Aerodynamic Shape Optimization for Unsteady Flows: Some Benchmark Problems

Kwesi P. Apponsah* and David W. Zingg.†
Institute for Aerospace Studies, University of Toronto, Toronto, Ontario, M3H 5T6, Canada

We present an efficient aerodynamic shape optimization framework for optimization problems under unsteady flow conditions. The optimization framework consists of a parallel Newton-Krylov flow solver for multi-block grids and an integrated geometry parameterization and mesh-deformation algorithm based on linear elasticity. We apply the adjoint method to the discretized governing equations to compute the gradients required by the sequential quadratic programming optimization algorithm. We propose two lift-constrained drag minimization problems for the purposes of testing and evaluating the framework. First, we consider a laminar flow airfoil optimization problem at a Reynolds number of 800 and also investigate the convexity of the optimization problem. We show that the optimizer is capable of reducing the drag for this problem by about 23% and produces a nearly steady flow compared to the vortex shedding observed for the baseline geometry at the required lift target. The second benchmark case is a lift-constrained drag minimization of an aspect ratio eight rectangular wing in a laminar flow at a Reynolds number of 800. Section shape, twist, and angle of attack are free. Although only partially converged at the time of writing, the preliminary results show a 20% drag reduction.

I. Nomenclature

$\mathbf{B}$	mesh-deformation variables
C_l, C_L	lift coefficient
C_d, C_D	drag coefficient
c	chord
Δt	time step
$\mathbf{E}$, $\mathbf{F}$, $\mathbf{G}$	inviscid fluxes
$\mathbf{E_v}$, $\mathbf{F_v}$, $\mathbf{G_v}$	viscous fluxes
$\mathcal{F}$	mesh-deformation and flow residuals
$\mathcal{J}$	objective function
$\mathcal{L}$	Lagrangian function
m	number of mesh-deformation increments
$\mathcal{M}$	mesh-deformation residual
N	number of time steps
$\mathbf{Q}$	conserved flow variables
$\mathcal{R}$	flow residual
$\tilde{\mathcal{R}}$	temporal component of flow residual
$\hat{\mathcal{R}}$	spatial component of flow residual
$\mathbf{V}$	mesh-deformation and conserved flow variables
$\mathbf{X}$	design variables
x, y, z	Cartesian coordinates
ξ, η, ζ	curvilinear coordinates

*PhD Candidate, AIAA Student Member (kwesi.apponsah@mail.utoronto.ca)
†University of Toronto Distinguished Professor of Computational Aerodynamics and Sustainable Aviation, Director, Centre for Research in Sustainable Aviation, Director, Centre for Computational Science and Engineering, AIAA Associate Fellow (dwz@utias.utoronto.ca)

II. Introduction

Computational tools currently play an important role in helping scientists and engineers understand the behaviour of physical systems. This is a result of improvements in both computing power and numerical algorithms. With the availability of high performance computing resources, the study of a wide range of scientific problems has become practical, of which fluid dynamics related problems have gained considerable attention. Over the last three decades, the use of computational fluid dynamics (CFD) algorithms has grown from the analysis of fluid problems to encompass sensitivity studies as well as optimization of aerodynamic shapes.

In terms of capability, CFD simulations have also evolved in fidelity over the years to allow engineers to model more complex problems as dictated by the level of computing resources available. Low fidelity CFD tools such as panel methods [1, 2] for analysis and design are well established. On the medium fidelity side, CFD algorithms solving the Euler, Reynolds-Averaged Navier-Stokes (RANS) and Unsteady RANS (URANS) equations for the purposes of analysis are also well established, while in terms of design, these tools are considered to be the state-of-the-art. High fidelity CFD simulations, dominated mainly by Detached Eddy Simulations (DES), Large-Eddy Simulations (LES), Implicit LES and Direct Numerical Simulations (DNS) are considered the state-of-the-art for the purposes of analysis. On the design end, such high-fidelity simulations are an active research area.

While the analysis of fluid problems has been central to the application of CFD tools, using such tools to drive the optimization of aerodynamic shapes is indispensable to pushing the design envelope for aerospace applications. The work by Hicks and Henne [3] was among the first to explore numerical optimization within the context of aerodynamic design, using finite differences to compute the gradient of the objective with respect to the design variables. Although this pioneering work hinted at the potential of aerodynamic design, using finite differences limits the number of design variables that can be used for practical problems since the cost of computing gradients is proportional to the number of design variables. Using control theory, Pironneau [4] introduced the adjoint approach and showed that the gradients can be computed at a cost equivalent to that of a flow evaluation, independent of the number of design variables. Jameson *et al.* [5], were among the first to explore shape optimization using the Euler and Navier-Stokes equations for steady flows. Using the RANS equations and the one-equation Spalart-Allmaras turbulent model, Anderson and Bonhaus [6] formulated a discrete adjoint methodology for aerodynamic design. The above contributions and many others have made the adjoint method widely used for optimization under steady flow conditions.

Unlike steady-state flow optimization problems, aerodynamic shape optimization for unsteady flows has not received as much attention. This is primarily due to the high cost associated with such problems and the challenge of gradient computation for chaotic flows [7]. However, practical design problems associated with turbo-machinery blades, helicopter blades, high-lift devices, aeroelastic flutter, transonic buffeting and low Reynolds number vortex-shedding all require unsteady flow analysis.

Some of the earlier work on optimization under unsteady flow conditions was carried out by Yee *et al.* [8]. They conducted aerodynamic shape optimization for rotor airfoils using the unsteady viscous flow equations. Nadarajah and Jameson [9] applied the adjoint method to the URANS equations and optimized airfoils to minimize time-averaged drag at a fixed time-averaged lift. Mani and Mavriplis [10] performed shape design on deforming structured meshes in two dimensions using the unsteady Euler equations. Nielsen *et al.* [11] applied a discrete adjoint method to the URANS equations on dynamic unstructured grids for the purposes of aerodynamic design. Further, Rumpfkeil and Zingg [12], also applied the discrete adjoint methodology to a remote inverse optimization problem of a laminar flow around the multi-element NLR 7301 configuration at a high angle of attack. The algorithm was also extended to allow for efficient far-field noise minimization through shape design changes [13].

The goal of this work is to extend the two-dimensional unsteady optimization framework developed in [12] to three-dimensions using an efficient optimization framework [14–18] for steady flows and an unsteady flow analysis algorithm [19]. We will also discuss benchmark cases to verify and characterize methodologies for two- and three-dimensional unsteady optimization problems. Over the last several years, the AIAA Aerodynamic Design Optimization Discussion Group (ADODG) has sought to provide an increasingly complex set of benchmark test cases suitable for testing the capabilities of aerodynamic optimization methods. Currently, all of the test cases provided by the ADODG focus on optimization under steady flow conditions; hence we propose some benchmark cases as a basis for an expanded ADODG test suite that includes test cases for aerodynamic optimization under unsteady conditions. The inclusion of benchmark cases for aerodynamic optimization under unsteady conditions will assist researchers to verify and evaluate their algorithms.

III. Flow Solution Algorithm

The flow solver is largely based on work in [14, 16, 19]. The governing equations and a brief description of the solution strategy for the unsteady flow problem is presented below.

A. Governing Equations

1. Navier-Stokes Equations

The Navier-Stokes equations in generalized curvilinear coordinates (i.e. $(x, y, z) \to (\xi, \eta, \zeta)$) are given by

$$\partial_t \hat{\mathbf{Q}} + \partial_\xi \hat{\mathbf{E}} + \partial_\eta \hat{\mathbf{F}} + \partial_\zeta \hat{\mathbf{G}} = Re^{-1} \left(\partial_\xi \hat{\mathbf{E}}_\mathbf{v} + \partial_\eta \hat{\mathbf{F}}_\mathbf{v} + \partial_\zeta \hat{\mathbf{G}}_\mathbf{v} \right) \tag{1}$$

where

$$\hat{\mathbf{Q}} = J^{-1}\mathbf{Q}$$

$$\hat{\mathbf{E}} = J^{-1} \left(\xi_x \mathbf{E} + \xi_y \mathbf{F} + \xi_z \mathbf{G} \right), \quad \hat{\mathbf{F}} = J^{-1} \left(\eta_x \mathbf{E} + \eta_y \mathbf{F} + \eta_z \mathbf{G} \right), \quad \hat{\mathbf{G}} = J^{-1} \left(\zeta_x \mathbf{E} + \zeta_y \mathbf{F} + \zeta_z \mathbf{G} \right),$$

$$\hat{\mathbf{E}}_\mathbf{v} = J^{-1} \left(\xi_x \mathbf{E}_\mathbf{v} + \xi_y \mathbf{F}_\mathbf{v} + \xi_z \mathbf{G}_\mathbf{v} \right), \quad \hat{\mathbf{F}}_\mathbf{v} = J^{-1} \left(\eta_x \mathbf{E}_\mathbf{v} + \eta_y \mathbf{F}_\mathbf{v} + \eta_z \mathbf{G}_\mathbf{v} \right), \quad \hat{\mathbf{G}}_\mathbf{v} = J^{-1} \left(\zeta_x \mathbf{E}_\mathbf{v} + \zeta_y \mathbf{F}_\mathbf{v} + \zeta_z \mathbf{G}_\mathbf{v} \right)$$

The variable $\mathbf{Q}$ is a vector containing the conserved variables, and J is the metric Jacobian resulting from the coordinate transformation from Cartesian to curvilinear coordinates. The variables $\mathbf{E}$, $\mathbf{F}$ and $\mathbf{G}$ are vectors containing the inviscid fluxes, while $\mathbf{E}_\mathbf{v}$, $\mathbf{F}_\mathbf{v}$ and $\mathbf{G}_\mathbf{v}$ are vectors containing the viscous fluxes. The term ∂_x is a shorthand for $\frac{\partial}{\partial x}$, and ξ_x is a shorthand for $\frac{\partial \xi}{\partial x}$, and so on. Further details on the full constituents of the governing equations and coordinate transformation can be found in [20].

2. Turbulence Model

We use the "negative" variant of Spalart-Allmaras (SA) one-equation turbulence model for URANS problems. Details of the implementation can be found in [16].

B. Discretization and Solution of Governing Equations

1. Spatial Discretization

The Navier-Stokes equations are discretized on structured multi-block grids using second-order summation-by-parts (SBP) operators, and at block interfaces and boundaries, simultaneous approximation terms (SATs) are used to couple blocks and apply boundary conditions in an accurate and stable manner [21]. Further details on this discretization strategy can be found in [14, 16]

2. Time Marching

The semi-discrete form of the governing equations is given as

$$\frac{d\hat{\mathbf{Q}}}{dt} + \hat{\mathcal{R}}(\hat{\mathbf{Q}}) = \mathbf{0} \tag{2}$$

where $\hat{\mathcal{R}}$ results from discretization of the spatial terms. We recast the semi-discrete equation (2) into a fully-discrete equation of the form

$$\tilde{\mathcal{R}}(\hat{\mathbf{Q}}) + \hat{\mathcal{R}}(\hat{\mathbf{Q}}) = \mathbf{0} \tag{3}$$

where $\tilde{\mathcal{R}}$ represents the contribution to the discrete residual associated with the time-marching method.

For time integration, the second-order backward formula (BDF2) and the explicit singly-diagonal implicit Runge-Kutta (ESDIRK) schemes are considered. At each time step n, the flow residual $\mathcal{R}_n$ for a linear multi-step method such as BDF2 can be expressed as

$$\mathcal{R}_n = \tilde{\mathcal{R}}_n + \hat{\mathcal{R}}_n \quad \text{for } n = 1, \ldots, N \tag{4}$$

where N is the number of time steps and

$$\tilde{\mathcal{R}}_n = \frac{3\hat{\mathbf{Q}}_n - 4\hat{\mathbf{Q}}_{n-1} + \hat{\mathbf{Q}}_{n-2}}{2\Delta t}$$

If we consider a multi-stage time marching method like the ESDIRK4, the discrete residual is given as

$$\mathcal{R}_j^{(n)} = \frac{1}{a_{jj}}\tilde{\mathcal{R}}_j^{(n)} + \hat{\mathcal{R}}_j^{(n)} + \sum_{k=1}^{j-1} \frac{a_{jk}}{a_{jj}}\hat{\mathcal{R}}_k^{(n)} \quad \text{for } n = 1,\dots,N \quad j = 2,\dots,6 \tag{5}$$

where

$$\tilde{\mathcal{R}}_j^{(n)} = \frac{\hat{\mathbf{Q}}_j^{(n)} - \hat{\mathbf{Q}}^{(n-1)}}{\Delta t}$$

and a_{jj} and a_{jk} are the coefficients associated with the ESDIRK scheme. The explicit first stage is specified as $\hat{\mathbf{Q}}_1^{(n)} = \hat{\mathbf{Q}}^{(n-1)}$, and the solution at time step n is given as $\hat{\mathbf{Q}}^{(n)} = \hat{\mathbf{Q}}_6^{(n)}$.

3. Solution of the Discrete Equations

At each time step n or stage j, a system of nonlinear equations $\mathcal{R}_n(\hat{\mathbf{Q}}_1,\dots,\hat{\mathbf{Q}}_{n-1},\hat{\mathbf{Q}}_n) = \mathbf{0}$ is solved using an inexact-Newton method. The resulting linear problem at each Newton iteration is solved using a Krylov subspace method [22].

IV. Optimization Problem and Gradient Computation

A. Optimization Problem

We introduce the variable $\mathbf{V}$ as the vector containing all of the mesh-deformation variables $\mathbf{B}_j$ at each mesh-deformation increment and all of the conserved flow variables $\mathbf{Q}_n$ at each time step. In addition, we introduce $\mathcal{F}$ as the vector containing all the mesh-deformation residuals $\mathcal{M}_j$ at each mesh-deformation increment, and all the flow state residuals $\mathcal{R}_n$ at each time step. Lumping the state variables and residuals together into $\mathbf{V}$ and $\mathcal{F}$ allows us to formulate the discrete adjoint method in a compact form. The mesh-deformation variables, $\mathbf{B}$, are the control points of the B-spline volumes used to parameterize the CFD mesh. A linear elasticity algorithm is used to deform the parameterized mesh after geometry modifications [15]. Since the mesh deformation is performed in m increments, the state residual at each increment is represented by $\mathcal{M}_j$, where j is the index of the mesh-deformation increment. Performing the mesh deformation in increments improves robustness by allowing for large geometric changes to be propagated through the mesh without a breakdown in the linear elasticity assumption that drives the algorithm.

The constrained optimization problem is defined as

$$\min_{\mathbf{X}} \quad \mathcal{J}(\mathbf{X},\mathbf{V}) \quad \text{subject to} \quad \begin{cases} \mathcal{F}(\mathbf{X},\mathbf{V}) = \mathbf{0} & \\ c_i(\mathbf{X},\mathbf{V}) = 0 & i \in \mathcal{E} \\ c_i(\mathbf{X},\mathbf{V}) \leq 0 & i \in \mathcal{I} \end{cases} \tag{6}$$

where $\mathbf{X}$ represents the design variables. The functions c_i, $i \in \mathcal{E}$ are *equality constraints*, while c_i, $i \in \mathcal{I}$ are *inequality constraints* (these constraints are smooth real-valued linear/nonlinear functions). For unsteady flows, the objective function, $\mathcal{J}$, can be a time-averaged functional such as lift, drag or lift-to-drag ratio, or the functional at the final time step N. We use the SNOPT optimizer [23] in our framework. To compute the gradients required by the optimizer, we use the discrete adjoint method.

Since $\mathcal{F}$ is zero for all $\mathbf{X}$, the optimization problem in equation (6) is reformulated as a Lagrangian problem of the form

$$\min_{\mathbf{X}} \quad \mathcal{L}(\mathbf{X},\mathbf{V},\mathbf{\Lambda}) = \mathcal{J}(\mathbf{X},\mathbf{V}) + \mathbf{\Lambda}^T\mathcal{F}(\mathbf{X},\mathbf{V}) \tag{7}$$

$$\text{subject to} \quad \begin{cases} c_i(\mathbf{X},\mathbf{V}) = 0 & i \in \mathcal{E} \\ c_i(\mathbf{X},\mathbf{V}) \leq 0 & i \in \mathcal{I} \end{cases}$$

where $\mathbf{\Lambda}$ is the adjoint vector containing the flow adjoints $\boldsymbol{\psi}_n$ for each time step and mesh adjoints $\boldsymbol{\lambda}_j$ for each mesh-deformation increment. A reduced-space approach is used to solve the optimization problem above. This approach assumes that the state variables $\mathbf{B}$ and $\mathbf{Q}$ are implicit functions of the control variable $\mathbf{X}$ and as such, for any valid $\mathbf{X}$, there is a unique solution to $\mathbf{B}(\mathbf{X})$ and $\mathbf{Q}(\mathbf{X},\mathbf{B})$. A consequence of this approach is that, once $\mathcal{F} = \mathbf{0}$ is satisfied, the optimizer only has to update the design variables, $\mathbf{X}$, at each iteration.

For the time-marching methods considered, the Lagrangian problem is given in discrete form as

$$\mathcal{L} = \sum_{n=1}^{N} \left[\omega_n \mathcal{J}_n + \boldsymbol{\psi}_n^T \mathcal{R}_n \right] + \sum_{j=1}^{m} \boldsymbol{\lambda}_j^T \mathcal{M}_j \tag{8}$$

$$\text{for } n = 1, \ldots, N \quad \text{and for } j = 1, \ldots, m$$

for the BDF2 time marching scheme, and

$$\mathcal{L} = \sum_{n=1}^{N} \left[\omega_n \mathcal{J}_n + \sum_{k=2}^{6} \left(\boldsymbol{\psi}_k^{(n)} \right)^T \mathcal{R}_k^{(n)} \right] + \sum_{j=1}^{m} \boldsymbol{\lambda}_j \mathcal{M}_j \tag{9}$$

$$\text{for } n = 1, \ldots, N \quad \text{and for } j = 1, \ldots, m$$

for the ESDIRK4 method. Here our objective function $\mathcal{J}$ has been cast in its discrete form as $\sum_{n=1}^{N} \omega_n \mathcal{J}_n$, where the ω_n are typically quadrature weights and $\mathcal{J}_n$ are the objective function contributions at each time step n.

B. Gradient Computation

The gradient, $\mathcal{G}$, of the Lagrangian in equation (7) with respect to the design variables is given as

$$\mathcal{G} = \left(\frac{\partial \mathcal{J}}{\partial \mathbf{X}} + \boldsymbol{\Lambda}^T \frac{\partial \mathcal{F}}{\partial \mathbf{X}} \right) + \left(\frac{\partial \mathcal{J}}{\partial \mathbf{V}} + \boldsymbol{\Lambda}^T \frac{\partial \mathcal{F}}{\partial \mathbf{V}} \right) \frac{d\mathbf{V}}{d\mathbf{X}} \tag{10}$$

To derive a formulation consistent with obtaining the first-order (necessary) optimality conditions for the optimization problem, the adjoint variables are chosen such that

$$\left[\frac{\partial \mathcal{F}}{\partial \mathbf{V}} \right]^T \boldsymbol{\Lambda} + \left[\frac{\partial \mathcal{J}}{\partial \mathbf{V}} \right]^T = \mathbf{0} \tag{11}$$

and in the process we avoid computing the sensitivities of the state variables with respect to the design variables (i.e. $d\mathbf{V}/d\mathbf{X}$). Equation (11) is referred to as the adjoint problem. Our compact notation shows a coupled flow adjoint and mesh adjoint problem but the two systems are decoupled in practice. The full constituents of the adjoint problem together with three examples are presented in the appendix. Once the governing equations and adjoint equations are satisfied, the gradient of the optimization problem can be computed from equation (10) using

$$\mathcal{G} = \frac{\partial \mathcal{J}}{\partial \mathbf{X}} + \boldsymbol{\Lambda}^T \frac{\partial \mathcal{F}}{\partial \mathbf{X}} \tag{12}$$

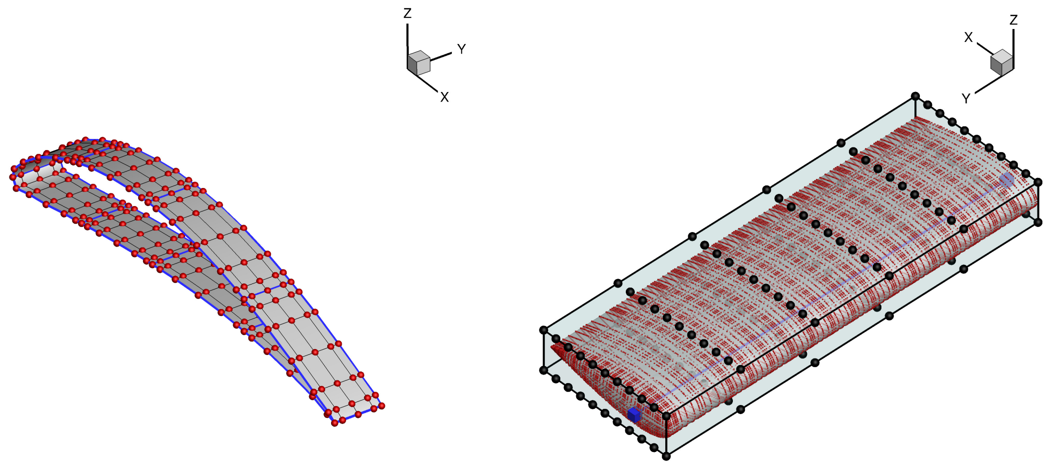
We use a modified and flexible variant of the generalized conjugate residual with inner orthogonalization and outer truncation method GCROT(m, k) [24] to solve the flow adjoint equation at each time step. For the mesh adjoint equations, a preconditioned conjugate gradient (PCG) algorithm is used to solve the linear system at each mesh-deformation increment. While the forward problem proceeds from the first mesh-deformation increment to the last flow time step, the presence of the transpose in the adjoint problem requires a backward solution of a sequence of linear problems that proceed from the last flow time step to the first mesh-deformation increment.

The gradient formulations for the time-marching methods considered are presented here. The governing equations contained in $\mathcal{F}$ are broken up into the flow $\mathcal{R}$ and mesh $\mathcal{M}$ related residuals in the formulations. For BDF2, the gradient can be computed using the following expression:

$$\mathcal{G} = \sum_{n=1}^{N} \left[\omega_n \frac{\partial \mathcal{J}_n}{\partial \mathbf{X}} + \boldsymbol{\psi}_n^T \frac{\partial \mathcal{R}_n}{\partial \mathbf{X}} \right] + \sum_{j=1}^{m} \boldsymbol{\lambda}_j \frac{\partial \mathcal{M}_j}{\partial \mathbf{X}} \tag{13}$$

In the case of ESDIRK4, the expression is:

$$\mathcal{G} = \sum_{n=1}^{N} \left[\omega_n \frac{\partial \mathcal{J}_n}{\partial \mathbf{X}} + \sum_{k=2}^{6} \left(\boldsymbol{\psi}_k^{(n)} \right)^T \frac{\partial \mathcal{R}_k^{(n)}}{\partial \mathbf{X}} \right] + \sum_{j=1}^{m} \boldsymbol{\lambda}_j \frac{\partial \mathcal{M}_j}{\partial \mathbf{X}} \tag{14}$$



(a) B-spline parameterization

(b) Free-Form and Axial deformation

Fig. 1 Geometry control systems

C. Geometry Control

We use B-splines to parameterize the geometries for the simple two-dimensional optimization problems. The methodology is based on the work in [15]. For our airfoil optimization problems, the geometry is parameterized using B-splines and the resulting control points are used to modify the geometry during the optimization process. We show a B-spline parameterization of an extruded airfoil in Figure 1 (a). The red spheres represent the B-spline control points while the blue lines show the patch interfaces. Generally, a subset of the control points along the airfoil are used as the design variables. For the three-dimensional design problems, we use the Free-Form (FFD) and axial deformation framework developed in [18] to control the geometries while retaining the B-spline parameterization. FFD combined with axial deformation provides a more intuitive (i.e. we can easily define constraints for features such as taper, sweep, dihedral etc.) and robust approach for modifying the geometry during optimization [25]. For this methodology the underlying surface parameterization (i.e. B-spline surfaces in our case) is embedded inside an FFD volume. The FFD control points are used to modify the geometry, and the analytical representation of the geometry is retained. We show this approach in Figure 1 (b). The black lines and spheres represent the FFD volume edges and control points respectively. The thick blue line running through the quarter-chord of the geometry and the cube at the ends show the axial deformation parameters. Furthermore, the tiny red spheres on the geometry represent the underlying B-spline parameterization of the geometry.

V. Test Cases

In this section we present two benchmark cases that can be used to test and evaluate a design framework for optimizing aerodynamic shapes under unsteady flow conditions. The first set of cases investigates airfoil design optimization for laminar flows in two dimensions. We also consider a three dimensional case involving a wing design optimization for laminar flows. We use the BDF2 time-marching method for all of the studies presented here.

A. Airfoil Optimization - Laminar Flow

1. Problem Definition

We consider a laminar flow at a Mach number of 0.2 and a Reynolds number of 800. The design objective is to minimize the *time-averaged drag coefficient* at a fixed *time-averaged lift coefficient* for the time window $t = [46, 76]$. The time-averaged lift coefficient constraint is set to 0.75. At the required lift target the flow over the baseline airfoil separates, which leads to vortex shedding in the wake region. The separation of the flow causes a large drag on the airfoil, while the periodic nature of the trailing vortices causes oscillations in the aerodynamic quantities, thereby

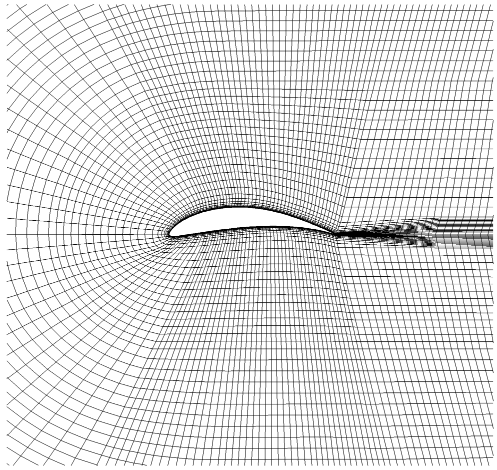


Fig. 2 Airfoil optimization grid

necessitating the use of an unsteady flow algorithm to solve the optimization problem.

We use the Chuch Hollinger 10 smoothed (CH10SM) airfoil as the baseline geometry for this study. We investigate three different geometric constraints; the first set of constraints (case 1) allow the thickness to not decrease by more than 15% of the initial value. In the second case (case 2), the thickness is allowed to decrease by no more than 5% of the initial value, while the final case (case 3) enforces a minimum area constraint, with the minimum area set to that of the baseline airfoil.

Figure 2 shows the grid used for the CH10SM airfoil flow problem. The computational mesh is divided into 120 blocks with 15×15 grid points in each two-dimensional block. From flow analysis on finer grids we observed that the largest error in the mean aerodynamic quantities between the grid used and the finer grids was about 2%. Hence this grid size provides a good balance between accuracy and turnaround time. The mesh blocks are parameterized using 7×5 B-spline surfaces, and there are 34 "effective" geometric design variables in addition to the angle of attack. Tables 1 and 2 show the mesh-deformation and flow parameters used for the cases in this section.

We note that while the cases we present here are two-dimensional in nature, in practice we set up these cases to test the three-dimensional capability of our flow and adjoint problems. To do this we extrude the airfoil in the spanwise direction by $0.1c$ using 11 mesh nodes. We then enforce a symmetry boundary condition on either side of the span to ensure the flow remains two-dimensional. A plot of the Q-criterion along spanwise stations shows that no three-dimensional structures are formed along the span and the flow remains two-dimensional (see Figure 3). For the optimization problem a linear constraint is enforced at spanwise stations to ensure the sections remain the same; hence our use of the term "effective" design variables in the preceding paragraph. The motivation was to study a problem that captured the essence of a three-dimensional problem in a *computational sense* but with a shorter turnaround time compared to a wing optimization problem. It is also important to note that solving this benchmark problem as two-dimensional should suffice for the purposes of algorithm characterization.

The control window for the optimization problem is constrained to the time domain $t = [46, 76]$, which is the last $3,000$ time steps of the flow problem. The time domain $t = [0, 46]$ is considered to be the flow adjusting phase and as such flow solutions in that domain do not contribute to the objective function. In practice, ω_n is set to zero for the time steps in the flow adjusting phase, and then assigned a non-zero value based on the trapezoidal rule to compute the objective value in the control window. Figure 4 shows the time history of the aerodynamic quantities with the results limited to the control window for the optimization problem. The horizontal lines in the plots show the mean values for the aerodynamic quantities.

2. Time-Stepping and Checkpointing

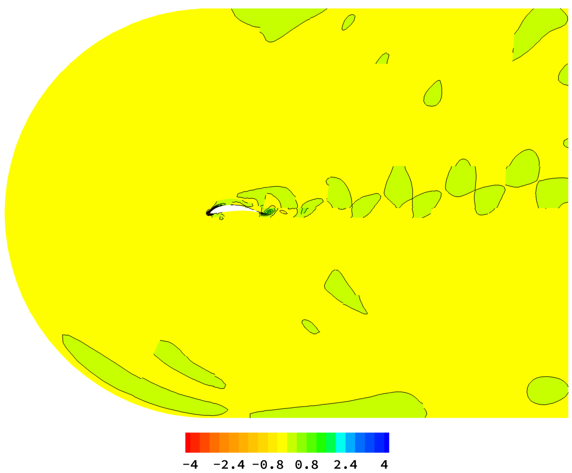
To traverse the transient phase of the unsteady flow problem quickly, we prescribe different time steps for different windows for the flow problem. We use a time step of 1.4 times the reference value defined in Table 2 for the first $1,000$

Parameter	CH10SM
mesh size	27,000
blocks	120
block size	15×15
control points	7×5
mesh-move tolerance	10^{-12}
adjoint tolerance	10^{-12}

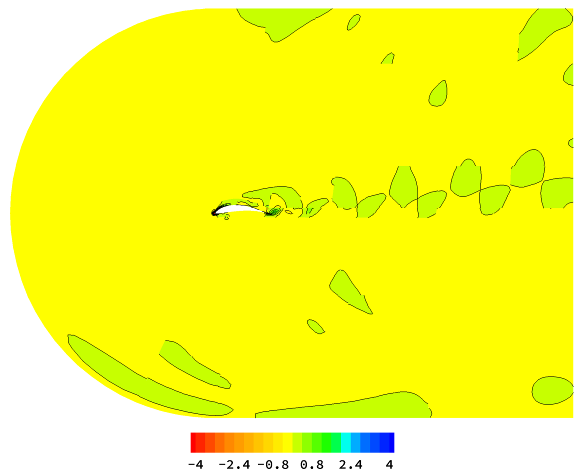
Table 1 Mesh-deformation parameters

Parameter	CH10SM
Reynolds number	800
initial angle of attack	8°
reference time step - Δt_{ref}	0.05
flow solve relative tolerance	10^{-9}
adjoint tolerance	10^{-12}
time steps	7,000

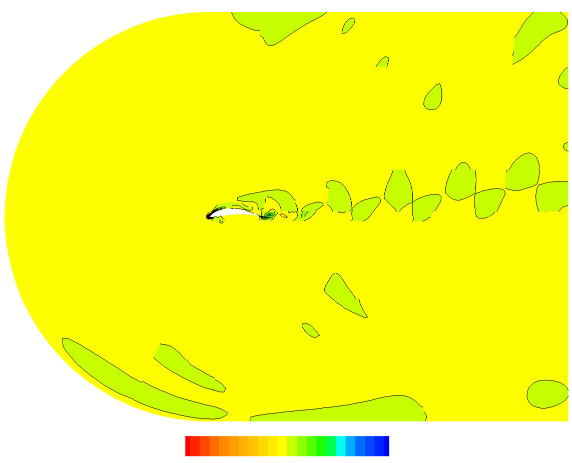
Table 2 Flow solution parameters



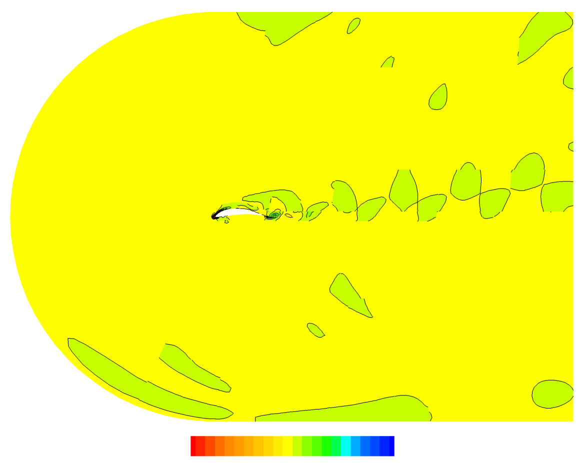
(a) Station $y = 0$.



(b) Station $y = 0.02c$

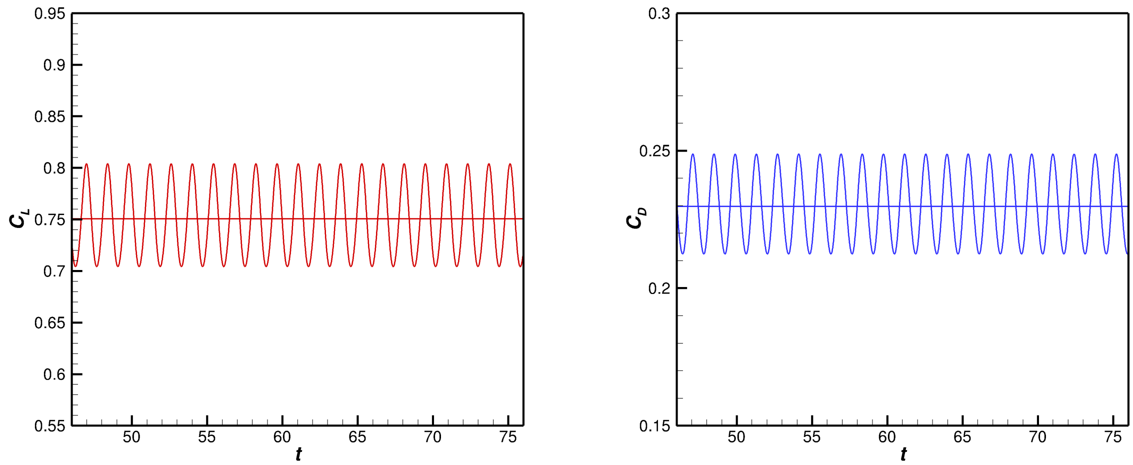


(c) Station $y = 0.06c$



(d) Station $y = 0.1c$

Fig. 3 Q-criterion at the final time step - baseline geometry



(a) Lift coefficient

(b) Drag coefficient

Fig. 4 Aerodynamic coefficients in the control window

steps. For the next $1{,}000$ steps we use 1.2 times the reference value, and the reference value for the last $5{,}000$ steps. All time step values are non-dimensionalized based on the speed of sound and the chord length.

A static checkpointing strategy is used to allow for long running jobs to restart from where they left off. Flow solutions are written to disk and reloaded back to compute the adjoint solutions backward in time. A set of checkpointing files are written to disk to allow the problem to proceed from the last known state prior to reaching the specified wall time (i.e. 24 hours). On the system used, disk storage is available, and the cost of I/O at each time step is significantly cheaper than recomputing the solutions between checkpoints for the adjoint problem, as is done in dynamic checkpointing [26].

3. Gradient Test

Directional derivatives are used to compare the gradients computed using the discrete adjoint approach and the gradients approximated with finite differences. The use of directional directives allows for the finite-difference approximation to be computed in one sweep for all design variables instead of verifying for each design variable separately, which is expensive. For a given direction $\vec{v}$, the directional derivative from the adjoint method is given as

$$D_{\vec{v}}\mathcal{J} = \frac{\partial \mathcal{J}}{\partial \mathbf{X}}\vec{v} \tag{15}$$

while the second-order finite-difference approximation is given as

$$D_{\vec{v}}\mathcal{J} = \frac{\mathcal{J}(\mathbf{X}+\epsilon\vec{v}) - \mathcal{J}(\mathbf{X}-\epsilon\vec{v})}{2\epsilon} + O(\epsilon^2)$$

where ϵ is the step size. The vector $\vec{v}$ is chosen as

$$v_i = \operatorname{sign}\left[\left(\frac{\partial \mathcal{J}}{\partial \mathbf{X}}\right)_i\right]$$

which gives a directional derivative equal to the L^1 norm of the gradient. Figure 5 shows the relative difference between the gradients computed using the discrete adjoint approach and the second-order finite-difference for the CH10SM airfoil using BDF2 time marching method. The gradient test shows that our discrete adjoint implementation computes accurate gradients for the cases presented. All additional gradient checks were performed using SNOPT's "cheap" gradient test. This test is similar to our directional directive test except SNOPT compares the adjoint and finite difference gradients using a single step size instead of sweeping through several step sizes.

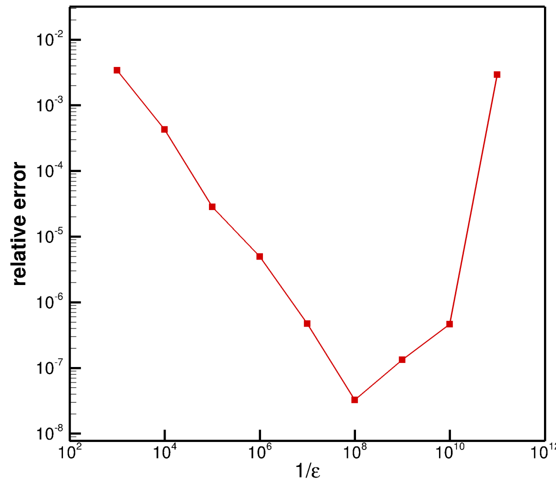


Fig. 5 Directional derivative gradient test

B. Wing Design - Laminar Flow

Here the objective is again to minimize the *time-averaged drag coefficient* at a fixed *time-averaged lift coefficient* with respect to a fixed time window. We use a grid with about 7.1 million nodes split into $1,152$ blocks of equal size. The baseline geometry is a NACA0012 wing with an aspect ratio of 8. The grid used for this problem is shown in Figure 6. We embed the baseline geometry in a FFD volume resulting in 138 geometric design variables in addition to angle of attack for this case. The optimizer was allowed to vary the vertical coordinates of the section, while the thickness at the sections is constrained to not decrease by more than 25% of the initial value. In addition, the optimizer is allowed to vary the twist of the sections as well. The sections with section shape and twist freedom along the semi-span are located at $y = [0, 0.8, 1.6, 2.4, 3.2, 4]$. Twist is applied about the axial curve, which is located at the quarter-chord.

We simulate the flow problem at a Mach number of 0.2, Reynolds number of 800 and an initial angle of attack of 12°. The flow separates and vortices shed in the wake region periodically at these conditions. The periodic vortex shedding influences the aerodynamic quantities thereby necessitating the use of an unsteady flow simulation to optimize the geometry. The *time-averaged lift coefficient* constraint is set to 0.4125. The time domain for the control window is $t = [16, 22]$ which covers 4 periods of the oscillating aerodynamic coefficients. Figure 7 shows the time history of the aerodynamic coefficients. We use a time step of 0.02 for the time domain $t = [0, 16]$ and time step of 0.01 for the control window.

VI. Results

A. Airfoil Design - Laminar Flow

1. CH10SM Design Problem

We present the optimization results in Figure 8. In the plots, optimality is defined as a measure of the gradient of the SNOPT Lagrangian merit function, where the merit function is a composite function of the objective function and any nonlinear constraint violations. Feasibility is defined as a measure of the nonlinear constraint violations, which is the lift constraint for the cases considered. All the cases converged in under 60 function evaluations, with optimality dropping by about 6 orders of magnitude. Feasibility also dropped to about 10^{-11} for all the cases and the optimized geometries had similar profiles with slight variations in the area as a result of the different constraints used.

The flow on the upper surface of the baseline airfoil separates downstream of about $0.5c$ (see Figure 9) when the lift constraint is satisfied. In addition to the large separation region, we observe vortex shedding in the wake region as shown in Figure 10. The optimizer primarily reduces the airfoil thickness and camber to delay flow separation downstream of $0.6c$ (see Figure 9) on the upper surface. We observe a flap-like feature at the trailing edge which helps

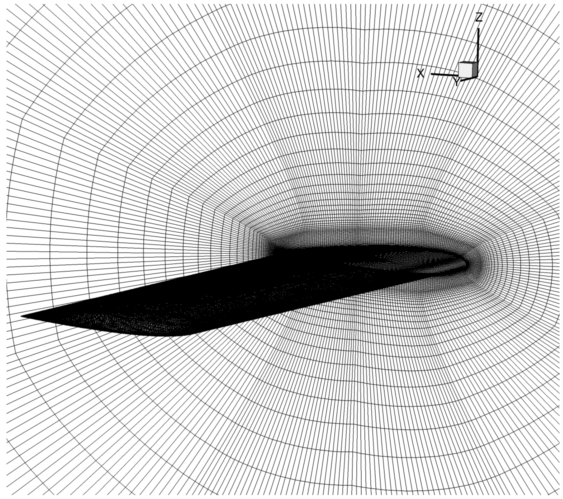
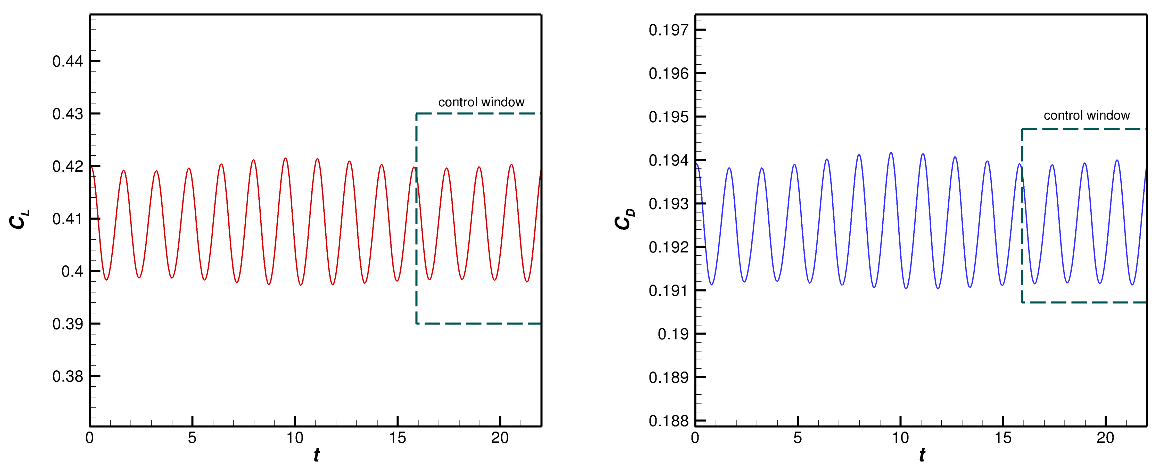


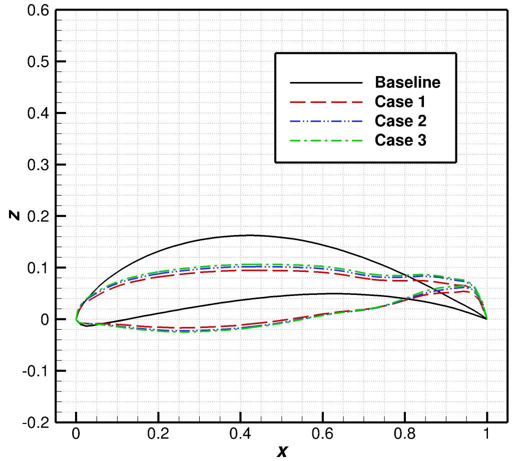
Fig. 6 Wing optimization grid



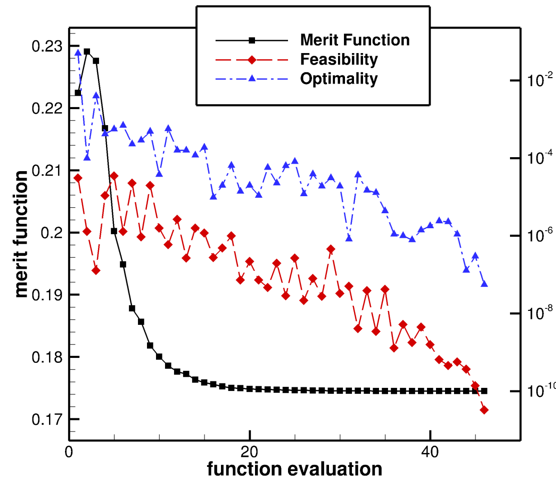
(a) Lift coefficient

(b) Drag coefficient

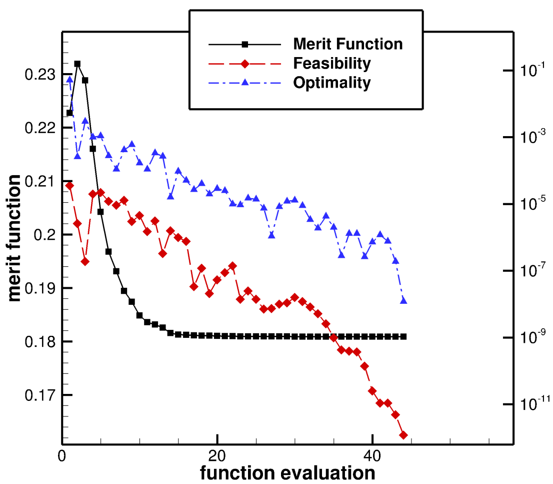
Fig. 7 NACA0012 wing - aerodynamic coefficients



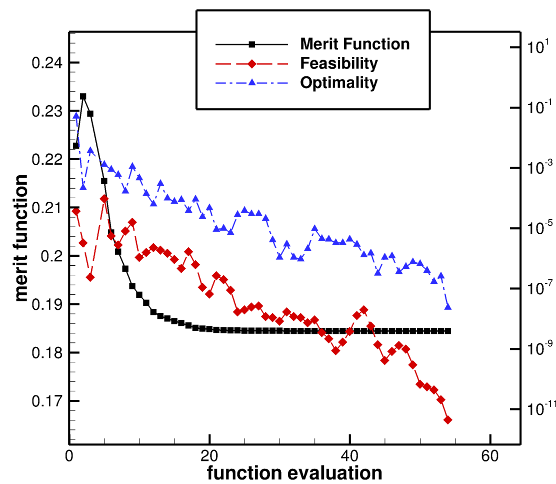
(a) Optimized geometries



(b) Optimization history - case 1



(c) Optimization history - case 2



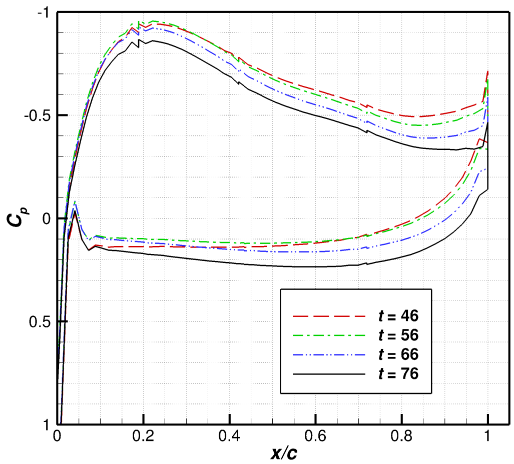
(d) Optimization history - case 3

Fig. 8 CH10SM optimization results

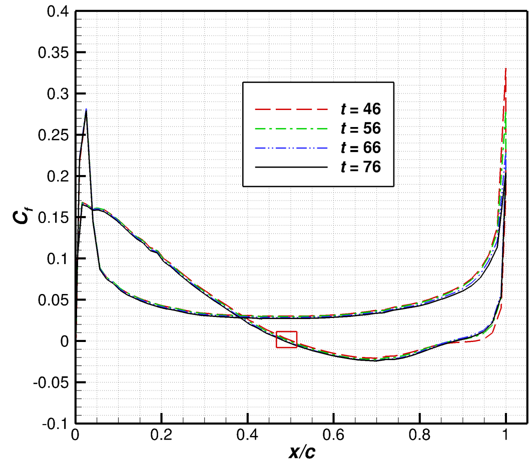
achieve the lift-target. The optimized geometry produces a nearly steady flow for the case where we allow the section thickness to vary between $\pm 15\%$ of the initial value (see Figures 10 and 11). For case 2 (thickness bounded between -5% and $+10\%$ on the initial value) and case 3 (minimum area constraint set to the initial value), the amplitude of the shed vortices is reduced significantly, as shown in Figure 11. For the first two cases, the constraints remain active at the lower bound while the minimum area constraint for the third case remains active as well. In Table 3 we show that the optimized geometries have significantly lower drag compared to the baseline airfoil at the design point.

2. Effect of Initial Angle of Attack

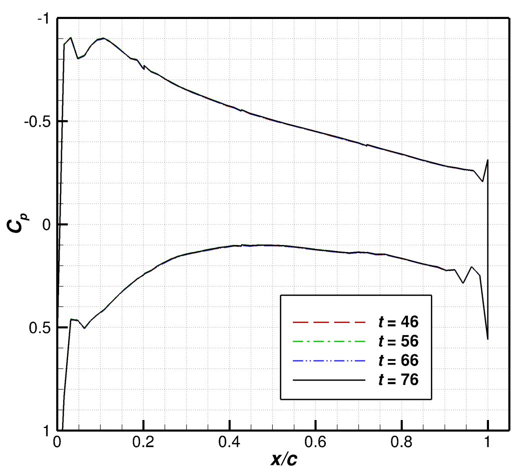
Here we looked at the effect of initial angle of attack on the optimized geometry. The initial angle of attack was set to $\alpha = 3°$ and the optimization was repeated for cases 1 and 2. For both cases, the optimized geometries realized are very similar to the optimized geometries obtained starting at $\alpha = 8°$. We show the optimized geometries in Figure 12.



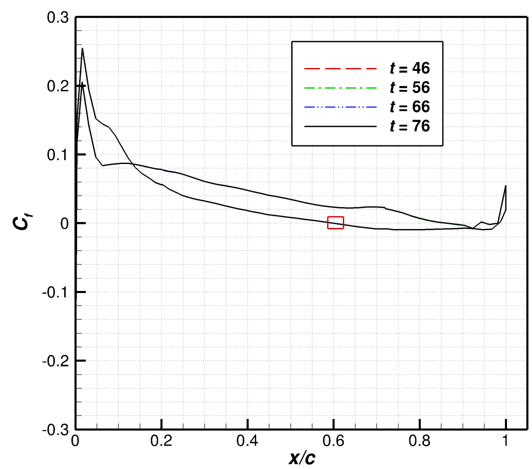
(a) Pressure coefficient - baseline geometry



(b) Friction coefficient - baseline geometry



(c) Pressure coefficient - optimized geometry (case 1)

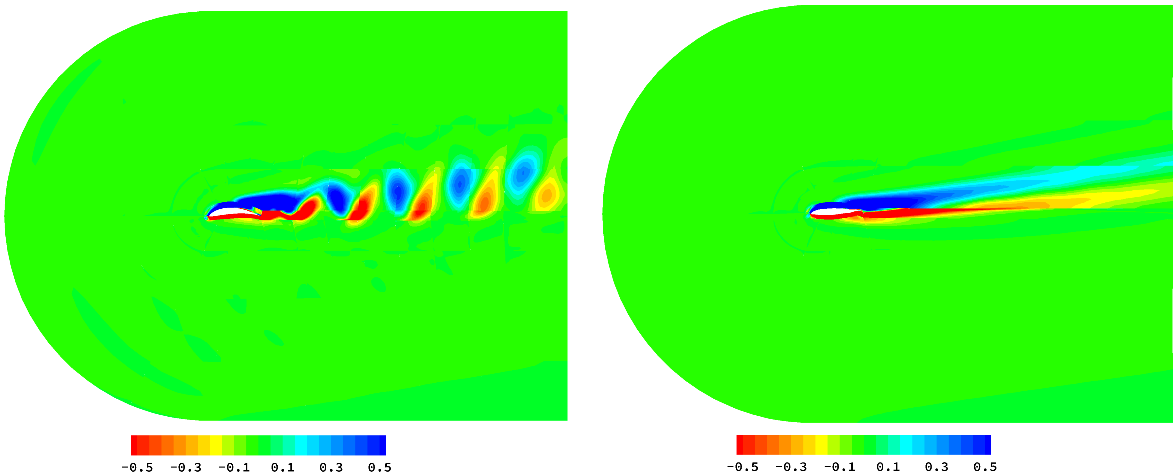


(d) Friction coefficient - optimized geometry (case 1)

Fig. 9 Pressure and friction coefficients

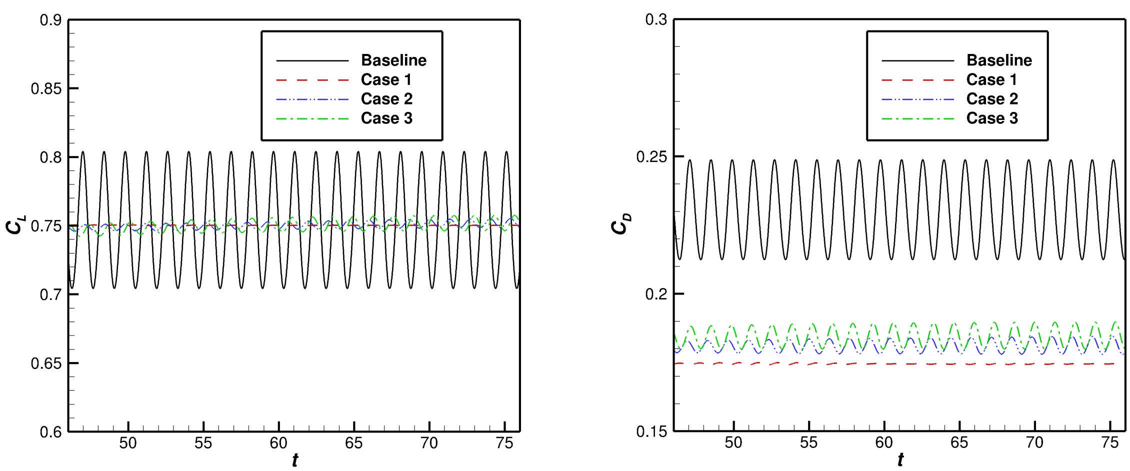
Case	α	C_L	C_D	Drag Reduction
Initial	8.60°	0.7500	0.2288	-
Case 1	7.37°	0.7500	0.1745	23.73%
Case 2	7.65°	0.7500	0.1809	20.94%
Case 3	7.62°	0.7500	0.1845	19.36%

Table 3 Summary of results



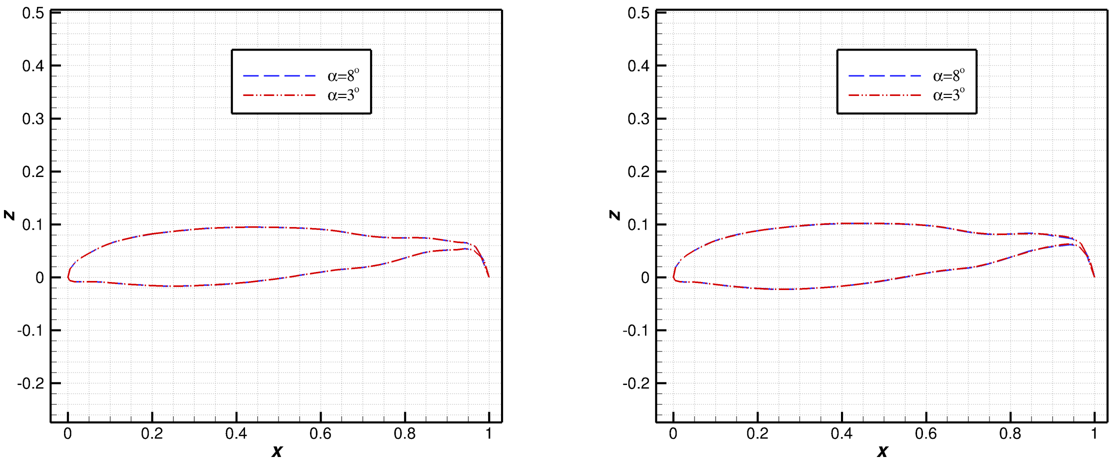
(a) Baseline geometry (b) Optimized geometry

Fig. 10 Y-vorticity contours at the final time step



(a) Lift coefficient (b) Drag coefficient

Fig. 11 Aerodynamic coefficient histories for optimized geometries



(a) Optimized geometries - case 1

(b) Optimized geometries - case 2

Fig. 12 Initial angle of attack comparison

B. Wing Design - Laminar Flow

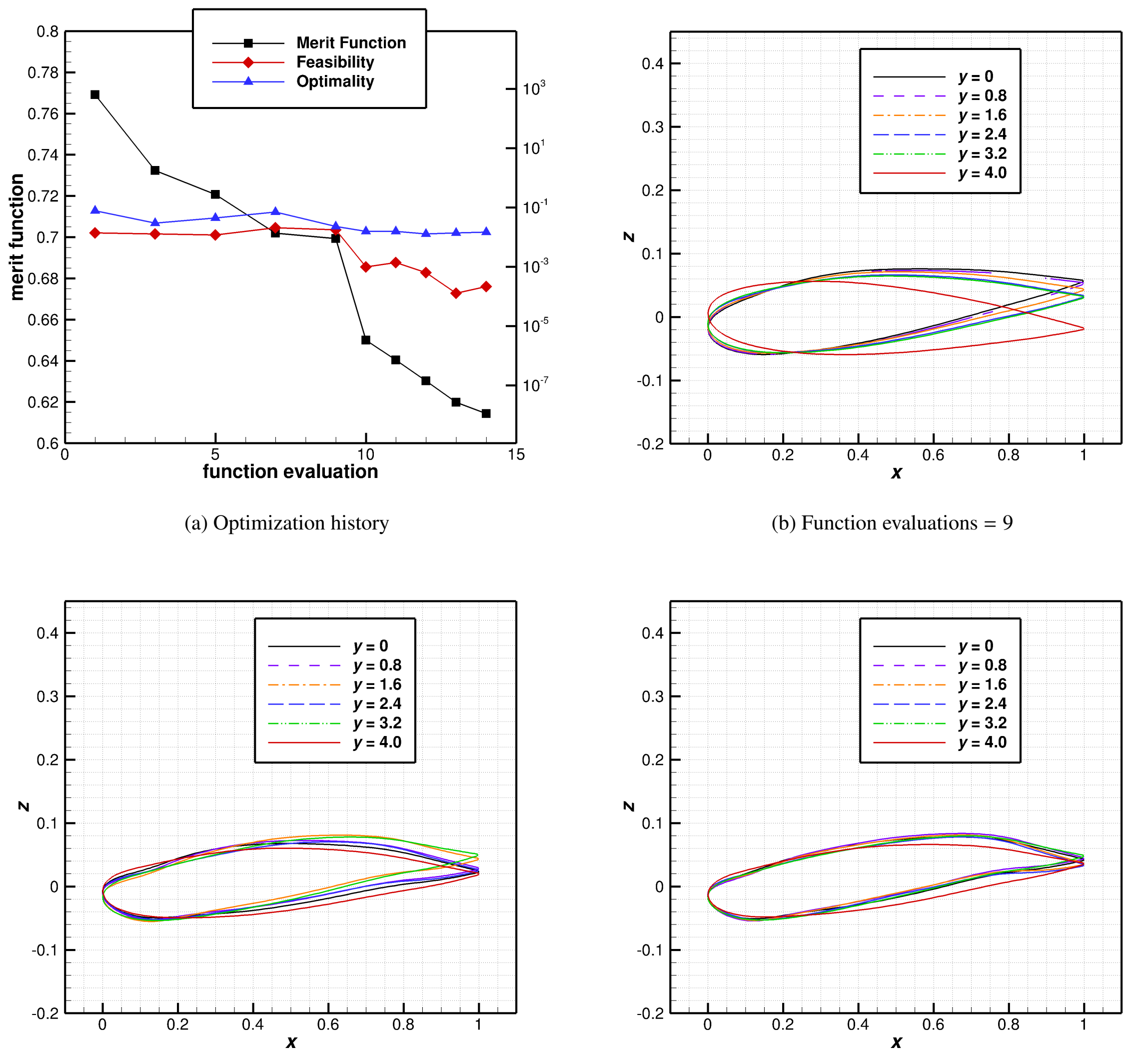
We present some preliminary results for this optimization problem. This case is quite computationally intensive, and only nine optimization iterations were completed at the time of writing. Figure 13 shows the convergence history, including optimality, feasibility, and the merit function, as well as the associated geometric changes. Initial iterations show primarily changes in twist, with later iterations showing section shape changes, with a flap-like feature beginning to emerge as in the two-dimensional case. As of the ninth optimization iteration, the time-averaged lift coefficient is 0.4118 compared to the 0.4125 target set. The drag has been reduced from 0.1925 to 0.1535, which represents about 20% drag reduction compared to the baseline geometry. The amplitude of the oscillations in the aerodynamic coefficients has also been reduced significantly after just seven iterations and by the ninth iteration we see a nearly steady flow similar to the two-dimensional case (see Figure 14).

VII. Conclusions

The use of aerodynamic optimization tools continues to be an integral part of the design process in the aerospace sector. As computing power increases, more complex problems requiring unsteady flow solutions will become tractable in this context. Hence, it is important to develop efficient algorithms and benchmark cases that allow researchers to evaluate their algorithms in order to reproduce consistent results. As part of this effort, the ADODG provides a solid foundation of benchmark cases for testing aerodynamic optimization methods that can be expanded to include unsteady flow problems. To this end, we have presented an efficient framework for optimizing aerodynamic shapes under unsteady flow conditions. We have verified the objective function gradients computed using the discrete adjoint method against the second order finite difference method. Additionally, we have proposed two benchmark cases to be considered for the ADODG test suite for aerodynamic optimization. We hope that the algorithm presented and benchmark cases proposed will help others develop, test, and evaluate their aerodynamic optimization algorithms for unsteady flow conditions.

Appendix

We present the full constituents of the coupled adjoint problem from equation (11). First we generalize the flow and mesh-deformation adjoint problems and then we show some examples using a fixed number of time steps and a fixed number of mesh-deformation increments.



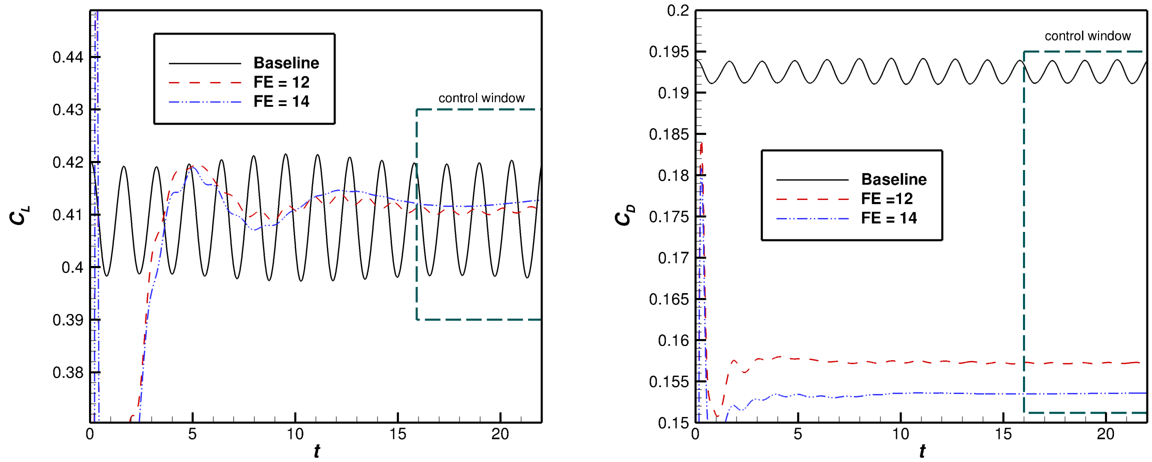
(a) Optimization history

(b) Function evaluations = 9

(c) Function evaluations = 12

(d) Function evaluations = 14

Fig. 13 Wing optimization results



(a) Lift coefficient

(b) Drag coefficient

Fig. 14 Aerodynamic coefficient histories at selected function evaluations (FE)

A. Adjoint Equations - BDF2

The generalized discrete adjoint problem for BDF2 method is given as

$$\left[\frac{\partial \mathcal{R}_n}{\partial \mathbf{Q}_n}\right]^T \boldsymbol{\psi}_n = -\left[\omega_n \frac{\partial \mathcal{J}_n}{\partial \mathbf{Q}_n}\right]^T \quad \text{for } n = N \tag{16}$$

$$\left[\frac{\partial \mathcal{R}_n}{\partial \mathbf{Q}_n}\right]^T \boldsymbol{\psi}_n = -\left[\omega_n \frac{\partial \mathcal{J}_n}{\partial \mathbf{Q}_n} + \sum_{i=1}^{p} \boldsymbol{\psi}_{n+i}^T \frac{\partial \mathcal{R}_{n+i}}{\partial \mathbf{Q}_n}\right]^T \quad \text{for } n = N-1,\ldots,1 \;,\; p = \min\left[N-n, r\right] \tag{17}$$

$$\left[\frac{\partial \mathcal{M}_j}{\partial \mathbf{B}_j}\right]^T \boldsymbol{\lambda}_j = -\underbrace{\left[\sum_{n=1}^{N} \left(\omega_n \frac{\partial \mathcal{J}_n}{\partial \mathbf{B}_j} + \boldsymbol{\psi}_n^T \frac{\partial \mathcal{R}_n}{\partial \mathbf{B}_j}\right)\right]^T}_{\text{last mesh-deformation increment RHS}} \quad \text{for } j = m \tag{18}$$

$$\left[\frac{\partial \mathcal{M}_j}{\partial \mathbf{B}_j}\right]^T \boldsymbol{\lambda}_j = -\left[\boldsymbol{\lambda}_{j+1}^T \frac{\partial \mathcal{M}_{j+1}}{\partial \mathbf{B}_j}\right]^T \quad \text{for } j = m-1,\ldots,1 \tag{19}$$

where r is the number of time levels of previous solutions required for the time marching method. For BDF2, r is equal to 2 (r can be varied to generalize the above expressions for any linear multi-step time marching method). Equations (16) and (17) constitute the flow adjoint, while equations (18) and (19) constitute the mesh adjoint equations.

B. Adjoint Equations - EDIRK4

The generalized adjoint equations for the ESDIRK4 time marching method are given as

$$\left[\frac{\partial \mathcal{R}_k^{(n)}}{\partial \mathbf{Q}_k^{(n)}}\right]^T \boldsymbol{\psi}_k^{(n)} = -\left[\omega_n \frac{\partial \mathcal{J}_n}{\partial \mathbf{Q}_k^{(n)}}\right]^T \quad \text{for } n = N \;, \quad k = 6 \tag{20}$$

$$\left[\frac{\partial \mathcal{R}_k^{(n)}}{\partial \mathbf{Q}_k^{(n)}}\right]^T \boldsymbol{\psi}_k^{(n)} = -\left[\sum_{i=k+1}^{6} \left(\boldsymbol{\psi}_i^{(n)}\right)^T \frac{\partial \mathcal{R}_i^{(n)}}{\partial \mathbf{Q}_k^{(n)}}\right]^T \quad \text{for } n = N \;, \quad k = 5,\ldots,2 \tag{21}$$

$$\left[\frac{\partial \mathcal{R}_k^{(n)}}{\partial \mathbf{Q}_k^{(n)}}\right]^T \boldsymbol{\psi}_k^{(n)} = -\left[\omega_n \frac{\partial \mathcal{J}_n}{\partial \mathbf{Q}_k^{(n)}} + \sum_{i=2}^{6} \left(\boldsymbol{\psi}_i^{(n+1)}\right)^T \frac{\partial \mathcal{R}_i^{(n+1)}}{\partial \mathbf{Q}_k^{(n)}} - \boldsymbol{\psi}_1^{(n+1)}\right]^T \quad \text{for } n = N-1,\ldots,1 \;, \quad k = 6 \tag{22}$$

$$\left[\frac{\partial \mathcal{R}_k^{(n)}}{\partial \mathbf{Q}_k^{(n)}}\right]^T \boldsymbol{\psi}_k^{(n)} = -\left[\sum_{i=k+1}^{6} \left(\boldsymbol{\psi}_i^{(n)}\right)^T \frac{\partial \mathcal{R}_i^{(n)}}{\partial \mathbf{Q}_k^{(n)}}\right]^T \quad \text{for } n = N-1,\ldots,1\,, \quad k = 5,\ldots,2 \tag{23}$$

$$\boldsymbol{\psi}_1^{(n)} = -\left[\sum_{i=2}^{6} \left(\boldsymbol{\psi}_i^{(n)}\right)^T \frac{\partial \mathcal{R}_i^{(n)}}{\partial \mathbf{Q}_1^{(n)}}\right]^T \tag{24}$$

$$\left[\frac{\partial \mathcal{M}_j}{\partial \mathbf{B}_j}\right]^T \boldsymbol{\lambda}_j = \underbrace{-\left[\sum_{n=1}^{N}\left(\omega_n \frac{\partial \mathcal{J}_n}{\partial \mathbf{B}_j} + \sum_{k=2}^{6}\left(\boldsymbol{\psi}_k^{(n)}\right)^T \frac{\partial \mathcal{R}_k^{(n)}}{\partial \mathbf{B}_j}\right)\right]^T}_{\text{last mesh-deformation increment RHS}} \quad \text{for } j = m \tag{25}$$

$$\left[\frac{\partial \mathcal{M}_j}{\partial \mathbf{B}_j}\right]^T \boldsymbol{\lambda}_j = -\left[\frac{\partial \mathcal{M}_{j+1}}{\partial \mathbf{B}_j}\boldsymbol{\lambda}_{j+1}\right]^T \quad \text{for } j = m-1,\ldots,1 \tag{26}$$

Equations (20), (21), (22), (23) and (24) constitute the flow adjoint problem, while equations (25) and (26) constitute the mesh adjoint problem for the ESDIRK4 time marching scheme.

C. Mesh Adjoint RHS Contribution from Flow Problem

At each time step, the right-hand side contribution to the the last mesh-deformation increment adjoint problem can be computed and stored. Let $\mathbf{Y}(\mathbf{B}_m)$ and $\mathbf{N}(\mathbf{Y})$ represent the grid points and grid metrics respectively of the flow solution mesh. From equation (18), the component of interest is derived as follows:

$$\begin{aligned}
\left[\sum_{n=1}^{N}\left(\omega_n \frac{\partial \mathcal{J}_n}{\partial \mathbf{B}_m} + \boldsymbol{\psi}_n^T \frac{\partial \mathcal{R}_n}{\partial \mathbf{B}_m}\right)\right]^T &= \sum_{n=1}^{N}\left[\left(\omega_n \left.\frac{\partial \mathcal{J}_n}{\partial \mathbf{Y}}\right|_{\mathbf{N}} + \omega_n \left.\frac{\partial \mathcal{J}_n}{\partial \mathbf{N}}\right|_{\mathbf{Y}} \frac{\partial \mathbf{N}}{\partial \mathbf{Y}} + \boldsymbol{\psi}_n^T \frac{\partial \mathcal{R}_n}{\partial \mathbf{N}} \frac{\partial \mathbf{N}}{\partial \mathbf{Y}}\right) \frac{\partial \mathbf{Y}}{\partial \mathbf{B}_m}\right]^T \\
&= \sum_{n=1}^{N}\left[\left\{\omega_n \left.\frac{\partial \mathcal{J}_n}{\partial \mathbf{Y}}\right|_{\mathbf{N}} + \left(\omega_n \left.\frac{\partial \mathcal{J}_n}{\partial \mathbf{N}}\right|_{\mathbf{Y}} + \boldsymbol{\psi}_n^T \frac{\partial \mathcal{R}_n}{\partial \mathbf{N}}\right) \frac{\partial \mathbf{N}}{\partial \mathbf{Y}}\right\} \frac{\partial \mathbf{Y}}{\partial \mathbf{B}_m}\right]^T \\
&= \frac{\partial \mathbf{Y}}{\partial \mathbf{B}_m}^T \sum_{n=1}^{N}\left[\omega_n \left.\frac{\partial \mathcal{J}_n}{\partial \mathbf{Y}}\right|_{\mathbf{N}} + \left(\omega_n \left.\frac{\partial \mathcal{J}_n}{\partial \mathbf{N}}\right|_{\mathbf{Y}} + \boldsymbol{\psi}_n^T \frac{\partial \mathcal{R}_n}{\partial \mathbf{N}}\right) \frac{\partial \mathbf{N}}{\partial \mathbf{Y}}\right]^T \\
&= \frac{\partial \mathbf{Y}}{\partial \mathbf{B}_m}^T \underbrace{\sum_{n=1}^{N}\left[\omega_n \left.\frac{\partial \mathcal{J}_n}{\partial \mathbf{Y}}\right|_{\mathbf{N}} + \left\{\omega_n \left.\frac{\partial \mathcal{J}_n}{\partial \mathbf{N}}\right|_{\mathbf{Y}} + \boldsymbol{\psi}_n^T \left(\boxed{\frac{\partial \tilde{\mathcal{R}}_n}{\partial \mathbf{N}}} + \frac{\partial \hat{\mathcal{R}}_n}{\partial \mathbf{N}}\right)\right\} \frac{\partial \mathbf{N}}{\partial \mathbf{Y}}\right]^T}_{\text{compute and add contribution for each time step}}
\end{aligned} \tag{27}$$

The final expression from equation (27) is used to compute the RHS for the last mesh-deformation increment for a single stage scheme like BDF2. For ESDIRK4, the expression is modified to

$$\begin{aligned}
&\left[\sum_{n=1}^{N}\left(\omega_n \frac{\partial \mathcal{J}_n}{\partial \mathbf{B}_j} + \sum_{k=2}^{6}\left(\boldsymbol{\psi}_k^{(n)}\right)^T \frac{\partial \mathcal{R}_k^{(n)}}{\partial \mathbf{B}_j}\right)\right]^T = \\
&\frac{\partial \mathbf{Y}}{\partial \mathbf{B}_m}^T \underbrace{\sum_{n=1}^{N}\left[\omega_n \left.\frac{\partial \mathcal{J}_n}{\partial \mathbf{Y}}\right|_{\mathbf{N}} + \left\{\omega_n \left.\frac{\partial \mathcal{J}_n}{\partial \mathbf{N}}\right|_{\mathbf{Y}} + \sum_{k=1}^{6}\left(\frac{1}{a_{kk}}\left(\boldsymbol{\psi}_k^{(n)}\right)^T \boxed{\frac{\partial \tilde{\mathcal{R}}_k^{(n)}}{\partial \mathbf{N}}} + \sum_{i=k}^{6}\frac{a_{ik}}{a_{kk}}\left(\boldsymbol{\psi}_i^{(n)}\right)^T \frac{\partial \hat{\mathcal{R}}_k^{(n)}}{\partial \mathbf{N}}\right)\right\} \frac{\partial \mathbf{N}}{\partial \mathbf{Y}}\right]^T}_{\text{compute and add contribution for each time step}}
\end{aligned} \tag{28}$$

with $\frac{1}{a_{11}} = 0$ and $\frac{a_{11}}{a_{11}} = 0$. $\tilde{\mathcal{R}}$ and $\hat{\mathcal{R}}$ retain their definitions from equation (3). Besides the boxed expressions (derivatives of the temporal residual) in equations (27) and (28) all the expressions present are available from the existing steady state optimization framework. The derivative of the temporal residual $\tilde{\mathcal{R}}(\hat{\mathbf{Q}}) = \tilde{\mathcal{R}}(J^{-1}(\mathbf{N})\mathbf{Q})$ with respect to the grid metrics $\mathbf{N}$ is given as

$$\frac{\partial \tilde{\mathcal{R}}}{\partial \mathbf{N}} = \frac{\partial \tilde{\mathcal{R}}}{\partial J} \frac{\partial J}{\partial \mathbf{N}} \tag{29}$$

D. Example Adjoint Problem - BDF2, Implicit Euler Startup, 3 Mesh-Deformation Increments and 5 Time Steps

1. Jacobian of Forward Problem, $\frac{\partial \mathcal{F}}{\partial \mathbf{V}}$

$$
\frac{\partial \mathcal{F}}{\partial \mathbf{V}} = \begin{pmatrix}
\frac{\partial \mathcal{M}_1}{\partial \mathbf{B}_1} & & & & & & & & \\
\frac{\partial \mathcal{M}_2}{\partial \mathbf{B}_1} & \frac{\partial \mathcal{M}_2}{\partial \mathbf{B}_2} & & & & & & & \\
 & \frac{\partial \mathcal{M}_3}{\partial \mathbf{B}_2} & \frac{\partial \mathcal{M}_3}{\partial \mathbf{B}_3} & & & & & & \\
 & & - & \mathcal{I} & & & & & \\
 & & \frac{\partial \mathcal{R}_2}{\partial \mathbf{B}_3} & -\frac{1}{\Delta t}\mathcal{I} & \frac{\partial \mathcal{R}_1}{\partial \mathbf{Q}_1} & & & & \\
 & & \frac{\partial \mathcal{R}_2}{\partial \mathbf{B}_3} & \frac{1}{2\Delta t}\mathcal{I} & -\frac{4}{2\Delta t}\mathcal{I} & \frac{\partial \mathcal{R}_2}{\partial \mathbf{Q}_2} & & & \\
 & & \frac{\partial \mathcal{R}_3}{\partial \mathbf{B}_3} & & \frac{1}{2\Delta t}\mathcal{I} & -\frac{4}{2\Delta t}\mathcal{I} & \frac{\partial \mathcal{R}_3}{\partial \mathbf{Q}_3} & & \\
 & & \frac{\partial \mathcal{R}_4}{\partial \mathbf{B}_3} & & & \frac{1}{2\Delta t}\mathcal{I} & -\frac{4}{2\Delta t}\mathcal{I} & \frac{\partial \mathcal{R}_4}{\partial \mathbf{Q}_4} & \\
 & & \frac{\partial \mathcal{R}_5}{\partial \mathbf{B}_3} & & & & \frac{1}{2\Delta t}\mathcal{I} & -\frac{4}{2\Delta t}\mathcal{I} & \frac{\partial \mathcal{R}_5}{\partial \mathbf{Q}_5}
\end{pmatrix}
$$

For BDF2

$$\frac{\partial \mathcal{R}_n}{\partial \mathbf{Q}_n} = \frac{3}{2\Delta t}\mathcal{I} + \frac{\partial \hat{\mathcal{R}}^n}{\partial \mathbf{Q}^n} \quad \text{for } n > 1$$

and for Implicit Euler

$$\frac{\partial \mathcal{R}_n}{\partial \mathbf{Q}_n} = \frac{1}{\Delta t}\mathcal{I} + \frac{\partial \hat{\mathcal{R}}^n}{\partial \mathbf{Q}^n} \quad \text{for } n = 1$$

2. Components of the Adjoint Problem

$$\left[\frac{\partial \mathcal{F}}{\partial \mathbf{V}}\right]^T \mathbf{\Lambda} = -\left[\frac{\partial \mathcal{J}}{\partial \mathbf{V}}\right]^T$$

$$
\begin{bmatrix}
\frac{\partial \mathcal{M}_1}{\partial \mathbf{B}_1}^T & \frac{\partial \mathcal{M}_2}{\partial \mathbf{B}_1}^T & & & & & & & & \\
 & \frac{\partial \mathcal{M}_2}{\partial \mathbf{B}_2}^T & \frac{\partial \mathcal{M}_3}{\partial \mathbf{B}_2}^T & & & & & & & \\
 & & \frac{\partial \mathcal{M}_3}{\partial \mathbf{B}_2}^T & - & \mathbf{0} & \frac{\partial \mathcal{R}_1}{\partial \mathbf{B}_3}^T & \frac{\partial \mathcal{R}_2}{\partial \mathbf{B}_3}^T & \frac{\partial \mathcal{R}_3}{\partial \mathbf{B}_3}^T & \frac{\partial \mathcal{R}_4}{\partial \mathbf{B}_3}^T & \frac{\partial \mathcal{R}_5}{\partial \mathbf{B}_3}^T \\
 & & & & \mathcal{I} & -\frac{1}{\Delta t}\mathcal{I} & \frac{1}{2\Delta t}\mathcal{I} & & & \\
 & & & & & \frac{\partial \mathcal{R}_1}{\partial \mathbf{Q}_1}^T & -\frac{4}{2\Delta t}\mathcal{I} & \frac{1}{2\Delta t}\mathcal{I} & & \\
 & & & & & & \frac{\partial \mathcal{R}_2}{\partial \mathbf{Q}_2}^T & -\frac{4}{2\Delta t}\mathcal{I} & \frac{1}{2\Delta t}\mathcal{I} & \\
 & & & & & & & \frac{\partial \mathcal{R}_3}{\partial \mathbf{Q}_3}^T & -\frac{4}{2\Delta t}\mathcal{I} & \frac{1}{2\Delta t}\mathcal{I} \\
 & & & & & & & & \frac{\partial \mathcal{R}_4}{\partial \mathbf{Q}_4}^T & -\frac{4}{2\Delta t}\mathcal{I} \\
 & & & & & & & & & \frac{\partial \mathcal{R}_5}{\partial \mathbf{Q}_5}^T
\end{bmatrix}
\begin{bmatrix}
\boldsymbol{\lambda}_1 \\ \boldsymbol{\lambda}_2 \\ \boldsymbol{\lambda}_3 \\ - \\ \boldsymbol{\psi}^{(1)} \\ \boldsymbol{\psi}^{(2)} \\ \boldsymbol{\psi}^{(3)} \\ \boldsymbol{\psi}^{(4)} \\ \boldsymbol{\psi}^{(5)}
\end{bmatrix}
=
\begin{bmatrix}
\mathbf{0} \\ \mathbf{0} \\ \sum_{n=1}^{5} \omega_n \frac{\partial \mathcal{J}_n}{\partial \mathbf{B}_3} \\ - \\ \omega_1 \frac{\partial \mathcal{J}_1}{\partial \mathbf{Q}_1} \\ \omega_2 \frac{\partial \mathcal{J}_2}{\partial \mathbf{Q}_2} \\ \omega_3 \frac{\partial \mathcal{J}_3}{\partial \mathbf{Q}_3} \\ \omega_4 \frac{\partial \mathcal{J}_4}{\partial \mathbf{Q}_4} \\ \omega_5 \frac{\partial \mathcal{J}_5}{\partial \mathbf{Q}_5}
\end{bmatrix}
$$

E. Example Adjoint Problem - BDF2, ESDIRK4 Startup, 3 Mesh-Deformation Increments and 5 Time Steps

1. Jacobian of Forward Problem, $\frac{\partial \mathcal{F}}{\partial \mathbf{V}}$

$$
\frac{\partial \mathcal{F}}{\partial \mathbf{V}} = \begin{pmatrix}
\frac{\partial \mathcal{M}_1}{\partial \mathbf{B}_1} \\
\frac{\partial \mathcal{M}_2}{\partial \mathbf{B}_1} & \frac{\partial \mathcal{M}_2}{\partial \mathbf{B}_2} \\
 & \frac{\partial \mathcal{M}_3}{\partial \mathbf{B}_2} & \frac{\partial \mathcal{M}_3}{\partial \mathbf{B}_3} \\
 & & - & \mathcal{I} \\
 & & \mathbf{0} & -\mathcal{I} & \mathcal{I} \\
 & & \frac{\partial \mathcal{R}_2^1}{\partial \mathbf{B}_3} & -\frac{1}{a_{22}\Delta t}\mathcal{I} & \frac{a_{21}}{a_{22}}\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1} & \frac{\partial \mathcal{R}_2^1}{\partial \mathbf{Q}_2^1} \\
 & & \frac{\partial \mathcal{R}_2^1}{\partial \mathbf{B}_3} & -\frac{1}{a_{33}\Delta t}\mathcal{I} & \frac{a_{31}}{a_{33}}\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1} & \frac{a_{32}}{a_{33}}\frac{\partial \hat{\mathcal{R}}_2^1}{\partial \mathbf{Q}_2^1} & \frac{\partial \mathcal{R}_3^1}{\partial \mathbf{Q}_3^1} \\
 & & \frac{\partial \mathcal{R}_4^1}{\partial \mathbf{B}_3} & -\frac{1}{a_{44}\Delta t}\mathcal{I} & \frac{a_{41}}{a_{44}}\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1} & \frac{a_{42}}{a_{44}}\frac{\partial \hat{\mathcal{R}}_2^1}{\partial \mathbf{Q}_2^1} & \frac{a_{43}}{a_{44}}\frac{\partial \hat{\mathcal{R}}_3^1}{\partial \mathbf{Q}_3^1} & \frac{\partial \mathcal{R}_4^1}{\partial \mathbf{Q}_4^1} \\
 & & \frac{\partial \mathcal{R}_5^1}{\partial \mathbf{B}_3} & -\frac{1}{a_{55}\Delta t}\mathcal{I} & \frac{a_{51}}{a_{55}}\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1} & \frac{a_{52}}{a_{55}}\frac{\partial \hat{\mathcal{R}}_2^1}{\partial \mathbf{Q}_2^1} & \frac{a_{53}}{a_{55}}\frac{\partial \hat{\mathcal{R}}_3^1}{\partial \mathbf{Q}_3^1} & \frac{a_{54}}{a_{55}}\frac{\partial \hat{\mathcal{R}}_4^1}{\partial \mathbf{Q}_4^1} & \frac{\partial \mathcal{R}_5^1}{\partial \mathbf{Q}_5^1} \\
 & & \frac{\partial \mathcal{R}_6^1}{\partial \mathbf{B}_3} & -\frac{1}{a_{66}\Delta t}\mathcal{I} & \frac{a_{61}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1} & \frac{a_{62}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_2^1}{\partial \mathbf{Q}_2^1} & \frac{a_{63}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_3^1}{\partial \mathbf{Q}_3^1} & \frac{a_{64}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_4^1}{\partial \mathbf{Q}_4^1} & \frac{a_{65}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_5^1}{\partial \mathbf{Q}_5^1} & \frac{\partial \mathcal{R}_6^1}{\partial \mathbf{Q}_6^1} \\
 & & \frac{\partial \mathcal{R}_2}{\partial \mathbf{B}_3} & \frac{1}{2\Delta t}\mathcal{I} & & & & & & -\frac{4}{2\Delta t}\mathcal{I} & \frac{\partial \mathcal{R}_2}{\partial \mathbf{Q}_2} \\
 & & \frac{\partial \mathcal{R}_3}{\partial \mathbf{B}_3} & & & & & & & \frac{1}{2\Delta t}\mathcal{I} & -\frac{4}{2\Delta t}\mathcal{I} & \frac{\partial \mathcal{R}_3}{\partial \mathbf{Q}_3} \\
 & & \frac{\partial \mathcal{R}_4}{\partial \mathbf{B}_3} & & & & & & & & \frac{1}{2\Delta t}\mathcal{I} & -\frac{4}{2\Delta t}\mathcal{I} & \frac{\partial \mathcal{R}_4}{\partial \mathbf{Q}_4} \\
 & & \frac{\partial \mathcal{R}_5}{\partial \mathbf{B}_3} & & & & & & & & & \frac{1}{2\Delta t}\mathcal{I} & -\frac{4}{2\Delta t}\mathcal{I} & \frac{\partial \mathcal{R}_5}{\partial \mathbf{Q}_5}
\end{pmatrix}
$$

Due to space limitations, $\mathcal{R}_k^{(n)}$ is represented as $\mathcal{R}_k^n$.

2. Components of the Adjoint Problem

$$
\left[\frac{\partial \mathcal{F}}{\partial \mathbf{V}}\right]^T \mathbf{\Lambda} = -\left[\frac{\partial \mathcal{J}}{\partial \mathbf{V}}\right]^T
$$

$$
\left[\frac{\partial \mathcal{F}}{\partial \mathbf{Q}}\right]^T = \begin{pmatrix}
\frac{\partial \mathcal{M}_1}{\partial \mathbf{B}_1}^T & \frac{\partial \mathcal{M}_2}{\partial \mathbf{B}_1}^T \\
 & \frac{\partial \mathcal{M}_2}{\partial \mathbf{B}_2}^T & \frac{\partial \mathcal{M}_3}{\partial \mathbf{B}_2}^T \\
 & & \frac{\partial \mathcal{M}_3}{\partial \mathbf{B}_2}^T & - & \mathbf{0} & \frac{\partial \mathcal{R}_2^1}{\partial \mathbf{B}_3}^T & \frac{\partial \mathcal{R}_3^1}{\partial \mathbf{B}_3}^T & \frac{\partial \mathcal{R}_4^1}{\partial \mathbf{B}_3}^T & \frac{\partial \mathcal{R}_5^1}{\partial \mathbf{B}_3}^T & \frac{\partial \mathcal{R}_6^1}{\partial \mathbf{B}_3}^T & \frac{\partial \mathcal{R}_2}{\partial \mathbf{B}_3}^T & \frac{\partial \mathcal{R}_3}{\partial \mathbf{B}_3}^T & \frac{\partial \mathcal{R}_4}{\partial \mathbf{B}_3}^T & \frac{\partial \mathcal{R}_5}{\partial \mathbf{B}_3}^T \\
 & & & \mathcal{I} & -\mathcal{I} & -\frac{1}{a_{22}\Delta t}\mathcal{I} & -\frac{1}{a_{33}\Delta t}\mathcal{I} & -\frac{1}{a_{44}\Delta t}\mathcal{I} & -\frac{1}{a_{55}\Delta t}\mathcal{I} & -\frac{1}{a_{66}\Delta t}\mathcal{I} & \frac{1}{2\Delta t}\mathcal{I} \\
 & & & & \mathcal{I} & \frac{a_{21}}{a_{22}}\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1}^T & \frac{a_{31}}{a_{33}}\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1}^T & \frac{a_{41}}{a_{44}}\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1}^T & \frac{a_{51}}{a_{55}}\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1}^T & \frac{a_{61}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1}^T \\
 & & & & & \frac{\partial \mathcal{R}_2^1}{\partial \mathbf{Q}_2^1}^T & \frac{a_{32}}{a_{33}}\frac{\partial \hat{\mathcal{R}}_2^1}{\partial \mathbf{Q}_2^1}^T & \frac{a_{42}}{a_{44}}\frac{\partial \hat{\mathcal{R}}_2^1}{\partial \mathbf{Q}_2^1}^T & \frac{a_{62}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_2^1}{\partial \mathbf{Q}_2^1}^T & \frac{a_{62}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_2^1}{\partial \mathbf{Q}_2^1}^T \\
 & & & & & & \frac{\partial \mathcal{R}_3^1}{\partial \mathbf{Q}_3^1}^T & \frac{a_{43}}{a_{44}}\frac{\partial \hat{\mathcal{R}}_3^1}{\partial \mathbf{Q}_3^1}^T & \frac{a_{53}}{a_{55}}\frac{\partial \hat{\mathcal{R}}_3^1}{\partial \mathbf{Q}_3^1}^T & \frac{a_{63}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_3^1}{\partial \mathbf{Q}_3^1}^T \\
 & & & & & & & \frac{\partial \mathcal{R}_4^1}{\partial \mathbf{Q}_4^1}^T & \frac{a_{54}}{a_{55}}\frac{\partial \hat{\mathcal{R}}_4^1}{\partial \mathbf{Q}_4^1}^T & \frac{a_{64}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_4^1}{\partial \mathbf{Q}_4^1}^T \\
 & & & & & & & & \frac{\partial \mathcal{R}_5^1}{\partial \mathbf{Q}_5^1}^T & \frac{a_{65}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_5^1}{\partial \mathbf{Q}_5^1}^T \\
 & & & & & & & & & \frac{\partial \mathcal{R}_6^1}{\partial \mathbf{Q}_6^1}^T & -\frac{4}{2\Delta t}\mathcal{I} & \frac{1}{2\Delta t}\mathcal{I} \\
 & & & & & & & & & & \frac{\partial \mathcal{R}_2}{\partial \mathbf{Q}_2}^T & -\frac{4}{2\Delta t}\mathcal{I} & \frac{1}{2\Delta t}\mathcal{I} \\
 & & & & & & & & & & & \frac{\partial \mathcal{R}_3}{\partial \mathbf{Q}_3}^T & -\frac{4}{2\Delta t}\mathcal{I} & \frac{1}{2\Delta t}\mathcal{I} \\
 & & & & & & & & & & & & \frac{\partial \mathcal{R}_4}{\partial \mathbf{Q}_4}^T & -\frac{4}{2\Delta t}\mathcal{I} \\
 & & & & & & & & & & & & & \frac{\partial \mathcal{R}_5}{\partial \mathbf{Q}_5}^T
\end{pmatrix}
$$

$$
\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & \lambda_2 & \lambda_3 & - & \boldsymbol{\psi}_1^{(1)} & \boldsymbol{\psi}_2^{(1)} & \boldsymbol{\psi}_3^{(1)} & \boldsymbol{\psi}_4^{(1)} & \boldsymbol{\psi}_5^{(1)} & \boldsymbol{\psi}_6^{(1)} & \boldsymbol{\psi}^{(2)} & \boldsymbol{\psi}^{(3)} & \boldsymbol{\psi}^{(4)} & \boldsymbol{\psi}^{(5)} \end{bmatrix}^T
$$

$$
\left[\frac{\partial \mathcal{J}}{\partial \mathbf{V}}\right]^T = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \sum_{n=1}^{5}\omega_n \frac{\partial \mathcal{J}_n}{\partial \mathbf{B}_3} & - & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \omega_1 \frac{\partial \mathcal{J}_1}{\partial \mathbf{Q}_1} & \omega_2 \frac{\partial \mathcal{J}_2}{\partial \mathbf{Q}_2} & \omega_3 \frac{\partial \mathcal{J}_3}{\partial \mathbf{Q}_3} & \omega_4 \frac{\partial \mathcal{J}_4}{\partial \mathbf{Q}_4} & \omega_5 \frac{\partial \mathcal{J}_5}{\partial \mathbf{Q}_5} \end{bmatrix}^T
$$

F. Example Adjoint Problem - ESDIRK4, 3 Mesh-Deformation Increments and 2 Time Steps

1. Jacobian of Forward Problem, $\frac{\partial \mathcal{F}}{\partial \mathbf{V}}$

$$
\frac{\partial \mathcal{F}}{\partial \mathbf{V}} = \left(\begin{array}{cccccccccccccccc}
\frac{\partial \mathcal{M}_1}{\partial \mathbf{B}_1} &&&&&&&&&&&&&&& \\
\frac{\partial \mathcal{M}_2}{\partial \mathbf{B}_1} & \frac{\partial \mathcal{M}_2}{\partial \mathbf{B}_2} &&&&&&&&&&&&&& \\
& \frac{\partial \mathcal{M}_3}{\partial \mathbf{B}_2} & \frac{\partial \mathcal{M}_3}{\partial \mathbf{B}_3} &&&&&&&&&&&&& \\
&& - & \mathcal{I} &&&&&&&&&&&& \\
&& \mathbf{0} & -\mathcal{I} & \mathcal{I} &&&&&&&&&&& \\
&& \frac{\partial \mathcal{R}_2^1}{\partial \mathbf{B}_3} & -\frac{1}{a_{22}\Delta t}\mathcal{I} & \frac{a_{21}}{a_{22}}\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1} & \frac{\partial \mathcal{R}_2^1}{\partial \mathbf{Q}_2^1} &&&&&&&&&& \\
&& \frac{\partial \mathcal{R}_3^1}{\partial \mathbf{B}_3} & -\frac{1}{a_{33}\Delta t}\mathcal{I} & \frac{a_{31}}{a_{33}}\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1} & \frac{a_{32}}{a_{33}}\frac{\partial \hat{\mathcal{R}}_2^1}{\partial \mathbf{Q}_2^1} & \frac{\partial \mathcal{R}_3^1}{\partial \mathbf{Q}_3^1} &&&&&&&&& \\
&& \frac{\partial \mathcal{R}_4^1}{\partial \mathbf{B}_3} & -\frac{1}{a_{44}\Delta t}\mathcal{I} & \frac{a_{41}}{a_{44}}\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1} & \frac{a_{42}}{a_{44}}\frac{\partial \hat{\mathcal{R}}_2^1}{\partial \mathbf{Q}_2^1} & \frac{a_{43}}{a_{44}}\frac{\partial \hat{\mathcal{R}}_3^1}{\partial \mathbf{Q}_3^1} & \frac{\partial \mathcal{R}_1^4}{\partial \mathbf{Q}_1^4} &&&&&&&& \\
&& \frac{\partial \mathcal{R}_5^1}{\partial \mathbf{B}_3} & -\frac{1}{a_{55}\Delta t}\mathcal{I} & \frac{a_{51}}{a_{55}}\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1} & \frac{a_{52}}{a_{55}}\frac{\partial \hat{\mathcal{R}}_2^1}{\partial \mathbf{Q}_2^1} & \frac{a_{53}}{a_{55}}\frac{\partial \hat{\mathcal{R}}_3^1}{\partial \mathbf{Q}_3^1} & \frac{a_{54}}{a_{55}}\frac{\partial \hat{\mathcal{R}}_4^1}{\partial \mathbf{Q}_4^1} & \frac{\partial \mathcal{R}_1^5}{\partial \mathbf{Q}_1^5} &&&&&&& \\
&& \frac{\partial \mathcal{R}_6^1}{\partial \mathbf{B}_3} & -\frac{1}{a_{66}\Delta t}\mathcal{I} & \frac{a_{61}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1} & \frac{a_{62}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_2^1}{\partial \mathbf{Q}_2^1} & \frac{a_{63}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_3^1}{\partial \mathbf{Q}_3^1} & \frac{a_{64}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_4^1}{\partial \mathbf{Q}_4^1} & \frac{a_{65}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_5^1}{\partial \mathbf{Q}_5^1} & \frac{\partial \mathcal{R}_1^6}{\partial \mathbf{Q}_1^6} &&&&&& \\
&& \mathbf{0} &&&&&& -\mathcal{I} & \mathcal{I} &&&&&& \\
&& \frac{\partial \mathcal{R}_2^2}{\partial \mathbf{B}_3} &&&&&&& -\frac{1}{a_{22}\Delta t}\mathcal{I} & \frac{a_{21}}{a_{22}}\frac{\partial \hat{\mathcal{R}}_1^2}{\partial \mathbf{Q}_1^2} & \frac{\partial \mathcal{R}_2^2}{\partial \mathbf{Q}_2^2} &&&& \\
&& \frac{\partial \mathcal{R}_3^2}{\partial \mathbf{B}_3} &&&&&&& -\frac{1}{a_{33}\Delta t}\mathcal{I} & \frac{a_{31}}{a_{33}}\frac{\partial \hat{\mathcal{R}}_1^2}{\partial \mathbf{Q}_1^2} & \frac{a_{32}}{a_{33}}\frac{\partial \hat{\mathcal{R}}_2^2}{\partial \mathbf{Q}_2^2} & \frac{\partial \mathcal{R}_2^3}{\partial \mathbf{Q}_2^3} &&& \\
&& \frac{\partial \mathcal{R}_4^2}{\partial \mathbf{B}_3} &&&&&&& -\frac{1}{a_{44}\Delta t}\mathcal{I} & \frac{a_{41}}{a_{44}}\frac{\partial \hat{\mathcal{R}}_1^2}{\partial \mathbf{Q}_1^2} & \frac{a_{42}}{a_{44}}\frac{\partial \hat{\mathcal{R}}_2^2}{\partial \mathbf{Q}_2^2} & \frac{a_{43}}{a_{44}}\frac{\partial \hat{\mathcal{R}}_3^2}{\partial \mathbf{Q}_3^2} & \frac{\partial \mathcal{R}_2^4}{\partial \mathbf{Q}_2^4} && \\
&& \frac{\partial \mathcal{R}_5^2}{\partial \mathbf{B}_3} &&&&&&& -\frac{1}{a_{55}\Delta t}\mathcal{I} & \frac{a_{51}}{a_{55}}\frac{\partial \hat{\mathcal{R}}_1^2}{\partial \mathbf{Q}_1^2} & \frac{a_{52}}{a_{55}}\frac{\partial \hat{\mathcal{R}}_2^2}{\partial \mathbf{Q}_2^2} & \frac{a_{53}}{a_{55}}\frac{\partial \hat{\mathcal{R}}_3^2}{\partial \mathbf{Q}_3^2} & \frac{a_{54}}{a_{55}}\frac{\partial \hat{\mathcal{R}}_4^2}{\partial \mathbf{Q}_4^2} & \frac{\partial \mathcal{R}_2^5}{\partial \mathbf{Q}_2^5} & \\
&& \frac{\partial \mathcal{R}_2^6}{\partial \mathbf{B}_3} &&&&&&& -\frac{1}{a_{66}\Delta t}\mathcal{I} & \frac{a_{61}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_1^2}{\partial \mathbf{Q}_1^2} & \frac{a_{62}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_2^2}{\partial \mathbf{Q}_2^2} & \frac{a_{63}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_3^2}{\partial \mathbf{Q}_3^2} & \frac{a_{64}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_4^2}{\partial \mathbf{Q}_4^2} & \frac{a_{65}}{a_{66}}\frac{\partial \hat{\mathcal{R}}_5^2}{\partial \mathbf{Q}_5^2} & \frac{\partial \mathcal{R}_2^6}{\partial \mathbf{Q}_2^6}
\end{array}\right)
$$

Due to space limitations, $\mathcal{R}_k^{(n)}$ is represented as $\mathcal{R}_k^n$.

$$\left[\frac{\partial \mathcal{F}}{\partial \mathbf{V}}\right]^T \mathbf{\Lambda} = -\left[\frac{\partial \mathcal{J}}{\partial \mathbf{V}}\right]^T$$

$$\left[\frac{\partial \mathcal{F}}{\partial \mathbf{Q}}\right]^T = \left(\begin{array}{cccccccccccccccc}
{\frac{\partial \mathcal{M}_1}{\partial \mathbf{B}_1}}^T & {\frac{\partial \mathcal{M}_2}{\partial \mathbf{B}_1}}^T & & & & & & & & & & & & & & \\
 & {\frac{\partial \mathcal{M}_2}{\partial \mathbf{B}_2}}^T & {\frac{\partial \mathcal{M}_3}{\partial \mathbf{B}_2}}^T & & & & & & & & & & & & & \\
 & & {\frac{\partial \mathcal{M}_3}{\partial \mathbf{B}_2}}^T & - & \mathbf{0} & {\frac{\partial \mathcal{R}_1^2}{\partial \mathbf{B}_3}}^T & {\frac{\partial \mathcal{R}_1^3}{\partial \mathbf{B}_3}}^T & {\frac{\partial \mathcal{R}_1^4}{\partial \mathbf{B}_3}}^T & {\frac{\partial \mathcal{R}_1^5}{\partial \mathbf{B}_3}}^T & {\frac{\partial \mathcal{R}_1^6}{\partial \mathbf{B}_3}}^T & \mathbf{0} & {\frac{\partial \mathcal{R}_2^2}{\partial \mathbf{B}_3}}^T & {\frac{\partial \mathcal{R}_2^3}{\partial \mathbf{B}_3}}^T & {\frac{\partial \mathcal{R}_2^4}{\partial \mathbf{B}_3}}^T & {\frac{\partial \mathcal{R}_2^5}{\partial \mathbf{B}_3}}^T & {\frac{\partial \mathcal{R}_2^6}{\partial \mathbf{B}_3}}^T \\
 & & & \mathcal{I} & -\mathcal{I} & -\frac{1}{a_{22}\Delta t}\mathcal{I} & -\frac{1}{a_{33}\Delta t}\mathcal{I} & -\frac{1}{a_{44}\Delta t}\mathcal{I} & -\frac{1}{a_{55}\Delta t}\mathcal{I} & -\frac{1}{a_{66}\Delta t}\mathcal{I} & & & & & & \\
 & & & & \mathcal{I} & \frac{a_{21}}{a_{22}}{\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1}}^T & \frac{a_{31}}{a_{33}}{\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1}}^T & \frac{a_{41}}{a_{44}}{\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1}}^T & \frac{a_{51}}{a_{55}}{\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1}}^T & \frac{a_{61}}{a_{66}}{\frac{\partial \hat{\mathcal{R}}_1^1}{\partial \mathbf{Q}_1^1}}^T & & & & & & \\
 & & & & & {\frac{\partial \mathcal{R}_2^1}{\partial \mathbf{Q}_2^1}}^T & \frac{a_{32}}{a_{33}}{\frac{\partial \hat{\mathcal{R}}_2^1}{\partial \mathbf{Q}_2^1}}^T & \frac{a_{42}}{a_{44}}{\frac{\partial \hat{\mathcal{R}}_2^1}{\partial \mathbf{Q}_2^1}}^T & \frac{a_{62}}{a_{66}}{\frac{\partial \hat{\mathcal{R}}_2^1}{\partial \mathbf{Q}_2^1}}^T & \frac{a_{62}}{a_{66}}{\frac{\partial \hat{\mathcal{R}}_2^1}{\partial \mathbf{Q}_2^1}}^T & & & & & & \\
 & & & & & & {\frac{\partial \mathcal{R}_3^1}{\partial \mathbf{Q}_3^1}}^T & \frac{a_{43}}{a_{44}}{\frac{\partial \hat{\mathcal{R}}_3^1}{\partial \mathbf{Q}_3^1}}^T & \frac{a_{53}}{a_{55}}{\frac{\partial \hat{\mathcal{R}}_3^1}{\partial \mathbf{Q}_3^1}}^T & \frac{a_{63}}{a_{66}}{\frac{\partial \hat{\mathcal{R}}_3^1}{\partial \mathbf{Q}_3^1}}^T & & & & & & \\
 & & & & & & & {\frac{\partial \mathcal{R}_4^1}{\partial \mathbf{Q}_4^1}}^T & \frac{a_{54}}{a_{55}}{\frac{\partial \hat{\mathcal{R}}_4^1}{\partial \mathbf{Q}_4^1}}^T & \frac{a_{64}}{a_{66}}{\frac{\partial \hat{\mathcal{R}}_4^1}{\partial \mathbf{Q}_4^1}}^T & & & & & & \\
 & & & & & & & & {\frac{\partial \mathcal{R}_5^1}{\partial \mathbf{Q}_5^1}}^T & \frac{a_{65}}{a_{66}}{\frac{\partial \hat{\mathcal{R}}_5^1}{\partial \mathbf{Q}_5^1}}^T & & & & & & \\
 & & & & & & & & & {\frac{\partial \mathcal{R}_6^1}{\partial \mathbf{Q}_6^1}}^T & -\mathcal{I} & -\frac{1}{a_{22}\Delta t}\mathcal{I} & -\frac{1}{a_{33}\Delta t}\mathcal{I} & -\frac{1}{a_{44}\Delta t}\mathcal{I} & -\frac{1}{a_{55}\Delta t}\mathcal{I} & -\frac{1}{a_{66}\Delta t}\mathcal{I} \\
 & & & & & & & & & & \mathcal{I} & \frac{a_{21}}{a_{22}}{\frac{\partial \hat{\mathcal{R}}_1^2}{\partial \mathbf{Q}_1^2}}^T & \frac{a_{31}}{a_{33}}{\frac{\partial \hat{\mathcal{R}}_1^2}{\partial \mathbf{Q}_1^2}}^T & \frac{a_{41}}{a_{44}}{\frac{\partial \hat{\mathcal{R}}_1^2}{\partial \mathbf{Q}_1^2}}^T & \frac{a_{51}}{a_{55}}{\frac{\partial \hat{\mathcal{R}}_1^2}{\partial \mathbf{Q}_1^2}}^T & \frac{a_{61}}{a_{66}}{\frac{\partial \hat{\mathcal{R}}_1^2}{\partial \mathbf{Q}_1^2}}^T \\
 & & & & & & & & & & & {\frac{\partial \mathcal{R}_2^2}{\partial \mathbf{Q}_2^2}}^T & \frac{a_{32}}{a_{33}}{\frac{\partial \hat{\mathcal{R}}_2^2}{\partial \mathbf{Q}_2^2}}^T & \frac{a_{42}}{a_{44}}{\frac{\partial \hat{\mathcal{R}}_2^2}{\partial \mathbf{Q}_2^2}}^T & \frac{a_{52}}{a_{55}}{\frac{\partial \hat{\mathcal{R}}_2^2}{\partial \mathbf{Q}_2^2}}^T & \frac{a_{62}}{a_{66}}{\frac{\partial \hat{\mathcal{R}}_2^2}{\partial \mathbf{Q}_2^2}}^T \\
 & & & & & & & & & & & & {\frac{\partial \mathcal{R}_3^2}{\partial \mathbf{Q}_3^2}}^T & \frac{a_{43}}{a_{44}}{\frac{\partial \hat{\mathcal{R}}_3^2}{\partial \mathbf{Q}_3^2}}^T & \frac{a_{53}}{a_{55}}{\frac{\partial \hat{\mathcal{R}}_3^2}{\partial \mathbf{Q}_3^2}}^T & \frac{a_{63}}{a_{66}}{\frac{\partial \hat{\mathcal{R}}_3^2}{\partial \mathbf{Q}_3^2}}^T \\
 & & & & & & & & & & & & & {\frac{\partial \mathcal{R}_4^2}{\partial \mathbf{Q}_4^2}}^T & \frac{a_{54}}{a_{55}}{\frac{\partial \hat{\mathcal{R}}_4^2}{\partial \mathbf{Q}_4^2}}^T & \frac{a_{64}}{a_{66}}{\frac{\partial \hat{\mathcal{R}}_4^2}{\partial \mathbf{Q}_4^2}}^T \\
 & & & & & & & & & & & & & & {\frac{\partial \mathcal{R}_5^2}{\partial \mathbf{Q}_5^2}}^T & \frac{a_{65}}{a_{66}}{\frac{\partial \hat{\mathcal{R}}_5^2}{\partial \mathbf{Q}_5^2}}^T \\
 & & & & & & & & & & & & & & & {\frac{\partial \mathcal{R}_6^2}{\partial \mathbf{Q}_6^2}}^T
\end{array}\right)$$

$$\mathbf{\Lambda} = \left[\begin{array}{cccccccccccccccc} \boldsymbol{\lambda}_1 & \boldsymbol{\lambda}_2 & \boldsymbol{\lambda}_3 & - & \boldsymbol{\psi}_1^{(1)} & \boldsymbol{\psi}_2^{(1)} & \boldsymbol{\psi}_3^{(1)} & \boldsymbol{\psi}_4^{(1)} & \boldsymbol{\psi}_5^{(1)} & \boldsymbol{\psi}_6^{(1)} & \boldsymbol{\psi}_1^{(2)} & \boldsymbol{\psi}_2^{(2)} & \boldsymbol{\psi}_3^{(2)} & \boldsymbol{\psi}_4^{(2)} & \boldsymbol{\psi}_5^{(2)} & \boldsymbol{\psi}_6^{(2)} \end{array}\right]^T$$

$$\left[\frac{\partial \mathcal{J}}{\partial \mathbf{V}}\right]^T = \left[\begin{array}{cccccccccccccccc} \mathbf{0} & \mathbf{0} & \omega_1\frac{\partial \mathcal{J}_1}{\partial \mathbf{B}_3} + \omega_2\frac{\partial \mathcal{J}_2}{\partial \mathbf{B}_3} & - & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \omega_1\frac{\partial \mathcal{J}_1}{\partial \mathbf{Q}_1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \omega_2\frac{\partial \mathcal{J}_2}{\partial \mathbf{Q}_2} \end{array}\right]^T$$

Acknowledgements

The authors are grateful for the financial support from the Natural Sciences and Engineering Research Council (NSERC), Centre for Aerial Robotics Research and Education (CARRE) and the Dr. Lorne Heuckroth Scholarship towards this work. All computations were performed on the Niagara supercomputer at the SciNet HPC Consortium. SciNet is funded by: the Canada Foundation for Innovation, the Government of Ontario, Ontario Research Fund - Research Excellence, and the University of Toronto.

References

[1] Hess, J. L., "Development and Application of Panel Methods," *Advanced Boundary Element Methods*, International Union of Theoretical and Applied Mechanics, Springer–Verlag, Berlin, 1988, pp. 165–177.

[2] Drela, M., "XFOIL: An Analysis and Design System for Low Reynolds Number Airfoils," *Low Reynolds Number Aerodynamics*, Lecture Notes in Engineering, Vol. 54, Springer–Verlag, Berlin, 1989, pp. 1–12.

[3] Hicks, R. M., and Henne, P. A., "Wing Design by Numerical Optimization," *Journal of Aircraft*, Vol. 15, 1978, pp. 407–412.

[4] Pironneau, O., "On Optimum Design in Fluid Mechanics," *Journal Fluid Mechanics*, Vol. 64, 1974, pp. 97–110.

[5] Jameson, A., Pierce, N. A., and Martinelli, L., "Optimum Aerodynamic Design Using the Navier-Stokes Equations," *Theoretical and Computational Fluid Dynamics*, Vol. 10, No. 1, 1998, pp. 213–237.

[6] Anderson, W. K., and Bonhaus, D. L., "Airfoil Design on Unstructured Grids for Turbulent Flows," *AIAA Journal*, Vol. 37, 1999, pp. 185–191.

[7] Wang, Q., Hu, R., and Blonigan, P., "Least Squares Shadowing Sensitivity Analysis of Chaotic Limit Cycle Oscillations," *Journal Computational Physics*, Vol. 267, 2014, pp. 210–224.

[8] Yee, K., Kim, K., and Lee, D., "Aerodynamic Shape Optimization of Rotor Airfoils Undergoing Unsteady Motion," AIAA Paper 99-3107, 1999.

[9] Nadarajah, S., and Jameson, A., "Optimal Control of Unsteady Flows using a Time Accurate Method," AIAA Paper 2002-5436, 2002.

[10] Mani, K., and Mavriplis, D. J., "Unsteady Discrete Adjoint Formulation for Two-Dimensional Flow Problems with Deforming Meshes," *AIAA Journal*, Vol. 46, No. 6, 2008, pp. 1351–1364.

[11] Nielsen, E. J., Diskin, B., and Yamaleev, N. K., "Discrete Adjoint-Based Design Optimization of Unsteady Turbulent Flows on Dynamic Unstructured Grids," *AIAA Journal*, Vol. 48, No. 6, 2010, pp. 1195–1206.

[12] Rumpfkeil, M., and D. W. Zingg, D. W., "The Optimal Control of Unsteady Flows with a Discrete Adjoint Method," *Optimization and Engineering*, Vol. 11, No. 1, 2010, pp. 5–22.

[13] Rumpfkeil, M., and Zingg, D. W., "A Hybrid Algorithm for Far-Field Noise Minimization," *Computers and Fluids*, Vol. 39, No. 9, 2010, pp. 1516–1528.

[14] Hicken, H. E., and Zingg, D. W., "A Parallel Newton-Krylov Solver for the Euler Equations Discretized Using Simultaneous Approximation Terms," *AIAA Journal*, Vol. 46, No. 11, 2008, pp. 2273–2786.

[15] Hicken, H. E., and Zingg, D. W., "Aerodynamic Optimization Algorithm with Integrated Geometry Parameterization and Mesh Movement," *AIAA Journal*, Vol. 48, No. 2, 2010, pp. 401–413.

[16] Osusky, M., and Zingg, D. W., "Parallel Newton-Krylov-Schur Solver for Navier-Stokes Equations Discretized Using Summation-By-Parts Operators," *AIAA Journal*, Vol. 51, No. 12, 2013.

[17] Osusky, L., Buckley, H., Reist, T., and Zingg, D. W., "Drag Minimization Based on the Navier-Stokes Equations Using A Newton-Krylov Approach," *AIAA Journal*, Vol. 53, No. 6, 2015, pp. 1672–1694.

[18] Gagnon, H., and Zingg, D. W., "Two-Level Free-Form and Axial Deformation for Exploratory Aerodynamic Shape Optimization," *AIAA Journal*, Vol. 53, No. 7, 2015, pp. 2015–2026.

[19] Boom, P. D., and Zingg, D. W., "Time-Accurate Flow Simulations Using an Efficient Newton-Krylov-Schur Approach with High-Order Temporal and Spatial Discretization," AIAA Paper 2013-0383, 2013.

[20] Pulliam, T. H., and Zingg, D. W., *Fundamental Algorithms in Computational Fluid Dynamics*, Springer-Verlag, 2003.

[21] Del Rey Fernández, D. C., Hicken, J., and Zingg, D. W., "Review of Summation-By-Parts Operators with Simultaneous Approximation Terms for the Numerical Solution of Partial Differential Equations," *Computers and Fluids*, Vol. 95, 2014, pp. 171–196.

[22] Saad, Y., and Schultz, M. H., "GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems," *SIAM Journal on Scientific and Statistical Computing*, Vol. 7, No. 3, 1986, pp. 856–869.

[23] Gill, P. E., Murray, W., and Saunders, M. A., "SNOPT: An SQP algorithm for Large-Scale Constrained Optimization," *SIAM Journal of Optimization*, Vol. 12, 2002, pp. 979–1006.

[24] Hicken, J. E., and Zingg, D. W., "A Simplified and Flexible Variant of GCROT for Solving Nonsymmetric Linear Systems," *SIAM Journal on Scientific Computing*, Vol. 32, No. 3, 2010, pp. 1672–1694.

[25] Lee, C., Koo, D., and Zingg, D. W., "Comparison of B-spline Surface and Free-form Deformation Control Methods for Aerodynamic Shape Optimization," *AIAA Journal*, Vol. 55, No. 1, 2017, pp. 228–240.

[26] Wang, Q., Moin, P., and Iaccarino, G., "Minimal Repetition Dynamic Checkpointing Algorithm for Unsteady Adjoint Calculation," *SIAM J. Scientific Computing*, Vol. 31, 2009, pp. 2549–2567.