q}}^T \phi \tag{3.8}$$

The left-hand side of flow-sensitivity equation (3.7) is the same as in the solution of the analysis problem, and is evaluated at the converged solution. An important property of the direct approach in (3.7) is that this system must be solved once for every design variable being considered. However, the result yields the gradients for as many constraint functions as desired, since ϕ is independent of $\mathcal{J}$.

In the second approach, called the adjoint method, an alternative intermediate problem is solved

$$\left(\frac{\partial \mathcal{R}}{\partial \mathcal{Q}} \right)^T \psi = \frac{\partial \mathcal{J}}{\partial \mathcal{Q}} \tag{3.9}$$

for ψ, or the adjoint vector, which is then substituted back into the gradient expression given in (3.6), which yields:

$$\frac{\mathrm{d}\mathcal{J}}{\mathrm{d}X} = \frac{\partial \mathcal{J}}{\partial X} - \psi^T \frac{\partial \mathcal{R}}{\partial X} \tag{3.10}$$

The left-hand side of the adjoint equation (3.9) differs from the left-hand side of the analysis problem by a transpose operator. The primary advantage of the adjoint approach is that the adjoint problem needs only to be solved once per function of interest, rather than once per design variable as in the direct approach. This is particularly beneficial for aero-structural optimization problems in which many design variables are considered.

The additional partial derivatives used in (3.7)-(3.10) are evaluated analytically where possible. Finite-difference approximations can also be used to evaluate the partial derivative terms with respect to each design variable, X, according to

$$\frac{\partial \mathcal{J}}{\partial X} \approx \frac{\mathcal{J}(\mathcal{Q}, X + \Delta X) - \mathcal{J}(\mathcal{Q}, X - \Delta X)}{2\Delta X} + O(\Delta X^2) \tag{3.11}$$

$$\frac{\partial \mathcal{R}}{\partial X} \approx \frac{\mathcal{R}(\mathcal{Q}, X + \Delta X) - \mathcal{R}(\mathcal{Q}, X - \Delta X)}{2\Delta X} + O(\Delta X^2) \tag{3.12}$$

where ΔX is the finite-difference step size. As well, the entries in the partial derivative of the objective, $\mathcal{J}$, with respect to the solution variables, $\mathcal{Q}$, are approximated by

$$\frac{\partial \mathcal{J}}{\partial \mathcal{Q}_i} \approx \frac{\mathcal{J}(\mathcal{Q} + \Delta\mathcal{Q}_i e_i, X) - \mathcal{J}(\mathcal{Q} - \Delta\mathcal{Q}_i e_i, X)}{2\Delta\mathcal{Q}_i} + O({\Delta\mathcal{Q}_i}^2) \tag{3.13}$$

where $\mathcal{Q}_i$ is the ith entry in the solution vector $\mathcal{Q}$, $\Delta\mathcal{Q}_i$ is the finite-difference step size, and e_i is a vector of all zeros except for a value of unity in the ith entry. Each of these partial derivative computations requires two objective or two residual vector evaluations respectively, and no additional aero-structural solves are required. As such, the entire adjoint method is primarily dependent on the number of state variables of the system, with minimal dependence on the number of design variables being considered in the optimization process.

Monolithic approach

If we consider a discrete system with multiple subsystems, such that

$$\mathcal{Q} = \begin{bmatrix} \mathcal{Q}_1 \\ \mathcal{Q}_2 \\ \vdots \\ Q_N \end{bmatrix}, \qquad \mathcal{R}(\mathcal{Q}) = \begin{bmatrix} \mathcal{R}_1(\mathcal{Q}_1, \mathcal{Q}_2, \cdots, \mathcal{Q}_N) \\ \mathcal{R}_2(\mathcal{Q}_1, \mathcal{Q}_2, \cdots, \mathcal{Q}_N) \\ \vdots \\ \mathcal{R}_N(\mathcal{Q}_1, \mathcal{Q}_2, \cdots, \mathcal{Q}_N) \end{bmatrix} \tag{3.14}$$

then we can express the adjoint equation (3.9) in the following form:

$$\begin{bmatrix} \frac{\partial \mathcal{R}_1}{\partial \mathcal{Q}_1}^T & \frac{\partial \mathcal{R}_2}{\partial \mathcal{Q}_1}^T & \cdots & \frac{\partial \mathcal{R}_N}{\partial \mathcal{Q}_1}^T \\ \frac{\partial \mathcal{R}_1}{\partial \mathcal{Q}_2}^T & \frac{\partial \mathcal{R}_2}{\partial \mathcal{Q}_2}^T & \cdots & \frac{\partial \mathcal{R}_N}{\partial \mathcal{Q}_2}^T \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \mathcal{R}_1}{\partial \mathcal{Q}_N}^T & \frac{\partial \mathcal{R}_2}{\partial \mathcal{Q}_N}^T & \cdots & \frac{\partial \mathcal{R}_N}{\partial \mathcal{Q}_N}^T \end{bmatrix} \begin{bmatrix} \psi_1 \\ \psi_2 \\ \vdots \\ \psi_N \end{bmatrix} = \begin{bmatrix} \frac{\partial \mathcal{J}}{\partial \mathcal{Q}_1} \\ \frac{\partial \mathcal{J}}{\partial \mathcal{Q}_2} \\ \vdots \\ \frac{\partial \mathcal{J}}{\partial \mathcal{Q}_N} \end{bmatrix} \tag{3.15}$$

where the transpose operator has been applied to the block sub-matrices on the left-hand side. We refer to (3.15) as the monolithic discrete-adjoint equations, which are solved once per function of interest (i.e., objective function or constraint). Once solved, the gradient of the objective function with respect to the design variables is found by evaluating (3.10).

Similarly, if we consider the same multi-system approach, we can express the flow-

sensitivity equations in the following form:

$$\begin{bmatrix} \frac{\partial \mathcal{R}_1}{\partial \mathcal{Q}_1} & \frac{\partial \mathcal{R}_1}{\partial \mathcal{Q}_2} & \cdots & \frac{\partial \mathcal{R}_1}{\partial \mathcal{Q}_N} \\ \frac{\partial \mathcal{R}_2}{\partial \mathcal{Q}_1} & \frac{\partial \mathcal{R}_2}{\partial \mathcal{Q}_2} & \cdots & \frac{\partial \mathcal{R}_2}{\partial \mathcal{Q}_N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \mathcal{R}_N}{\partial \mathcal{Q}_1} & \frac{\partial \mathcal{R}_N}{\partial \mathcal{Q}_2} & \cdots & \frac{\partial \mathcal{R}_N}{\partial \mathcal{Q}_N} \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_N \end{bmatrix} = \begin{bmatrix} \frac{\partial \mathcal{R}_1}{\partial X} \\ \frac{\partial \mathcal{R}_2}{\partial X} \\ \vdots \\ \frac{\partial \mathcal{R}_N}{\partial X} \end{bmatrix} \tag{3.16}$$

which is solved for each design variable, X. We refer to (3.16) as the monolithic flow-sensitivity equations. After solving for ϕ, the gradient is found by evaluating (3.8).

Partitioned approach

We define a partitioned (staggered) approach to solving the discrete-adjoint equations by writing (3.15) in the following form [22]:

$$\left(\frac{\partial \mathcal{R}_M}{\partial \mathcal{Q}_M}\right)^T \psi_M^{(n+1)} = \frac{\partial \mathcal{J}}{\partial \mathcal{Q}_M} - \sum_{k=1}^{M-1} \left(\frac{\partial \mathcal{R}_k}{\partial \mathcal{Q}_M}\right)^T \psi_k^{(n+1)} - \sum_{k=M+1}^{N} \left(\frac{\partial \mathcal{R}_k}{\partial \mathcal{Q}_M}\right)^T \psi_k^{(n)} \tag{3.17}$$

for each equation $M = 1, \ldots, N$, where the equations are solved in order at each iteration, n, until some convergence criterion is achieved, and the solution ψ has converged. The staggered solution scheme is a block Gauss-Seidel iterative solution of (3.15). Once the adjoint vector, ψ, has been calculated the gradient, $\mathcal{G}$, is calculated from (3.10) in the following form:

$$\mathcal{G} = \frac{\partial \mathcal{J}}{\partial X} - \psi_1^T \frac{\partial \mathcal{R}_1}{\partial X} - \psi_2^T \frac{\partial \mathcal{R}_2}{\partial X} - \cdots - \psi_N^T \frac{\partial \mathcal{R}_N}{\partial X} \tag{3.18}$$

In a similar manner we can express a block Gauss-Seidel iterative solution of the monolithic flow-sensitivity equation (3.16) in the following form [22]:

$$\left(\frac{\partial \mathcal{R}_M}{\partial \mathcal{Q}_M}\right) \phi_M^{(n+1)} = \frac{\partial \mathcal{R}_M}{\partial X} - \sum_{k=1}^{M-1} \left(\frac{\partial \mathcal{R}_M}{\partial \mathcal{Q}_k}\right) \phi_k^{(n+1)} - \sum_{k=M+1}^{N} \left(\frac{\partial \mathcal{R}_M}{\partial \mathcal{Q}_k}\right) \phi_k^{(n)} \tag{3.19}$$

for each equation $M = 1, \ldots, N$, where the equations are solved in order at each iteration, n, until a converged solution of ϕ is obtained. At this point, the gradient is calculated from (3.8) in the following form:

$$\mathcal{G} = \frac{\partial \mathcal{J}}{\partial X} - \frac{\partial \mathcal{J}}{\partial \mathcal{Q}_1}^T \phi_1 - \frac{\partial \mathcal{J}}{\partial \mathcal{Q}_2}^T \phi_2 - \cdots - \frac{\partial \mathcal{J}}{\partial \mathcal{Q}_N}^T \phi_N \tag{3.20}$$

3.2 Optimization algorithms

3.2.1 BFGS method

The goal of the optimizer is to modify the each of the design variables, X, in order to minimize an objective function, $\mathcal{J}(X)$. When the objective function is minimized, then the L_2-norm of the gradient vector is zero. For this work, the BFGS (Broyden-Fletcher-Goldfarb-Shanno) algorithm is used to drive the L_2-norm of the gradient vector to zero. The BFGS algorithm features a quasi-Newton approach coupled with a line search procedure. A brief description of the algorithm is provided herein; for further details the reader is directed to [28, 38, 19, 27].

The optimization scheme starts with an initial set of design variables, X_0, and the objective value, $\mathcal{J}(X_0)$, evaluated at the initial design variable values. The optimizer determines a search vector, s_p, at each optimization iteration, p, in the direction of the local optimum according to

$$s_p = -\mathcal{H}_p \mathcal{G}_p \tag{3.21}$$

where $\mathcal{H}$ is the approximation of the inverse Hessian matrix, and $\mathcal{G}$ is the gradient of the objective function. Then, once the search vector is determined, the design variables are updated with

$$X_{p+1} = X_p + \beta_p s_p \tag{3.22}$$

where β_p is a step size determined by a line-search procedure. As the BFGS algorithm is a quasi-Newton method, an approximation of the inverse Hessian is used rather than the exact inverse Hessian. A BFGS secant update is used to generate a series of increasingly-accurate approximations to the inverse Hessian, $\mathcal{H}$, based on the successive values of the gradients throughout the optimization scheme [28]:

$$\mathcal{H}_{p+1} = \mathcal{H}_p - \frac{\mathcal{H}_p v_p (\mathcal{H}_p v_p)^T}{{v_p}^T \mathcal{H}_p v_p} + \frac{\delta_p {\delta_p}^T}{{\delta_p}^T v_p} + {v_p}^T \mathcal{H}_p v_p (r r^T) \tag{3.23}$$

where

$$r = \frac{\delta_p}{{\delta_p}^T v_p} - \frac{\mathcal{H}_p v_p}{{v_p}^T \mathcal{H}_p v_p} \tag{3.24}$$

and

$$\delta_p = X_{p+1} - X_p \tag{3.25}$$

$$v_p = \mathcal{G}_{p+1} - \mathcal{G}_p \tag{3.26}$$

At the first optimizer iteration, when previous gradient information is not available to approximate the inverse Hessian, the identity matrix is used ($\mathcal{H}_0 = I$), which sets the search vector to the direction of steepest descent. After the first iteration but before the first inverse Hessian approximation according to (3.23), the initial inverse Hessian is set to:

$$\mathcal{H}_0 = \frac{{v_p}^T s_p}{{v_p}^T v_p} I \tag{3.27}$$

A full-Newton step requires $\beta = 1$, however at times during the iteration (particularly during the initial iterations), a full-Newton step may not result in sufficient decrease of the objective function, or may lead to divergence of the optimization algorithm. A line search procedure is used to find a new step size, β, until an acceptable update of the design variables is found. In particular, an acceptable step size satisfies the strong Wolfe conditions: the *sufficient decrease condition* and the *curvature condition* [28]. We define a function describing the objective along the search direction

$$\phi(\beta) = \mathcal{J}(X_p + \beta s_p) \tag{3.28}$$

and the derivative of this function is expressed as

$$\phi'(\beta) = \mathcal{G}(X_p + \beta s_p)^T s_p \tag{3.29}$$

We express the sufficient decrease and curvature conditions as

$$\phi(\beta) \leq \phi(0) + \beta c_1 \phi'(0) \tag{3.30}$$

$$|\phi'(\beta)| \leq c_2 |\phi'(0)| \tag{3.31}$$

where $c_1 = 10^{-4}$ and $c_2 = 0.9$ are used in this work [28]. If the step size, β, does not satisfy both conditions, then a line search procedure is used to determine an updated step size. In this work, the line search procedure of Moré and Thuente [26] is used, which uses a sequence of polynomial interpolations to compute the step size updates. The line search allows for $\beta > 1$. This procedure is repeated until the conditions in (3.30) and (3.31) are satisfied to a user-defined maximum number of step-search iterations. If the conditions cannot be satisfied within the maximum number of iterations, then the optimizer is considered stalled, and the inverse Hessian is reset to the identity matrix.

The BFGS scheme implemented in this work is the large-scale bound-constrained optimization algorithm (L-BFGS-B), which accepts constraints on the design variables in the form

$$l \leq X \leq u \tag{3.32}$$

where l and u are constant lower and upper bounds, respectively [38].

3.2.2 SQP method

Sequential quadratic programming (SQP) methods can be used to solve nonlinear optimization problems with a series of linear or nonlinear inequality constraints. An SQP method uses a smooth Lagrangian merit function consisting of a combination of a quadratic model of the objective and a linearization of the constraint functions. For this work, the SNOPT optimization package is used to solve constrained aero-structural optimization problems [7]. SNOPT uses a BFGS method to approximate the inverse Hessian matrix and a reduced-Hessian semi-definite QP solver for the QP sub-problems. For additional information the reader is directed to [7].

3.3 Aero-structural constraints

Constraints are required for practical aero-structural optimization problems to ensure the optimum value returned by the optimization scheme is contained within the feasible design space. Constraints are one important component of a full aero-structural optimization problem definition, which also requires a set of governing equations, design variables, and objective function. Typical values to be constrained in an aero-structural optimization include, but are not limited to: structural weight, geometric constraints (i.e. minimum thickness to fit structural members, nodal tolerances to maintain topology), lift, pitching moment, and maximum stress within a structure. The governing equations are described in Chapters 4 and 5, and the optimization problem definitions, including the objective functions and design variables, are given in Sections 5.2.1-5.2.3. Constraints used in this study are described next.

3.3.1 Maximum elemental stress constraint

A maximum elemental stress constraint is required to ensure that the stress in an individual structural element does not exceed some maximum design yield stress. One could impose individual stress constraints on each element in the domain, but this would result in many constraints ($\geq 10^3$), each of which requiring separate derivatives to be handled individually by the optimizer.

To avoid the cost of large numbers of constraints, aggregate methods can be used to lump multiple constraints into a single, conservative constraint function. One such method is through the use of the Kreisselmeier-Steinhauser (KS) function. Consider elemental stress constraints in the following form

$$g_i = 1 - \frac{\sigma_{Vi}}{\sigma_Y} \geq 0 \tag{3.33}$$

where σ_{Vi} is the elemental von Mises stress, and σ_Y is the yield stress of the material. We lump the constraints for each element using a KS function in the following form

$$\mathcal{K} = -\frac{1}{\rho_\mathcal{K}} \ln \left(\sum_i \mathrm{e}^{-\rho_\mathcal{K} g_i} \right) \tag{3.34}$$

where i indicates the structural element index, and $\rho_\mathcal{K}$ is a positive weighting parameter. The function represents a lower bound envelope of all of the stress inequality constraints, g_i, and $\rho_\mathcal{K}$ is used to set how close the KS constraint is to the minimum of each of the elemental constraints individually. For this work a value of $\rho_\mathcal{K} = 100$ is used. For $\mathcal{K} \geq 0$ this conservative lumping method ensures that each of the individual constraints are satisfied. However, the optimum determined using the aggregate KS function is not the same optimum as would be found by handling the constraints separately [22].

3.3.2 Lift constraint

A lift constraint is used to ensure that each steady solution obtained from the aero-structural optimization scheme maintains some target lift coefficient value to ensure a feasible design in flight conditions. The equality constraint is expressed in the following form

$$C_l - C_l^* = 0 \tag{3.35}$$

where C_l is the lift coefficient evaluated over the entire aerodynamic surface, and C_l^* is the target lift coefficient value permitted.

Chapter 4

One-Dimensional Model Problem

In the development of a full-scale FSI solver, the use of simplified model problems is a useful exercise to gain experience and overall understanding, with the idea that the solution strategies developed can be extended to practical 2-D and 3-D problems. Nordström and Eriksson examined a simple 1-D fluid-spring problem in their fluid-structure interaction analysis [29], Lefrançois and Boufflet solved a 1-D FSI problem with a piston-spring configuration [17], and Blom solved a similar configuration with a monolithic approach [3]. These ideas motivate the use of a 1-D model problem. In this work, the 1-D problem is used to develop a monolithic framework and implementation methodology, and to test the approach.

4.1 Analysis

The model problem consists of a 1-D fluid flow interacting with a 1-D elastic bar finite element structure. For a base case, the fluid domain consists of 50 nodes, while the structural domain consists of 10 nodes. The number of nodes in each domain can be varied between cases. The fluid exerts a pressure at the fluid-structure interface, causing a structural deformation, which in turn affects the fluid solution and the pressure that is exerted. A diagram of the model problem is provided in Figure 4.1.

4.1.1 Formulation and governing equations

The fluid domain is governed by the 1-D Euler equations in conservative form:

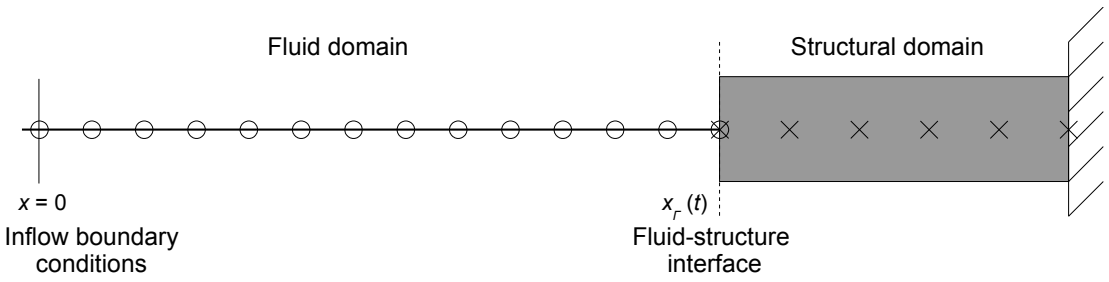

Figure 4.1: A diagram of the one-dimensional FSI model problem

$$\frac{\partial Q_A}{\partial t} + \frac{\partial E}{\partial x} = 0, \qquad 0 \leq x \leq x_\Gamma(t) \tag{4.1}$$

where

$$Q_A = \begin{bmatrix} \rho \\ \rho u \\ e \end{bmatrix}, \qquad E = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ u(e+p) \end{bmatrix} \tag{4.2}$$

and ρ is the fluid density, u the velocity, e the total energy per unit volume, and p the pressure. The relation between e and p is supplied by the equation of state of an ideal gas

$$p = (\gamma - 1)(e - \rho u^2/2) \tag{4.3}$$

where γ is the ratio of specific heats, C_p/C_v.

The structural domain is governed by a finite-element bar model, described by

$$M\ddot{d} + Kd = f(t) \tag{4.4}$$

where M is the global consistent mass matrix, K is the global stiffness matrix, d is the vector of nodal displacements, and $f(t)$ is the global force vector. For an individual bar element, we have

$$M_e = \frac{\rho_\mathrm{s} AL}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}, \qquad K_e = \frac{AE_\mathrm{s}}{L}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}, \qquad f_e = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}, \qquad d_e = \begin{bmatrix} d_1 \\ d_2 \end{bmatrix} \tag{4.5}$$

where A is the cross-sectional area of the bar, L is the length of the bar element, ρ_s is the structural density, E_s is Young's modulus, and the e subscript indicates elemental quantities. We express (4.4) in first-order form:

$$Y_0\dot{\mathcal{Q}}_S + Y_1\mathcal{Q}_S = F_\mathrm{s} \tag{4.6}$$

where

$$Y_0 = \begin{bmatrix} K & 0 \\ 0 & M \end{bmatrix}, \quad Y_1 = \begin{bmatrix} 0 & -K \\ K & 0 \end{bmatrix}, \quad \mathcal{Q}_S = \begin{bmatrix} d \\ \dot{d} \end{bmatrix}, \quad F_{\mathrm{s}} = \begin{bmatrix} 0 \\ f(t) \end{bmatrix} \tag{4.7}$$

and d and $\dot{d}$ are vectors of nodal displacement and velocity, respectively, and $f(t)$ is a vector of applied nodal forces. For this problem, the force vector is zero for each node except for the first node (at the fluid structure interface).

4.1.2 Fluid-structure coupling and boundary conditions

Coupling of the fluid and structure is required to account for the mutual interactions between the two domains. Two coupling conditions must be satisfied in the model problem. First, the pressure exerted by the fluid at the interface must yield the force exerted on the structure at the interface:

$$Ap_\Gamma = f_\Gamma \tag{4.8}$$

where f_Γ is the force exerted on the structure at the interface, p_Γ the fluid pressure at the interface, and A the area of the structure at the interface. Second, the velocity of the structure at the interface must equal the velocity of the fluid at the interface:

$$\dot{d}_\Gamma = u_\Gamma \tag{4.9}$$

where $\dot{d}_\Gamma$ is the structural velocity at the interface, and u_Γ is the fluid velocity at the interface.

The inflow boundary is specified with a periodic pressure boundary condition

$$p_{BC} = p_{\mathrm{amp}} \sin(\omega t) + p_0 \tag{4.10}$$

where p_{amp} is the specified amplitude of pressure fluctuations, p_0 is the mean pressure, and ω is the frequency. The displacement and velocity of the fixed structural node at the far right of the problem configuration are both zero.

4.1.3 Coordinate transformation

A time-varying coordinate transformation is applied to (4.1) such that the computational domain remains uniform regardless of the position of the fluid-structure interface. We introduce a coordinate transformation:

$$\begin{aligned} \tau &= t \\ \xi &= \xi(x,t) \end{aligned} \tag{4.11}$$

where τ and ξ are the transformed temporal and spatial variables, respectively, mapping each point in physical space to a point in computational space. Chain-rule expansions are used to represent the derivatives according to

$$\begin{aligned} \frac{\partial}{\partial x} &= \frac{\partial \xi}{\partial x}\frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial t} &= \frac{\partial}{\partial \tau} + \frac{\partial \xi}{\partial t}\frac{\partial}{\partial \xi} \end{aligned} \tag{4.12}$$

We select a transformation to a fixed spatial domain such that

$$\xi = \frac{x}{x_\Gamma(t)} \tag{4.13}$$

and we use (4.13) to evaluate the partial derivatives appearing in (4.12) analytically:

$$\frac{\partial \xi}{\partial x} = \frac{1}{x_\Gamma(t)}, \qquad \frac{\partial \xi}{\partial t} = -\frac{x x_\Gamma{}'(t)}{x_\Gamma(t)^2} \tag{4.14}$$

where $x_\Gamma{}'(t)$ is the velocity of the fluid-structure interface. Applying the chain-rule derivatives to (4.1) we obtain the coordinate-transformed 1-D Euler equations:

$$x_\Gamma(t)\frac{\partial Q_A}{\partial \tau} + \frac{\partial \tilde{E}}{\partial \xi} = 0, \qquad 0 \le \xi \le 1 \tag{4.15}$$

where

$$\tilde{E} = E - \xi x_\Gamma{}'(t) Q_A \tag{4.16}$$

Many fluid-structure interaction problems, and other fluid flow problems with a deforming fluid mesh, are solved using an Arbitrary Lagrangian Eulerian (ALE) approach, where each point in the fluid domain is moved at an arbitrary velocity [17]. The ALE formulation is a combination of an Eulerian reference frame, in which the observer is fixed and observes the fluid particles passing, and a Lagrangian approach, where the observer moves at the velocity of the particles themselves. If we consider the physical implications of our time-dependent coordinate-transformation, each point in the fluid domain is moving at a velocity, w, as the fluid-structure interface moves and the nodes remain uniformly-spaced. A point in the fluid domain at the interface moves at the velocity of the interface, while a point at the inflow of the fluid domain remains fixed. Each point in between moves at some fraction of the interface velocity. Noting that $\xi x_\Gamma{}'(t) = w$, we can express (4.16) as

$$\tilde{E} = E - wQ_A = \begin{bmatrix} \rho(u-w) \\ \rho u(u-w) + p \\ e(u-w) + up \end{bmatrix} \tag{4.17}$$

The coordinate-transformed Euler equations (4.15) are equivalent to an ALE formulation of the 1-D Euler equations, where the fluid domain moves at an arbitrary velocity, w. We also note that the Eulerian description can be recovered by setting $w = 0$, while a pure Lagrangian description is obtained with $w = u$.

4.1.4 Spatial discretization

Fluid discretization

The transformed 1-D Euler equations (4.15) are discretized on a uniform fluid mesh containing N_A nodes from $\xi = 0$ to $\xi = 1$, where $j = 1$ is the fluid left-hand boundary node and $j = N_A$ is the node on the fluid-structure interface. The equations are written in semi-discrete form at each interior node in the fluid mesh, j, where $\tau = t$ from (4.11):

$$\frac{\mathrm{d}}{\mathrm{d}t}(Q_A)_j = -\frac{1}{x_\Gamma(t)}\left[\delta_\xi \tilde{E}_j - (D_\xi)_j\right] \qquad j = 2, \ldots, N_A - 1 \tag{4.18}$$

where δ_ξ is a second-order spatial difference operator such that

$$\delta_\xi \tilde{E}_j = \frac{\tilde{E}_{j+1} - \tilde{E}_{j-1}}{2\Delta\xi} \tag{4.19}$$

and D_ξ is the scalar artificial dissipation term containing both second- and fourth-difference terms [31]. We write (4.18) in the following form:

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{Q}_A^{\text{interior}} + \mathcal{R}_A^{\text{interior}}(\mathcal{Q}_A) = 0 \tag{4.20}$$

where $\mathcal{Q}_A$ is a column vector of discrete flow variables, $(Q_A)_j$, $\mathcal{Q}_A^{\text{interior}}$ is a column vector of discrete flow variables on interior nodes, and $\mathcal{R}_A^{\text{interior}}$ is a column vector of discrete residuals for interior nodes, $(\mathcal{R}_A^{\text{interior}})_j$. We express our discrete residual vector on the interior nodes as

$$\mathcal{R}_A^{\text{interior}}(\mathcal{Q}_A, \mathcal{Q}_S) = \frac{1}{x_\Gamma(t)}\left[\delta_\xi \tilde{\mathcal{E}} - \mathcal{D}_\xi\right] \tag{4.21}$$

where $\mathcal{E}$ is a column vector of inviscid fluxes at each node, E_j, and $\mathcal{D}_\xi$ is a column vector of artificial dissipation terms at each node, $(D_\xi)_j$. We express the position of the fluid-structure interface, $x_\Gamma(t)$, in terms of the structural system from

$$x_\Gamma(t) = x_\Gamma(0) + d_1(t) \tag{4.22}$$

where $d_1(t)$ is the displacement of the structural node on the fluid-structure interface. We also note that the transformed inviscid flux vector, $\tilde{\mathcal{E}}$, is a function of the structural

velocity at the fluid-structure interface from (4.16):

$$\tilde{\mathcal{E}} = \mathcal{E} - \xi \dot{d}_1 \mathcal{Q}_A \tag{4.23}$$

At the left-hand fluid boundary, the pressure is specified from the periodic pressure boundary condition (4.10) and momentum is extrapolated from the interior. For inflow ($u \geq 0$) density is specified as free-stream density, ρ_∞, and for outflow ($u < 0$) the density is extrapolated from the interior. The discretized boundary conditions at the left-hand fluid boundary are written as

$$\begin{aligned} \rho_1 &= \rho^{(\text{left})} \\ (\rho u)_1 &= (\rho u)_2 \\ p_1 &= p_{BC} \end{aligned} \tag{4.24}$$

where $\rho^{(\text{left})} = \rho_\infty$ for inflow and $\rho^{(\text{left})} = \rho_2$ for outflow. The discrete residual vector of the left-hand fluid boundary node is expressed as

$$\mathcal{R}_A^{\text{left}}(\mathcal{Q}_A) = \begin{bmatrix} \rho_1 - \rho^{(\text{left})} \\ u_1 - u_2 \\ p_1 - p_{BC} \end{bmatrix} \tag{4.25}$$

At the fluid-structure interface, the velocity of the fluid is specified from the structural velocity at the interface (4.9) and the density and total energy are extrapolated from the interior. The discretized boundary conditions at the fluid-structure interface are written as

$$\begin{aligned} \rho_{N_A} &= \rho_{N_A - 1} \\ u_{N_A} &= \dot{d}_1 \\ e_{N_A} &= e_{N_A - 1} \end{aligned} \tag{4.26}$$

and the discrete residual vector of the fluid node on the interface is expressed as

$$\mathcal{R}_A^{(\Gamma)}(\mathcal{Q}_A, \mathcal{Q}_S) = \begin{bmatrix} \rho_{N_A} - \rho_{N_A - 1} \\ u_{N_A} - \dot{d}_1 \\ e_{N_A} - e_{N_A - 1} \end{bmatrix} \tag{4.27}$$

and $\dot{d}_1$ is the velocity of the first structural node (i.e. the structural node on the fluid-structure interface). We express our complete discrete residual vector for each node in the fluid domain as

$$\mathcal{R}_A(\mathcal{Q}_A, \mathcal{Q}_S) = \begin{bmatrix} \mathcal{R}_A^{\text{left}}(\mathcal{Q}_A) \\ \mathcal{R}_A^{\text{interior}}(\mathcal{Q}_A, \mathcal{Q}_S) \\ \mathcal{R}_A^{(\Gamma)}(\mathcal{Q}_A, \mathcal{Q}_S) \end{bmatrix} \tag{4.28}$$

Structural discretization

The first-order form of the structural equations (4.6) is discretized uniformly to N_S nodes using $N_S - 1$ 1-D bar elements. We express the structural equations in semi-discrete form for each structural node as follows:

$$\frac{\mathrm{d}\mathcal{Q}_S}{\mathrm{d}t} = Y_0^{-1}(F_\mathrm{s} - Y_1\mathcal{Q}_S) \tag{4.29}$$

where $\mathcal{Q}_S$ is a column vector of nodal displacements, d_j, followed by nodal velocities, $\dot{d}_j$. The structural node at the fixed boundary is not considered a degree of freedom and is not included as a structural variable. Therefore, this system contains $N_S - 1$ nodes and $2 * (N_S - 1)$ degrees of freedom. Writing (4.29) in the following form on interior nodes:

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{Q}_S^{\text{interior}} + \mathcal{R}_S^{\text{interior}}(\mathcal{Q}_S) = 0 \tag{4.30}$$

and noting that there are no applied nodal forces on any of the interior structural nodes ($F = 0$), we can express our discrete residual column vector on the interior structural nodes, $\mathcal{R}_S^{\text{interior}}$, as

$$\mathcal{R}_S^{\text{interior}}(\mathcal{Q}_S) = (Y_0^{-1}Y_1)'\mathcal{Q}_S \tag{4.31}$$

where $(Y_0^{-1}Y_1)'$ is a matrix containing the rows of $Y_0^{-1}Y_1$ corresponding to rows of interior structural nodes. At the fluid-structure interface, the force applied to the structural node is equal to the force exerted by the fluid at the interface, as described in (4.8). The discretized boundary condition at the first structural node is written as a function of the fluid variables at the interface

$$f_1 = Ap_{N_A} \tag{4.32}$$

where p_{N_A} is a function of the fluid variables at node N_A from the ideal gas law (4.3). Noting the semi-discrete form of the structural equations (4.29) applies to the first structural node, we then express the discrete residual vector of the structural node on the interface as

$$\mathcal{R}_S^{(\Gamma)}(\mathcal{Q}_A, \mathcal{Q}_S) = (Y_0^{-1}Y_1)''\mathcal{Q}_S - (Y_0^{-1})''Ap_{N_A} \tag{4.33}$$

where $(Y_0^{-1}Y_1)''$ and $(Y_0^{-1})''$ contain only the rows corresponding to interface structural nodes of $Y_0^{-1}Y_1$ and Y_0^{-1}, respectively. The boundary condition at the structural node at the fixed wall ($d = \dot{d} = 0$) is handled through the construction of the global mass and stiffness matrices. Constructing the complete discrete residual vector for each structural

node we get

$$\mathcal{R}_S(\mathcal{Q}_A, \mathcal{Q}_S) = \begin{bmatrix} \mathcal{R}_S^{(\Gamma)}(\mathcal{Q}_A, \mathcal{Q}_S) \\ \mathcal{R}_S^{\text{interior}}(\mathcal{Q}_S) \end{bmatrix} \tag{4.34}$$

4.1.5 Monolithic approach

The semi-discrete fluid and structural domains are assembled into a single fluid-structure equation, described by the following coupled system:

$$\frac{\mathrm{d}\mathcal{Q}}{\mathrm{d}t} = -\mathcal{R}(\mathcal{Q}) \tag{4.35}$$

where

$$\mathcal{Q} = \begin{bmatrix} \mathcal{Q}_A \\ \mathcal{Q}_S \end{bmatrix}, \qquad \mathcal{R}(\mathcal{Q}) = \begin{bmatrix} \mathcal{R}_A(\mathcal{Q}) \\ \mathcal{R}_S(\mathcal{Q}) \end{bmatrix} \tag{4.36}$$

The monolithic system of coupled fluid-structure equations is discretized in time by applying implicit Euler time marching with local time linearization to (4.35):

$$\left[\frac{I}{\Delta t} + \left(\frac{\partial \mathcal{R}}{\partial \mathcal{Q}} \right)^{(n)} \right] \Delta \mathcal{Q}^{(n)} = -\mathcal{R}(\mathcal{Q})^{(n)} \tag{4.37}$$

where $\Delta \mathcal{Q}^{(n)} = \mathcal{Q}^{(n+1)} - \mathcal{Q}^{(n)}$ is the solution update at each time step, n, and I is the identity matrix. Expressing (4.37) in terms of the individual fluid and structural sub-systems, we obtain our fully-discrete 1-D monolithic system:

$$\begin{bmatrix} \frac{I}{\Delta t} + \left(\frac{\partial \mathcal{R}_A}{\partial \mathcal{Q}_A} \right) & \left(\frac{\partial \mathcal{R}_A}{\partial \mathcal{Q}_S} \right) \\ \left(\frac{\partial \mathcal{R}_S}{\partial \mathcal{Q}_A} \right) & \frac{I}{\Delta t} + \left(\frac{\partial \mathcal{R}_S}{\partial \mathcal{Q}_S} \right) \end{bmatrix}^{(n)} \begin{bmatrix} \Delta \mathcal{Q}_A \\ \Delta \mathcal{Q}_S \end{bmatrix}^{(n)} = - \begin{bmatrix} \mathcal{R}_A(\mathcal{Q}_A, \mathcal{Q}_S) \\ \mathcal{R}_S(\mathcal{Q}_A, \mathcal{Q}_S) \end{bmatrix}^{(n)} \tag{4.38}$$

While in this case the same time-marching method has been applied to both the fluid and structural sub-systems, this is not a necessity for a monolithic approach.

The fluid Jacobian sub-matrix, $\partial \mathcal{R}_A / \partial \mathcal{Q}_A$, evaluated at interior fluid nodes, is obtained from (4.21):

$$\left(\frac{\partial \mathcal{R}_A^{\text{interior}}}{\partial \mathcal{Q}_A} \right)^{(n)} = \frac{1}{x_\Gamma(t)} \left[\delta_\xi \tilde{\mathcal{A}}^{(n)} - \left(\frac{\partial D_\xi}{\partial \mathcal{Q}_A} \right)^{(n)} \right] \tag{4.39}$$

where $\tilde{\mathcal{A}}$ is the transformed flux Jacobian given by

$$\tilde{\mathcal{A}} = \frac{\partial \tilde{\mathcal{E}}}{\partial \mathcal{Q}_A} = \frac{\partial \mathcal{E}}{\partial \mathcal{Q}_A} - wI \tag{4.40}$$

Both $\partial\mathcal{R}_A^{\text{left}}/\partial\mathcal{Q}_A$ and $\partial\mathcal{R}_A^{(\Gamma)}/\partial\mathcal{Q}_A$ are evaluated analytically from Eqs. (4.25) and (4.27), respectively.

The fluid coupling sub-matrix, $\partial\mathcal{R}_A/\partial\mathcal{Q}_S$, contains contributions from both the $\mathcal{R}_A^{\text{interior}}$ and $\mathcal{R}_A^{(\Gamma)}$ terms. The partial derivative of the interior node discrete residual with respect to the structural variables is constructed column-wise for derivatives with respect to the only structural variables included in (4.21): the displacement and velocity at the fluid-structure interface (d_1 and $\dot{d}_1$). We evaluate these columns of the coupling sub-matrix by differentiating (4.21) with respect to the interface displacement using the following chain rule form:

$$\left(\frac{\partial\mathcal{R}_A^{\text{interior}}}{\partial d_1}\right)^{(n)} = \left(\frac{\partial\mathcal{R}_A^{\text{interior}}}{\partial x_\Gamma(t)}\right)^{(n)} \left(\frac{\partial x_\Gamma(t)}{\partial d_1}\right)^{(n)} = \frac{-1}{x_\Gamma(t)^2}\left[\delta_\xi\tilde{\mathcal{E}}^{(n)} - \mathcal{D}_\xi^{(n)}\right] \tag{4.41}$$

The partial derivative of the interior residual vector with respect to the interface velocity is obtained by writing the residual (4.21) in the following form:

$$\mathcal{R}_A^{\text{interior}}(\mathcal{Q}_A, \mathcal{Q}_S) = \frac{1}{x_\Gamma(t)}\left[\delta_\xi\mathcal{E} - x_\Gamma{}'(t)\delta_\xi(\xi\mathcal{Q}_A) - \mathcal{D}_\xi\right] \tag{4.42}$$

and differentiating with respect to interface velocity to obtain

$$\left(\frac{\partial\mathcal{R}_A^{\text{interior}}}{\partial \dot{d}_1}\right)^{(n)} = \frac{-1}{x_\Gamma(t)}\left[\delta_\xi(\xi\mathcal{Q}_A)\right] \tag{4.43}$$

The contributions to the fluid coupling sub-matrix from the interface residual term, $\mathcal{R}_A^{(\Gamma)}$ is computed by differentiating (4.27) with respect to structural variables. The resulting $\partial\mathcal{R}_A^{(\Gamma)}/\partial\mathcal{Q}_S$ term is zero for every entry except for a value of negative unity at the index corresponding to the row of the fluid velocity at the interface and the column of the structural nodal velocity at the interface.

The structural Jacobian sub-matrix, $\partial\mathcal{R}_S/\partial\mathcal{Q}_S$, for interior structural nodes, is determined analytically from (4.31):

$$\left(\frac{\partial\mathcal{R}_S^{\text{interior}}}{\partial\mathcal{Q}_S}\right)^{(n)} = \left(Y_0^{-1}Y_1\right)' \tag{4.44}$$

The structural contributions from the interface are included in the $\partial\mathcal{R}_S^{(\Gamma)}/\partial\mathcal{Q}_S$ term, evaluated analytically from (4.33) in the same manner. The structural coupling sub-matrix, $\partial\mathcal{R}_S/\partial\mathcal{Q}_A$, is zero except for the $\partial\mathcal{R}_S^{(\Gamma)}/\partial\mathcal{Q}_A$ term, calculated analytically from (4.33) using the ideal gas law (4.3) to relate fluid pressure to fluid flow variables at the interface. The structural coupling at the fluid-structure interface is expressed as

$$\left(\frac{\partial\mathcal{R}_S^{(\Gamma)}}{\partial\mathcal{Q}_A}\right)^{(n)} = -\left(Y_0^{-1}\right)'' A\frac{\partial p_{N_A}}{\partial Q_A} \tag{4.45}$$

where the fluid node, $j = N_A$, is at the fluid-structure interface.

The left-hand side of the monolithic system (4.38) is stored explicitly as a dense 2-D array. The system is solved directly using a full LU-decomposition scheme with partial pivoting.

4.1.6 Partitioned approach

The partitioned approach is obtained by handling the fluid and structural sub-systems separately. In this case, we describe our system with two systems of coupled ODEs:

$$\frac{\mathrm{d}\mathcal{Q}_A}{\mathrm{d}t} = -\mathcal{R}_A(\mathcal{Q}_A, \mathcal{Q}_S) \tag{4.46}$$

$$\frac{\mathrm{d}\mathcal{Q}_S}{\mathrm{d}t} = -\mathcal{R}_S(\mathcal{Q}_A, \mathcal{Q}_S) \tag{4.47}$$

Applying implicit Euler time marching with local time linearization, and taking the solution variables of the opposite sub-system to be constant at each time step, we obtain the following system of equations:

$$\left[\frac{I}{\Delta t} + \left(\frac{\partial \mathcal{R}_A}{\partial \mathcal{Q}_A}\right)^{(n)}\right] \Delta\mathcal{Q}_A^{(n)} = -\mathcal{R}_A(\mathcal{Q}_A^{(n)}, \mathcal{Q}_S^{(n)}) \tag{4.48}$$

$$\left[\frac{I}{\Delta t} + \left(\frac{\partial \mathcal{R}_S}{\partial \mathcal{Q}_S}\right)^{(n)}\right] \Delta\mathcal{Q}_S^{(n)} = -\mathcal{R}_S(\mathcal{Q}_A^{(n+1)}, \mathcal{Q}_S^{(n)}) \tag{4.49}$$

The partitioned fluid-structure system described by (4.46) and (4.47) is solved by successively solving (4.48) and (4.49) in order for the solution at the next time step, $n + 1$. This approach is a block Gauss-Seidel iterative solution of the system.

Both the fluid and structural left-hand side matrices are stored explicitly in dense 2-D arrays and the systems are solved using a full LU-decomposition scheme with partial pivoting.

4.2 Optimization

With the monolithic system formed in the previous section, a steady aero-structural optimization problem is developed. In the steady 1-D formulation, a constant inlet pressure boundary condition is applied to the fluid domain, and the fluid-structure system is converged to steady-state before computing the gradients.

4.2.1 Formulation and governing equations

A model aero-structural optimization problem is proposed in the following form:

$$\begin{aligned} \text{minimize} \qquad & \mathcal{J} = I_a(d_1 - d_1^*)^2 + I_b(\omega_\text{n} - \omega_\text{n}^*)^2 \\ \text{w.r.t.} \qquad & E_\text{s},\ \ \rho_\text{s} \end{aligned} \tag{4.50}$$

where d_1 is the displacement of the first structural node (the fluid-structure interface), ω_n is the natural frequency of the structure, I_a and I_b are scaling parameters, and d_1^* and ω_n^* are target values for d_1 and ω_n, respectively. The lowest natural frequency of the structure, ω_n, is found using

$$\left|K - \omega_\text{n}^2 M\right| = 0 \tag{4.51}$$

obtained from the solution of (4.4) with $f(t) = 0$, and assuming a solution in the form $d = \bar{d}e^{i\omega t}$ [21]. The design variables selected for this case are the Young's modulus, E_s and structural density, ρ_s. The selection of these design variables is primarily academic, as a designer would not be able to vary these parameters continuously in reality, but they could vary these parameters through the selection of different materials for the structure.

4.2.2 Computation of derivatives for gradient evaluation

When considering gradient calculation through the adjoint (3.9) or flow-sensitivity equations (3.7), we require four partial derivative terms: the partial derivatives of the discrete residual vector and objective function with respect to both the design variables and the solution variables. Considering the notation used in Section 3.1.2, $\mathcal{Q} = [\mathcal{Q}_A \ \ \mathcal{Q}_S]^T$, and the design variables, $X = [E_\text{s} \ \ \rho_\text{s}]^T$. In this case, the objective function depends explicitly on only the structural state variables, $\mathcal{Q}_S$, and in particular, only on the displacement at the first structural node (at the fluid-structure interface), d_1. Therefore, the partial derivative of the objective with respect to the system state variables, $\partial\mathcal{J}/\partial\mathcal{Q}_S$ is zero for each entry except for the index corresponding to d_1, where we calculate the value analytically from

$$\frac{\partial \mathcal{J}}{\partial d_1} = 2I_a(d_1 - d_1^*) \tag{4.52}$$

From (4.51) we note that the natural frequency of the structure, ω_n, depends explicitly on both design variables, E_s and ρ_s. The partial derivative of the objective with respect to the design variables, $\partial\mathcal{J}/\partial X$ is obtained from the following chain rule expansion

$$\frac{\partial \mathcal{J}}{\partial X} = \frac{\partial \mathcal{J}}{\partial \omega_\text{n}} \frac{\partial \omega_\text{n}}{\partial X} \tag{4.53}$$

The first term in (4.53) is evaluated analytically from

$$\frac{\partial \mathcal{J}}{\partial \omega_{\mathrm{n}}} = 2I_b(\omega_{\mathrm{n}} - \omega_{\mathrm{n}}^*) \tag{4.54}$$

and the second term in (4.53) is evaluated using finite-differences from (4.51). In the finite-difference evaluation, each design variable is perturbed a small value ($\Delta X = 10^{-4}$), the global stiffness and mass matrices are updated from the new design variable values, and the change in natural frequency, $\Delta\omega_{\mathrm{n}}$, is determined. We then approximate our partial derivative from

$$\frac{\partial \omega_{\mathrm{n}}}{\partial X} = \frac{\Delta \omega_{\mathrm{n}}}{\Delta X} \tag{4.55}$$

The partial derivative of the discrete residuals of the governing equations with respect to the design variables, $\partial\mathcal{R}/\partial X$, is also required in the gradient calculation. In this case, the two design variables, E_{s} and ρ_{s}, are both structural variables, and so only appear explicitly in the residuals of the structural equations, $\mathcal{R}_S$. The partial derivative is approximated using finite-differences by perturbing each design variable a small value ($\Delta X = 10^{-4}$), updating the global stiffness and mass matrices, and evaluating the change in structural residual from (4.34):

$$\frac{\partial \mathcal{R}_S}{\partial X} = \frac{\Delta \mathcal{R}_S}{\Delta X} \tag{4.56}$$

4.2.3 Discrete-adjoint equations

The discrete-adjoint equations for the monolithic system are written in the following form, from (3.15):

$$\left(\frac{\partial \mathcal{R}}{\partial \mathcal{Q}}\right)^T \psi = \frac{\partial \mathcal{J}}{\partial \mathcal{Q}} \tag{4.57}$$

where the left-hand side matrix is the same as in the monolithic system used in the aero-structural analysis (4.38) at steady-state with a transpose operator applied, and $\partial\mathcal{J}/\partial\mathcal{Q}$ is found from the description in Section 4.2.2. Expanding this system to show the fluid and structural sub-system components, we get

$$\begin{bmatrix} \left(\frac{\partial \mathcal{R}_A}{\partial \mathcal{Q}_A}\right)^T & \left(\frac{\partial \mathcal{R}_S}{\partial \mathcal{Q}_A}\right)^T \\ \left(\frac{\partial \mathcal{R}_A}{\partial \mathcal{Q}_S}\right)^T & \left(\frac{\partial \mathcal{R}_S}{\partial \mathcal{Q}_S}\right)^T \end{bmatrix} \begin{bmatrix} \psi_A \\ \psi_S \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{\partial \mathcal{J}}{\partial \mathcal{Q}_S} \end{bmatrix} \tag{4.58}$$

where $\partial\mathcal{J}/\partial\mathcal{Q}_A = 0$ from the description in Section 4.2.2. This system is solved for the adjoint vector, ψ, and the gradient of the objective with respect to the design variables

is computed from (3.10) in the form

$$\mathcal{G} = \frac{\partial \mathcal{J}}{\partial X} - \psi^T \frac{\partial \mathcal{R}}{\partial X} \tag{4.59}$$

The partitioned approach is defined by writing the monolithic adjoint system (4.58) as

$$\left(\frac{\partial \mathcal{R}_A}{\partial \mathcal{Q}_A}\right)^T \psi_A^{(n+1)} = -\left(\frac{\partial \mathcal{R}_S}{\partial \mathcal{Q}_A}\right)^T \psi_S^{(n)} \tag{4.60}$$

$$\left(\frac{\partial \mathcal{R}_S}{\partial \mathcal{Q}_S}\right)^T \psi_S^{(n+1)} = \frac{\partial \mathcal{J}}{\partial \mathcal{Q}_S} - \left(\frac{\partial \mathcal{R}_A}{\partial \mathcal{Q}_S}\right)^T \psi_A^{(n+1)} \tag{4.61}$$

where the equations are solved in order at each iteration, n, until the solutions of ψ_A and ψ_S have converged. This solution scheme is a block Gauss-Seidel iterative solution of (4.58). The gradient of the objective with respect to the design variables is then computed from (3.10) in the following form

$$\mathcal{G} = \frac{\partial \mathcal{J}}{\partial X} - \psi_A^T \frac{\partial \mathcal{R}_A}{\partial X} - \psi_S^T \frac{\partial \mathcal{R}_S}{\partial X} \tag{4.62}$$

In both the monolithic system (4.58) and the staggered system (4.60)-(4.61), the left-hand side matrices are stored explicitly in 2-D arrays and each system is solved directly using LU-decomposition with partial pivoting.

4.2.4 Flow-sensitivity (direct) equations

We can also express the monolithic flow-sensitivity equations (3.7) for this system as

$$\left(\frac{\partial \mathcal{R}}{\partial \mathcal{Q}}\right) \phi = \frac{\partial \mathcal{R}}{\partial X} \tag{4.63}$$

where the left-hand side matrix is the same as used in the aero-structural analysis (4.38) at steady-state, and $\partial \mathcal{R}/\partial X$ is evaluated for each design variable according to Section 4.2.2. Expanding to see the individual fluid and structural sub-systems we get

$$\begin{bmatrix} \left(\frac{\partial \mathcal{R}_A}{\partial \mathcal{Q}_A}\right) & \left(\frac{\partial \mathcal{R}_A}{\partial \mathcal{Q}_S}\right) \\ \left(\frac{\partial \mathcal{R}_S}{\partial \mathcal{Q}_A}\right) & \left(\frac{\partial \mathcal{R}_S}{\partial \mathcal{Q}_S}\right) \end{bmatrix} \begin{bmatrix} \phi_A \\ \phi_S \end{bmatrix} = \begin{bmatrix} \frac{\partial \mathcal{R}_A}{\partial X} \\ \frac{\partial \mathcal{R}_S}{\partial X} \end{bmatrix} \tag{4.64}$$

which is solved once for each design variable, and the gradient of the objective with respect to the design variable is calculated from (3.8) according to

$$\mathcal{G} = \frac{\partial \mathcal{J}}{\partial X} - \frac{\partial \mathcal{J}}{\partial \mathcal{Q}}^T \phi \tag{4.65}$$

We also express the partitioned flow-sensitivity approach by applying a block Gauss-Seidel iterative solution scheme to the monolithic flow-sensitivity system (4.64) in the following form:

$$\left(\frac{\partial \mathcal{R}_A}{\partial \mathcal{Q}_A}\right)\phi_A^{(n+1)} = \frac{\partial \mathcal{R}_A}{\partial X} - \left(\frac{\partial \mathcal{R}_A}{\partial \mathcal{Q}_S}\right)\phi_S^{(n)} \tag{4.66}$$

$$\left(\frac{\partial \mathcal{R}_S}{\partial \mathcal{Q}_S}\right)\phi_S^{(n+1)} = \frac{\partial \mathcal{R}_S}{\partial X} - \left(\frac{\partial \mathcal{R}_S}{\partial \mathcal{Q}_A}\right)\phi_A^{(n+1)} \tag{4.67}$$

where the equations are solved in order until converged solutions of ϕ_A and ϕ_S have been obtained. We then compute the objective gradient from (3.8) in the form

$$\mathcal{G} = \frac{\partial \mathcal{J}}{\partial X} - \frac{\partial \mathcal{J}}{\partial \mathcal{Q}_A}^T \phi_A - \frac{\partial \mathcal{J}}{\partial \mathcal{Q}_S}^T \phi_S \tag{4.68}$$

As in the solution of the discrete-adjoint equations, both the monolithic system (4.64) and the staggered system (4.66)-(4.67) have their left-hand side matrices stored explicitly in 2-D arrays and then each system is solved directly using LU-decomposition with partial pivoting.

Chapter 5

Two-Dimensional Aero-structural Problem

A 2-D model problem for steady aero-structural analysis and optimization is formulated. The geometry consists of a symmetric airfoil with finite trailing-edge thickness and a horizontal, rectangular tab located at the trailing edge. The airfoil geometry is based on the symmetric NACA 0012 airfoil, with 12% maximum thickness compared to the chord length, as described by

$$y = a_0\sqrt{x} + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4 \tag{5.1}$$

where y is the airfoil half-thickness, x the chord-wise position, $a_0 = 0.2969$, $a_1 = -0.1260$, $a_2 = 0.3516$, $a_3 = 0.2843$, and $a_4 = -0.1015$ [15]. The geometry was slightly modified to incorporate a trailing edge thickness equal to 1% chord length by using a value of $a_4 = -0.0986$ in (5.1). The horizontal tab has a length of 10% of the original chord length. In this work, the chord length refers to the horizontal distance from the leading edge to the blunt trailing edge of the airfoil, not including the tab structure. This is also the chord length used in C_l calculations. A horizontal tab at the trailing edge of the airfoil can have significant impact on pressure distributions and moment characteristics, where tabs of this nature are common in helicopter blades to control moment and damping characteristics [16]. However, the geometry in this case was selected primarily as a model aero-structural problem in which an elastic finite-element structure can be incorporated with a rigid common aerodynamic application. For a practical application, this horizontal tab could be replaced by another structure of interest, such as a vertical Gurney flap or a separate airfoil flap structure. Alternatively, as in the time-accurate studies by Horáček

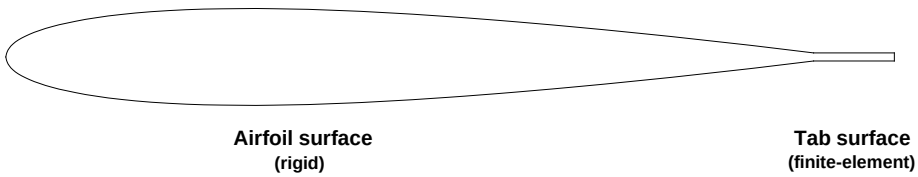

Figure 5.1: Airfoil-tab geometry used in two-dimensional model problem

et al. [12], the rear portion of the airfoil itself can be considered the elastic structure.

The surfaces of the horizontal tab define the extent of a 2-D finite-element structure composed of triangular plane stress elements. As such, the horizontal tab surfaces are free to move through the solution of the finite-element equations described in this section. The fluid flow over the airfoil-tab configuration exerts force on the tab structure, causing structural deformation, which in turn affects the fluid solution through the inclusion of a fluid mesh-movement scheme. A diagram of the airfoil-tab geometry is illustrated in Figure 5.1.

The fluid mesh consists of a 9-block structured H-grid topology. The top and bottom surfaces of both the airfoil and the tab are defined on separate blocks. The blunt trailing edge of the tab is meshed using a separate block which grows linearly along the positive x-direction from the trailing edge of the airfoil-tab geometry. Mesh density is increased near block interfaces at the leading and trailing edges of the geometry. An off-wall spacing of 1×10^{-3} chord lengths was used. Eighty nodes are used in the streamwise direction along the top and bottom airfoil surfaces, and 20 nodes are used along the top and bottom tab surfaces. Twenty nodes with uniform spacing are used on the blunt trailing edge of the tab. Forty nodes are used in the geometry normal direction towards the far-field boundaries. A total of 13 400 nodes are used in the fluid domain. The fluid mesh is shown in Figure 5.2.

The finite-element structural mesh consists of triangular plane stress elements. The fluid nodes on the tab surfaces are used to define coincident structural nodes on the fluid-structure interface. These interface nodes are used to define all other structural nodes. Structural nodes along the blunt trailing edge of the airfoil are created with uniform spacing to match the nodes at the trailing edge of the tab. Lines are drawn between matching nodes on opposite surfaces to divide the structure into quadrilaterals. Additional structural nodes are created at the centroid of each quadrilateral. Element edges are drawn from each quadrilateral node to each adjacent centroid node to create

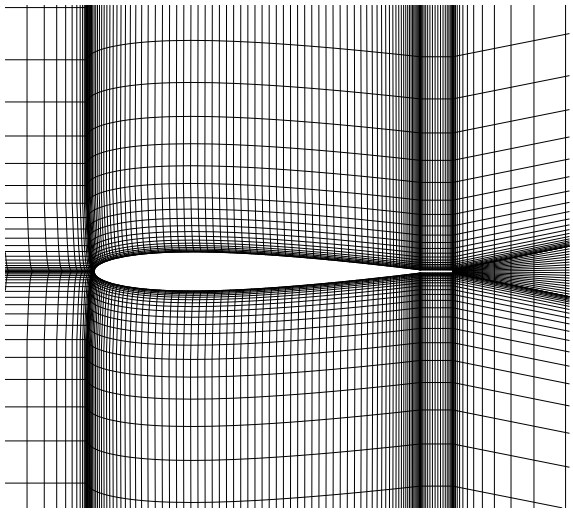

Figure 5.2: Airfoil-tab fluid mesh

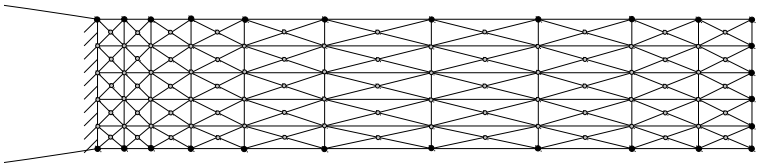

Figure 5.3: Schematic of structural mesh

the triangular elements. A schematic of this structural mesh scheme is shown in Figure 5.3, where the structural nodes are indicated by shaded circles, and the larger, darker circles are nodes coincident with the fluid nodes. A total of 1 444 structural elements and 761 structural nodes were used.

A local structural elemental and nodal numbering scheme was used to organize the finite-element structure. Nodes are numbered from top to bottom, moving left to right, alternating through columns of quadrilateral and centroid nodes. Elements are numbered in the same direction, with triangles along the left quadrilateral edges numbered first, then right, top, and bottom triangles, respectively.

The fluid and structural sub-systems are coupled to a fluid mesh-movement scheme. The mesh-movement uses the structural displacements computed from the fluid forces on the tab surfaces to move the fluid mesh nodes to reflect the updated structural position. This process is explained in more detail in Section 5.1.1.

5.1 Analysis

For 2-D aero-structural analysis, the problem has been expressed using a three-field formulation, which takes into consideration the individual fluid, structural, and fluid mesh-movement sub-systems. Each of these domains has its own set of governing equations, which are described next.

5.1.1 Formulation and governing equations

The fluid domain is governed by the Euler equations, which is written in conservative form in two dimensions as

$$\frac{\partial Q_A}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} = 0 \tag{5.2}$$

where Q_A is the vector of conserved aerodynamic quantities (mass, momentum in x- and y-directions, and energy), and E and F are the inviscid convective flux vectors, given by

$$Q_A = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ e \end{bmatrix}, \qquad E = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(e+p) \end{bmatrix}, \qquad F = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(e+p) \end{bmatrix} \tag{5.3}$$

where ρ is the fluid density, u and v are the Cartesian components of the fluid velocity, e is the energy of the fluid, and p is the fluid pressure. In order to close the system, an equation of state must be used to relate p to the fluid variables. In this case, the ideal gas law provides the relation

$$p = (\gamma - 1)\left[e - \rho\frac{u^2 + v^2}{2}\right] \tag{5.4}$$

where γ is the ratio of specific heats of the fluid ($\gamma = 1.4$ for air).

The structural domain is governed by the steady case of the second-order equation of motion (4.4), which reduces to the linear stiffness problem given by

$$Kd = f \tag{5.5}$$

where K is the global stiffness matrix of the structure, f is the applied nodal force vector, and d is the vector of nodal displacements. For the 2-D airfoil-tab problem, the structural domain is discretized using triangular plane stress elements. In the case of 2-D plane stress, the stress-strain relationship can be expressed as

$$\sigma = D_{\mathrm{s}}\epsilon \tag{5.6}$$

where σ is the stress tensor in vector form, ϵ is the strain tensor in vector form, and D_{s} is the constitutive law matrix. On an elemental basis these quantities are expressed as

$$\sigma_e = \begin{bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{bmatrix}, \qquad \epsilon_e = \begin{bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{bmatrix}, \qquad D_{\mathrm{s}e} = \frac{E_{\mathrm{s}}}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \tag{5.7}$$

where E_{s} is the Young's modulus and ν the Poisson's ratio. The strain-displacement relationship is expressed as

$$\epsilon = B_{\mathrm{s}} d \tag{5.8}$$

where d is the vector of nodal displacements, and B_{s} is the strain-displacement matrix. For a single 2-D triangular plane stress element this is defined by

$$B_{\mathrm{s}e} = \frac{1}{2A} \begin{bmatrix} y_j - y_m & 0 & y_m - y_i & 0 & y_i - y_j & 0 \\ 0 & x_m - x_j & 0 & x_i - x_m & 0 & x_j - x_i \\ x_m - x_j & y_j - y_m & x_i - x_m & y_m - y_i & x_j - x_i & y_i - y_j \end{bmatrix} \tag{5.9}$$

where i,j, and m are the vertices of each triangular plane stress element (ordered clockwise), A is the element area, and x and y are coordinates of the i,j,m element vertices, respectively. From these relationships the elemental stiffness matrix is expressed as [21]:

$$k = tAB_{\mathrm{s}}^T D_{\mathrm{s}} B_{\mathrm{s}} \tag{5.10}$$

where t is the elemental thickness.

The fluid mesh-movement is based on the linear elasticity method developed by Truong et al. [36, 37]. The nodes in the fluid mesh define the elements and nodes of a pseudo-structure consisting of quadrilateral plane stress elements. The stiffness of each element in the fluid mesh varies depending on a measure of mesh quality of that particular element. By varying the stiffness of the elements of the fluid mesh, mesh quality can be preserved for large deformations. For more information, the reader is directed to [36].

The mesh-movement is governed by a linear stiffness problem in the form

$$K_M d_M = f_M \tag{5.11}$$

where K_M is the global stiffness matrix of the fluid mesh pseudo-structure, f_M is the effective nodal force applied to the fluid mesh nodes, and d_M is the vector of fluid nodal displacements. In this case, f_M simply consists of the displacements of the nodes on the fluid-structure interface. In this work, each node in the fluid domain has two degrees of

freedom (x- and y-displacement), with the exception of those nodes that lie on the rigid airfoil surface and those on the fluid far-field boundaries. This represents a worst-case scenario in terms of computational cost, as the effect of structural movement is felt by each fluid node in the domain through the solution of (5.11). Computational cost can be reduced by selecting a rigid boundary of fluid nodes that surrounds the structure, inside which the structural movement can occur and the mesh-movement can be calculated without a significant negative impact on fluid mesh quality.

5.1.2 Fluid-structure-mesh-movement coupling and boundary conditions

Coupling of the fluid, structure, and mesh-movement domains is required to account for the interdependence of the sub-systems in this three-field formulation. In general, two coupling conditions per sub-system (i.e. one coupling condition for each dependence to another sub-system) are possible. In this scenario, each sub-system is coupled to one other sub-system for a total of three coupling conditions. First, the position of the fluid domain in physical space depends explicitly on the solution variables of the mesh-movement domain, such that

$$x = x^{(0)} + d_M^{(x)} \tag{5.12}$$

$$y = y^{(0)} + d_M^{(y)} \tag{5.13}$$

where x and y represent the position of the fluid domain in Cartesian coordinates, $x^{(0)}$ and $y^{(0)}$ are initial coordinates, and $d_M^{(x)}$ and $d_M^{(y)}$ are mesh-movement values in the x- and y-direction, respectively. The position of the fluid domain at far-field boundaries and along the rigid airfoil surface is fixed. Second, the force applied on the surface of the tab structure by the fluid is proportional to the fluid pressure, p, such that

$$f = CLp \tag{5.14}$$

where f is the applied force, C is a scaling parameter, and L is the area per unit depth (length) that the force acts on. The force is acts in the normal direction to the surface, and the direction is set by selecting a positive or negative C value. Only the y-components of the force on the top and bottom surfaces of the tab are considered in this work. There are no forces acting on the interior of the tab structure. Third, the effective force applied

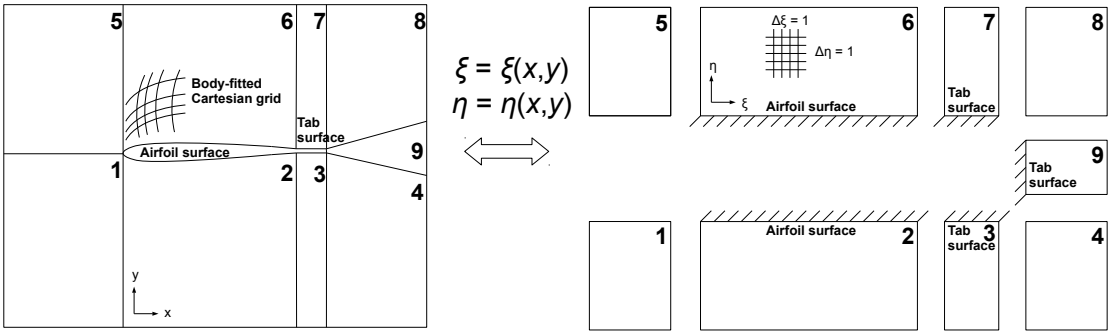

Figure 5.4: Nine-block airfoil and tab coordinate transformation from Cartesian (x,y) to uniform computational coordinates (ξ,η)

to the fluid mesh-movement scheme is equal to the displacement of the structure along the tab surface, according to

$$f_M = d \tag{5.15}$$

where f_M is the effective force, and d is the structural displacement. The effective force is equal to zero for all other locations.

At far-field boundaries the scaled fluid velocity is set according to the Mach number and the angle of attack. At the solid boundaries a slip-wall boundary condition is applied such that the velocity component normal to the surface is zero. Where the structural tab connects to the rigid airfoil is fixed, so the displacement is zero. For the fluid mesh movement, locations along the rigid airfoil surface and fluid far-field boundaries are fixed, so mesh-movement displacements are zero for these locations.

5.1.3 Coordinate transformation

A generalized steady curvilinear coordinate transformation is used to map the physical fluid domain (curvilinear body-fitted structured grid in Cartesian coordinates x and y) to the computational domain (uniform-spaced square grid in transformed coordinates ξ and η)

$$\xi = \xi(x, y), \qquad \eta = \eta(x, y) \tag{5.16}$$

The transformation, based on Pulliam [31] and Pulliam and Steger [30], is illustrated in Figure 5.4.

The transformation is applied to (5.2), which yields

$$\frac{\partial \hat{Q}_A}{\partial t} + \frac{\partial \hat{E}}{\partial \xi} + \frac{\partial \hat{F}}{\partial \eta} = 0 \tag{5.17}$$

where

$$\hat{Q}_A = J^{-1}\begin{bmatrix}\rho\\ \rho u\\ \rho v\\ e\end{bmatrix}, \qquad \hat{E} = J^{-1}\begin{bmatrix}\rho U\\ \rho u U + \xi_x p\\ \rho v U + \xi_y p\\ U(e+p)\end{bmatrix}, \qquad \hat{F} = J^{-1}\begin{bmatrix}\rho V\\ \rho u V + \eta_x p\\ \rho v V + \eta_y p\\ V(e+p)\end{bmatrix} \tag{5.18}$$

where U and V are the contravariant velocities, ξ_x, ξ_y, η_x and η_y are partial derivatives of the transformed spatial coordinates (ξ and η) with respect to the Cartesian coordinates (x and y) known as the grid metrics, and J is the metric Jacobian. The grid metrics are calculated from

$$\xi_x = J y_\eta, \qquad \eta_x = -J y_\xi, \qquad \xi_y = -J x_\eta, \qquad \eta_y = J x_\xi \tag{5.19}$$

and the metric Jacobian, J, is

$$J = (x_\xi y_\eta - y_\xi x_\eta)^{-1} \tag{5.20}$$

The partial derivative terms in (5.19) and (5.20) are straightforward to calculate using finite-differences, as they are computational space derivatives. Finally, the contravariant velocities are calculated from the grid metrics according to

$$U = \xi_x u + \eta_x v, \qquad V = \xi_y u + \eta_y v \tag{5.21}$$

5.1.4 Spatial discretization

Fluid discretization

The transformed 2-D Euler equations (5.17) are written in semi-discrete form at each interior node, (j,k) as

$$\frac{1}{J}\frac{\mathrm{d}}{\mathrm{d}t}(Q_A)_{j,k} = -\left[\delta_\xi \hat{E}_{j,k} + \delta_\eta \hat{F}_{j,k} - (D_\xi)_{j,k} - (D_\eta)_{j,k}\right] \tag{5.22}$$

where D_ξ and D_η are artificial dissipation terms in the ξ and η directions, respectively, and δ is a spatial difference operator such that

$$\delta_\xi \hat{E}_{j,k} = \frac{\hat{E}_{j+1,k} - \hat{E}_{j-1,k}}{2\Delta\xi} \tag{5.23}$$

$$\delta_\eta \hat{F}_{j,k} = \frac{\hat{F}_{j,k+1} - \hat{F}_{j,k-1}}{2\Delta\eta} \tag{5.24}$$

We now write (5.22) in the following form:

$$\frac{1}{J}\frac{\mathrm{d}\mathcal{Q}_A^{\text{interior}}}{\mathrm{d}t} + \mathcal{R}_A^{\text{interior}}(\mathcal{Q}_A, \mathcal{Q}_M) = 0 \tag{5.25}$$

where $\mathcal{Q}_A$ and $\mathcal{Q}_M$ are column vectors of discrete flow variables, $(Q_A)_{j,k}$, and discrete mesh-movement variables, $(Q_M)_{j,k}$, respectively, and $\mathcal{R}_A^{\text{interior}}$ is a column vector of discrete fluid residuals, $(R_A)_{j,k}$, for each interior node. We write our discrete residual vector on interior nodes as

$$\mathcal{R}_A^{\text{interior}}(\mathcal{Q}_A, \mathcal{Q}_M) = \left[\delta_\xi \hat{\mathcal{E}} + \delta_\eta \hat{\mathcal{F}} - \mathcal{D}_\xi - \mathcal{D}_\eta\right] \tag{5.26}$$

where $\hat{\mathcal{E}}$ and $\hat{\mathcal{F}}$ are column vectors of inviscid flux terms at each interior node, $\hat{E}_{j,k}$ and $\hat{F}_{j,k}$, and $\mathcal{D}_\xi$ and $\mathcal{D}_\eta$ are column vectors of artificial dissipation terms at each interior node, $(D_\xi)_{j,k}$ and $(D_\eta)_{j,k}$, respectively. The explicit dependence of the discrete fluid residual vector, $\mathcal{R}_A$, on the discrete mesh-movement variables, $\mathcal{Q}_M$, occurs through the grid metrics and metric Jacobian contained in $\hat{\mathcal{E}}$ and $\hat{\mathcal{F}}$, which depend explicitly on the position of the fluid nodes through (5.19) and (5.20), respectively. This coupling condition is described by (5.12) and (5.13).

At far-field and solid boundaries, as well as at block interfaces, the appropriate spatial difference operators are modified to one-sided differences and SATs are used to apply a penalty term. Additional details can be found in [9, 19]. The discrete fluid residual at boundary and block interface nodes is given by $\mathcal{R}_A^{(\text{bc})}$. In this case, we can express our complete discrete residual for each node in the fluid domain as

$$\mathcal{R}_A(\mathcal{Q}_A, \mathcal{Q}_M) = \begin{bmatrix} \mathcal{R}_A^{\text{interior}}(\mathcal{Q}_A, \mathcal{Q}_M) \\ \mathcal{R}_A^{(\text{bc})}(\mathcal{Q}_A, \mathcal{Q}_M) \end{bmatrix} \tag{5.27}$$

Structural discretization

We express the structural governing equations (5.5), discretized using 2-D plane stress triangular elements, in discrete form for each structural node in the following form:

$$K\mathcal{Q}_S - f = 0 \tag{5.28}$$

where $\mathcal{Q}_S$ is a column vector of nodal displacements, d_m, for each structural node, m, f is a vector of applied nodal forces, and K is the global stiffness matrix. Writing (5.28) in the following form:

$$\mathcal{R}_S^{\text{interior}}(\mathcal{Q}_S) = 0 \tag{5.29}$$

and noting there are no applied nodal forces on any of the interior structural nodes, we express our discrete residual column vector on the structural interior nodes as

$$\mathcal{R}_S^{\text{interior}}(\mathcal{Q}_S) = K'\mathcal{Q}_S \tag{5.30}$$

where K' is a matrix containing the rows of the global stiffness matrix, K, corresponding to the interior structural nodes. At the fluid-structure interface the force applied to the structure is proportional to the fluid dynamic pressure, as described by the coupling condition in (5.14). We express the discretized boundary condition at the structural nodes on the fluid-structure interface as

$$f_m = C_m L_m p_{(j,k)} \tag{5.31}$$

where m is the structural node on the interface, and (j,k) is the coincident fluid node on the interface. Writing (5.28) in the following form:

$$\mathcal{R}_S^{(\Gamma)}(\mathcal{Q}_S, \mathcal{Q}_A) = 0 \tag{5.32}$$

we can then express our discrete residual column vector on the fluid structure interface nodes as

$$\mathcal{R}_S^{(\Gamma)}(\mathcal{Q}_S, \mathcal{Q}_A) = K\mathcal{Q}_S - C_m L_m p_{(j,k)} \tag{5.33}$$

The boundary condition at the fixed structural/airfoil boundary ($\mathcal{Q}_S = 0$) is contained within the global stiffness matrix. We can now express the complete discrete residual vector for each structural node:

$$\mathcal{R}_S(\mathcal{Q}_S, \mathcal{Q}_A) = \begin{bmatrix} \mathcal{R}_S^{(\Gamma)}(\mathcal{Q}_S, \mathcal{Q}_A) \\ \mathcal{R}_S^{\text{interior}}(\mathcal{Q}_S) \end{bmatrix} \tag{5.34}$$

Mesh-movement discretization

We express the fluid mesh-movement equations (5.11), discretized to match the discretization of the fluid domain using 2-D quadrilateral plane stress elements, in discrete form for each mesh-movement node as follows:

$$K_M\mathcal{Q}_M - f_M = 0 \tag{5.35}$$

where $\mathcal{Q}_M$ is a column vector of mesh-movement nodal displacements, d_{M_i} for each mesh-movement node, i, and f_M is the vector of effective nodal forces. K_M is the global mesh-movement stiffness matrix. We express (5.35) in the following form:

$$\mathcal{R}_M(\mathcal{Q}_M) = 0 \tag{5.36}$$

and we note that there are no effective nodal forces on the interior mesh-movement nodes, so we can express our discrete residual vector on the interior mesh-movement nodes as follows:

$$\mathcal{R}_M^{\text{interior}}(\mathcal{Q}_M) = K_M{}'\mathcal{Q}_M \tag{5.37}$$

where $K_M{}'$ contains the rows of the mesh-movement stiffness matrix, K, corresponding to interior mesh-movement nodes. At the fluid-structure interface the effective nodal force applied to the mesh-movement system is equal to the structural nodal displacements on the interface. This mesh-movement coupling condition is described in (5.15). In this case, we express the boundary condition in discretized form such that

$$f_{M_i} = d_m \tag{5.38}$$

where i is the mesh-movement node on the interface and m is the coincident structural node. We can now write (5.35) as

$$\mathcal{R}_M^{(\Gamma)}(\mathcal{Q}_M, \mathcal{Q}_S) = 0 \tag{5.39}$$

and we express our effective nodal force from (5.38) to obtain

$$\mathcal{R}_M^{(\Gamma)}(\mathcal{Q}_M, \mathcal{Q}_S) = K_M\mathcal{Q}_M^{(\Gamma)} - d \tag{5.40}$$

where d is a column vector containing structural displacements from $\mathcal{Q}_S$ in rows corresponding to coincident mesh-movement nodes. The boundary condition at the rigid airfoil surface and at the far-field boundaries ($\mathcal{Q}_M = 0$) is applied inherently through the global mesh-movement stiffness matrix. We express our complete discrete residual vector for each node in the mesh-movement domain as

$$\mathcal{R}_M(\mathcal{Q}_M, \mathcal{Q}_S) = \begin{bmatrix} \mathcal{R}_M^{(\Gamma)}(\mathcal{Q}_M, \mathcal{Q}_S) \\ \mathcal{R}_M^{\text{interior}}(\mathcal{Q}_M) \end{bmatrix} \tag{5.41}$$

5.1.5 Monolithic approach

At a steady-state solution, time derivative terms must equal zero such that

$$\frac{\mathrm{d}\mathcal{Q}}{\mathrm{d}t} = 0 \tag{5.42}$$

We can then assemble the discrete fluid, structural, and mesh-movement domains into a single equation, described by the coupled system:

$$\mathcal{R}(\mathcal{Q}) = 0 \tag{5.43}$$

where

$$\mathcal{Q} = \begin{bmatrix} \mathcal{Q}_A \\ \mathcal{Q}_S \\ \mathcal{Q}_M \end{bmatrix}, \qquad \mathcal{R}(\mathcal{Q}) = \begin{bmatrix} \mathcal{R}_A(\mathcal{Q}) \\ \mathcal{R}_S(\mathcal{Q}) \\ \mathcal{R}_M(\mathcal{Q}) \end{bmatrix} \tag{5.44}$$

From an initial solution vector quantity, $\mathcal{Q}^{(0)}$, we desire a solution update

$$\Delta\mathcal{Q}^{(n)} = \mathcal{Q}^{(n+1)} - \mathcal{Q}^{(n)} \tag{5.45}$$

such that

$$\mathcal{R}(\mathcal{Q}^{(n)} + \Delta\mathcal{Q}^{(n)}) \approx 0 \tag{5.46}$$

Introducing the notation $\mathcal{R}^{(n)} = \mathcal{R}(\mathcal{Q}^{(n)})$, we perform a Taylor series expansion of $\mathcal{R}^{(n+1)}$ about $\mathcal{Q}^{(n)}$:

$$\mathcal{R}^{(n+1)} \approx \mathcal{R}^{(n)} + \left(\frac{\partial\mathcal{R}}{\partial\mathcal{Q}}\right)^{(n)} \Delta\mathcal{Q}^{(n)} \tag{5.47}$$

and setting the right-hand side of (5.47) to zero we obtain the following linear system

$$\left(\frac{\partial\mathcal{R}}{\partial\mathcal{Q}}\right)^{(n)} \Delta\mathcal{Q}^{(n)} = -\mathcal{R}^{(n)} \tag{5.48}$$

which is solved for the solution update, $\Delta\mathcal{Q}^{(n)}$. This system is solved at each iteration, n, until the steady-state solution is obtained.

Considering now the three-field fluid-structure-mesh-movement system, and also the definitions of discrete residual vectors in (5.27), (5.34), and (5.41), we can express (5.48) in the following form:

$$\begin{bmatrix} \mathbf{A} & 0 & \left(\frac{\partial\mathcal{R}_A}{\partial\mathcal{Q}_M}\right) \\ \left(\frac{\partial\mathcal{R}_S}{\partial\mathcal{Q}_A}\right) & \left(\frac{\partial\mathcal{R}_S}{\partial\mathcal{Q}_S}\right) & 0 \\ 0 & \left(\frac{\partial\mathcal{R}_M}{\partial\mathcal{Q}_S}\right) & \left(\frac{\partial\mathcal{R}_M}{\partial\mathcal{Q}_M}\right) \end{bmatrix}^{(n)} \begin{bmatrix} \Delta\mathcal{Q}_A \\ \Delta\mathcal{Q}_S \\ \Delta\mathcal{Q}_M \end{bmatrix}^{(n)} = - \begin{bmatrix} \mathcal{R}_A(\mathcal{Q}_A, \mathcal{Q}_M) \\ \mathcal{R}_S(\mathcal{Q}_S, \mathcal{Q}_A) \\ \mathcal{R}_M(\mathcal{Q}_M, \mathcal{Q}_S) \end{bmatrix}^{(n)} \tag{5.49}$$

where $\mathbf{A} = \partial\mathcal{R}_A/\partial\mathcal{Q}_A$, and the coupling sub-matrices equal to zero have not been included.

The solution of the monolithic system in (5.49) largely follows the strategy of the fluid solver *Diablo* [9], with a few notable exceptions. *Diablo* is a parallel, implicit finite-difference CFD solver which uses second-order accurate summation-by-parts (SBP) operators for spatial discretization and simultaneous approximation terms (SATs) at block interfaces and domain boundaries. The system is solved iteratively using an inexact-Newton method. For additional information on *Diablo*, the reader is directed to [9, 10].

The application of Newton's method to (5.43) shown in (5.49) converges quadratically when the initial iterate, $\mathcal{Q}^{(0)}$, is sufficiently close to the steady solution. However, finding a suitable initial iterate can be difficult. As such, the solution algorithm is broken into two phases: an approximate Newton start-up phase to find a suitable initial iterate, and an inexact-Newton phase to iterate to steady-state. The start-up phase uses a pseudo-transient formulation similar to applying implicit Euler time-marching to (5.22) with a first-order Jacobian matrix obtained by eliminating fourth-difference dissipation terms and increasing the coefficient for second-difference dissipation. This yields a modified fluid sub-matrix, $\mathbf{A}'$, which replaces $\mathbf{A}$ in (5.49) during the start-up phase:

$$\mathbf{A}'^{(n)} = \mathbf{T}^{(n)} + \left(\frac{\partial \mathcal{R}_A}{\partial \mathcal{Q}_A}\right)_1^{(n)} \tag{5.50}$$

where the subscript 1 indicates a first-order approximate Jacobian, and $\mathbf{T}$ is a diagonal matrix of inverse local time steps for each equation. The local, spatially-varying time step at each node, i, for each iteration, n, is computed from

$$\Delta t_i^{(n)} = \frac{\Delta t_{\text{ref}}^{(n)}}{J_i(1 + \sqrt[3]{J_i})} \tag{5.51}$$

where the reference time step, Δt_{ref}, for each iteration, n, is given as

$$\Delta t_{\text{ref}}^{(n)} = ab^n \tag{5.52}$$

where values of $a = 0.1$ and $b = 1.5$ are used in this work [9].

The algorithm switches to the inexact-Newton phase after the L_2-norm of the residual has been reduced one order of magnitude. In this phase, the fluid sub-matrix becomes:

$$\mathbf{A}'^{(n)} = \mathbf{T}^{(n)} + \left(\frac{\partial \mathcal{R}_A}{\partial \mathcal{Q}_A}\right)^{(n)} \tag{5.53}$$

where the local, spatially-varying time step is computed from (5.51) and the reference time step is computed now from

$$\Delta t_{\text{ref}}^{(n)} = \max\left[\alpha(||\mathcal{R}^{(n)}||_2)^{-\beta}, \Delta t_{\text{ref}}^{(n-1)}\right] \tag{5.54}$$

where $\alpha = 1.0$ and $\beta = 1.75$ is used in this work. As the iterations progress and the residual is reduced, the time step increases to infinity and a full-Newton step is recovered [19]. For additional information the reader is directed to [9]. The two phase approach is applied only to the fluid sub-matrix in the monolithic system.

In this work, all matrices are formed explicitly and stored in memory in compressed sparse row (CSR) format. Subroutines in *Diablo* were modified to form the fluid Jacobian matrix and store it in CSR format along with the other sub-matrices. The monolithic system is solved using an ILU-preconditioned GMRES algorithm [35]. The preconditioning consists of an incomplete LU factorization with dual truncation strategy (ILUT), with a fill-in level equal to the dimension of the left-hand side matrix and with a drop tolerance of 1.0×10^{-5}. These parameters were selected to balance efficiency and robustness. The residual of the monolithic system is reduced one and two orders of magnitude for the approximate and inexact-Newton phases, respectively. A convergence criterion of reducing the L_2-norm of the residual below 1.0×10^{-11} is used. The current implementation is completed for serial execution only.

Fluid sub-matrices

The fluid Jacobian sub-matrix, $\partial \mathcal{R}_A / \partial \mathcal{Q}_A$, evaluated at interior nodes is obtained from (5.26):

$$\left(\frac{\partial \mathcal{R}_A^{\text{interior}}}{\partial \mathcal{Q}_A} \right)^{(n)} = \left[\delta_\xi \hat{\mathcal{A}}^{(n)} + \delta_\eta \hat{\mathcal{B}}^{(n)} - \left(\frac{\partial D_\xi}{\partial \mathcal{Q}_A} \right)^{(n)} - \left(\frac{\partial D_\eta}{\partial \mathcal{Q}_A} \right)^{(n)} \right] \tag{5.55}$$

where $\hat{\mathcal{A}}$ and $\hat{\mathcal{B}}$ are the coordinate-transformed flux Jacobian matrices given by

$$\hat{\mathcal{A}} = \frac{\partial \hat{\mathcal{E}}}{\partial \mathcal{Q}_A}, \qquad \hat{\mathcal{B}} = \frac{\partial \hat{\mathcal{F}}}{\partial \mathcal{Q}_A} \tag{5.56}$$

As well, $\partial \mathcal{R}_A^{(\text{bc})} / \partial \mathcal{Q}_A$ is determined in the same manner as in (5.55). The fluid coupling sub-matrix, $\partial \mathcal{R}_A / \partial \mathcal{Q}_M$, is obtained using a finite-difference approximation. The aerodynamic residual, $\mathcal{R}_A$, depends on the mesh-movement variables, $\mathcal{Q}_M$, through the grid metrics contained in the transformed inviscid convective flux vectors, $\hat{\mathcal{E}}$ and $\hat{\mathcal{F}}$. Each mesh-movement displacement value is individually perturbed a small distance, ΔQ_M, and the updated fluid mesh node positions are computed from:

$$x_{(j,k)} = x_{(j,k)}^{(0)} + \Delta(\mathcal{Q}_M^{(x)})_i \tag{5.57}$$

$$y_{(j,k)} = y_{(j,k)}^{(0)} + \Delta(\mathcal{Q}_M^{(y)})_i \tag{5.58}$$

for each fluid node, (j,k), and corresponding mesh-movement node, i. The perturbed grid metrics are calculated at the updated fluid node positions, and the change in residual vector is computed from (5.26). The fluid coupling matrix is constructed column-wise for each perturbation of the fluid mesh-movement nodal displacement values. Due to the

cost of computing this term from finite-differences, $\partial\mathcal{R}_A/\partial\mathcal{Q}_M$ is not included in the left-hand side matrix of (5.49) during the aero-structural analysis. The absence of this term does not affect the steady solution, but may affect the rate of convergence. For gradient calculation in aero-structural optimization, this term is included, as the discrete-adjoint and flow-sensitivity (direct) equations require an exact Jacobian matrix at the converged solution.

Structural sub-matrices

The structural Jacobian sub-matrix, $\partial\mathcal{R}_S/\partial\mathcal{Q}_S$ is obtained analytically from (5.34) for each structural node:

$$\left(\frac{\partial\mathcal{R}_S}{\partial\mathcal{Q}_S}\right)^{(n)} = K \tag{5.59}$$

and equal to the global stiffness matrix. The structural coupling sub-matrix, $\partial\mathcal{R}_S/\partial\mathcal{Q}_A$, is obtained analytically using the definition for the structural residual vector (5.34). Noting that the only discrete residual components with an explicit dependence on fluid solution variables are on the fluid-structure interface, we use (5.33) to obtain:

$$\left(\frac{\partial\mathcal{R}_S^{(\Gamma)}}{\partial\mathcal{Q}_A}\right)^{(n)} = -\left(\frac{\partial C_m L_m p_{(j,k)}}{\partial\mathcal{Q}_A}\right)^{(n)} \tag{5.60}$$

where m indicates structural nodes along the fluid-structure interface, and (j,k) indicates the coincident fluid nodes. We express the fluid pressure, p, from the ideal gas law (5.4) as:

$$p = (\gamma - 1)\left[q_4 - \frac{q_2^2 + q_3^2}{2q_1}\right] \tag{5.61}$$

where $q_1 = \rho$, $q_2 = \rho u$, $q_3 = \rho v$, and $q_4 = e$. In this case we can compute the individual components of the structural coupling sub-matrix from

$$\frac{\partial\mathcal{R}_S^{(\Gamma)}}{\partial q_1} = -C_m L_m(\gamma - 1)\left(\frac{q_2^2 + q_3^2}{2q_1^2}\right) \tag{5.62}$$

$$\frac{\partial\mathcal{R}_S^{(\Gamma)}}{\partial q_2} = C_m L_m(\gamma - 1)\left(\frac{q_2}{q_1}\right) \tag{5.63}$$

$$\frac{\partial\mathcal{R}_S^{(\Gamma)}}{\partial q_3} = C_m L_m(\gamma - 1)\left(\frac{q_3}{q_1}\right) \tag{5.64}$$

$$\frac{\partial\mathcal{R}_S^{(\Gamma)}}{\partial q_4} = -C_m L_m(\gamma - 1) \tag{5.65}$$

The resulting structural coupling sub-matrix is sparse, with elements only in structural rows corresponding to surface structural nodes, and in fluid columns corresponding to the q_1, q_2, q_3, and q_4 values in surface fluid nodes.

Mesh-movement sub-matrices

The mesh-movement Jacobian sub-matrix, $\partial\mathcal{R}_M/\partial\mathcal{Q}_M$ is obtained from the discrete residual vector in (5.41) for each mesh-movement node:

$$\left(\frac{\partial\mathcal{R}_M}{\partial\mathcal{Q}_M}\right)^{(n)} = K_M \tag{5.66}$$

and equal to the global mesh-movement stiffness matrix. The mesh-movement coupling sub-matrix, $\partial\mathcal{R}_M/\partial\mathcal{Q}_S$, is obtained analytically using the definition for the mesh-movement residual vector, (5.41), and noting that only the discrete residual components at the interface (5.40) have an explicit dependence on structural solution variables, we get

$$\left(\frac{\partial\mathcal{R}_M^{(\Gamma)}}{\partial\mathcal{Q}_S}\right)^{(n)} = -\left(\frac{\partial d_m}{\partial\mathcal{Q}_S}\right)^{(n)} \tag{5.67}$$

where m indicates structural nodes on the fluid-structure interface. Since $\mathcal{Q}_S = d_m$, the resulting sparse sub-matrix contains values of negative unity at the intersection of mesh-movement rows of surface nodes with the structural columns of corresponding surface nodes. All other entries in the mesh-movement coupling sub-matrix are zero.

5.1.6 Partitioned approach

We develop a partitioned approach to the fluid-structure-mesh-movement system by treating each sub-system individually. We write our system with three systems of coupled equations in the form:

$$\mathcal{R}_A(\mathcal{Q}_A, \mathcal{Q}_M) = 0 \tag{5.68}$$

$$\mathcal{R}_S(\mathcal{Q}_S, \mathcal{Q}_A) = 0 \tag{5.69}$$

$$\mathcal{R}_M(\mathcal{Q}_M, \mathcal{Q}_S) = 0 \tag{5.70}$$

Performing a Taylor series expansion of each sub-system in the form of (5.47), and taking the solution variables of all other sub-systems as constant, we obtain the following linear

system of equations:

$$\mathbf{A}^{(n)}\Delta\mathcal{Q}_A^{(n)} = -\mathcal{R}_A(\mathcal{Q}_A^{(n)}, \mathcal{Q}_M^{(n)}) \tag{5.71}$$

$$\left(\frac{\partial\mathcal{R}_S}{\partial\mathcal{Q}_S}\right)^{(n)}\Delta\mathcal{Q}_S^{(n)} = -\mathcal{R}_S(\mathcal{Q}_S^{(n)}, \mathcal{Q}_A^{(n+1)}) \tag{5.72}$$

$$\left(\frac{\partial\mathcal{R}_M}{\partial\mathcal{Q}_M}\right)^{(n)}\Delta\mathcal{Q}_M^{(n)} = -\mathcal{R}_M(\mathcal{Q}_M^{(n)}, \mathcal{Q}_S^{(n+1)}) \tag{5.73}$$

where the partitioned fluid-structure-mesh-movement system is solved by solving the three systems, in order, for the solution update. This is a block Gauss-Seidel iterative solution scheme of the system (5.68)-(5.70). The fluid, structural, and mesh-movement Jacobian matrices are obtained using the strategy described in Section 5.1.5.

The fluid and mesh-movement systems in the partitioned scheme is solved using an ILU-preconditioned GMRES algorithm [35], and the structural system is solved using the conjugate gradient method. The incomplete LU factorization is computed using a dual truncation strategy (ILUT), with a fill-in level equal to the dimension of the matrix and the drop tolerance equal to 1.0×10^{-3} and 1.0×10^{-8} for the fluid and mesh-movement systems, respectively. Jacobian matrices are stored directly in CSR format. The residual of the fluid system for each GMRES variable update solve is reduced one and two orders of magnitude corresponding to the approximate and inexact-Newton phases, respectively. The structural and mesh-movement systems are solved completely (to a residual reduction of twelve orders of magnitude) at each solution variable update. Globally the fluid system is considered converged after a reduction of two orders of magnitude. The coupled fluid-structure-mesh-movement system is converged when the L_2-norm of the each of the system residuals is reduced below 1.0×10^{-11}.

5.2 Optimization

The 2-D airfoil-tab case is used as the basis of several aero-structural optimization problems of increasing complexity and relevance. Two design variable types are considered in this study. The first is an aerodynamic variable, the angle of attack, α, in degrees. The second is a structural variable, z_m, where m indicates the structural design variable index ($m = 1, 2$). In this case, the two z_m values fully define the initial, undeflected shape of the tab structure, prior to the aerostructural analysis. z_1 and z_2 are perturbation magnitudes in the y-direction of the structural nodes at the tip of the tab on the

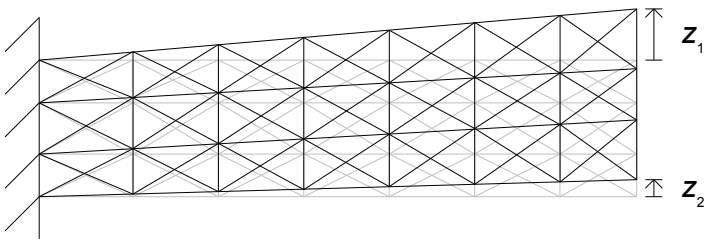

Figure 5.5: Diagram of structural movement scheme used in optimization

top and bottom surfaces, respectively. The perturbation magnitudes, in units of chord lengths, are measured with respect to the base tab geometry case described in previously in Chapter 5. The position of all other structural nodes are determined from the position of the tip nodes defined by z_1 and z_2. The tab top and bottom surfaces are defined by a straight line from the trailing edge of the airfoil (constant thickness) to the tab tip nodes. The node distribution along the top and bottom tab surfaces remains constant in the x-direction from the initial structural mesh. The y-distribution of structural nodes on the tab tip edge remains uniform along the perturbed tab tip length. The remaining structural nodes are stretched linearly to fill the new tab shape. A diagram of the structural movement scheme is shown in Figure 5.5, where the base geometry is shown in light grey and the updated geometry is shown in black. The updated position of the structural nodes on the fluid-structure interface is sent to the mesh-movement to update the initial position of the fluid mesh. All displacements in the solution of the problem are measured from the new initial positions as determined from z_1 and z_2.

Gradient-based optimization requires the partial derivatives of the residuals of the governing equations with respect to the design variables considered. The first design variable, angle of attack (α), is an aerodynamic variable which appears explicitly only in the residuals of the aerodynamic governing equations. In particular, the angle of attack appears only in the equations of the far-field fluid boundary nodes. In this study, the angle of attack derivative, $\partial\mathcal{R}/\partial\alpha$ is obtained with finite-differences by perturbing the angle of attack a small value ($\Delta\alpha = 10^{-4}$) and evaluating the change in aerodynamic residuals, $\Delta\mathcal{R}_A$, which gives

$$\frac{\partial\mathcal{R}}{\partial\alpha} = \begin{bmatrix} \frac{\partial\mathcal{R}_A}{\partial\alpha} \\ \frac{\partial\mathcal{R}_S}{\partial\alpha} \\ \frac{\partial\mathcal{R}_M}{\partial\alpha} \end{bmatrix} = \begin{bmatrix} \frac{\partial\mathcal{R}_A}{\partial\alpha} \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{\Delta\mathcal{R}_A}{\Delta\alpha} \\ 0 \\ 0 \end{bmatrix} \tag{5.74}$$

The structural design variables, z_m, do not appear explicitly in the residuals of the

governing equations of any sub-system in the present form. However, the structural movement defines the initial, undeflected position of each of the structural nodes. We express the structural residual in the following form

$$\mathcal{R}_S = Kd - f = K(x - x_o) - f \tag{5.75}$$

where x is the position of the displaced structural nodes and x_o is the initial position of the structural nodes. Assuming a constant stiffness matrix, K, we obtain the partial derivative term $\partial\mathcal{R}/\partial z_m$ using finite-differences by applying a small perturbation to the structural movement variable ($\Delta z_m = 10^{-6}$), determining the perturbed positions of the structural nodes, and evaluating the change in residual of the structural equations, evaluated from

$$\Delta\mathcal{R}_S = \mathcal{R}_S^+ - \mathcal{R}_S = K(x - x_o^+) - f - K(x - x_o) + f = -K(x_o^+ - x_o) \tag{5.76}$$

where $+$ indicates a perturbed quantity. The structural movement also updates the initial position of the mesh-movement nodes, so we must also consider

$$\mathcal{R}_M = K_M d_M - f_M = K_M(y - y_o) - (x - x_o) \tag{5.77}$$

where y is the position of the displaced mesh-movement nodes, y_o is the initial position of the mesh-movement nodes, and the effective nodal force vector, f_M, is a vector with entries equal to the structural displacement for mesh-movement nodes on the fluid-structure interface or zero otherwise. In this case, if we consider the change in mesh-movement residuals due to a perturbation in structural movement variables, Δz_m, we get

$$\Delta\mathcal{R}_M = \mathcal{R}_M^+ - \mathcal{R}_M = -K_M d_M^+ + d^+ \tag{5.78}$$

where $d_M^+ = y_o^+ - y_o$ is the displacement of the mesh-movement nodes due to the perturbation and $d^+ = x_o^+ - x_o$ is the displacement of the structural nodes due to the perturbation. We obtain the displacement of the mesh-movement nodes due to structural perturbation, d_M^+, from the solution of the mesh-movement problem:

$$K_M d_M^+ = d^+ \tag{5.79}$$

We note that (5.78) must equal zero from the solution of the system in (5.79).

We now evaluate our partial derivative with respect to z_m in the following form

$$\frac{\partial\mathcal{R}}{\partial z_m} = \begin{bmatrix} \frac{\partial\mathcal{R}_A}{\partial z_m} \\ \frac{\partial\mathcal{R}_S}{\partial z_m} \\ \frac{\partial\mathcal{R}_M}{\partial z_m} \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{\partial R_S}{\partial z_m} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{\Delta R_S}{\Delta z_m} \\ 0 \end{bmatrix} \tag{5.80}$$

The individual optimization problems considered are described below, followed by the monolithic and partitioned formulations of the discrete-adjoint and flow-sensitivity equations.

5.2.1 Unconstrained optimization with one design variable

An unconstrained aero-structural optimization problem with one design variable in the following form is considered

$$\begin{aligned} \text{minimize} \qquad & \mathcal{J} = C(d_{\text{tip}} - d^*_{\text{tip}})^2 \\ \text{w.r.t.} \qquad & \alpha \end{aligned} \tag{5.81}$$

where $C = 10^8$ is a scaling parameter, d_{tip} is the y-displacement of the structural node at the tip of the tab on the top surface, and $d^*_{\text{tip}} = 2.0 \times 10^{-4}$ is the target displacement value. The scaling parameter is selected so that at the first iteration the objective value is of the order of unity. The design variable is the angle of attack, α. The target displacement value, d^*_{tip}, was selected to be close to the steady displacement value obtained with an analysis run at the initial conditions ($M = 0.6$, $\alpha = 2.0°$). The quadratic objective function was chosen such that when the target displacement is achieved, the objective value is zero. A single design variable was chosen to maintain simplicity, as the purpose of this problem is to test the ability of the optimization algorithm to correctly vary one design variable to obtain a particular solution variable quantity.

To complete the gradient-based optimization, the partial derivatives of the objective function, $\mathcal{J}$, with respect to the state variables, $\mathcal{Q}$, and the design variable, α, must be computed. The value of $\partial\mathcal{J}/\partial\mathcal{Q}$ was computed analytically. Noting that the only state variable appearing explicitly in the objective function is d_{tip}, and so the result vector is zero for every entry except for at the y-displacement index of the tip, where we obtain the single entry in our result vector from

$$\frac{\partial \mathcal{J}}{\partial d_{\text{tip}}} = 2C(d_{\text{tip}} - d^*_{\text{tip}}) \tag{5.82}$$

The partial derivative of the objective function with respect to the angle of attack, $\partial\mathcal{J}/\partial\alpha$ is zero, as the angle of attack does not appear explicitly in the objective function.

5.2.2 Inverse design optimization

As a second problem to test the aero-structural optimization implementation, an inverse design scenario is considered, in which a unique combination of design variable values is

used to obtain a target solution. This target solution is reflected in the objective function, while the optimizer adjusts the design variables to independently obtain the values used to calculate the target solution. In this case, the following problem was considered

$$\begin{aligned} \text{minimize} \quad & \mathcal{J} = \sum_i (f_i - f_i^*)^2 \\ \text{w.r.t.} \quad & \alpha, \;\; z_m \qquad \text{for} \quad m = 1, 2 \end{aligned} \tag{5.83}$$

where f_i are the aerodynamic forces applied to each structural node, i, on the fluid-structure interface, and f_i^* are the target aerodynamic forces obtained with a unique set of design variables. The objective function is a sum of the squared difference between the current nodal force and the target nodal force. When the target force distribution is obtained, the value of the objective function is zero.

The partial derivatives of the objective function with respect to the state variables, $\mathcal{Q}$, and the design variables, α and z_m, are required to complete the optimization. The forces in the objective function explicitly depend only on the aerodynamic state variables, $\mathcal{Q}_A$, and so we require only the partial derivative term $\partial\mathcal{J}/\partial\mathcal{Q}_A$, obtained analytically from

$$\frac{\partial \mathcal{J}}{\partial \mathcal{Q}_A} = \frac{\partial \mathcal{J}}{\partial f_i} \frac{\partial f_i}{\partial \mathcal{Q}_A} \tag{5.84}$$

where

$$\frac{\partial \mathcal{J}}{\partial f_i} = 2(f_i - f_i^*) \tag{5.85}$$

for each structural node, i, on the fluid-structure interface (and $\partial\mathcal{J}/\partial f_i = 0$ otherwise), and $\partial f_i/\partial\mathcal{Q}_A$ is equivalent to the structural coupling sub-matrix with the coupling condition (5.14) applied, with entries obtained using Equations (5.62)-(5.64).

The partial derivative of the objective function with respect to the design variables, α and z_m, are both equal to zero, as the design variables do not appear explicitly in the objective function.

5.2.3 Constrained aero-structural optimization

A constrained aero-structural optimization problem is considered in the following form

$$\begin{aligned} \text{minimize} \quad & \mathcal{J} = I_W W + I_T T \\ \text{w.r.t.} \quad & \alpha, \;\; z_m \qquad \text{for} \quad m = 1, 2 \\ \text{subject to} \quad & C_l - C_l^* = 0 \\ & \mathcal{K} \geq 0 \end{aligned} \tag{5.86}$$

where W is the structural weight defined by

$$W = \sum_i \rho_s A_i \tag{5.87}$$

where i is the elemental index and the summation is performed over each structural element and A_i is the elemental area, and T is the tab tip deflection defined by

$$T = d_{\text{tip}} \tag{5.88}$$

I_W and I_T are scaling parameters to keep structural weight and tip deflection at the same order of magnitude, and to study the trade-off between W and T in the objective function. $\mathcal{K}$ is the maximum elemental stress constraint expressed in the form of an aggregate KS function. The target lift constraint is expressed in terms of the lift coefficient, C_l, and the target lift coefficient, C_l^*. d_{tip} is the y-displacement of the structural node on the top surface at the free end of tab.

The partial derivatives of the objective function, lift constraint, and KS function with respect to the state variables, $\mathcal{Q}$, and design variables, z_1 and z_2 are required for the gradient-based optimization. The objective function depends explicitly on only the structural state variables, $\mathcal{Q}_S$, and in particular, only the tab tip y-displacement value, so the derivative with respect to the state variables is zero everywhere except at the tab tip y-displacement, where it is calculated analytically from

$$\frac{\partial \mathcal{J}}{\partial d_{\text{tip}}} = I_T \tag{5.89}$$

The derivative of the objective with respect to the design variables is determined from

$$\frac{\partial \mathcal{J}}{\partial z_m} = I_W \frac{\partial W}{\partial z_m} + I_T \frac{\partial T}{\partial z_m} \tag{5.90}$$

for $m = 1, 2$. We note the explicit dependence of tab deflection, T, on z_m by writing in the following form:

$$T = d_{\text{tip}} = (y_{\text{tip}} - y_{\text{tip}}^{(0)}) \tag{5.91}$$

where y_{tip} is the y-position and $y_{\text{tip}}^{(0)}$ is the initial y-position of the tab tip top node. Now we express this initial y-position as

$$y_{\text{tip}}^{(0)} = y_{\text{tip}}^{(\text{base})} + z_1 \tag{5.92}$$

where $y_{\text{tip}}^{(\text{base})}$ is the y-position of the tab tip top node in the base geometry from which z_1 is measured. Finally we express our tip deflection from initial geometry, T, as

$$T = y_{\text{tip}} - y_{\text{tip}}^{(\text{base})} - z_1 \tag{5.93}$$

and from (5.93) we arrive at

$$\frac{\partial T}{\partial z_1} = -1 \qquad \frac{\partial T}{\partial z_2} = 0 \tag{5.94}$$

We evaluate (5.90) from finite-differences by perturbing the design variables, z_m, a small value ($\Delta z_m = 10^{-6}$) and calculating the change in structural weight, ΔW

$$\frac{\partial W}{\partial z_m} = \frac{\Delta W}{\Delta z_m} \tag{5.95}$$

The lift constraint depends explicitly on the aerodynamic state variables, $\mathcal{Q}_A$, and the mesh-movement variables, $\mathcal{Q}_M$, only, and specifically, only on the fluid and mesh-movement nodes on the aerodynamic surface. The $\partial C_l/\partial \mathcal{Q}_A$ and $\partial C_l/\partial \mathcal{Q}_M$ terms are calculated from finite-differences by perturbing the fluid solution variables ($\mathcal{Q}_A$) and the fluid mesh-movement solution variables ($\mathcal{Q}_M$) on the aerodynamic surface a small value ($\Delta \mathcal{Q}_A = 10^{-8}$, $\Delta \mathcal{Q}_M = 10^{-8}$) and calculating the change in lift coefficient, ΔC_l

$$\frac{\partial C_l}{\partial \mathcal{Q}_A} = \frac{C_l^+ - C_l}{\Delta \mathcal{Q}_A} \tag{5.96}$$

$$\frac{\partial C_l}{\partial \mathcal{Q}_M} = \frac{C_l^+ - C_l}{\Delta \mathcal{Q}_M} \tag{5.97}$$

and $\partial C_l/\partial \mathcal{Q}_A = 0$, $\partial C_l/\partial \mathcal{Q}_M = 0$ for all fluid and mesh-movement variables not on the aerodynamic surface. The derivatives with respect to the design variables, z_1 and z_2, are both equal to zero, as there is no explicit dependence. The derivative with respect to the angle of attack is calculated from finite-differences.

The KS constraint depends explicitly on the structural state variables, $\mathcal{Q}_S$, through the stress-displacement relationship. The partial derivative of the KS function with respect to the structural state variables, $\mathcal{Q}_S = d$, is determined analytically from the following chain rule expansion:

$$\frac{\partial \mathcal{K}}{\partial d} = \frac{\partial \mathcal{K}}{\partial g_i}\frac{\partial g_i}{\partial \sigma_{V\,i}}\frac{\partial \sigma_{V\,i}}{\partial \sigma}\frac{\partial \sigma}{\partial d} \tag{5.98}$$

The first derivative in (5.98) is evaluated by differentiating the KS function given in Equation (3.34) with respect to the g_i, which yields

$$\frac{\partial \mathcal{K}}{\partial g_i} = \left[\sum_i \mathrm{e}^{-\rho_{\mathcal{K}} g_i}\right]^{-1} \mathrm{e}^{-\rho_{\mathcal{K}} g_i} \tag{5.99}$$

The second derivative in (5.98) is evaluated by differentiating the definition of g_i given in Equation (3.33) with respect to the elemental von Mises stress, $\sigma_{V\,i}$, which produces

$$\frac{\partial g_i}{\partial \sigma_{V\,i}} = -\frac{1}{\sigma_Y} \tag{5.100}$$

The elemental von Mises stress for a plane stress element is given by

$$\sigma_V = \sqrt{{\sigma_x}^2 + {\sigma_y}^2 - \sigma_x\sigma_y + 3{\tau_{xy}}^2} \tag{5.101}$$

where σ_x, σ_y, and τ_{xy} are the components of the elemental stress tensor, σ. From this definition we obtain the third derivative in (5.98) by differentiating Equation (5.101) as follows

$$\frac{\partial \sigma_{Vi}}{\partial \sigma} = \frac{-1}{2\sqrt{{\sigma_x}^2 + {\sigma_y}^2 - \sigma_x\sigma_y + 3{\tau_{xy}}^2}} \begin{bmatrix} 2\sigma_x - \sigma_y \\ 2\sigma_y - \sigma_x \\ 6\tau_{xy} \end{bmatrix}^T \tag{5.102}$$

Considering next the strain-displacement (5.8) and stress-strain (5.6) relationships, we can express the elemental stress tensor as a function of the nodal displacements and we obtain the following stress-displacement relationship

$$\sigma = D_{\mathrm{s}} B_{\mathrm{s}} d \tag{5.103}$$

where D_{s} is the constitutive law matrix and B_{s} is the strain-displacement matrix. The final term in (5.98) is now found by differentiating (5.103) with respect to the structural displacement, d:

$$\frac{\partial \sigma}{\partial d} = D_{\mathrm{s}} B_{\mathrm{s}} \tag{5.104}$$

The derivative of the KS function with respect to the design variables, z_m, is found from a similar chain rule expansion to (5.98):

$$\frac{\partial \mathcal{K}}{\partial z_m} = \frac{\partial \mathcal{K}}{\partial g_i} \frac{\partial g_i}{\partial \sigma_{Vi}} \frac{\partial \sigma_{Vi}}{\partial \sigma} \frac{\partial \sigma}{\partial z_m} \tag{5.105}$$

The first three terms are calculated analytically from (5.99), (5.100), and (5.102) respectively. The final term, the derivative of the elemental stress tensor with respect to the design variables, z_m, is found from finite-differences by perturbing the design variables a small amount ($\Delta z_m = 10^{-6}$) and calculating the change in stress tensor, σ, from the following expression

$$\sigma = D_{\mathrm{s}} B_{\mathrm{s}} (x - x_o) \tag{5.106}$$

where x is the position of the displaced structure and x_o is the initial position of the structure. The partial derivative term approximation is then calculated from

$$\frac{\partial \sigma}{\partial z_m} = \frac{\Delta \sigma}{\Delta z_m} = \frac{D_{\mathrm{s}}[B_{\mathrm{s}}^+ (x - x_o^+) - B_{\mathrm{s}}(x - x_o)]}{\Delta z_m} \tag{5.107}$$

where the $+$ superscript indicates a quantity calculated from a perturbed design variable value.

5.2.4 Discrete-adjoint equations

The gradient of the objective function with respect to the design variables can be computed from the solution of the discrete-adjoint equations, with a computational cost proportional to the number of functions being considered. The discrete-adjoint equations (3.9) can be expressed for our monolithic system (5.49) in the following form

$$\begin{bmatrix} \mathbf{A}^T & \left(\frac{\partial \mathcal{R}_S}{\partial Q_A}\right)^T & 0 \\ 0 & \left(\frac{\partial \mathcal{R}_S}{\partial \mathcal{Q}_S}\right)^T & \left(\frac{\partial \mathcal{R}_M}{\partial Q_S}\right)^T \\ \left(\frac{\partial \mathcal{R}_A}{\partial Q_M}\right)^T & 0 & \left(\frac{\partial \mathcal{R}_M}{\partial \mathcal{Q}_M}\right)^T \end{bmatrix} \begin{bmatrix} \psi_A \\ \psi_S \\ \psi_M \end{bmatrix} = \begin{bmatrix} \frac{\partial \mathcal{J}}{\partial Q_A} \\ \frac{\partial \mathcal{J}}{\partial Q_S} \\ \frac{\partial \mathcal{J}}{\partial Q_M} \end{bmatrix} \tag{5.108}$$

where the transpose operator has been applied to the block sub-matrices on the left-hand side. The discrete-adjoint equations can also be solved with a staggered (partitioned) approach by expressing (5.108) in the following form

$$[\mathbf{A}]^T \psi_A^{(n+1)} = \frac{\partial \mathcal{J}}{\partial Q_A} - \left(\frac{\partial \mathcal{R}_S}{\partial Q_A}\right)^T \psi_S^{(n)} \tag{5.109}$$

$$\left(\frac{\partial \mathcal{R}_M}{\partial \mathcal{Q}_M}\right)^T \psi_M^{(n+1)} = \frac{\partial \mathcal{J}}{\partial Q_M} - \left(\frac{\partial \mathcal{R}_A}{\partial Q_M}\right)^T \psi_A^{(n+1)} \tag{5.110}$$

$$\left(\frac{\partial \mathcal{R}_S}{\partial \mathcal{Q}_S}\right)^T \psi_S^{(n+1)} = \frac{\partial \mathcal{J}}{\partial Q_S} - \left(\frac{\partial \mathcal{R}_M}{\partial Q_S}\right)^T \psi_M^{(n+1)} \tag{5.111}$$

where the equations are solved in order at each iteration, n, until some stopping criterion is met such that the solution ψ is converged. The solution of the system in this form is a block Gauss-Seidel iterative solution of (5.108). In either the monolithic or partitioned formulation, the gradient is computed from the adjoint vector, ψ, from (3.10).

The monolithic adjoint system (5.108) is solved in a similar manner as the monolithic aero-structural analysis system (5.49), with ILU-preconditioned GMRES algorithm. The incomplete LU factorization is performed with a dual truncation strategy (ILUT) with the fill-in level equal to the matrix dimension and the drop tolerance equal to 1.0×10^{-5}. The transpose operator is applied to the left-hand side matrix and stored explictly in CSR format. The residual L_2-norm is reduced 16 orders of magnitude. The staggered system equations in (5.109)-(5.111) are each solved sequentially using an ILU-preconditioned GMRES solution scheme for (5.109) and (5.110) and a conjugate gradient method for (5.111). The incomplete LU factorization (ILUT) uses a fill-in level equal to the matrix dimension and a drop tolerance of 1.0×10^{-3} and 1.0×10^{-8} for the fluid and mesh-movement systems, respectively. Each matrix is stored explicitly in CSR format. The residual of each system is reduced 16 orders of magnitude.

5.2.5 Flow-sensitivity (direct) equations

The objective function gradient with respect to the design variables can be computed from the solution of the flow-sensitivity (direct) equations. In this case, the computational cost is proportional to the number of design variables being considered. The flow-sensitivity equations (3.7) can be expressed for our monolithic system (5.49) as

$$\begin{bmatrix} \mathbf{A} & 0 & \left(\frac{\partial \mathcal{R}_A}{\partial Q_M}\right) \\ \left(\frac{\partial \mathcal{R}_S}{\partial Q_A}\right) & \left(\frac{\partial \mathcal{R}_S}{\partial \mathcal{Q}_S}\right) & 0 \\ 0 & \left(\frac{\partial \mathcal{R}_M}{\partial Q_S}\right) & \left(\frac{\partial \mathcal{R}_M}{\partial \mathcal{Q}_M}\right) \end{bmatrix} \begin{bmatrix} \phi_A \\ \phi_S \\ \phi_M \end{bmatrix} = \begin{bmatrix} \frac{\partial \mathcal{R}_A}{\partial X} \\ \frac{\partial \mathcal{R}_S}{\partial X} \\ \frac{\partial \mathcal{R}_M}{\partial X} \end{bmatrix} \tag{5.112}$$

where the left-hand side of (5.112) is the same left-hand side of our monolithic system, (5.49), evaluated at the steady-state solution. We can also solve the flow-sensitivity equations with a staggered (partitioned) approach by expressing (5.112) as

$$\mathbf{A}\phi_A^{(n+1)} = \frac{\partial \mathcal{R}_A}{\partial X} - \left(\frac{\partial \mathcal{R}_A}{\partial Q_M}\right)\phi_M^{(n)} \tag{5.113}$$

$$\left(\frac{\partial \mathcal{R}_S}{\partial \mathcal{Q}_S}\right)\phi_S^{(n+1)} = \frac{\partial \mathcal{R}_S}{\partial X} - \left(\frac{\partial \mathcal{R}_S}{\partial Q_A}\right)\phi_A^{(n+1)} \tag{5.114}$$

$$\left(\frac{\partial \mathcal{R}_M}{\partial \mathcal{Q}_M}\right)\phi_M^{(n+1)} = \frac{\partial \mathcal{R}_M}{\partial X} - \left(\frac{\partial \mathcal{R}_M}{\partial Q_S}\right)\psi_S^{(n+1)} \tag{5.115}$$

where the equations are solved in order at each iteration, n, until convergence to a solution of ϕ. In this form, the staggered solution scheme is a block Gauss-Seidel iteration of the monolithic system in (5.112). The gradient is then computed from the flow-sensitivity vector, ϕ, from (3.8).

The monolithic flow-sensitivity system (5.112), like the discrete-adjoint system, is solved in much the same manner as the monolithic aero-structural analysis system (5.49), with an ILU-preconditioned GMRES approach.The left-hand side matrix is stored explictly in CSR format and the residual L_2-norm is reduced 16 orders of magnitude. The staggered system equations given in (5.113)-(5.115) are solved sequentially using the same ILU-preconditioned GMRES solution scheme for (5.113) and (5.115) and the conjugate gradient method for (5.114), with each system sub-matrix stored in CSR format. The residual of each system is reduced 16 orders of magnitude until a converged solution is achieved. The incomplete LU factorization with dual truncation (ILUT) is used for both monolithic and partitioned cases, with a fill-in level equal to the matrix dimension and a drop tolerance of 1.0×10^{-5} for the monolithic case and 1.0×10^{-3} and 1.0×10^{-8} for the partitioned fluid and mesh-movement systems, respectively.

Chapter 6

Results

6.1 One-dimensional aero-structural analysis

6.1.1 Sample results

The monolithic 1-D fluid-structure solver is used to obtain results for various test cases. Each test case is a variation of the specified parameters of an arbitrary base case. The first case is run with an inlet velocity frequency of $\omega = \frac{8}{5}\omega_{\mathrm{n}}$ (where ω_{n} is the lowest natural frequency of the structure) and a Young's modulus of $E_{\mathrm{s}} = 1 \times 10^6$ Pa, while the second case is run with an inlet velocity frequency of $\omega = \omega_{\mathrm{n}}$ and a Young's modulus of $E_{\mathrm{s}} = 1 \times 10^7$ Pa. In the second case with $\omega = \omega_{\mathrm{n}}$, the magnitude of the oscillations in the displacement of the fluid-structure interface increases significantly over time. The interface displacement and inlet pressure histories are shown in Figure 6.1 and Figure 6.2 for the first and second test case, respectively.

6.2 One-dimensional aero-structural optimization

6.2.1 Sample results

The monolithic aero-structural adjoint solver is used to complete the optimization test cases. A constant inlet pressure fluid boundary condition is used to calculate a steady structural displacement and to evaluate the objective function. The gradients are calculated using the aero-structural adjoint method, and the L-BFGS-B algorithm is used to compute the design variable updates. The accuracy of the gradients is tested with

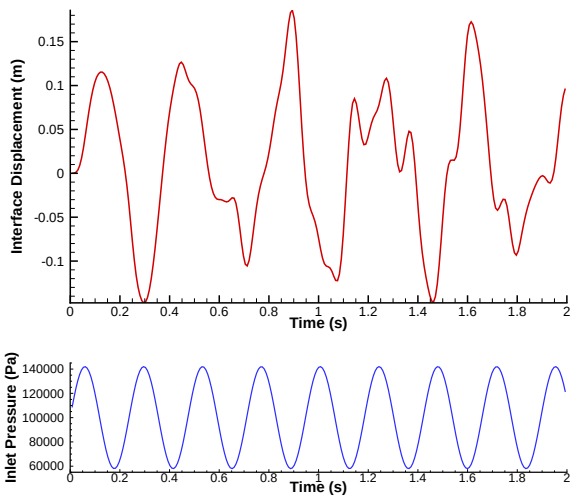

Figure 6.1: Displacement of fluid-structure interface for $\omega = \frac{8}{5}\omega_{\mathrm{n}}$, $E_{\mathrm{s}} = 1 \times 10^6$ Pa

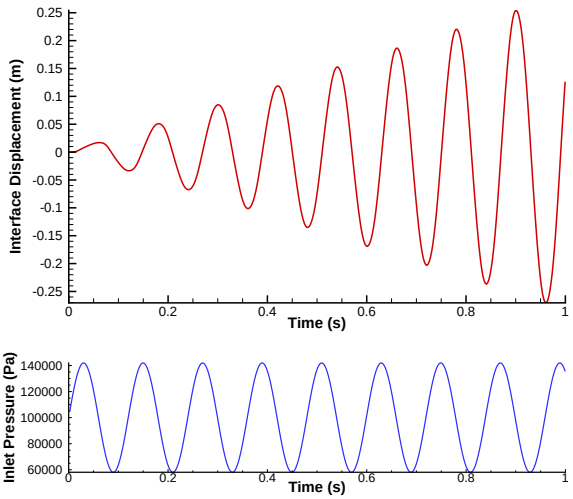

Figure 6.2: Displacement of fluid-structure interface for $\omega = \omega_{\mathrm{n}}$, $E_{\mathrm{s}} = 1 \times 10^7$ Pa

finite-differences in Section 6.2.2. Test parameters are selected as follows: $I_a = 100$, $I_b = 10$, $d_1^* = 0.02$ m, $\omega_{\mathrm{n}}^* = 0.10$ s^{-1}. The initial values of the design variables are $E_{\mathrm{s}}^{(0)} = 1.00 \times 10^7$ Pa and $\rho_{\mathrm{s}}{}^{(0)} = 1.00 \times 10^3$ kg/m^3. Design variables are scaled by their initial values. The convergence criterion is selected as a reduction in objective function value by ten orders of magnitude. The optimization case converged in 17 iterations. The calculated values of the design variables are $E_{\mathrm{s}} = 1.26 \times 10^7$ Pa and $\rho_{\mathrm{s}} = 2.87 \times 10^3$ kg/m^3. The convergence history of the objective function value and the magnitude of the gradient is given in Figure 6.3, where the shaded squares indicate outer iterations of the L-BFGS-B algorithm and the smaller, blanked squares indicate inner iterations that do not satisfy both the sufficient decrease and curvature conditions. The evolution of design variable values through the optimization run on a contour plot of objective function values is shown in Figure 6.4, where the initial iterate position is given at ($X_1 = 1.0$, $X_2 = 1.0$), and the first nine L-BFGS-B iterations are labelled.

6.2.2 Gradient accuracy

The accuracy of the computed gradient from the adjoint approach is evaluated by comparing to a value calculated using finite-differences. Both design variables, E_{s} and ρ_{s},

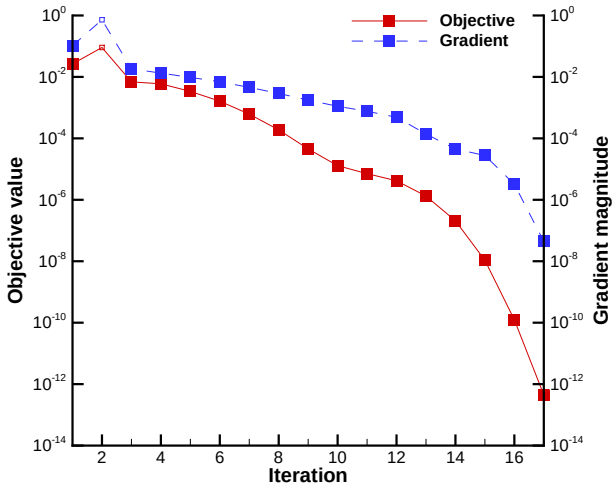

Figure 6.3: Convergence history for one-dimensional optimization case

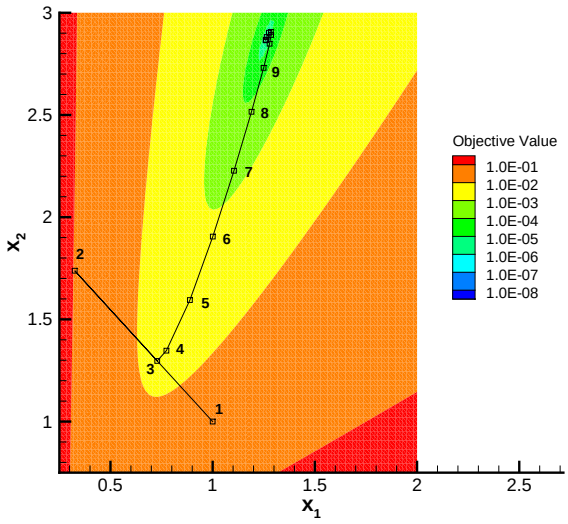

Figure 6.4: Design variable evolution on objective function value contours

are perturbed separately by a small value, $\Delta X = 10^{-7}$. The perturbation of the design variables occurs after being scaled by their initial values. The objective function values are calculated after completing one steady analysis iteration. The percent error in the objective gradient with respect to E_s and ρ_s are are acceptable and are summarized in Table 6.1.

Table 6.1: Gradient accuracy for 1-D optimization problem

X	Gradient (FD)	Gradient (adjoint)	% Error
E_s	7.00361541 $\times 10^{-9}$	7.00356984 $\times 10^{-9}$	0.000651
ρ_s	-7.66286387 $\times 10^{-5}$	-7.66211408 $\times 10^{-5}$	0.009785

6.3 Two-dimensional aero-structural analysis

6.3.1 Sample results

The 2-D aero-structural analysis problem is solved using both the partitioned and monolithic approaches at various operating conditions. In both cases the free-stream fluid density is $\rho_\infty = 1.2$ kg/m^3, and the Young's modulus of the tab is $E_s = 250$ Pa. The first operating condition considered is a fluid Mach number of 0.6 and angle of attack of 2.0°. In the partitioned approach, the fluid residual is reduced two orders of magnitude at each iteration, the structural residual is solved completely, and the mesh-movement residual is reduced eight orders of magnitude at each iteration. The partitioned system is converged when each sub-system residual is reduced by ten orders of magnitude at the start of the iteration. In the monolithic approach, the system residual is dropped ten orders of magnitude. The fluid solution contours are reasonably smooth, with no shocks being generated on the airfoil-tab surface. The deformation of the finite-element tab structure is relatively small, with the deflection in the y-direction of the structural node on the top surface at the tip (d_{tip}) equal to about 1.6×10^{-4} chord lengths. The lift coefficient of the steady solution is $C_l = 0.3287$. Dimensionless fluid density contours are shown in Figure 6.5, and tab deformation is shown in Figure 6.6.

A second scenario is considered for the 2-D aero-structural analysis problem consisting of a fluid Mach number of 0.75 and an angle of attack of 8.0°. As in the first case, the partitioned solver reduces the fluid residual two orders of magnitude at each iteration while the monolithic approach drops the system residual by ten orders of magnitude. The fluid solution contours in this case show a shock wave formed on the top surface of the airfoil. The deformation of the finite-element tab structure is considerable in this case, with the deflection in the y-direction of the structural node on the top surface at the tip (d_{tip}) equal to about 1.1×10^{-3} chord lengths. The steady C_l value is equal to 1.5111. Dimensionless fluid density contours are shown in Figure 6.7, and tab deformation is shown in Figure 6.8.

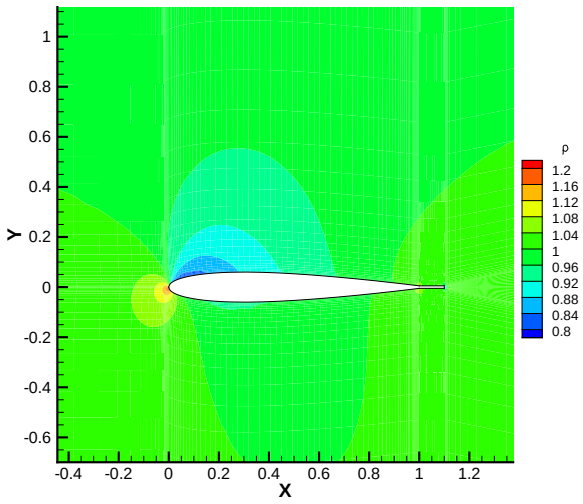

Figure 6.5: Density contours over airfoil-tab at $M = 0.6$, $\alpha = 2.0°$

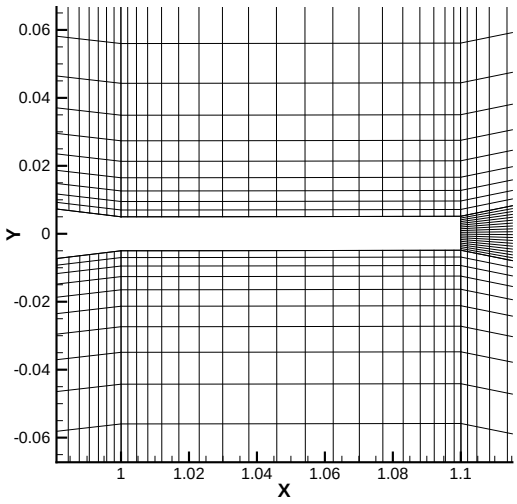

Figure 6.6: Tab deformation at $M = 0.6$, $\alpha = 2.0°$

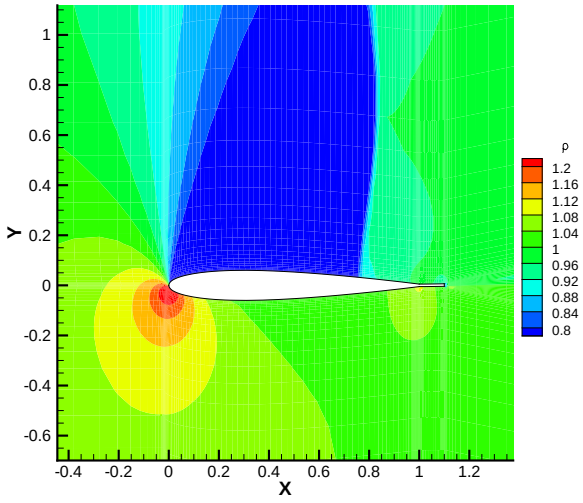

Figure 6.7: Density contours over airfoil-tab at $M = 0.75$, $\alpha = 8.0°$

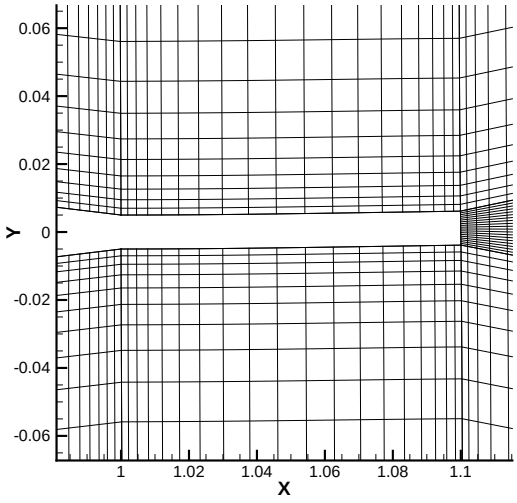

Figure 6.8: Tab deformation at $M = 0.75$, $\alpha = 8.0°$

6.3.2 Monolithic vs. partitioned performance

These two 2-D aero-structural analysis cases are solved with both the partitioned and monolithic solution schemes. For the first case with $M = 0.6$ and $\alpha = 2.0°$ the partitioned solution takes 2538 s to complete 15 full fluid-structure-mesh-movement iterations and

Table 6.2: Summary of partitioned and monolithic 2-D analysis results

M	$\alpha(^\circ)$	Partitioned time (s)	iterations	Monolithic time (s)	iterations
0.6	2.0	2538	15	1323	57
0.75	8.0	3896	21	1546	67

converge ten orders of magnitude. The monolithic approach converges in 1323 s, completing 57 iterations. For the second case with $M = 0.75$ and $\alpha = 8.0^\circ$ the partitioned solution takes 3896 s to complete 21 full iterations, while the monolithic approach converges in 1546 s, completing 67 iterations. These results are summarized in Table 6.2.

The convergence behaviour of the partitioned solver for the two cases are shown in Figure 6.9 and Figure 6.10, respectively, where the structural residual is given before the solution of the structural system. The convergence behaviour of the monolithic solver for the two cases is shown in Figures 6.11 and 6.12, respectively. The monolithic solution scheme switches from approximate-Newton phase to the inexact-Newton phase after a residual reduction of two orders of magnitude, affecting the fluid sub-matrix, at 44 and 53 iterations for the two cases, respectively.

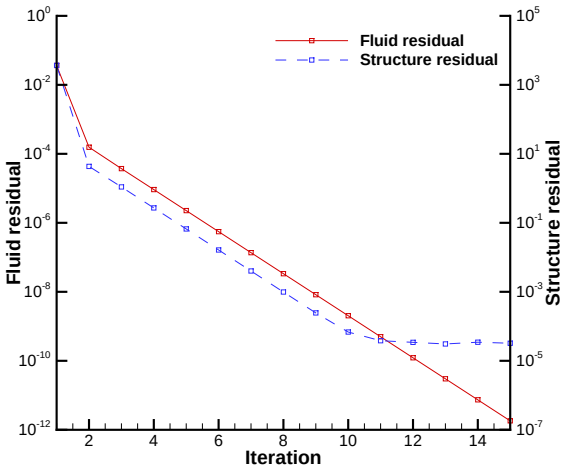

Figure 6.9: Partitioned solver convergence for $M = 0.60$, $\alpha = 2.0^\circ$

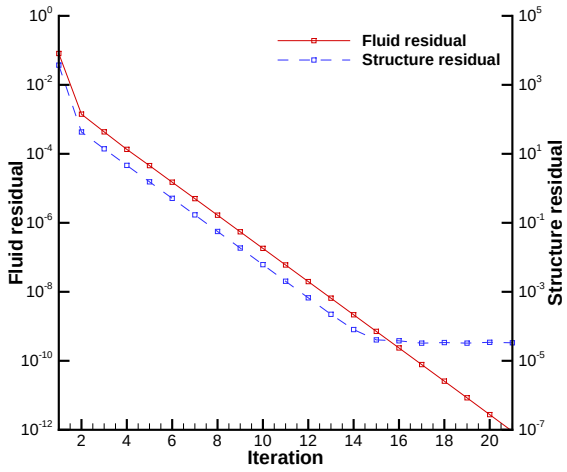

Figure 6.10: Partitioned solver convergence for $M = 0.75$, $\alpha = 8.0^\circ$

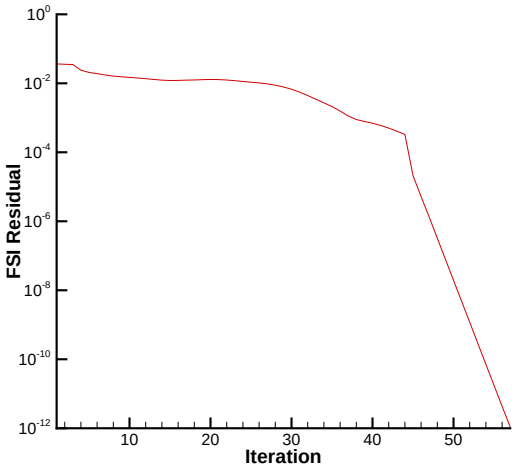

Figure 6.11: Monolithic solver convergence for $M = 0.60$, $\alpha = 2.0°$

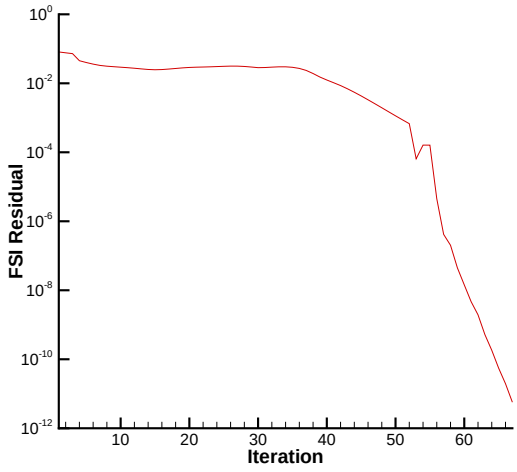

Figure 6.12: Monolithic solver convergence for $M = 0.75$, $\alpha = 8.0°$

6.4 Two-dimensional aero-structural optimization

6.4.1 Unconstrained optimization with single design variable

The unconstrained aero-structural optimization problem with one design variable, described in Section 5.2.1 is solved using the monolithic approach for the aero-structural analysis. The gradients are calculated using both the aero-structural adjoint approach and the direct approach. The adjoint and direct equations are solved with both monolithic and partitioned approaches. The optimization process, including the design variable updates, is performed using the L-BFGS-B algorithm. The fluid Mach number is 0.6 and the initial angle of attack is 2.0°. The fluid free-stream density is $\rho_\infty = 1.2$ kg/m^3 and the Young's modulus of the tab is $E_s = 250$ Pa. The optimization process converges in 7 total iterations, of which 5 are outer iterations that result in satisfying both the BFGS sufficient decrease condition and the curvature condition. The objective value is reduced roughly 18 orders of magnitude and the gradient magnitude is reduced roughly 10 orders of magnitude. The angle of attack at the local optimum is $\alpha = 2.57297°$, resulting in a tab tip y-deflection, d_{tip}, equal to 2.0×10^{-4} chord lengths. The lift coefficient of the airfoil-tab geometry is $C_l = 0.4238$. The convergence history of the optimization is shown in Figure 6.13, where shaded squares indicate outer iterations and the blanked squares indicate inner iterations that do not satisfy both the sufficient decrease and curvature

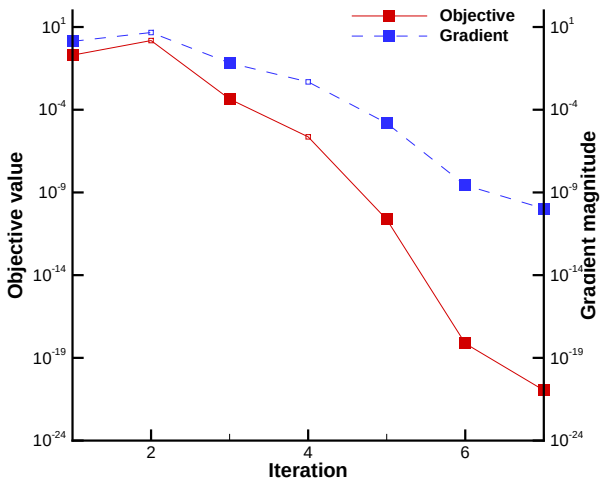

Figure 6.13: Convergence history for unconstrained optimization with single design variable

Table 6.3: Gradient accuracy for unconstrained optimization with single design variable

X	Gradient (FD)	Gradient (adjoint)	% Error
α	-0.678079095629	-0.678079208451	0.000017

conditions.

The accuracy of the computed gradient is tested using finite-differences. The design variable, α, is perturbed a small value ($\Delta\alpha = 1 \times 10^{-4}$) in both the positive and negative directions from the initial angle of attack, $\alpha = 2.0°$. The percent error in the gradient calculated from the adjoint approach is within an acceptable margin and is given in Table 6.3. For the monolithic direct and adjoint solves the residual is reduced by ten orders of magnitude, while for the partitioned solves each system residual is reduced by two orders of magnitude until the residual of each system is reduced by ten orders of magnitude at the start of the iteration.

The timing performance of the optimization is also measured. For each optimizer iteration, the primary contributions to the algorithm run time are divided into three components: time to perform the monolithic FSI solve, time to compute the fluid coupling sub-matrix with finite-differences, and time to solve the gradient equations (adjoint method or direct method with both monolithic and partitioned solves). The average it-

Table 6.4: Timing results for unconstrained optimization with single design variable

Task	Timing factor			Iterations
	ILU	GMRES	Total	
FSI solve	-	-	1.000	-
Fluid coupling	-	-	0.913	-
Gradient (direct) Monolithic	0.057	0.161	0.218	176
Gradient (direct) Partitioned	0.050	0.032	0.082	17
Gradient (adjoint) Monolithic	0.305	0.146	0.451	185
Gradient (adjoint) Partitioned	0.050	0.054	0.104	32

eration timing results, normalized to the FSI solve, are given in Table 6.4. The total time for the gradient calculation is divided into time for forming the ILU factors for preconditioning and the iterative GMRES solution. The partitioned solution scheme converges faster than the monolithic solution scheme for both the adjoint and direct approaches, and the direct solves are both faster than their equivalent adjoint solves. The number of iterations given correspond to the number of GMRES iterations for the monolithic cases and the number of Gauss-Seidel iterations for the partitioned cases.

6.4.2 Inverse design optimization

The inverse design aero-structural optimization problem described in Section 5.2.2 is solved. The aero-structural analysis is performed using the monolithic approach and the gradients are calculated using both the adjoint and direct formulations, which are each solved with monolithic and partitioned approaches. The design variable updates are calculated using the L-BFGS-B algorithm. Constant lower and upper bounds are placed on the structural design variables, z_m, to ensure the tab tip nodes do not cross over each other and produce an infeasible result. The free-stream fluid conditions consist of a Mach number of 0.6 and density of $\rho_\infty = 1.2$ kg/m^3, while the initial angle of attack is 2.0°. The Young's modulus of the tab is $E_\mathrm{s} = 250$ Pa. 13 total iterations are required for convergence, of which 12 are outer iterations that satisfy both the BFGS sufficient decrease and curvature condition. A reduction of objective value of about 16 orders of magnitude is achieved. The magnitude of the objective gradient is also reduced roughly 10 orders of magnitude. At convergence of the optimizer, the design variables are each equal (to at least ten significant figures) to the values used to obtain the target force

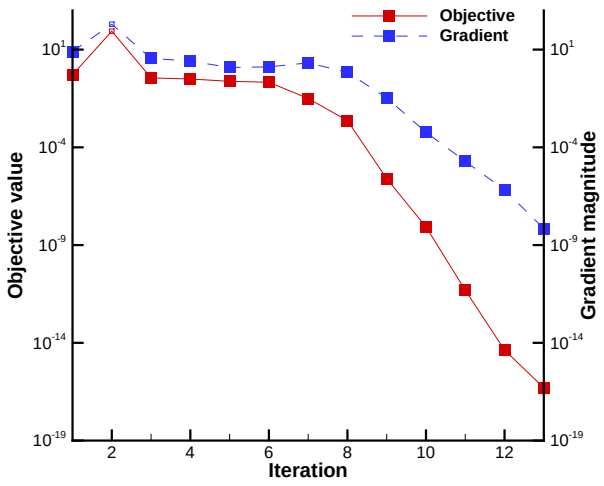

Figure 6.14: Convergence history for inverse design optimization

Table 6.5: Gradient accuracy for inverse design optimization

X	Gradient (FD)	Gradient (adjoint)	% Error
α	3.02540688989	3.02540881750	0.000064
z_1	-10113.3379624	-10113.3399738	0.000020
z_2	2390.269152279	2390.27541625	0.000262

vector, f_i^* : $\alpha = 2.25°$, $z_1 = 2.0 \times 10^{-4}$ chord lengths and $z_2 = 1.0 \times 10^{-4}$ chord lengths. The steady lift coefficient at the optimum is $C_l = 0.3656$. The convergence history of the optimization is given in Figure 6.14, where the shaded squares indicate outer optimizer iterations and the blanked squares indicate iterations that do not satisfy the sufficient decrease and curvature conditions.

Finite-differencing is used to evaluate the accuracy of the computed gradients, where step sizes of $\Delta\alpha = 1 \times 10^{-4}$ and $\Delta z_m = 1 \times 10^{-6}$ are used in the calculation. The percent error in the gradient calculated from the adjoint approach with respect to α, z_1 and z_2 are each acceptably small and are given in Table 6.5. As in the unconstrained optimization case, the monolithic residuals are reduced by ten orders of magnitude while the partitioned Gauss-Seidel iterations are considered complete when the residual norm of each individual system has been reduced by ten orders of magnitude at the start of the iteration.

The average iteration timing results, normalized to the length of the FSI solve, are given in Table 6.6. As before, the GMRES and Gauss-Seidel iterations are given for the monolithic and partitioned cases, respectively. In this case, the adjoint approaches are slightly faster than their corresponding direct methods, which follows the fact that the cost of the direct approach is proportional to the number of design variables. The partitioned approaches are each faster than their monolithic counterparts as well. As before, the gradient timing factors are divided into the one-time cost of forming the ILU factors and the time for the GMRES iterations, which scales with number of design variables for the direct method and number of functions (in this case just the objective) for the adjoint method.

Comparing the timing results for this case with those from the unconstrained optimization with a single design variable (see Section 6.4.1), and assuming consistent FSI solve times between the two cases, the cost per design variable of both the direct and adjoint approaches is approximated. The timing factor per design variable is approximated as the difference in timing factor between the two optimization cases divided by the increase in design variables (in this case, two). These timing factors per design variable values are also given in Table 6.6. For a monolithic approach, the adjoint solution is slightly faster with three design variables than the direct approach, and so for problems with three or more design variables, the adjoint approach should always be more efficient. For a partitioned approach, the direct method is faster than the adjoint method for three design variables, and so the cross-over point is estimated to be at about 7 design variables by considering the change in time between the cases with one and three design variables. These reference values, however, will vary when considering additional functions of interest (e.g., constraints), which will increase the cost of the adjoint method for each function considered.

6.4.3 Constrained optimization with multiple design variables

The constrained optimization problem with multiple design variables described in Section 5.2.3 is solved with monolithic aero-structural analysis approach. The gradients are calculated using both the discrete-adjoint approach and the direct approach. Both the adjoint and direct equations are solved with monolithic and partitioned approaches. The SNOPT algorithm is used to perform the optimization and compute the design variable updates. The maximum stress (KS function formulation) and target lift constraint are

Table 6.6: Timing results for inverse design optimization

Task	Timing factor				Iterations
	ILU	GMRES	Total	Per X	
FSI solve	-	-	1.000	-	-
Fluid coupling	-	-	0.913	-	-
Gradient (direct) Monolithic	0.057	0.501	0.558	0.170	554
Gradient (direct) Partitioned	0.050	0.088	0.138	0.028	50
Gradient (adjoint) Monolithic	0.305	0.150	0.455	0.002	189
Gradient (adjoint) Partitioned	0.050	0.040	0.090	-0.007	24

handled by the optimizer directly. Constant lower and upper bounds are placed on the structural design variables, z_m, to ensure the tab tip nodes do not cross over each other and produce an infeasible result. The scaling values in the objective function, I_W and I_T, are set to 1.0 and 1.0×10^3, respectively, such that the structural weight and tip deflection terms are of the same order of magnitude at the first iteration. The target lift coefficient value is set to $C_l^* = 0.4$. The yield stress, σ_Y, in the KS function is set to 700 Pa, which is 30% larger than the maximum elemental stress in the structure at the steady solution of the initial iterate. Like the previous two optimization cases, the fluid is specified with a Mach number of 0.6 and free-stream density of $\rho_\infty = 1.2$ kg/m^3. The initial angle of attack is 2.0°. The Young's modulus of the tab structure is $E_\mathrm{s} = 250$ Pa.

The optimization process runs for 18 iterations, of which 4 satisfy both the lift and elemental stress (KS function) constraints. The convergence history is given in Figure 6.15, where the shaded symbols indicate feasible iterations which satisfy both the target lift and maximum elemental stress constraints. The objective function consists of a structural weight term and a tab tip deflection from initial shape term. At the last iteration, the structural weight is reduced to 83.2% of the initial weight. The reduction in structural weight is achieved by decreasing the undeflected structural thickness at the free end of the tab, resulting in a reduction of tab area (in 2-D). The objective value is reduced from about 1.15 to about 0.34, with the majority of this decrease a result of reduction of the tab tip deflection term, T.

The design variables at the final iteration are $\alpha = 3.298°$, $z_1 = 2.898243 \times 10^{-3}$ chord lengths and $z_2 = 6.251193 \times 10^{-3}$ chord lengths. This structural shape is shown in Figure 6.16, with the initial, undeflected tab shape (shown in thick red lines) included

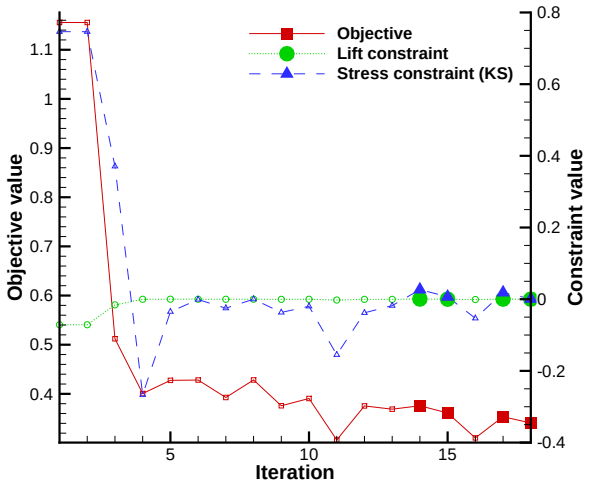

Figure 6.15: Convergence history for constrained optimization with multiple design variables

for reference. The reduction of tab tip deflection is achieved by displacing the initial structural tab tip locations in the positive y-direction such that the steady deflection is in the negative y-direction. At the final iteration the maximum elemental stress is about $\sigma_V^{(\max)} = 677$ Pa, or 3.3% less than the yield stress, and the lift coefficient is $C_l = 0.40005$. At the final iteration, the steady tab tip deflection magnitude (measured relative to the shape defined by z_1 and z_2), d_{tip}, is equal to -4.92×10^{-4} chord lengths.

The accuracy of computed gradients is evaluated with finite-differences. Step sizes of $\Delta\alpha = 1 \times 10^{-4}$ and $\Delta z_m = 1 \times 10^{-6}$ are used to to perturb the design variables and measure the change in objective function, maximum stress constraint (KS function) and target lift constraint. The percent error in the objective, stress constraint, and lift constraint gradients with respect to the design variables are each within are each acceptably small when calculated with the adjoint approach. The gradient accuracy is summarized in Table 6.7. As before, the monolithic gradient solution is converged by reducing the residual by ten orders of magnitude. The partitioned system is considered converged when the residual of each system is reduced by ten orders of magnitude at the start of the Gauss-Seidel iteration.

The average iteration timing results, normalized to the length of the FSI solve, are given in Table 6.8. As before, the total time for each gradient solution is divided into

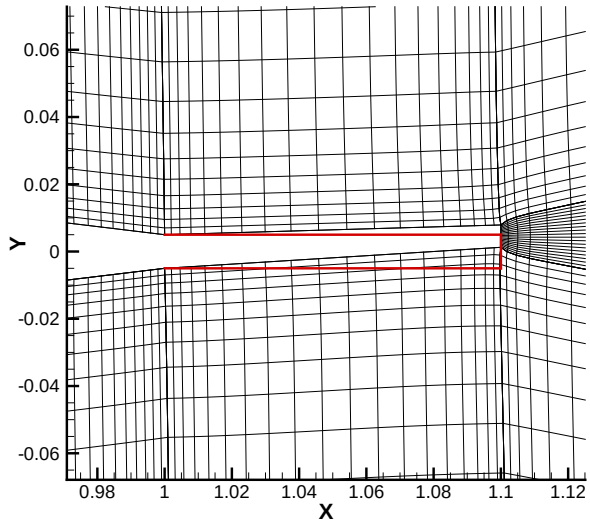

Figure 6.16: Undeflected structural shape result from constrained optimization with initial shape shown in red

time for the ILU decomposition and time for the GMRES iterations, and the number of GMRES and Gauss-Seidel iterations are provided for the monolithic and partitioned cases, respectively. In this case, the flow-sensitivity (direct) solutions are more efficient in calculating the objective and constraint gradients than the adjoint equations for both the partitioned and monolithic approaches. The computational cost of the direct solutions is comparable to the cost for the inverse design problem (also using three design variables), which is expected. The cost of the adjoint solution is greater than the cost for the inverse design problem (single function), which is also expected as the cost of the adjoint solve is proportional to the number of functions being considered (in this case, three). As well, in this case the partitioned adjoint calculation is more efficient than the monolithic.

We also compare the timing results to those from inverse optimization problem in Section 6.4.2, as both problems have the same number of design variables but different numbers of functions (objective and constraints). The time per function is computed by taking the difference between total time of gradient solves between the two cases and dividing by the difference in number of functions (in this case, two). As expected, the adjoint gradient solutions scale with the number of functions while the direct solutions do not. The cost per function is also used to estimate a cross-over point at which a direct or adjoint solution is more efficient. In this case, for a constant number of design variables,

Table 6.7: Gradient accuracy for constrained optimization with multiple design variables

	α	z_1	z_2
$\mathcal{J}$ gradient (FD)	0.076063370150	-1.3680100996	-114.99841210
$\mathcal{J}$ gradient (adjoint)	0.076063370852	-1.3680208900	-114.99824634
% error	0.000001	0.000789	0.000114
KS gradient (FD)	-0.096831676695	84.2834479098	63.6688661049
KS gradient (adjoint)	-0.096831676852	84.2842222063	63.6686678220
% error	0.0000002	0.000919	0.000311
Lift gradient (FD)	0.165382321123	-14.9222261805	-14.9379434061
Lift gradient (adjoint)	0.165380860335	-14.9221850730	-14.9378930603
% error	0.000883	0.000275	0.000337

Table 6.8: Timing results for constrained optimization with multiple design variables

Task	Timing factor				Iterations
	ILU	GMRES	Total	Per func.	
FSI solve	-	-	1.000	-	-
Fluid coupling	-	-	0.913	-	-
Gradient (direct) Monolithic	0.057	0.515	0.572	0.007	564
Gradient (direct) Partitioned	0.050	0.084	0.134	-0.002	53
Gradient (adjoint) Monolithic	0.305	0.428	0.733	0.139	541
Gradient (adjoint) Partitioned	0.050	0.098	0.148	0.029	63

the monolithic solution is faster with a direct approach than with the adjoint approach for two or more functions. The direct approach is faster than the adjoint approach for any number of functions in the partitioned case.

Chapter 7

Conclusions and Recommendations

7.1 Conclusions

A monolithic approach to aero-structural analysis and optimization was developed and implemented. Two test configurations were considered in this work: a 1-D unsteady FSI problem and a 2-D steady aero-structural problem based on a rigid NACA 0012 airfoil with a flexible, horizontal rectangular tab at the trailing edge. For the 1-D work, a new solver was created based on the 1-D version of the ARC2D algorithm. The fluid-structure coupling was handled through a spatial coordinate transformation and local time linearization. For the 2-D work, an existing flow solver, *Diablo*, was modified to handle a three-field aero-structural formulation including aerodynamic, structural, and fluid mesh-movement (linear elasticity) systems. Three-way coupling was developed systematically through the linearization of equation residuals with respect to the variables of each system. Steady optimization problems were also developed for both the 1-D and 2-D test configurations. Gradient-based optimization algorithms L-BFGS-B and SNOPT were used to calculate design variable updates. Sensitivity analysis was completed using the discrete aero-structural adjoint and flow-sensitivity (direct) approaches. Comparisons to a partitioned approach were performed, where appropriate.

In general, the monolithic approach showed to be a viable option for aero-structural analysis and optimization. Linear structural and mesh-movement formulations integrated well into the existing aerodynamic system formed in *Diablo*. From a 2-D aero-structural analysis perspective, the monolithic approach, on average, converged about 1.9 to 2.5 times faster than the partitioned approach, while enforcing solution compatibility across the system interfaces at each sub-iteration.

From an optimization perspective, the gradient calculation through the solution of the discrete-adjoint or flow-sensitivity equations was simplified by excluding the necessity of a partitioned approach. As well, the left-hand side from the monolithic aero-structural analysis was re-used in these calculations. However, the monolithic solution of the discrete-adjoint and flow-sensitivity equations was considerably slower than the partitioned approach for the test problems in this study.

The monolithic approach required considerable development and implementation effort over the partitioned approach, which can utilize existing fluid, structural, and mesh-movement solvers, each of which are tailored to their respective sub-problems.

7.2 Recommendations

Further advancements in the monolithic approach are required to solve larger, 3-D problems of interest in aero-structural analysis and design.

In terms of monolithic system development, it is recommended to:

1. Extend the 2-D monolithic approach in this work to handle larger, 3-D problems of practical interest. The most significant tasks in this item include, but are not limited to: parallelization of existing approach to handle larger problems, developing a relevant 3-D test problem geometry, discretizing the structural domain and expressing in a finite-element formulation, and extending the fluid mesh-movement scheme to 3-D.

2. Incorporate additional aerodynamic shape and structural degrees of freedom (design variables) in the aero-structural optimization to allow for more flexibility in aero-structural designs (e.g., B-spline control points/surfaces for aerodynamics, element thickness for structural members).

3. Compare the monolithic approach (3-D) to a partitioned approach by coupling *Diablo* to an existing structural solver and mesh-movement scheme.

In terms of further advancement of the current 2-D monolithic implementation, there exists some room for improvement in terms of accuracy, efficiency, and robustness. To further develop the existing implementation, it is recommended to:

1. Develop and implement an analytical approach to calculating the fluid coupling sub-matrix, $\partial R_A / \partial Q_M$, to increase the efficiency of the gradient calculation at

each optimizer iteration compared to the current finite-difference approach. If the cost of this calculation can be significantly reduced, then this term can be included in the aero-structural analysis as well, which may increase the rate of convergence of the monolithic system and further decrease computational cost.

2. Decrease the size of the fluid mesh-movement domain by designating fluid nodes sufficiently far from the structural deformation as rigid. The result is a smaller monolithic system to be solved at each iteration and a reduction in computational cost. As well, consider reducing the mesh-movement degrees of freedom by using a B-spline control points approach.

3. Remove the limitation of defining finite-element structural nodes from the position of fluid nodes on the fluid-structure interface. That is, incorporate a conservative fluid load transfer scheme to the structure and a structural displacement transfer scheme to the mesh-movement that does not require nodes on the interface to be coincident.

4. Incorporate additional aerodynamic shape and structural design variables for more flexibility in initial geometry and undeflected structural shape.

References

[1] Airline fuel and labour cost share. IATA economic briefing, International Air Transportation Association, 2010.

[2] Barcelos, M., Bavestrello, H., and Maute, K. A Schur-Newton-Krylov solver for stready-state aeroelastic analysis and design sensitivity analysis. *Computer Methods in Applied Mechanics and Engineering 195* (2006), 2050–2069.

[3] Blom, F. J. A monolithical fluid-structure interaction algorithm applied to the piston problem. *Computer Methods in Applied Mechanics and Engineering 167* (1998), 369–391.

[4] Degroote, J., Bathe, K.-J., and Vierendeels, J. Performance of a new partitioned scheme versus a monolithic procedure in fluid-structure interaction. *Computers and Structures 87* (2009), 793–801.

[5] Farhat, C., van der Zee, K. G., and Geuzaine, P. Provably second-order time-accurate loosely-coupled solution algorithms for transient nonlinear computational aeroelasticity. *Computer Methods in Applied Mechanics and Engineering 195* (2006), 1973–2001.

[6] Gee, M., Küttler, U., and Wall, W. Truly monolithic algebraic multigrid for fluid-structure interaction. *International Journal for Numerical Methods in Engineering 85* (2010), 987–1016.

[7] Gill, P. E., Murray, W., and Saunders, M. A. SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization. *SIAM Review 47*, 1 (2005), 99–131.

[8] Heil, M., Hazel, A. L., and Boyle, J. Solvers for large-displacement fluid-structure interaction problems: segregated versus monolithic approaches. *Computational Mechanics 43*, 1 (2008), 91–101.

[9] Hicken, J. E. *Efficient algorithms for future aircraft design: contributions to aerodynamic shape optimization*. PhD thesis, University of Toronto, 2009.

[10] Hicken, J. E., and Zingg, D. W. Parallel Newton-Krylov solver for the Euler equations discretized using simultaneous-approximation terms. *AIAA Journal 46*, 11 (2008), 2773–2786.

[11] Hicken, J. E., and Zingg, D. W. Induced-drag minimization of nonplanar geometries based on the Euler equations. *AIAA Journal 48* (2010), 2564–2575.

[12] Horáček, J., Sváček, P., Růžička, M., and Feistauer, M. Contribution to finite element modelling of airfoil aeroelastic instabilities. *Applied and Computational Mechanics 1*, 1 (2007), 43–52.

[13] Kennedy, G. J., and Martins, J. R. R. A. A regularized discrete laminate parametrization technique with applications to wing-box design optimization. In *53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference* (Honolulu, HI, April 2012).

[14] Kenway, G. K. W., Kennedy, G. J., and Martins, J. R. R. A. A scalable parallel approach for high-fidelity steady-state aeroelastic analysis and adjoint derivative computations. *AIAA Journal* (2012). (In press).

[15] Ladson, C. L., Brooks, C. W., Jr., Hill, A. S., and Sproles, D. W. Computer program to obtain ordinates for naca airfoils. *Journal of Statistical Physics 104* (1996).

[16] Lasauskas, E., Lutz, T., and Dietz, M. Influence of trailing edge tab on moment characteristics of NACA 23012 airfoil. *Aviation 11*, 4 (2007), 3–8.

[17] Lefrançois, E., and Boufflet, J.-P. An introduction to fluid-structure interaction: application to the piston problem. *SIAM Review 52*, 4 (2010), 747–767.

[18] Leung, T. M., and Zingg, D. W. Aerodynamic shape optimization of wings using a parallel Newton-Krylov approach. *AIAA Journal 50* (2012), 540–550.

[19] Leung, T. M.-M. *A Newton-Krylov approach to aerodynamic optimization in three dimensions*. PhD thesis, University of Toronto, 2010.

[20] Liu, B., and Haftka, R. Single-level composite wing optimization based on flexural lamination parameters. *Structural and Multidisciplinary Optimization 26* (2004), 111–120.

[21] Logan, D. L. *A first course in the finite element method*, 4 ed. Nelson, 2007.

[22] Martins, J. R. R. A., Alonso, J. J., and Reuther, J. J. High-fidelity aerostructural design optimization of a supersonic business jet. In *Proceedings of the 43rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference* (Denver, CO, April 2002). AIAA 2002-1483.

[23] Martins, J. R. R. A., Alonso, J. J., and Reuther, J. J. High-fidelity aerostructural design optimization of a supersonic business jet. *Journal of Aircraft 41*, 3 (2004), 523–530.

[24] Maute, K., Nikbay, M., and Farhat, C. Coupled analytical sensitivity analysis and optimization of three-dimensional nonlinear aeroelastic systems. *AIAA Journal 39*, 11 (2001), 2051–2061.

[25] Michler, C., Hulshoff, S. J., van Brummelen, E. H., and de Borst, R. A monolithic approach to fluid-structure interaction. *Computers & Fluids 33* (2004), 839–848.

[26] Moré, J. J., and Thuente, D. J. Line search algorithms with guaranteed sufficient decrease. *ACM Transactions on Mathematical Software 20* (1992), 286–307.

[27] Nemec, M. *Optimal shape design of aerodynamic configurations: a Newton-Krylov approach*. PhD thesis, University of Toronto, 2003.

[28] Nocedal, J., and Wright, S. J. *Numerical Optimization*, 2nd ed. Springer, New York, 2006.

[29] Nordström, J., and Eriksson, S. Fluid structure interaction problems: the necessity of a well posed, stable and accurate formulation. *Communications in Computational Physics 8*, 5 (2010), 1111–1138.

[30] Pulliam, T., and Steger, J. Implicit finite-difference simulations of three-dimensional compressible flow. *AIAA Journal 18*, 1 (1980), 159–167.

[31] Pulliam, T. H. Numerical techniques for viscous flow computation in turbomachinery bladings. In *Lecture notes for the von Kármán institute for fluid dynamics lecture series* (Rhode-St-Genese,Belgium, 1985), von Kármán institute.

[32] Reuther, J. J., Alonso, J. J., Martins, J. R. R. A., and Smith, S. C. A coupled aero-structural optimization method for complete aircraft configurations. AIAA Paper 99–0187, American Institute of Aeronautics and Astronautics, 1999.

[33] Reuther, J. J., Jameson, A., Alonso, J. J., , Rimlinger, M. J., and Saunders, D. Constrained multipoint aerodynamic shape optimization using an adjoint formulation and parallel computers, part 1. *Journal of Aircraft 36*, 1 (1999), 51–60.

[34] Robinson-Mosher, A., Schroeder, C., and Fedkiw, R. A symmetric positive definite formulation for monolithic fluid structure interaction. *Journal of Computational Physics 230* (2011), 1547–1566.

[35] Saad, Y. SPARSKIT: a basic tool kit for sparse matrix computations - Version 2, 1994.

[36] Truong, A. H. Development of grid movement algorithms suitable for aerodynamic optimization. Master's thesis, University of Toronto, 2005.

[37] Truong, A. H., Oldfield, C. A., and Zingg, D. W. Mesh movement for a discrete-adjoint Newton-Krylov algorithm for aerodynamic optimization. *AIAA Journal 46*, 7 (2008), 1695–1704.

[38] Zhu, C., Byrd, R. H., Lu, P., and Nocedal, J. Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on Mechanical Software 23*, 4 (Dec. 1997), 550–560.