Adaptive High-Order Fluid-Structure Interaction Simulations with a Study of Mesh-Motion Errors

Vivek Ojha[*], Krzysztof J. Fidkowski[†] and Carlos E. S. Cesnik[‡]

Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI 48109, USA

This paper demonstrates an adaptive approach for solving fluid-structure interaction problems using high-fidelity numerical methods along with a detailed analysis of mesh-motion errors. A high-order partitioned approach is applied to couple the fluid and the structural subsystems, where the fluid subsystem is discretized using a discontinuous Galerkin finite-element method, while the structures solver uses a continuous Galerkin discretization. An explicit mapping is used as the primary mesh deformation algorithm. High-order time integration schemes are used by the coupled solver to march forward in time, and the space-time mesh of the fluid subsystem is adapted using output-based methods. The error estimates for the unsteady outputs are evaluated by calculating the uncoupled, unsteady adjoint of the fluid subsystem. Adaptive meshing is used to demonstrate the importance of mesh-motion errors on output convergence and a comprehensive analysis is conducted to control such errors arising from the mesh deformation algorithm. The benefits of adaptive meshing are demonstrated on a cantilevered Euler-Bernoulli beam placed in a low-Reynolds number flow and on a two-dimensional pitching-plunging airfoil in a high-Reynolds number flow.

Nomenclature

$\mathbf{u}$	=	Conservative state vector
$\mathbb{R}$	=	Real coordinate space
$\hat{s}$	=	Number of states
d	=	Dimension
$\vec{\mathbf{F}}$	=	Flux vector
$\mathcal{G}$	=	Mapping between physical and reference domain
$\underline{G}$	=	Spatial Jacobian of the mapping
g	=	Determinant of the mapping

Presented as Paper 2019-3056 at the 2019 AIAA Aviation Forum, Dallas,TX, June 17–21, 2019

[*] Ph.D. Candidate, AIAA Student Member

[†] Associate Professor, AIAA Senior Member

[‡] Clarence L. “Kelly” Johnson Collegiate Professor of Aerospace Engineering, Fellow AIAA

$\bar{\mathbf{R}}$	=	Strong-form unsteady residual vector
$\bar{\mathbf{r}}$	=	Discrete spatial residual vector
$\bar{\mathbf{R}}$	=	Unsteady residual
$\mathbf{r}$	=	Discrete spatial residual
$\mathbf{M}$	=	Mass matrix
$\mathbf{C}$	=	Damping matrix of the structure
$\mathbf{K}$	=	Stiffness matrix of the structure
$\hat{\mathbf{F}}$	=	Vector of external forces on the structure
P	=	Blending polynomials for mesh deformation
r	=	Normalized radial distance
$\mathbf{t}$	=	Traction acting on the structure
$\bar{J}$	=	Unsteady output of interest
$\mathbf{\Psi}$	=	Adjoint
$\mathcal{L}$	=	Lagrangian
T	=	Final time of an unsteady simulation
α	=	Pitch degree of freedom
h	=	Plunge degree of freedom
c	=	Chord length
p	=	Spatial order of approximation
Re	=	Reynolds number
M	=	Mach number
M_f	=	Flutter Mach number
M_∞	=	Free-stream Mach number
f_{tot}	=	Growth factor
L	=	Lift
D	=	Drag
ρ	=	Density
A	=	Cross section area
E	=	Elastic modulus

Superscripts

f	=	Fluid subsystem
s	=	Structural subsystem

I. Introduction

Numerical simulation of fluid structure interaction (FSI) is a complex and challenging problem that finds relevance across disciplines. The focus of this paper is on the aeronautical applications of FSI in which numerical simulation can be used to improve the understanding and prediction of coupled instabilities observed in aircraft [1]. The interaction between the two systems is non-linear and involves multiple scales, thereby making the coupled system challenging to solve. Many approaches have been suggested for simulating fluid-structure interaction [2]. Numerical approaches for solving the coupled fluid-structure system can be broadly divided into two: monolithic and partitioned. The monolithic approach [3] is a fully-coupled approach where the two systems are solved simultaneously. This approach combines both systems into one large system of equations and solves them simultaneously, which often leads to accurate results but requires significant implementation effort and uses less efficient solvers. The second approach, generally referred to as the partitioned approach, uses two separate solvers to solve the subsystems and then couples them by communication between the solvers. This method facilitates software modularity and mathematical modelling. This work implements a high-order partitioned approach based on the Implicit Explicit Runge Kutta scheme (IMEX) presented in the work of Van Zuijlen et al. [4], which coupled the two subsystems in a high-order manner without sub-iterations.

The main motivation for developing a high-order FSI solver is to be able to accurately predict coupled outputs like lift, drag and structural deformation. Increasing the accuracy of a simulation by refining the entire space-time mesh makes the simulation computationally expensive and inefficient. A better strategy is to evaluate errors in the output of interest and to adapt the mesh in the regions of space-time which contribute most to the error. Such methods are called output-based adaptive methods. They offer a systematic approach for identifying regions of the domain that require more resolution for the prediction of scalar outputs of interest [5, 6].

Fluid structure interaction problems require a framework that can handle the evolving fluid domain imposed by the deforming structure. A common way of simulating such a problem is through Arbitrary Lagrangian Eulerian (ALE) methods. In the ALE framework, the fluid mesh can move and deform arbitrarily, with no correlation to the flow, which is useful for modeling problems in which objects move or deform. The ALE method uses a map between the moving physical domain and a static reference domain and solves transformed equations on the reference domain [7]. Fluid simulations based on the ALE formulation can use $r-$adaptation to obtain an optimized mesh close to solid boundaries [8] or use a mesh deformation technique to conform the fluid mesh to the moving boundaries. Several mesh deformation methods exist in the literature [9], and these can be classified into two main categories, 1) physical-analogy based techniques and 2) interpolation based techniques. Physical analogy methods [10] typically model each edge of the mesh as a spring with an individual stiffness value. On the other hand, interpolation based methods compute the movement of grid nodes as a function of boundary nodes, with no attached physical meaning. Radial basis function interpolation [11] and inverse distance methods [12] are some examples of interpolation-based techniques. An alternative to these methods is the use of an explicit expression for the mapping between the reference and physical domains, as introduced by Persson et al. [13]. The present paper considers primarily the mesh deformation techniques based on the explicit mapping. The mesh deformation strategy is chosen for its suitability for high-order simulations, where smoothness of the mapping is necessary for accuracy and optimal convergence. With the primary application of the paper being in aeroelasticity, the problems that the paper tackles are ones undergoing small deformation. Displacements are considered to be small, when ALE can be used with a single mesh, without the need for dynamic re-meshing.

Moving mesh nodes to accommodate the deformation at the boundary introduces errors into a fluid simulation. These errors are primarily due to the introduction of mesh velocities and distortion of elements, which lead to accuracy losses. One of the most common errors arises from a non-constant Jacobian of the ALE mapping. By virtue of the state representation in an ALE formulation, a non-constant (and non-polynomial) Jacobian leads to a constant free stream state not being exactly preserved. A solution for this inaccuracy, introduced by Lesoinne and Farhat [14], is known as enforcement of the Geometric Conservation Law (GCL), which conserves uniform flow by linearizing the mapping from the reference to the physical domain [15]. Other GCL formulations are possible, including ones that use an auxiliary equation to compute corrections to the GCL error [7].

In this paper, a mesh-motion algorithm based on explicit mappings is optimized to reduce discretization errors arising from mesh motion using error estimates and rates of output convergence. Using optimized mesh-motion parameters, a high-order fluid structure interaction solver is developed for coupled high-order simulations. Furthermore, the adaptive techniques are applied to one dimensional and two dimensional cases to validate the output-based adaptation strategy, and to demonstrate a reduction in computational cost. The outline of the remainder of the paper is as follows. Section II reviews the governing equations. Section III reviews the spatial and temporal coupling algorithm for the partitioned approach. Section IV reviews the error estimation and mesh adaptation strategies used in this work. Finally, Section V outlines the results generated using these methods, demonstrating the impact of mesh-motion on output convergence and demonstrating the use of output-based mesh adaptation in efficiently reducing the errors generated in an FSI simulation.

II. Governing Equations

A. Fluid Subsystem

The fluid subsystem is governed by the Navier-Stokes equations, given by

$$\left.\frac{\partial \mathbf{u}}{\partial t}\right|_x + \nabla \cdot \vec{\mathbf{F}}(\mathbf{u}, \nabla \mathbf{u}) = \mathbf{0}, \quad \vec{\mathbf{F}} = \vec{\mathbf{F}}^{i}(\mathbf{u}) - \vec{\mathbf{F}}^{\nu}(\mathbf{u}, \nabla \mathbf{u}), \tag{1}$$

where $\mathbf{u}(\vec{x}, t) \in \mathbb{R}^{\hat{s}}$ is the conservative state vector, $\vec{x} \in \mathbb{R}^{d}$ is the spatial coordinate, $t \in \mathbb{R}$, and $\vec{\mathbf{F}}^{i}$ and $\vec{\mathbf{F}}^{\nu}$ are the inviscid and viscous fluxes, respectively. In the case of a stationary domain, the fluid equations are solved numerically in the Eulerian frame of reference, where the computational grid is fixed and the fluid moves with respect to the grid. However, numerical simulations of fluid dynamics involving a moving and/or deforming domain, such as in the case of FSI, face issues due to the lack of precise interface definition and under-resolved flow features when solved in the Eulerian frame of reference. The Lagrangian approach, on the other hand, faces problems dealing with large distortions of the computational domain. To resolve these issues, an alternate method, the Arbitrary Lagrangian Eulerian approach, has been introduced and is applied in the present work.

B. Arbitrary Lagrangian-Eulerian Formulation

The Arbitrary Lagrangian Eulerian (ALE) approach combines advantages of both the Eulerian and Lagrangian approaches. In this method, the deformable physical domain is mapped to a fixed referenced domain by a time dependent mapping. A simple and effective ALE method for DG was introduced by Persson et al. [7] and a similar approach is followed in this work [16].

Let $\vec{x} = \mathscr{G}(\vec{X}, t)$ represent the one-to-one time-dependent mapping between the physical volume and the reference volume. Each point $\vec{X}$ in the static reference domain is mapped to a corresponding point $\vec{x}$ in the physical domain, based on the desired deformation of the mesh. The spatial Jacobian of the mapping, represented by $\underline{G}$, and the mapping velocity, $\vec{v}_X$, are given by

$$\underline{G} = \nabla_X \mathscr{G}, \quad \vec{v}_X = \left.\frac{\partial \mathscr{G}}{\partial t}\right|_X. \tag{2}$$

Let g=det($\underline{\mathrm{G}}$). The corresponding Navier-Stokes equations in the reference frame can be written as

$$\left.\frac{\partial \mathbf{u}_X}{\partial t}\right|_X + \nabla_X \cdot \vec{\mathbf{F}}_X(\mathbf{u}_X, \nabla_X \mathbf{u}_X) = \mathbf{0}, \quad \vec{\mathbf{F}}_X = \vec{\mathbf{F}}^{i}_X(\mathbf{u}_X) - \vec{\mathbf{F}}^{\nu}_X(\mathbf{u}_X, \nabla_X \mathbf{u}_X), \tag{3}$$

where the transformed vectors, derivatives, and fluxes in the reference frame are given by

$$\mathbf{u}_X = g\mathbf{u}, \tag{4}$$

$$\nabla_x \mathbf{u} = \nabla_X (g^{-1}\mathbf{u}_X)\underline{G}^{-T} = (g^{-1}\nabla_X \mathbf{u}_X - \mathbf{u}_X \nabla_X (g^{-1}))\underline{G}^{-T}, \tag{5}$$

$$\vec{\mathbf{F}}_X^i = g\underline{G}^{-1}\vec{\mathbf{F}}^i - \mathbf{u}_X \underline{G}^{-1}\vec{v}_X, \qquad \vec{\mathbf{F}}_X^v = g\underline{G}^{-1}\vec{\mathbf{F}}^v. \tag{6}$$

The explicit definition of the grid velocities requires knowledge of the mapping in advance, which is not available when solving for the coupled system in a partitioned approach. This issue is circumvented by assuming the structural subsystem to be stationary when solving for the unsteady fluid subsystem. This assumption helps in providing the mapping, used to define the grid position and velocities, which is lagging in time and is corrected using the predictor-corrector step, as described in the temporal coupling section. To simplify the notation, in the following sections, we drop the X subscript from the state and fluxes when referring to the fluid system, which is solved on a reference static domain but is mapped to the physical domain for coupling with the structural solver.

C. Spatial Discretization

To discretize the state equations (Eq. 3), a discontinuous Galerkin (DG) finite-element method is used in space. As a finite-element method, DG approximates the state **u** in functional form using linear combinations of basis functions on each element. No continuity constraints are imposed between adjacent elements. Denoting by T_h the set of N_e elements in a non-overlapping tessellation of the domain Ω, the state on element e, Ω_e, is approximated as

$$\mathbf{u}_h(\vec{x}(\vec{\xi}))\Big|_{\Omega_e} = \sum_{n=1}^{N_p} \mathbf{U}_{en}\phi_{en}(\vec{x}(\vec{\xi})). \tag{7}$$

In this equation, N_p is the number of basis functions per element, $\mathbf{U}_{en}$ is the vector of q coefficients for the n^{th} basis function on element e: $\phi_{en}(\vec{x}(\vec{\xi}))$, and q is the state rank. $\vec{x}$ denotes the global coordinates, and $\vec{\xi}$ denotes the reference-space coordinates in a master element. Formally, $\mathbf{u}_h \in \mathcal{V}_h = [\mathcal{V}_h]^q$, where, if the elements are not curved, $\mathcal{V}_h = \left\{u \in L_2(\Omega) : u|_{\Omega_e} \in \mathcal{P}^p \ \forall \Omega_e \in T_h\right\}$, and $\mathcal{P}^p$ denotes polynomials of order p on each element. With the spatial discretization described above, the governing equations can written in an abbreviated form as

$$\bar{\mathbf{R}}^f = \mathbf{M}^f \frac{\mathrm{d}\mathbf{U}^f}{\mathrm{d}t} - \mathbf{r}^f = \mathbf{0}, \tag{8}$$

where $\mathbf{r}^f$ is the (negative) discrete spatial residual vector, $\bar{\mathbf{R}}^f$ is the temporally strong-form unsteady residual, and the f superscript denotes that these equations apply to the fluids subsystem.

D. Structural Subsystem

The structural subsystem is governed by a set of partial differential equations derived on the basis of continuum modelling with a general non-isotropic constitutive law. For the initial development of the coupled solver, a linear structure model is assumed. Discretizing the governing equations in space using a continuous finite-element method, the semi discrete form of the equation is given by

$$\mathbf{M}\ddot{\mathbf{u}}^{s} + \mathbf{C}\dot{\mathbf{u}}^{s} + \mathbf{K}\mathbf{u}^{s} = \hat{\mathbf{F}}, \tag{9}$$

where $\mathbf{u}^s$ represents the vector of displacements, $\mathbf{M}$ is the mass matrix, and $\mathbf{K}$ is the stiffness matrix of the structure. $\mathbf{C}$ denotes the internal or external damping in the structure, and $\hat{\mathbf{F}}$ represents the vector of external forces acting on the structure. The semi-discrete form can be re-written as a system of two first-order ordinary differential equations

$$\begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \dot{\mathbf{U}}^{s} + \begin{bmatrix} \mathbf{C} & \mathbf{K} \\ -\mathbf{1} & \mathbf{0} \end{bmatrix} \mathbf{U}^{s} = \begin{bmatrix} \hat{\mathbf{F}} \\ \mathbf{0} \end{bmatrix} \quad \text{where} \quad \mathbf{U}^{s} = \begin{bmatrix} \dot{\mathbf{u}}^{s} \\ \mathbf{u}^{s} \end{bmatrix}, \tag{10}$$

where $\mathbf{1}$ denotes the identity matrix. The governing equations can be written in an abbreviated form as:

$$\bar{\mathbf{R}}^{s} = \mathbf{M}^{s}\frac{\mathrm{d}\mathbf{U}^{s}}{\mathrm{d}t} - \mathbf{r}^{s} = \mathbf{0}, \tag{11}$$

where $\mathbf{r}^s$ is the (negative) discrete spatial residual vector, and $\bar{\mathbf{R}}^s$ is the strong-form unsteady residual. The superscript s denotes that these equations apply to the structures subsystem.

III. Fluid-Structure Coupling

Solving the coupled fluid-structure problem using a high-order partitioned approach intrinsically requires separate high-order solvers for both the fluid and the structural subsystems. The coupling then occurs by communication between the two solvers. The Discontinuous Galerkin (DG) formulation of the Navier-Stokes equations coupled with high-order time stepping schemes such as ESDIRK4/ DIRK4 allows for the solution of the fluid subsystem with high-order accuracy in both space and time. Similarly, high-order elements in the continuous Galerkin formulation of the structural governing equations, coupled with high-order time stepping provide us with high-fidelity solutions of the structural subsystem. Thus, development of a high-order partitioned solver comes down to the order of coupling between the subsystems.

A. Spatial Coupling

The partitioned approach for FSI allows us to use a separate mesh discretization for each subsystem which offers the advantage of resolving regions of complex flow in the fluids or large displacement in the structures. This may result in meshes for the coupled problem which may not coincide at the interface. Thus, a spatial coupling algorithm is required to interpolate interface displacement and velocities from the structural mesh to the fluids and loads from the fluid mesh to the structures. For one-dimensional test cases, cubic splines provide high-order of accuracy for interpolation and have been used for the spatial coupling in this work. When dealing with complex structures for multi-dimensional problems, radial basis function (RBF) or nearest neighbor interpolation will be considered based on previous studies. [17]

B. Mesh Deformation

1. Explicit Mapping

For deforming domains, the ALE formulation of the Navier-Stokes equations requires a mapping between the reference and the deformed physical mesh. The mapping interpolates the boundary displacements to the interior of the fluid mesh. The explicit mapping presented by Persson et al. [13] does not require solving a system of equations for deforming the volume. It uses explicit expressions for the mapping that blend the motion, smoothly reducing it to the identity mapping away from the boundary. The mesh-motion algorithm described for the membrane [13] is employed for accommodating the deformation in the beam, the first problem considered in this paper, while the following mesh-deformation algorithm, described below, is used for the deformation in an airfoil, the second problem considered in this paper. For any rigid body deformation, the mesh motion algorithm divides the entire spatial domain into two regions based on inner and outer radii of the blending region. The region extending up to the inner radius from the center of the deformation marks the region of rigid deformation. Within this region, any deformation provided by the user is

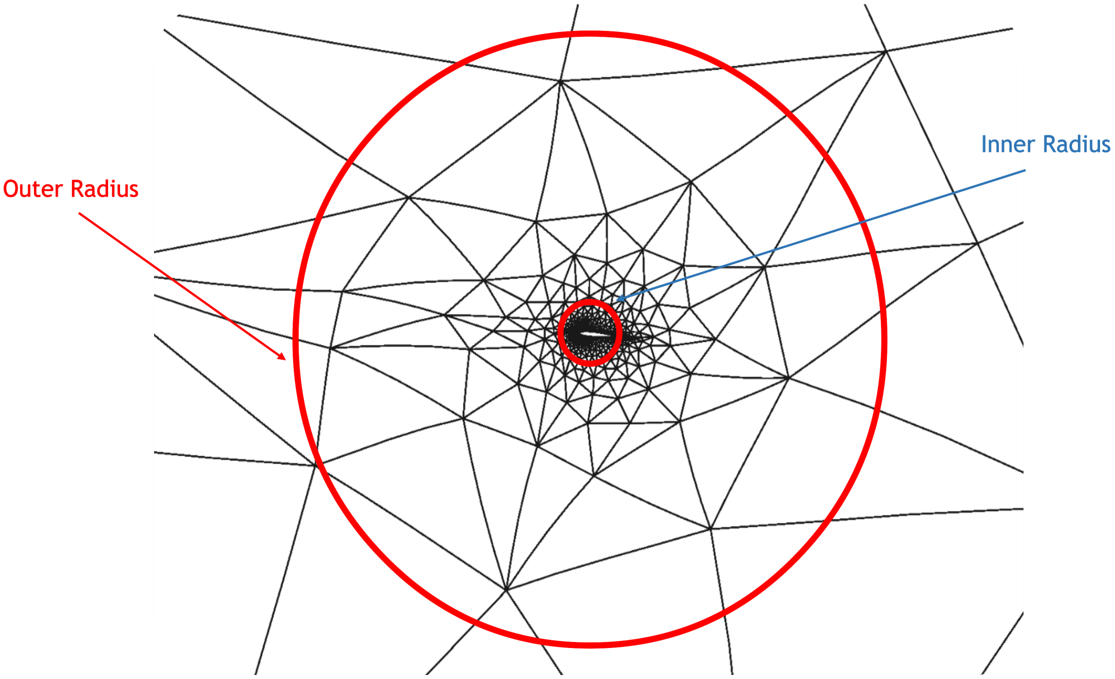


Fig. 1 Example of inner and outer radius of the blending region for the explicit mapping mesh motion.

applied to all of the mesh elements without any blending. The presence of the rigid region prevents errors such as mesh element inversion in highly-stretched elements, which face such errors when placed in a blending region. The blending region, which exists between the inner and outer radii from the center of the motion, uses a polynomial function to blend the deformation radially such that the deformation goes to zero at the outer radius. To achieve blending of the motion, polynomial blending functions $P(r)$ of odd degrees are used. The three different blending polynomials analyzed in this study are cubic, $P(r) = 3r^2 - 2r^3$, quintic, $P(r) = 10r^3 - 15r^4 + 6r^5$, and septic, $P(r) = -20r^7 + 70r^6 - 84r^5 + 35r^4$, where r is the normalized radial distance from the inner radius. Figure 1 shows the deformed mesh for an airfoil undergoing rigid-body pitch motion. The inner and outer radii are placed at one and ten chords away from the center of motion, which is at the quarter chord of the airfoil.

2. Errors Generated by Mesh Deformation

Mesh deformation algorithms generate spatial and temporal errors in a simulation by two main mechanisms. First, the spatial dependence of the motion required to accommodate the prescribed mesh deformation, provided by the structural subsystem, causes the elements in the fluid mesh to distort, which reduces their spatial approximation capability [18]. This is due to the reference-space polynomial approximation space no longer being polynomial after a nonlinear mapping into global space. These errors exist for both steady and unsteady simulations as they exist in the presence of any non-trivial deformation of the mesh. Second, for unsteady simulations involving mesh deformation, grid velocities are also introduced in the mesh, which are arbitrary in spatial variation and depend on the mesh motion parameters. Under-integration of terms in the weak form involving mesh motion, due to the use of finite quadrature and increased complexity of the integrands affected by mesh motion derivatives and velocities, introduces spatial and temporal errors in the simulation. In this work, a qualitative study is conducted to study the spatial errors generated only from distortion of mesh elements for a steady fluid simulation on a static but deformed mesh. A geometric conservation law, as introduced earlier, is not explicitly enforced in this work. Previous research [7, 16] discussing the relevance of the GCL has shown that for outputs on adapted meshes and even for outputs corrected by error estimates, the GCL is not critical. Even though the GCL explicitly enforces conservation, the errors made without it are on the order of discretization errors, which reduce with further refinement, making the ALE scheme asymptotically conservative. From the experience of the authors, the complexity of its implementation (particularly in the adjoint discretization) outweigh its benefits, and it is thus not implemented in this work.

C. Implicit Explicit Runge Kutta Scheme

The time integration method used in this study is the Implicit Explicit Runge Kutta scheme [19]. IMEX is a high-order time integration scheme developed for stiff time marching problems where the rate of change of the state can be split into a stiff and non-stiff component. The stiff component of the state time derivative, a term that can lead to rapid variation in the solution, is then solved implicitly in time, while the non-stiff component is solved explicitly. To demonstrate the scheme, consider a semi-discretized multiphysics equation,

$$\mathbf{M}\dot{\mathbf{u}} = \mathbf{r}(\mathbf{u}, t) = \mathbf{f}(\mathbf{u}, t) + \mathbf{g}(\mathbf{u}, t), \tag{12}$$

where $\mathbf{M}$ is the mass matrix, $\mathbf{u}$ is the state vector, and $\mathbf{r}$ is the residual, which is split into $\mathbf{f}$ and $\mathbf{g}$, the non-stiff and stiff component respectively. $\mathbf{f}$ is integrated with an n-stage explicit Runge-Kutta scheme and $\mathbf{g}$ is integrated with an n-stage diagonally-implicit Runge-Kutta scheme. The equations for an update in one time step are

$$\mathbf{u}_{n,j} = \mathbf{u}_{n-1} + \sum_{p=1}^{j-1} \hat{a}_{jp}\hat{k}_{n,p} + \sum_{p=1}^{j} a_{jp}k_{n,p} \tag{13}$$

$$\mathbf{M}\hat{k}_{n,j} = \Delta t_n \mathbf{f}(\mathbf{u}_{n,j}, t_{n-1} + \hat{c}_j \Delta t_n) \tag{14}$$

$$\mathbf{M}k_{n,j} = \Delta t_n \mathbf{g}(\mathbf{u}_{n,j}, t_{n-1} + c_j \Delta t_n) \tag{15}$$

$$\mathbf{u}_n = \mathbf{u}_{n-1} + \sum_{p=1}^{s} \hat{b}_p \hat{k}_{n,p} + \sum_{p=1}^{s} b_p k_{n,p} \tag{16}$$

The IMEX scheme considered in this paper consists of a fourth-order explicit Runge-Kutta (ERK) and a stiffly-accurate fourth order, explicit, first stage, singly-diagonally implicit Runge-Kutta (ESDIRK) scheme [20] .

D. Temporal Coupling

The coupled FSI problem can be represented in semi-discrete form as

$$\mathbf{M}\dot{\mathbf{u}} = \mathbf{r}(\mathbf{u}), \tag{17}$$

$$\mathbf{u} = \begin{bmatrix} \mathbf{u}^f \\ \mathbf{u}^s \end{bmatrix}, \quad \mathbf{r} = \begin{bmatrix} \mathbf{r}^f(\mathbf{u}^f; \mathbf{z}(\mathbf{u}^s)) \\ \mathbf{r}^s(\mathbf{u}^s; \mathbf{t}(\mathbf{u}^f)) \end{bmatrix}, \quad \mathbf{M} = \begin{bmatrix} \mathbf{M}^f & 0 \\ 0 & \mathbf{M}^s \end{bmatrix}, \tag{18}$$

where $\mathbf{u}$ is a combined vector of the fluid and structural states, and $\mathbf{z}$ and $\mathbf{t}$ denote the terms in the residual responsible for the coupling of the two subsystems. The unsteady residual of the structural subsystem, given in Eq. 11 may be separated as

$$\mathbf{r}^s(\mathbf{u}^s; \mathbf{t}(\mathbf{u}^f)) = \mathbf{r}^{ss}(\mathbf{u}^s) + \mathbf{r}^{sf}(\mathbf{t}(\mathbf{u}^f)). \tag{19}$$

The first term represents the effect of the current structural state, while the second term represents the effect of the traction, $\mathbf{t}$, from the fluid. Since the second term is linear in $\mathbf{t}$, a predicted value of the traction is introduced, $\tilde{\mathbf{t}}$, as presented in Froehle and Persson [21]. Thus, the residual can be re-written as

$$\mathbf{r}^s(\mathbf{u}^s;\mathbf{t}(\mathbf{u}^s)) = \mathbf{r}^s(\mathbf{u}^s;\tilde{\mathbf{t}}) + \mathbf{r}^{sf}(\mathbf{t}(\mathbf{u}^f) - \tilde{\mathbf{t}}). \tag{20}$$

Using the above formulation, Eq. 17 can be split as

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = \begin{bmatrix} 0 \\ \mathbf{r}^{sf}(\mathbf{t}(\mathbf{u}^f) - \tilde{\mathbf{t}}) \end{bmatrix} + \begin{bmatrix} \mathbf{r}^f(\mathbf{u}^f;\mathbf{z}(\mathbf{u}^s)) \\ \mathbf{r}^s(\mathbf{u}^s;\tilde{\mathbf{t}}) \end{bmatrix}. \tag{21}$$

With the introduction of a predictor, the coupled problem has been suitably modified into a form where the high-order IMEX scheme can be applied. As already mentioned, in the IMEX scheme the non-stiff (first) term is integrated explicitly and the stiff (second) term is integrated implicitly in time. However, the scheme differs slightly from IMEX as the evaluation of the explicit terms $\mathbf{r}^{sf}$ is avoided and instead the stage flux is updated for the structure equation using the corrected value of the coupling $\mathbf{t}(\mathbf{u}^f)$. To solve the subsystems implicitly, ESDIRK4, a fourth-order time accurate implicit scheme is employed. An explicit traction predictor, proposed by Van Zuijlen et al. [4], for the structural subsystem, at an implicit stage i is given by

$$\tilde{\mathbf{t}} = \sum_{j=1}^{i-1} \frac{\hat{a}_{ij} - a_{ij}}{a_{ii}} \mathbf{t}_j, \tag{22}$$

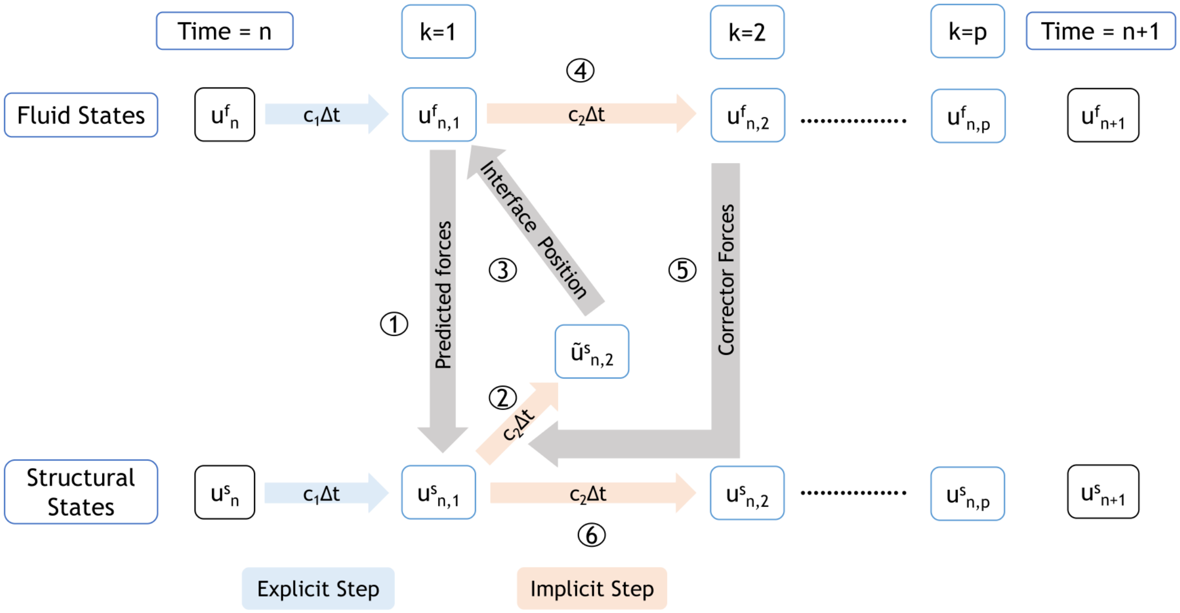


Fig. 2 IMEX algorithm for one coupled time step.

where $\hat{a}_{ij}$ and a_{ij} are the coefficients of the explicit and implicit Runge-Kutta integration schemes, respectively.

Figure 2 summarizes a single time step of the coupled solver. The coupled FSI problem uses a block Gauss-Seidel partitioning, in which the structural subsystem uses the explicit traction predictor (Step 1) and is integrated first, in an implicit manner (Step 2) . The spatial coupling algorithm transfers the predicted interface data to the fluid subsystem (Step 3). The predicted interface data are used by the mesh-motion algorithm to deform the fluid mesh and are then implicitly integrated (Step 4). Corrected traction obtained from the fluid solve is lastly used to correct the structural subsystem (Steps 5 and 6).

IV. Output-Based Mesh Adaptation

In the coupled FSI problem, the numerical error in the output results from discretization errors generated in both the fluid and structural subsystems. For this study, the errors generated in the structural subsystem are neglected and only the errors from the fluid discretization drive the adaptation process. To adapt the mesh, spatially discrete but continuous in time adjoints are used to estimate the error in the output of interest [22].

A. Continuous-in-Time Adjoint Evaluation

Consider an unsteady output of the form

$$\bar{J} = \int_0^T J(\mathbf{U}^f, \mathbf{U}^s, t)\, dt, \tag{23}$$

where J is a functional of the fluid, $\mathbf{U}^f$, and structural, $\mathbf{U}^s$, states. The continuous adjoints, $\mathbf{\Psi}^f$ and $\mathbf{\Psi}^s$, represent the sensitivity of the output to perturbations in the unsteady residuals $\bar{\mathbf{R}}^f$ (Eq. 8) and $\bar{\mathbf{R}}^s$ (Eq. 11), respectively. To derive the adjoint equations, a Lagrangian is defined as

$$\mathcal{L} = \bar{J} + \int_0^T \mathbf{\Psi}^{f\top} \bar{\mathbf{R}}^f\, dt + \int_0^T \mathbf{\Psi}^{s\top} \bar{\mathbf{R}}^s dt. \tag{24}$$

As the focus of this work is on the errors in the fluid discretization and thus the fluid adjoint, perturbations in the structural state are ignored and assumed to be independent of the perturbation in the fluid state. Substituting Eq. 23 into Eq. 24, integrating the second term by parts, and requiring stationarity of the Lagrangian with respect to the permissible state variations, which in this particular work is only in the fluid state, $\delta\mathbf{U}^f$, gives

$$\mathbf{\Psi}^{f\top} \mathbf{M}^f\, \delta\mathbf{U}^f \Big|_{t=T} - \mathbf{\Psi}^{f\top} \mathbf{M}^f\, \delta\mathbf{U}^f \Big|_{t=0} + \int_0^T \left[\frac{\partial J}{\partial \mathbf{U}^f} - \frac{\mathrm{d}\mathbf{\Psi}^{f\top}}{\mathrm{d}t} \mathbf{M}^f - \mathbf{\Psi}^{f\top} \frac{\partial \mathbf{r}^f}{\partial \mathbf{U}^f} \right] \delta\mathbf{U}^f\, dt = \mathbf{0}. \tag{25}$$

The middle term at $t = 0$ drops out since the initial condition on the primal fully specifies the state there, so $\delta\mathbf{U}^f = 0$ at $t = 0$. By transposing the time integrand and noting that the mass matrix is symmetric, the remaining terms yield the adjoint differential equation

$$-\mathbf{M}^f \frac{\mathrm{d}\mathbf{\Psi}^f}{\mathrm{d}t} - \frac{\partial \mathbf{r}^f}{\partial \mathbf{U}^f}^{\top} \mathbf{\Psi}^f + \frac{\partial J}{\partial \mathbf{U}^f}^{\top} = \mathbf{0}, \tag{26}$$

and the terminal condition

$$\mathbf{\Psi}^f(T) = \mathbf{0}. \tag{27}$$

Note that the mass matrix is non-singular. The adjoint equation is solved backward in time, starting from the given terminal condition. The time integration scheme used for both the primal and the adjoint equation for this study is ESDIRK4, but other time schemes can be used as well.

B. Error Estimation

The unsteady adjoint can be used to evaluate the error in the output of interest through the adjoint-weighted residual [5]. Let $\mathbf{U}_H^f$ be the approximate fluid solution obtained from the current space-time approximation space, denoted by subscript H, and $\mathbf{\Psi}_h^f$ be the fluid adjoint in the fine space denoted by h. The error in the output is defined as:

$$\delta \bar{J} = \bar{J}_H(\mathbf{U}_H^f) - \bar{J}_h(\mathbf{U}_h^f) \approx \frac{\partial \bar{J}_h}{\partial \mathbf{U}_h^f}\delta\mathbf{U}_h^f \approx -\int_0^T \mathbf{\Psi}_h^{f\top} \bar{\mathbf{R}}_h^f(\mathbf{U}_H^f)dt, \tag{28}$$

where $\delta\mathbf{U}_h^f = \mathbf{U}_h^{f,H} - \mathbf{U}_h^f$ is the primal error in the fluid state, and $\mathbf{U}_h^{f,H}$ is the injected solution from space H to h. The exact discrete unsteady adjoint, which is unavailable, is approximated in a finer space by increasing the degrees of freedom in the spatial discretization.

C. Mesh Adaptation

Unsteady error estimates in the space-time fluid mesh guide the adaptation process. Space-time elements selected for refinement or coarsening are chosen based on two factors: 1) the estimated error in the space-time element, and 2) the computational cost of refinement. These two aspects are combined into an adaptive indicator called the "figure of merit". The figure of merit is defined as the element error eliminated by refinement divided by the degrees of freedom introduced by the refinement. The cost is defined by the total degrees of freedom,

$$C \equiv C^{\text{space}} C^{\text{time}} \quad , \quad C^{\text{space}} \equiv \sum_{e=1}^{N_e} n(p_e) \quad , \quad C^{\text{time}} \equiv N_t n_r, \tag{29}$$

where $n(p_e)$ is the number of spatial degrees for an element of order p_e, N_t is the number of time steps, and n_r is the number of temporal degrees of freedom, i.e., system solves, per time step. For example, the five-stage ESDIRK4 method has $n_r = 5$. In this work, the adaptive strategy refines in space by hanging node refinement (h-adaptation) or spatial order refinement (p-adaptation) and in time by reducing the uniform time step.

V. Numerical Results

The focus of this section is two-fold: First, the impact of mesh motion algorithms on a high-fidelity fluid simulation with a deforming domain is studied. The existence of an optimum spatial location of the blending region in a mesh motion algorithm is investigated for two separate cases. Second, the high-order FSI solver is validated and the effectivity of the error estimate and the efficiency of the adapted space-time meshes are demonstrated. The two test cases chosen to showcase the benefits of adaptivity in aeroelastic problems are accurate drag estimation on a cantilevered beam subjected to a low Reynolds number flow, and accurate flutter prediction on a pitching-plunging airfoil in a high Reynolds number flow. The impact of mesh motion on FSI simulations is further highlighted in the flutter prediction study.

A. Optimizing Mesh Motion Algorithms

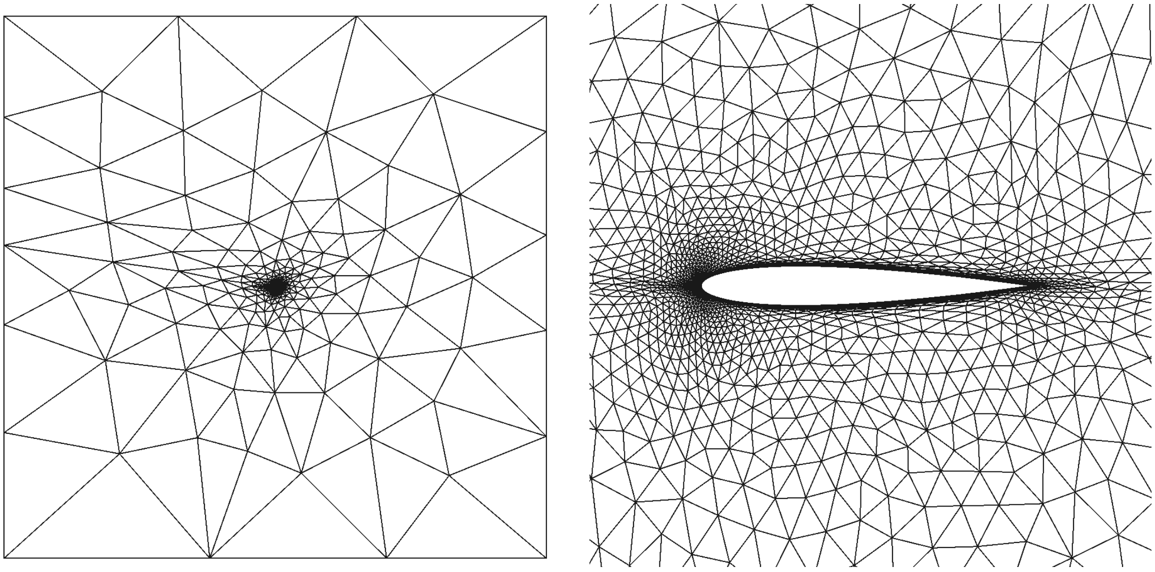
In this section, the impact of mesh motion algorithms on a high-fidelity fluid simulation with deforming domains is studied on an unstructured mesh. Two separate cases have been designed for quantifying and analyzing the errors arising from mesh deformation. Firstly, a free-stream preservation test, which is widely used for highlighting the impact of non-linear mapping in a free stream is modified to quantify the errors generated by the mesh deformation. In this case, the impact of the mesh deformation is characterized by entropy generation. Secondly, a more practical case of an airfoil undergoing rigid body deformation in a steady fluid flow is analyzed for engineering outputs of interest such as lift.

1. Free-stream preservation

Spatial and temporal errors generated in a fluid simulation with deforming domains arise from the corresponding space-time discretization and mesh deformation. As both sources of error propagate spatially and temporally in a simulation, it is difficult to separate the errors obtained from the output-based error estimate based on the source. However, for an arbitrary mesh motion applied to a free stream, the states are only contaminated by the errors arising from mesh deformation, as the space-time discretization without deformation conserves the free stream. Therefore, such a test acts as an ideal case to study the spatial and temporal distribution of mesh deformation errors. A steady and an unsteady mesh motion error quantification study are presented for a freestream undergoing mesh deformation. Error estimates of the entropy generated by the mesh deformation are used to demonstrate the existence of optimum blending region for the mesh motion algorithm.

Consider a uniform fluid flow around an airfoil placed centrally in square domain which spans $[-100c, 100c]$ in both dimensions, where c is the chord length of the airfoil. To simulate a uniform fluid flow, a coarse, unstructured, triangular mesh of 5489 elements is generated, as shown in Figure 3. The unstructured triangular fluid meshes used in this paper are generated using BAMG [23], an anisotropic 2D mesh generator. For fixed, user defined, degrees of freedom, BAMG is used to generate an h-adapted mesh, optimized using metric-based mesh adaptation. Freestream boundary conditions are applied at the farfield boundaries as well as on the airfoil. The airfoil boundary acts as the set of nodes where the deformation is prescribed but as the boundary condition on the airfoil is free-stream, it does not violate the preservation phenomenon. Despite this being a free-stream preservation test, the reason for choosing a viscous mesh is because most of the instabilities that occur in FSI simulations, such as flutter, occur at high Reynolds number flows. Thus, a mesh capable of simulating such as system is used to study the impact of mesh motion. Two degrees of freedom of the airfoil motion, the pitch, $\alpha(t)$, and plunge, $h(t)$, are prescribed using sinusoidal functions. An explicit mesh deformation algorithm, as mentioned in Section III, is applied to handle the deformations occurring in the fluid domain due to the moving airfoil. The mesh deformation algorithm depends on three variables: the inner radius, the outer radius, and the polynomial blending function, which are generally user-defined. The goal of this study is to investigate the existence of an optimal blending region and blending function for a viscous mesh undergoing arbitrary motion, with the goal of minimizing the error due to mesh deformation.

The study is conducted for both steady and unsteady cases. For the steady-state deformation, an inviscid simulation is conducted with $p = 3$ order polynomials with a constant pitch deformation of 5 degrees, centered at the quarter chord of the airfoil. The error estimate for the steady case is evaluated for an output defined as the domain integral of


(a)

(b)

Fig. 3 Unstructured viscous mesh for the free-stream preservation test.

the entropy, given by

$$\bar{J} = \int_{\Omega} E d\Omega. \tag{30}$$

Figure 4 compares the effect of order of blending polynomials on the entropy error estimate for inner and outer radii of $1c$ and $5c$. It shows that across different orders of dicretizations, increasing the order of blending polynomials leads to better representation of the deformation in the elements, which in turn leads to lower errors, provided a sufficiently high quadrature rule is used for integration when mesh motion is active. Figure 5(a) shows a contour representation of error estimates for varying inner and outer radii of the blending region with septic blending. The optimum inner and outer radii for the lowest error estimate come out to be $1c$ and $5c$, respectively. We can conclude that lowering the inner radius to be as low as possible while avoiding element inversion reduces the errors in the output due to mesh motion, because this results in a blending region located in the domain which is finer. Lowering the outer radius causes the blending to occur in a very small domain resulting in large gradients in the determinant of the mapping, g, within the elements, thereby causing higher errors. Similarly, increasing the outer radius of the blending region also results in an increase in the error because the number of elements affected by the blending region grows, as does the size of these elements. For this case, the optimum outer radius was identified at $5c$, which lies between the two extremes and leads to the lowest errors arising from mesh motion. The optimum parameters are specific to the case tested. However, similar results are expected for the blending region location for viscous meshes i.e, close to the deforming domain for the lowest mesh motion errors. A similar optimum blending region is observed for different amplitudes of the deformation and other degrees of freedom. Apart from error estimates, the absolute error in the average entropy in the entire domain is also studied for varying outer radii, as shown in Figure 5(b), which depicts a similar behaviour as the error estimate.

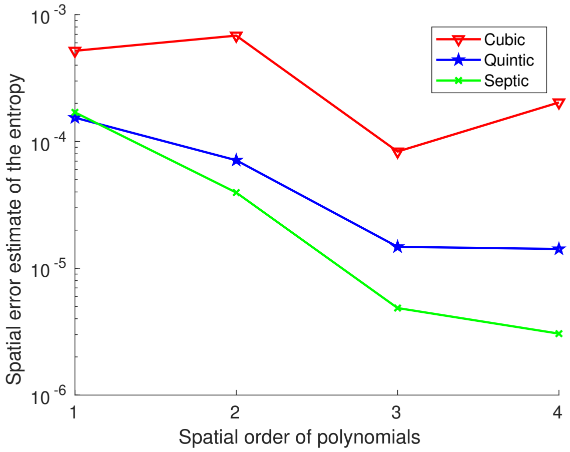


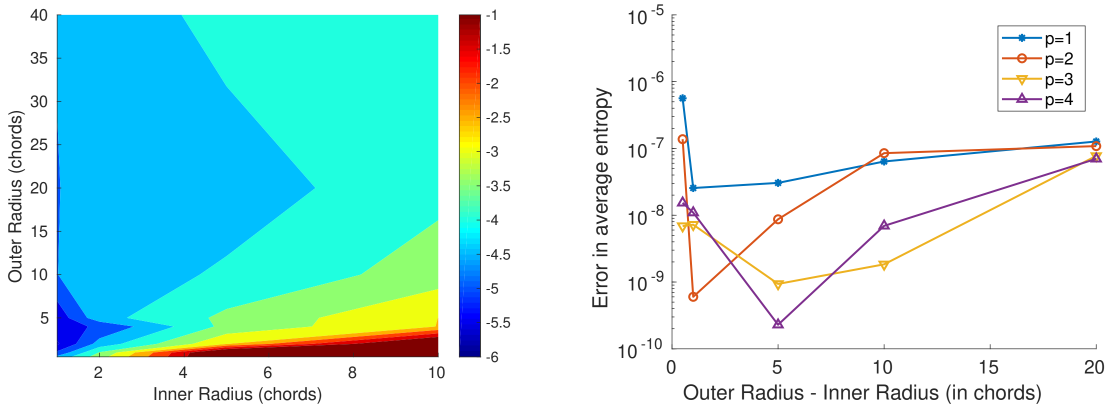
Fig. 4 Effect of the order of the blending polynomial on the error estimate.

2. *NACA 0012 airfoil with prescribed pitch deformation*

In this test case, mesh deformation is applied to an airfoil with prescribed rigid-body motion and the output of interest chosen for the study is the lift generated by the airfoil. The investigation is conducted for both viscous and inviscid flows. The main motivation behind this problem is to investigate the effect of the position of blending regions on the convergence of the output of interest. Secondly, the effect of different definitions of error estimates used for mesh adaption on the output convergence is also investigated for simulations undergoing mesh deformation.

Consider a NACA 0012 airfoil placed centrally in a circular mesh of radius 1000 chords. A pitch deformation of five degrees about the leading edge is provided to the airfoil using the mesh motion algorithm described in Section III. The effect of viscosity on the mesh deformation error is studied by considering two flow conditions, viscous and inviscid flow. For the viscous simulations, the Reynolds number is chosen to be Re=1000. The focus of this analysis is to quantify only the spatial errors generated by distortion of the mesh elements by the mesh motion algorithm. Therefore, all the simulations are steady in nature. The meshes used for this simulation use curved elements of order three, $q = 3$, to represent the airfoil geometry. Freestream boundary conditions are applied at the farfield boundaries and wall boundary conditions are applied at the airfoil boundary. The Mach number used for this analysis is $M = 0.345$. This subsonic Mach number is chosen based on the flutter estimation study conducted in the next subsection.

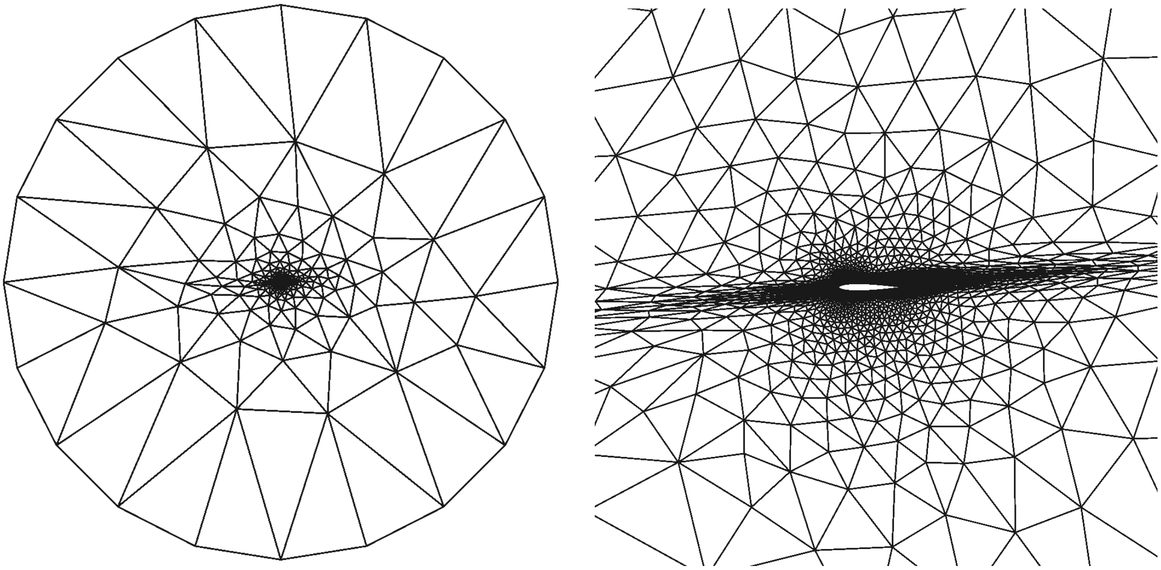
Multiple inner and outer radii combinations are used to vary the position of the blending region for the deformation to study the effect of the position of the blending region on the output convergence. For a particular blending region, the spatial errors in the simulation are quantified by comparing the output of interest against a reference



(a) Error estimates of the entropy, in log scale, for varying inner and outer radii with septic blending polynomials.

(b) Error in the average entropy for varying outer radius with a fixed inner radius of $1c$.

Fig. 5 Entropy error and error estimates.



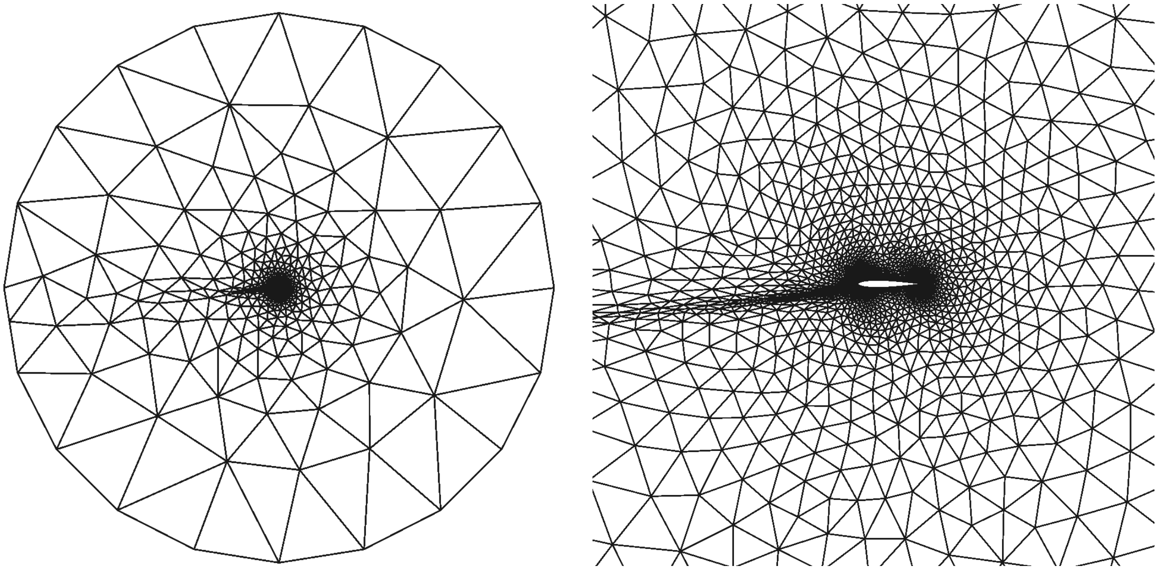
Refer

(a)

(b)

Fig. 6 ence mesh for lift evaluation for laminar flow.

case, which is unaffected by mesh motion. In the reference case, the desired angle of attack is achieved by changing the flow boundary conditions at the farfield without applying any mesh deformation to the airfoil. Figure 6 and Figure 7 present the meshes used for the laminar and inviscid reference cases, respectively. The reference lift is evaluated for a spatial discretization of order five, $p = 5$ using the mesh of around 5000 elements. Using an initial



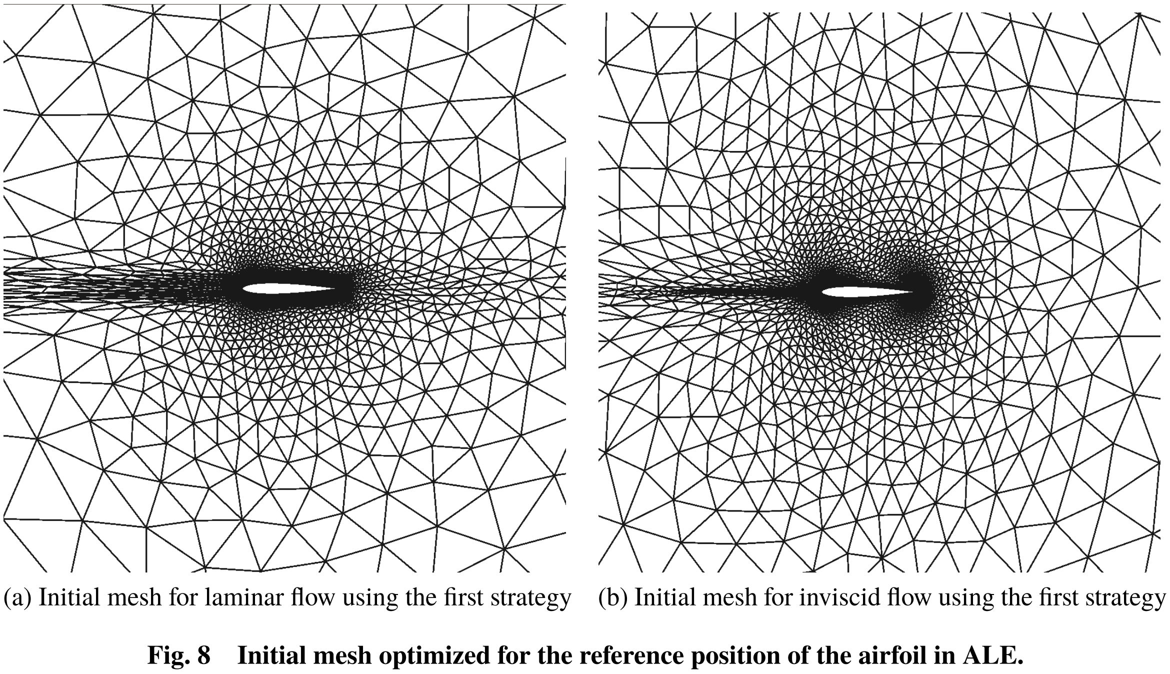
(a)

(b)

Fig. 7 Reference mesh for lift evaluation for inviscid flow.

mesh optimized for the flow configuration leads to better output convergence. However, in simulations involving FSI, the mesh deformation is often not known a priori to the user. Therefore, using a mesh optimized to reduce spatial discretization errors for the reference position of the airfoil can yield a good initial mesh. Using this strategy, a mesh optimized for the reference position of the airfoil in the ALE framework, where the airfoil is aligned with the flow, is obtained, as shown in Figure 8. An initial mesh optimized for the entire unsteady FSI simulation can also be used by incorporating the time evolution of the mapping, $\mathcal{G}$, in the error estimate [24, 25]. This concept has been used previously [22] for generating optimized meshes for unsteady gust simulations. Extending it to FSI simulation is the subject of ongoing research. Employing such a mesh, h optimized without mesh motion, can lead to different output error, depending on the mesh motion algorithm itself. Starting with this initial mesh, the mesh motion algorithm deforms the mesh for the various combinations of inner and outer radii. For the error analysis, discrete values of inner radius, $R_{\text{inner}} \in [1c, 5c, 20c, 40c, 100c]$ and blending distance, which is the distance between the inner and outer radius, $D_{\text{blending}} \in [1c, 5c, 20c, 40c, 100c, 400c, 800c]$, are used. As a single mesh is used for the error estimation for the various blending region, the initial spatial error arising only from the discretization is the same for all of the blending regions. To study the effect of mesh deformation, an output convergence study is conducted using output based mesh adaptation. Starting with uniform $p = 1$ elements in the entire domain, the initial mesh is adapted in spatial order by subjecting it to six cycles of p-adaptation. The growth factor is chosen to be two for each adaptation cycle.

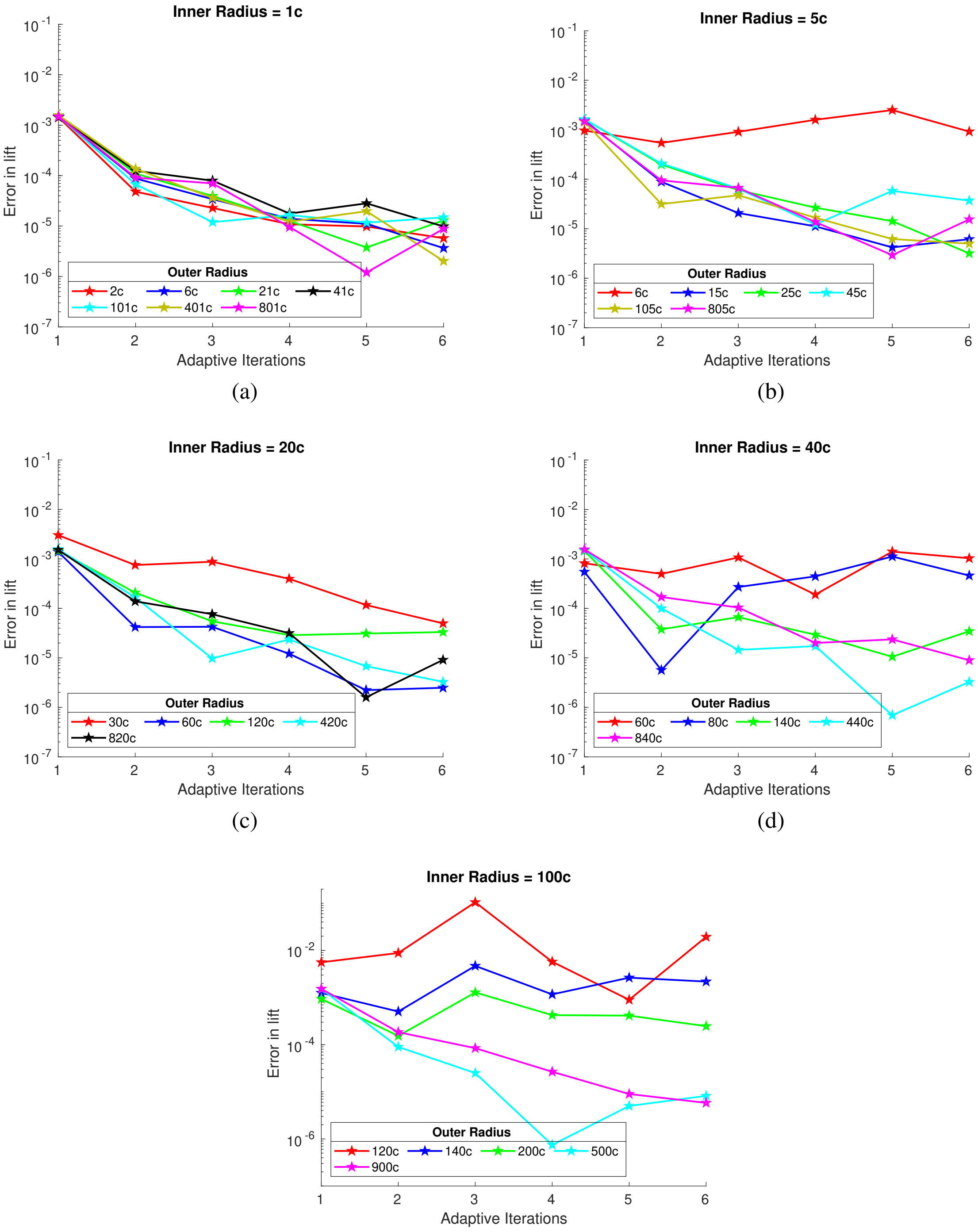
Figure 9 and Figure 10 show the output convergence for the different blending regions as a function of the adaptive



(a) Initial mesh for laminar flow using the first strategy (b) Initial mesh for inviscid flow using the first strategy

Fig. 8 Initial mesh optimized for the reference position of the airfoil in ALE.

iteration for the laminar and inviscid flows, respectively. The various plots track the absolute error in lift for a constant inner radius and varying outer radius. As the convergence of lift is not a monotonic function of the adaptive iteration, the sudden drop in the error of the output is not an indication of convergence but a sign of the output crossing the



(a)

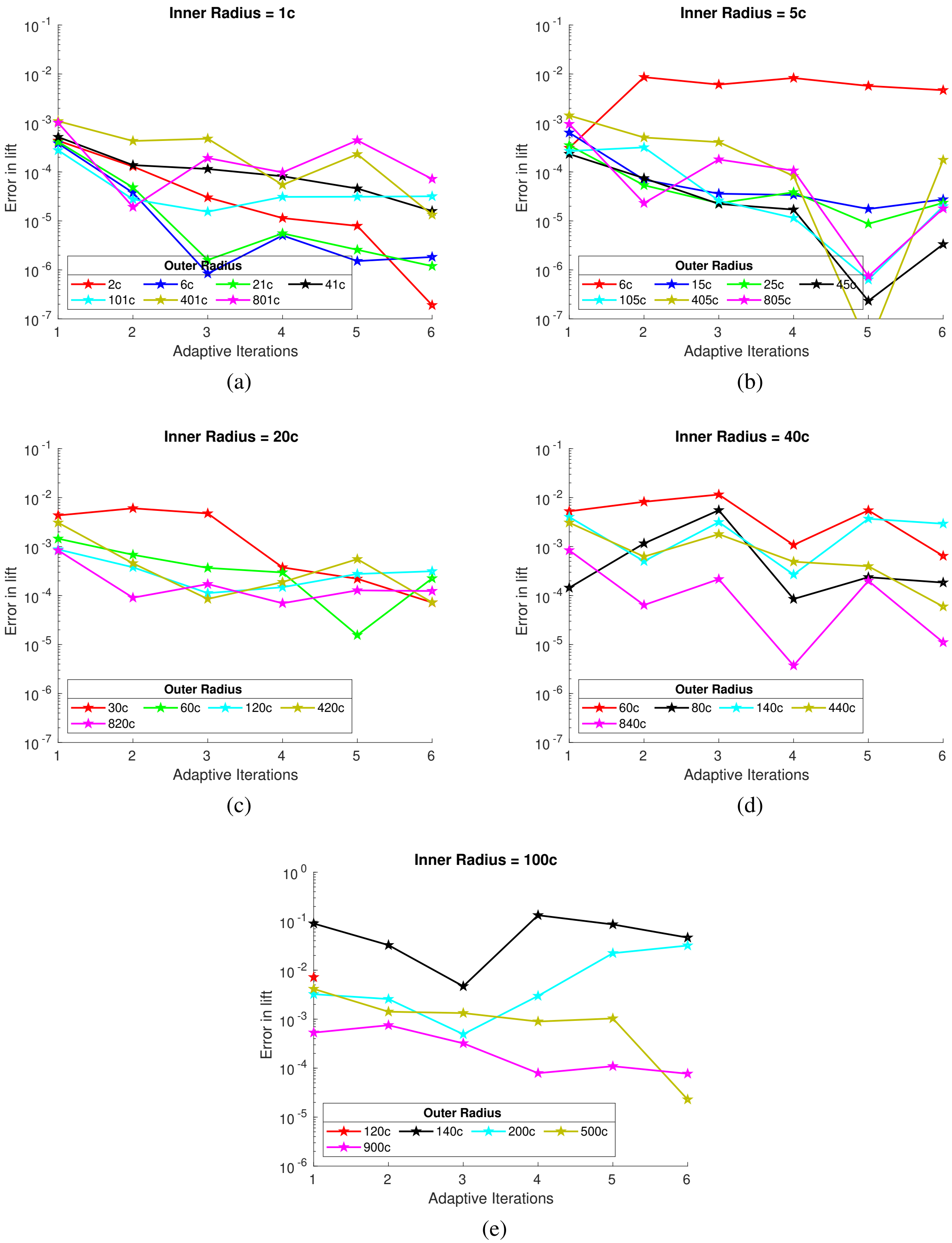
(b)

(c)

(d)

(e)

Fig. 9 Error in lift generated by the airfoil in a laminar flow as a function of the adaptive iterations.



(a)

(b)

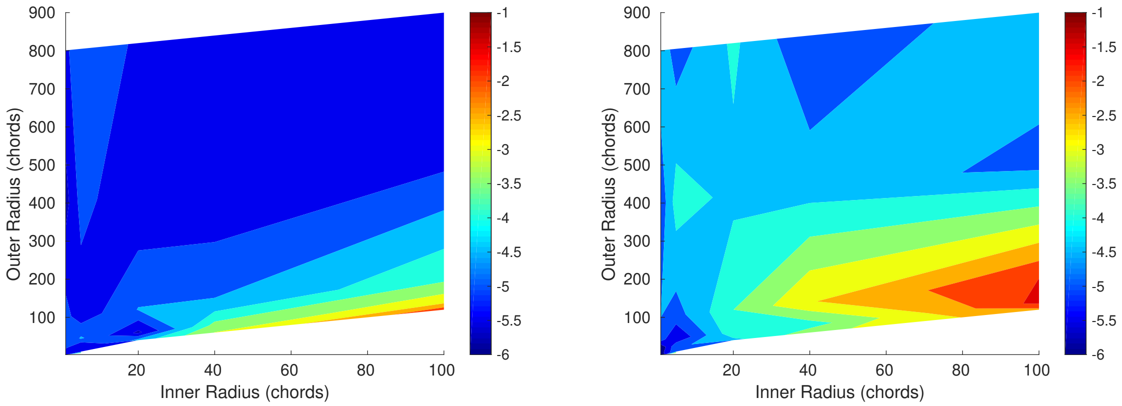
(c)

(d)

(e)

Fig. 10 Error in lift generated by the airfoil in an inviscid flow as a function of the adaptive iterations.

reference lift en route to convergence. The existence of two optimum blending regions can be observed from Figure 11(a), which shows the absolute error in the lift at the end of the adaptive iterations for laminar flow. The first optimum blending region is located close to the airfoil. This location of the blending region benefits from the mesh density of the



(a) Laminar

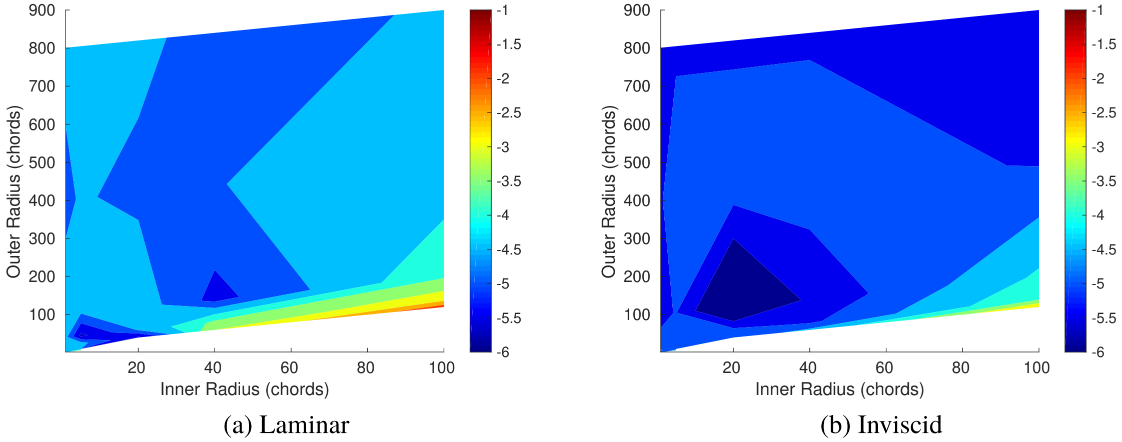
(b) Inviscid

Fig. 11 Error in lift post output-based adaptation.

initial mesh used for the simulations. The initial mesh optimized for the reference position leads to a finer mesh close to the airfoil, which effectively resolves the boundary layer. Similarly, lack of flow features far away from the airfoil results in larger element sizes there. Due to a finer mesh close to the airfoil, the blending region is well-resolved and leads to less distortion within each element, thereby resulting in better output convergence. The second optimum blending region exists where the outer radius extends far away from the airfoil. Irrespective of the inner radius, having a larger outer radius results in less distortion within an element, which in turn results in less mesh deformation error. Relatively slow convergence can be seen in the output for cases having small blending regions with larger inner radius. Pushing the inner radius away from the airfoil and keeping the blending region small results in the blending occurring primarily in elements of larger sizes. These large elements are incapable of resolving the blending well, thus leading to high errors. Similar optimum blending regions are observed for the inviscid case as well, as shown in Figure 11(b). For the inviscid case, the optimum blending region close to the airfoil outperforms the optimum blending region away from the airfoil due to the initial mesh used for the inviscid flow. The initial optimized mesh for the inviscid case is more isotropic compared to the viscous flow case because of the lack of a boundary layer. The isotropic nature of the mesh combined with the radial nature of the blending leads to a better resolution for the deformations blended close to the airfoil.

The output convergence study in this analysis is conducted using output-based mesh adaptation. This technique is successful in targeting the elements in the blending region for further adaptation because of the definition of the adaptive indicator. As described in Section IV, the adaptive indicator is a function of the adjoint and the residual evaluated by projecting the coarse space solution into the fine space. The elements inside the blending region, despite having a lower adjoint magnitude, suffer from the errors originating from mesh motion and have a higher projected residual. A second definition of the adaptive indicator, based solely on the residual is also tested for output convergence. Figure 12 shows the output convergence for the various blending regions as a function of the adaptive iteration, using residual-based adaptation. The output convergence for the various positions for the blending region in the laminar case is slower for residual-based adaptation when comparing it against output-based adaptation. The inclusion of the lift adjoint in the adaptive indicator focuses the adaptation on mesh elements important for lift evaluation irrespective of the mesh deformation errors. Residual-based adaptation, on the other hand, focuses more on the errors due to mesh deformation in the bigger elements leading to slower convergence. However, the opposite behavior is seen in the case of inviscid flow, where better convergence rates are seen with residual based adaptation. In output-based adaptation, the singularity of the adjoint along the stagnation streamline [26] leads to numerical noise in the adjoint evaluation. This causes excessive adaptation along the stagnation streamline, which is avoided in the case of residual-based adaptation, leading to more efficient output convergence. Mesh adaptation using residual-based adaptive indicators is comparatively much faster than output-based adaptive indicators due to the lack of the adjoint evaluation. It is also able to highlight some of the shortcomings of the output-based approach, where errors in adjoint evaluations can lead to slower output convergence for inviscid cases. However, such a definition of the error estimate is not useful for unsteady cases, where the information of characteristics, provided by the adjoint, is extremely useful.

Mesh-motion errors arise due to under-integration and loss of optimal polynomial approximation in global space due space-time deformation of the elements. Within the current scope of this paper, where mesh deformation is applied on a single mesh without re-meshing, this effect is mitigated by changing the blending parameter, which determine the region in the mesh where the mesh-motion errors are generated. The blending parameter affects elements of varying sizes differently, which explains the sensitivity of the outputs to the position of the blending region. The blending parameter, unique to this particular deformation algorithm, is equivalent to the term that signifies the region of influence of the



(a) Laminar

(b) Inviscid

Fig. 12 Error in lift post residual-based adaptation.

displacement at a particular boundary node for any mesh deformation scheme. Other deformation approaches, including interpolation methods and PDE based methods, also have tunable parameters and thus likely face similar issues.

B. High-fidelity FSI Solver

In this section, the high-order coupled solver is validated and the effectivity of the error estimate and the efficiency of the adapted space-time meshes are demonstrated.

1. Cantilevered beam

Consider the flow around an Euler-Bernoulli beam cantilevered at the leading edge. At the undeformed strain-free position, the beam makes an angle of attack of 5 degrees to the flow. The actual beam is considered two dimensional when represented in the fluid mesh, with a length of 1 m and a thickness of 0.005 m. The fluid mesh uses a structured grid consisting of 1954 nodes and 1840 quadrilateral elements. The domain consists of a rectangular box, [0,9]×[0,8], where the beam is cantilevered at the center of the domain and the fluid enters the domain from the left. Symmetry boundary conditions at the top and bottom of the domain and a static pressure boundary condition at the outlet are used. Figure 13 shows the computational fluid mesh. The one-dimensional beam is represented by 8 beam elements of uniform size in the structures solver. The fluid and structural meshes are constructed such that the interface nodes coincide, thus reducing the error in the structural coupling of the two subsystems. After adaptation, cubic splines are used as the spatial coupling algorithm to transfer data between the subsystems.

The fluid flow properties are Mach number $M = 0.2$ and Reynolds number $Re = 2000$, and the beam has an elastic modulus $E = 0.2$ Pa, area moment of inertia $I = 1$ kg m^2, $\rho = 1$ kg/m^3, cross section area $A = 1$ m^2, and

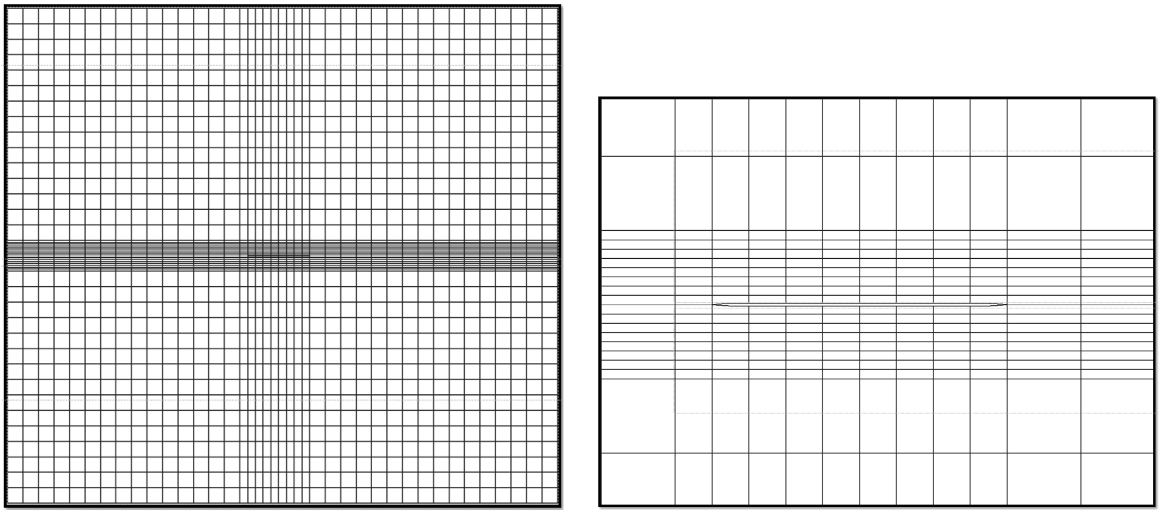
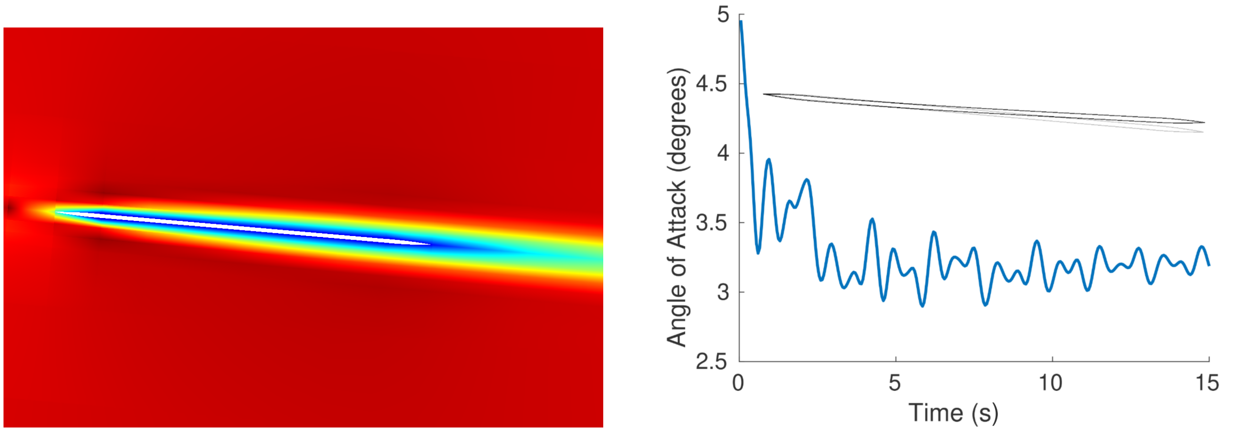


Fig. 13 The reference fluid mesh for the cantilevered beam test case.

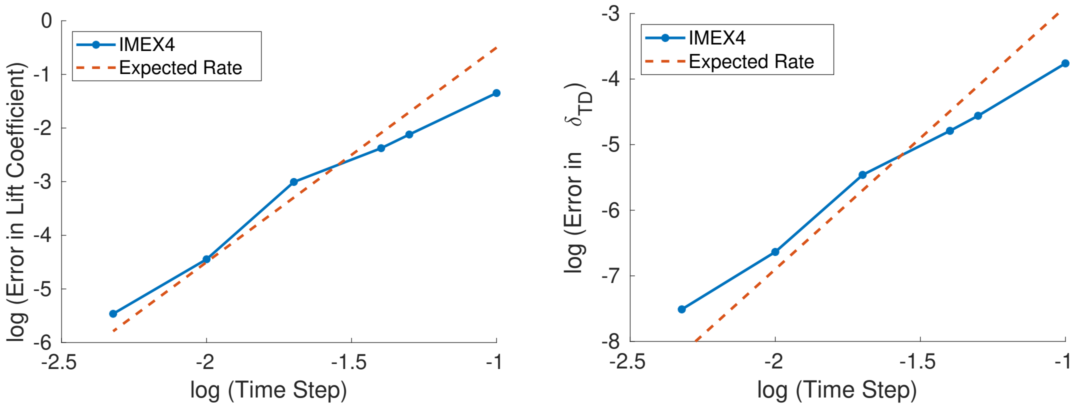


(a) x momentum at final time, $T = 15$s (b) At steady state the beam develops a negative camber

Fig. 14 Time evolution of the motion of the beam.

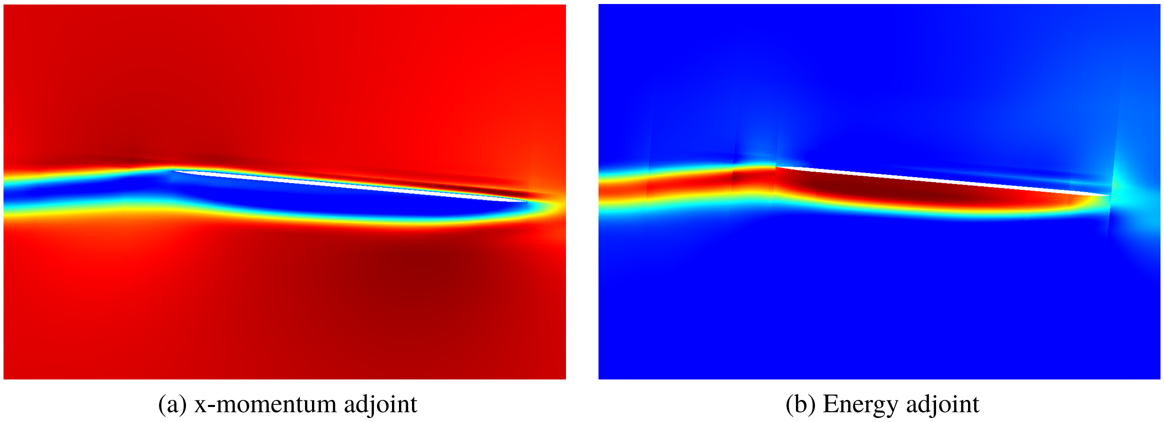
Rayleigh damping parameters, $\alpha = 1$ and $\beta = 1$. At time $t = 0$, the structure is at rest and the fluid state is obtained from a steady solve for the steady-state solution of the fluid. $p = 1$ order polynomials have been used for the fluid spatial discretization and the time scheme for both subsystems is ESDIRK4. The two subsystems are coupled temporally using the IMEX scheme. For this flow condition, a convex nature in the beam shape is expected at steady state, which is corroborated by Figure 14(a), where the final time is $T = 15$s. Figure 14(b) presents the evolution of the angle of attack on the beam over time for this flow condition. The angle of attack is measured as the angle made between the chord line and the horizontal flow direction.

To validate the temporal convergence of the high-order coupled solver, the absolute error in the lift coefficient



(a) Lift (b) Tip displacement

Fig. 15 Convergence study for the IMEX scheme.


(a) x-momentum adjoint

(b) Energy adjoint

Fig. 16 Uncoupled drag adjoint solution for the beam test case at initial time.

and non-dimensionalized tip displacement are measured at the final time, $T = 4$s. A plot of the observed relative error as a function of time step for the ESDIRK4 scheme is shown in Figure 15. Note that in each case, the scheme exhibits convergence at the designed rate. The plot demonstrates that this partitioned approach attains up to fourth-order accuracy in time with the IMEX coupling scheme. The output of interest for mesh adaptation is the time-integral of the drag force on the beam,

$$\bar{J} = \int_0^T D(t)dt. \tag{31}$$

Figure 16 shows the adjoint solution at the initial time of the simulation. The adjoint solution shows the sensitivity of the output to the perturbation in the x-momentum and energy conservation equations. Using the drag adjoint, an unsteady error estimate on the output was evaluated and used to adapt the fluid mesh. Figure 17 compares the output convergence of the drag, using the current adjoint-based approach, to uniform refinement. At every stage of uniform refinement, the spatial degrees of freedom of the fluid computational mesh and the number of time steps are doubled.

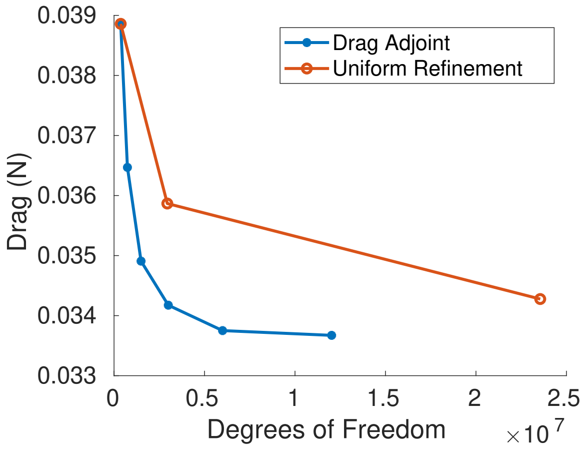
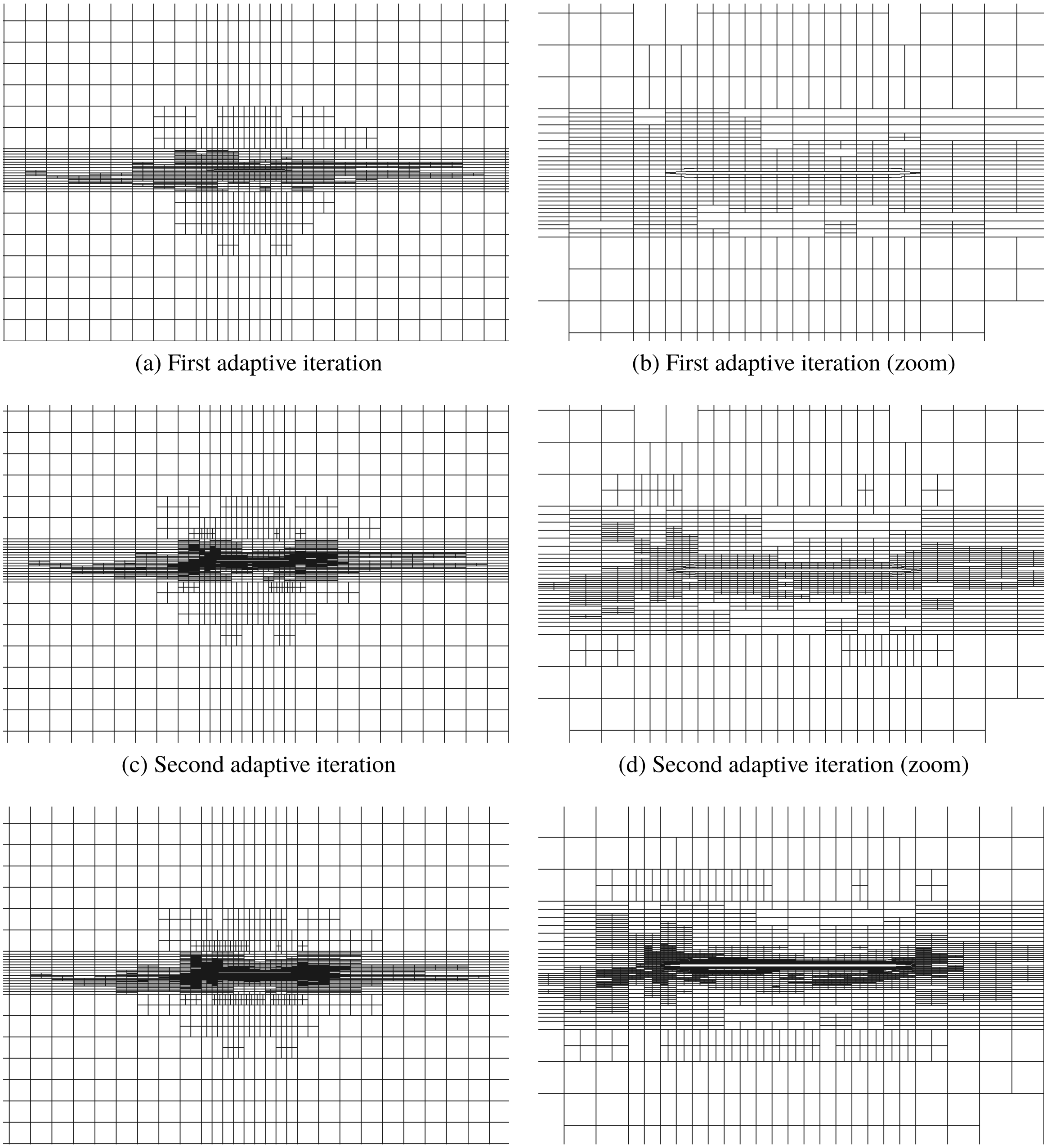


Fig. 17 Comparison of output-based adaption to uniform refinement.



(a) First adaptive iteration

(b) First adaptive iteration (zoom)

(c) Second adaptive iteration

(d) Second adaptive iteration (zoom)

(e) Third adaptive iteration

(f) Third adaptive iteration (zoom)

Fig. 18 Successive iterations of hanging node adaptation on the beam test case.

The growth factor for the total number of space-time degrees of freedom per adaptive iteration is set to f_{tot}= 2. The initial solution, on which adaptation takes place, is obtained with $p = 1$ order polynomial, 40 time steps, and a final time of $T = 4s$. The plot shows effectiveness of the drag adjoint relative to uniform refinement. The adapted meshes converge to a lower error with fewer degrees of freedom, thereby increasing accuracy and reducing computational cost. Figure 18 shows the adapted fluid mesh after each successive adaptive iteration. The adaptive iterations refine the regions close to the interface of the beam, where the drag adjoint is high, thereby reducing the spatial discretization error in the fluid mesh on the output.

2. Two-dimensional pitching-plunging NACA 0012 airfoil

A two-dimensional pitching-plunging airfoil is a common aeroelastic model which has been studied extensively [27]. The two degree-of-freedom system, when exposed to an airstream of Mach number M_f, exhibits a dynamic instability known as flutter where the two aeroelastic modes, pitch and plunge, coalesce, resulting in a self-sustained simple harmonic motion of the structure. The Mach number M_f represents the stability boundary for the system, above which the oscillations grow in amplitude.

To study the phenomenon of flutter, consider a NACA 0012 airfoil pinned at it's elastic axis in a free-stream flow of Mach number M_∞ as shown in Figure 19. The two degrees of freedom of the airfoil are the plunge h, which is taken to be positive in the downward direction, and the pitch angle α, which is considered positive clockwise (pitch up). The geometric properties of the airfoil are the chord c. Measured from the nose of the airfoil, the center of mass and the elastic axis are located at x_{cg} and x_{ea} respectively. The inertia properties of the airfoil are the mass, m, and the moment of inertia about the elastic axis, I_{ea}. The airfoil is connected to two springs at the elastic axis where the plunge spring represents the bending stiffness, K_h, of the structure, and the torsional spring represents the torsional stiffness, K_α. The equations of motion of the pitching-plunging airfoil are given as:

$$m\ddot{h} + S\ddot{\alpha} + K_h h = -L, \tag{32}$$

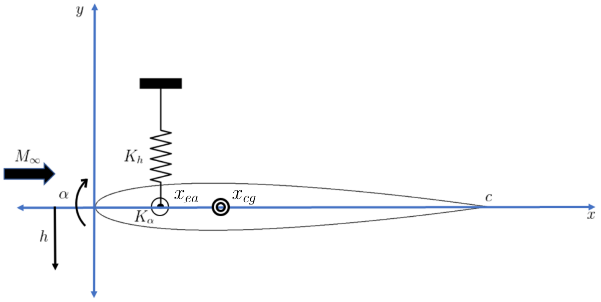


Fig. 19 Model of a two-degree-of-freedom pitching-plunging airfoil.

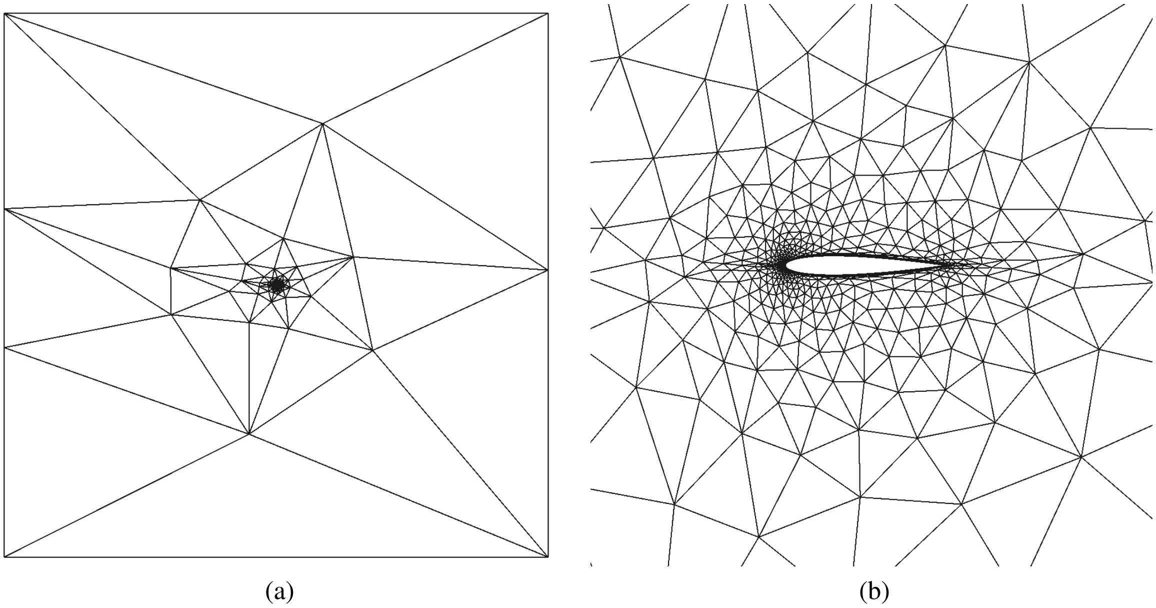


Fig. 20 Unstructured viscous mesh for the pitching plunging airfoil.

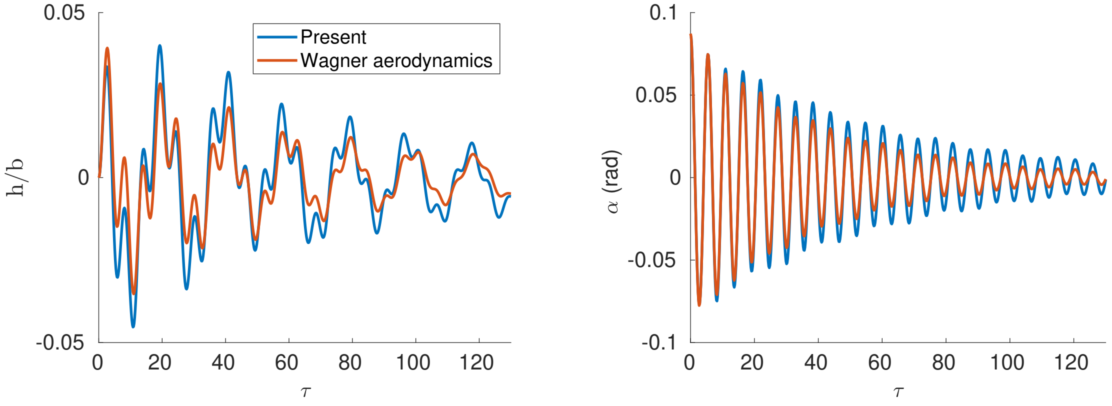
$$S\ddot{h} + I_{ea}\ddot{\alpha} + K_{\alpha}\alpha = M, \tag{33}$$

where S is the static unbalance defined by the product between $(x_{cg} - x_{ea})$ and the mass of the airfoil, L is the total lift acting on the airfoil and M is the net moment on the airfoil about the elastic axis. The non-dimensional parameters used to describe the model are defined as:

$$\mu = \frac{m}{\pi\rho_{\infty}b^2}, \quad \bar{\omega} = \frac{\omega_h}{\omega_{\alpha}}, \quad r_{\alpha} = \sqrt{\frac{I_f}{mb^2}}, \quad \chi = \frac{S}{mb}, \tag{34}$$

where μ is the mass ratio, $\bar{\omega}$ is the ratio of the uncoupled natural frequencies, and r_{α} and χ are the non-dimensional inertia and static balance, respectively. The uncoupled, in-vacuum, natural pitching and plunging frequency are defined as $\omega_h = \sqrt{K_h/m}$ and $\omega_{\alpha} = \sqrt{K_{\alpha}/I_f}$, respectively. The case setup has been taken from Sanchez et al. [28] where the non-dimensional parameters are set as $\mu = 100$, $\bar{\omega} = 0.3185$, $r_{\alpha} = 0.5$ and $\chi = 0.25$, to ensure a subsonic flutter. The pitching frequency is set to be $\omega_{\alpha} = 45$ rad/s and the elastic axis is located at the quarter chord of the airfoil. The Reynolds number is set to $Re = 4\times10^6$ and the chord, $c = 1$ m. The airfoil is set at an initial angle of attack of five degrees to the flow.

A coarse, unstructured, triangular, viscous mesh of 2766 elements is generated for the coupled simulation, as shown in Figure 20. The airfoil is located centrally in the domain, the boundary of which consists of a square box which spans from $-100c$ to $100c$ in both dimensions. The fluid flow is simulated using a RANS solver with the SA turbulence model [29]. The coupled system uses the fourth-order time scheme, ESDIRK4, introduced in the section III,



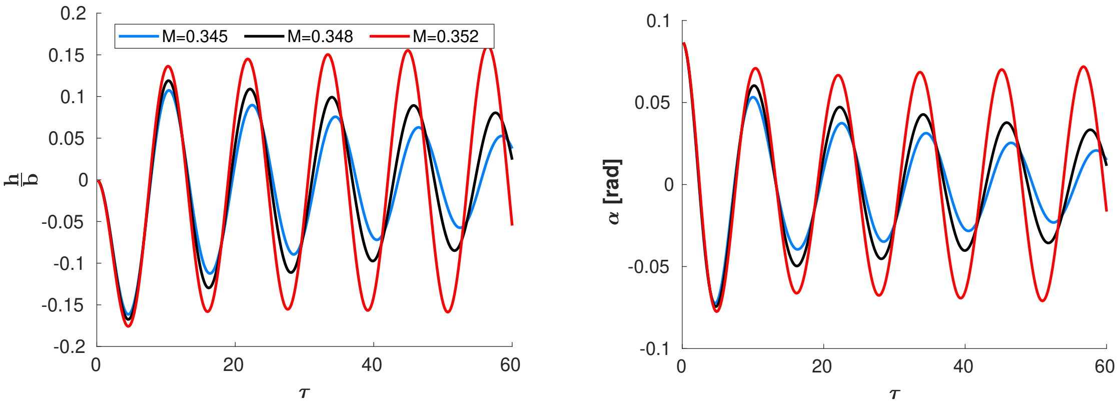
(a) Non-dimensional plunge displacement

(b) Pitch angle

Fig. 21 Aeroelastic response of the airfoil in the subcritical regime as a function of the non-dimensional time (τ).

to march forward in time. A steady-state flow solution is used as the initial condition for the simulation. A mesh motion algorithm is applied to handle the deformations occurring in the fluid subsystem due to the moving airfoil. The mesh motion algorithm divides the spatial domain into two sub-domains. The region extending up to a radial distance of one chord away from the elastic axis is deformed rigidly according to the deformation provided by the structures solver. Following the region of rigid deformation, a septic polynomial blends the deformation smoothly in the region extending between a radial distance of one chord and ten chords resulting in zero deformation at a radial distance of ten chords.

Using the parameters defined above, the aeroelastic response of the coupled system is verified by conducting a simulation at a subcritical Mach number, $M_\infty = 0.1$. The simulation is performed with $p = 3$ order polynomials



(a) Non-dimensional plunge displacement

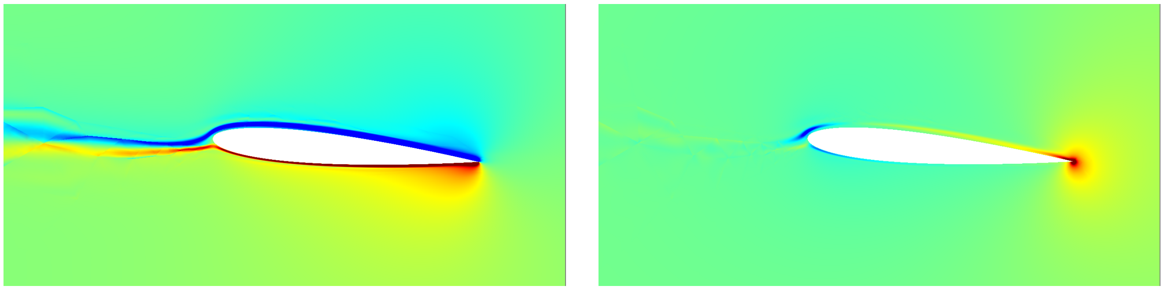
(b) Pitch angle

Fig. 22 Aeroelastic response of the airfoil near the flutter boundary for different values of the uniform flow Mach number.

for the fluid spatial discretization and 60 time steps per oscillation of the plunge mode of the airfoil. The response of the coupled system is compared against Wagner's incompressible aerodynamic model [30]. Figure 21 shows the non-dimensionalized unsteady pitch and plunge displacements as functions of the non-dimensional time, $\tau = \omega_\alpha t$. The unsteady pitch and plunge displacements are consistent with the theoretical aerodynamics model at the subcritical Mach number as well as other CFD solvers as shown in Sanchez et al. [28]. To determine the flutter boundary of the system, the aeroelastic response is recorded for varying Mach numbers. The damping coefficient of the plunge displacement in each run is evaluated by the logarithmic decrement approach and fitted by a quadratic polynomial to estimate the actual flutter Mach number. Figure 22 shows the aeroelastic response at three different Mach numbers close to flutter using $p = 3$ order polynomials for the spatial discretization. The responses at M_∞=0.345 and M_∞=0.348 lie in the sub-critical regime while the response at M_∞=0.352 is past the flutter boundary as the states grow without bound over time. The flutter Mach number is estimated to be at M_f =0.351 by interpolating using the damping evaluated at these Mach numbers.

The errors in the flutter boundary evaluation arise due to the spatial discretization errors in the fluid subsystem and the temporal discretization in both the fluid and the structural subsystem. A p-refinement study of the coupled simulation corroborated the effects of spatial discretization on the flutter boundary for the two-degree of freedom system. An equivalent time-refinement study did not result in a significant change in the flutter boundary due to the fourth-order order time scheme, ESDIRK4, and a small time step used in the coupled simulation. Therefore, the spatial discretization errors dominate the flutter boundary evaluation and an adaptive procedure is used to obtain an optimized fluid mesh.

The current implementation of the adaptation process allows for adaptation in the spatial order of elements i.e, p-adaptation for unstructured triangular meshes. The adaptation process begins with a coarse viscous mesh with $p = 1$ order elements in the entire domain. The adaptation process does not directly adapt on the flutter Mach number as this quantity depends on both subsystems and requires the evaluation of the coupled adjoint. However, as the current


(a) Conservation of the x-momentum adjoint

(b) Conservation of the y-momentum adjoint

Fig. 23 Uncoupled lift adjoint solution for the airfoil at the initial time.

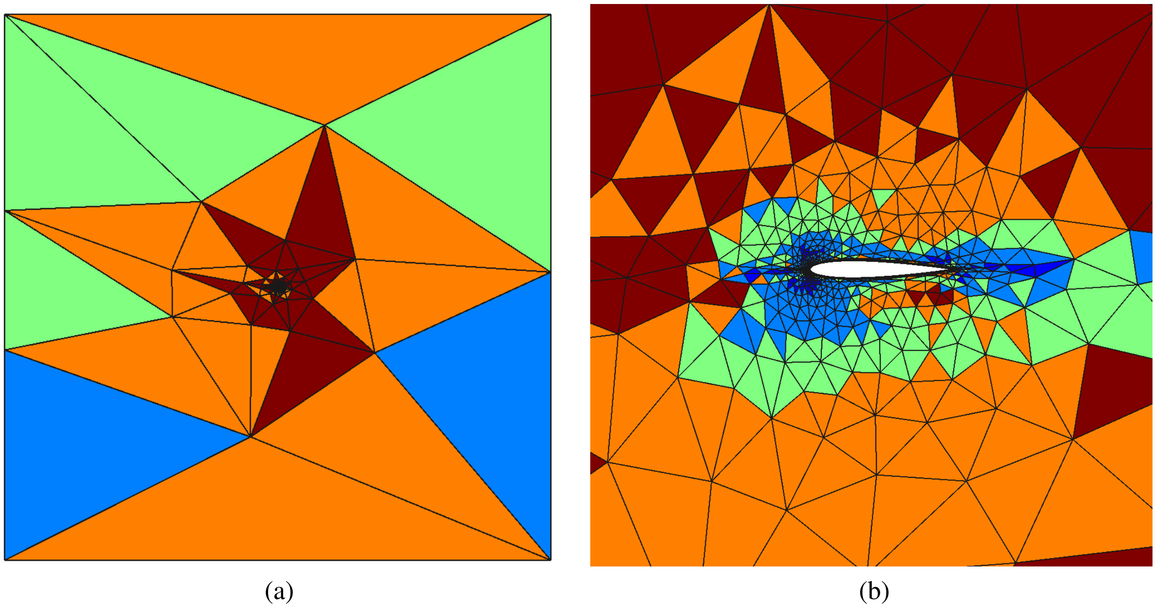

(a) (b)

Fig. 24 Element order distribution at the end of the adaptation process on the pitching-plunging airfoil. The elements colored dark blue denote $p = 1$, light blue denote $p = 2$, green denote $p = 3$, orange denote $p = 4$, and dark red denote $p = 5$ order elements.

implementation of the coupled solver provides the fluid adjoint only, the adaptation is based on the time-integrated lift.

$$\bar{J} = \int_0^T L(t)dt. \tag{35}$$

The accurate estimation of the flutter Mach number and the unsteady lift are both affected by the spatial errors in the fluid subsystem. As the governing equation of the structural subsystem depends on the lift, adapting on the unsteady lift will directly affect the flutter Mach number evaluation as well. The Mach number chosen for the adaptation cycle is $M_\infty = 0.345$, which lies in the subcritical regime. The final time chosen for each adaptation iteration is at

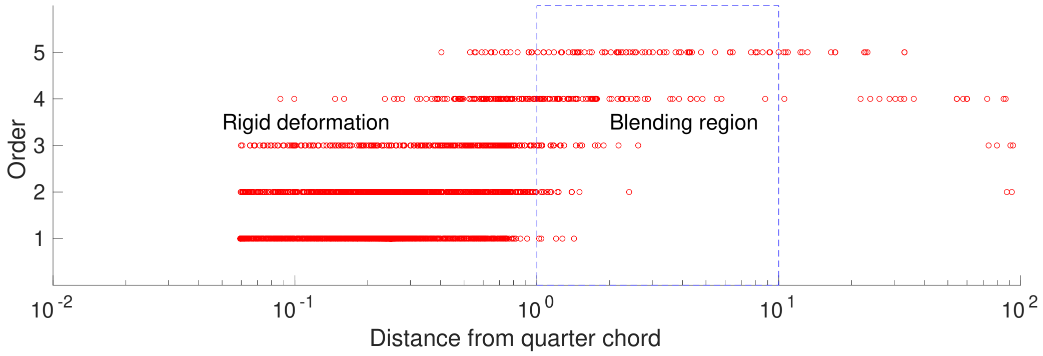


Fig. 25 Spatial distribution of the element spatial order at the end of the adaptation process.

the end of the first cycle of oscillation in the plunge displacement of the airfoil. Figure 23 shows the conservation of the x-momentum and y-momentum components of the lift adjoint at the initial time. The magnitude of the lift adjoint decreases away from the airfoil showing that the output is most sensitive to the residuals on the elements close to the airfoil. Using the lift adjoint, an unsteady error estimate for the output is evaluated and used to adapt the fluid mesh. The total number of degrees of freedom introduced after each adaptive iteration is set by the growth factor which is set to, $\mathrm{f}_{\mathrm{tot}} = 1.2$. The optimized mesh is obtained by subjecting the coarse mesh to four cycles of adaption.

Figure 24 shows the spatial order of interpolation after each adaptive iteration. The higher-order elements ($p = 4$ and $p = 5$) introduced in the adaptation process target the elements with the highest error estimates and lie primarily in the vicinity of the airfoil and inside the blending region, as shown in Figure 25. Adaptation occurring in the blending region seems counter-intuitive as the adjoints on these elements is not as large in magnitude as on the elements inside the rigid deformation region, as seen in Figure 25. The reason behind elements in the blending region being chosen for adaptation also goes back to the definition of the adaptive indicator which is a function of the adjoint and the residual evaluated by projecting the coarse space solution into the fine space. The elements inside the blending region, despite having a lower adjoint magnitude, suffer from the errors originating from mesh motion algorithm. These error result in a larger fine-space residual term in the adaptive indicator. The mesh motion errors become smaller with higher-order approximation [16] but are dominant at lower-order approximation of the state, where the adaptation begins. Similar adaptation is not observed in the cantilevered beam case, as the amount of structural displacement is smaller due to the high structural stiffness and added damping.

As the importance of mesh motion errors arising in the blending region is evident from the spatial distribution of the higher-order elements, the blending region is thereby varied to show its effect on output evaluation in FSI

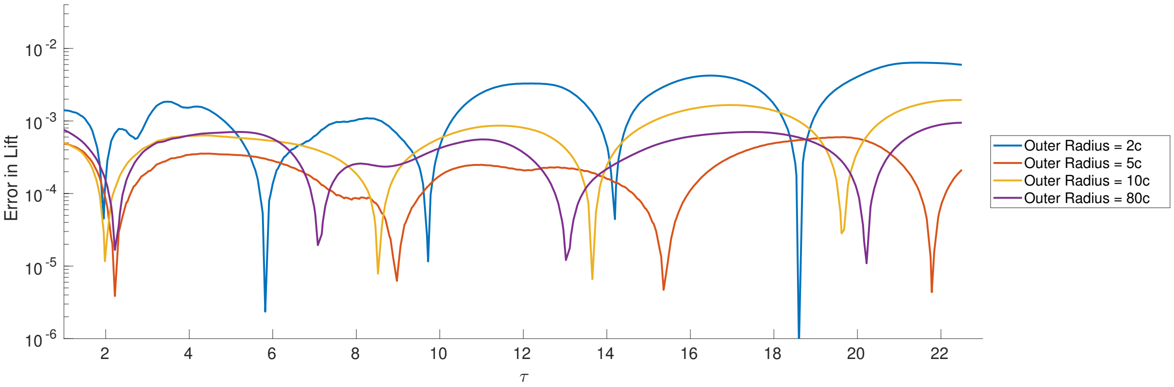
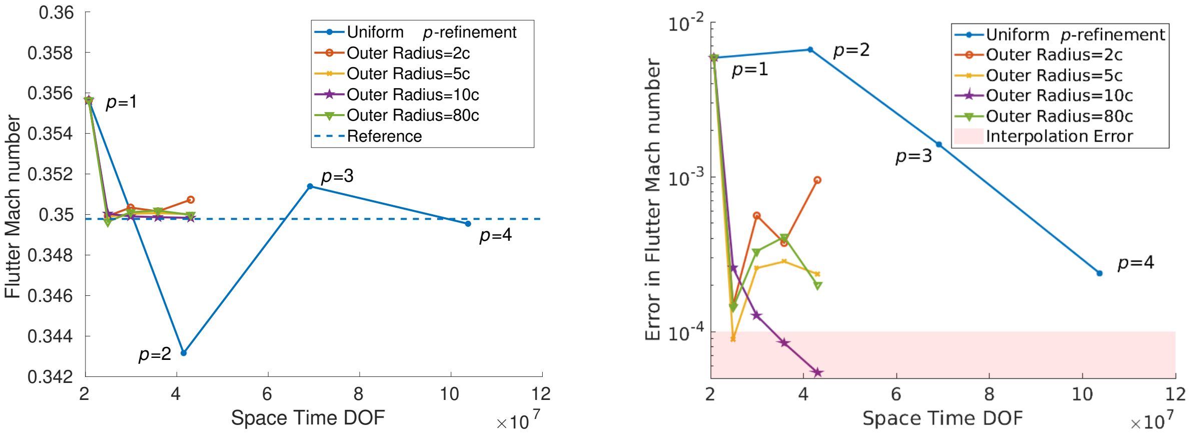


Fig. 26 Error in the unsteady lift for blending regions with constant inner radius of 1c and varying outer radii, post-adaptation.



(a) Convergence of flutter boundary for various blending regions

(b) Error in the flutter boundary for various blending regions

Fig. 27 Comparison of the flutter boundaries for output-based adaptation to uniform *p*-refinement.

simulations. Keeping the inner radius of the blending region constant at one chord, the outer radius is varied throughout the domain to show the effect of the position of the blending regions on the flutter Mach number calculations and unsteady lift evaluation. Figure 26 compares the absolute error in the time history of lift for the various blending regions, post-adaptation, against the reference unsteady lift. The results corroborate the existence of an optimum blending region close to the airfoil, seen previously in V.A. Figure 27(a) compares the convergence of the flutter boundary Mach number for the adapted meshes against uniform *p*-refinement. At every stage of uniform refinement, the spatial order of the elements in the entire domain is increased by one. The reference/truth flutter Mach number, $M_f = 0.3498$ and the unsteady lift is evaluated on a highly refined viscous mesh with 10492 elements with $p = 4$ elements in the entire spatial domain. The fine mesh is generated by refining the coarse mesh anisotropically. A metric is introduced for the size of each element which is scaled by a factor to obtain the refined mesh. The plot shows the effectiveness of the adaptation process relative to uniform refinement. Figure 27(b) compares the absolute error in flutter Mach number for the adapted meshes against uniform *p* refinement. The effect of the blending region is not very distinct due to the interpolation error, which dominates the evaluation of the flutter Mach number from the damping coefficient of the plunge displacements, as shown as the shaded region in the figure. The adapted meshes, irrespective of the position of the blending region, converge at a faster rate with fewer degrees of freedom, thereby increasing accuracy and reducing computational cost.

VI. Conclusion

In this paper, a high-order partitioned approach is combined with output-based adaptation to provide high-fidelity solutions for fluid-structure interaction problems. The cantilevered beam test case verifies the high-order coupling scheme, IMEX, by demonstrating the expected order of convergence. The two-dimensional pitching-plunging NACA

0012 airfoil is also simulated using the high-order coupled solver, for which the aeroelastic response is verified at low Mach numbers against Wagner's incompressible aerodynamic model. An adaptive meshing procedure is applied to obtain an accurate drag prediction for the beam case and flutter boundary for the two degree-of-freedom case. Further, output-based error estimation is also applied to control errors arising from mesh deformation. A two-dimensional free-stream preservation test in an inviscid flow is used to quantify error due to mesh-motion algorithms. An output-based error estimate on the entropy norm over the domain gives an estimate of the error due to the mesh deformation procedure. The error estimate is used to optimize the mesh motion algorithm by optimizing the variables used to blend the deformation. For an explicit mapping, an optimized inner and outer radius of the blending is obtained for a steady and unsteady deformation resulting in the least error in the output. A secondary case of an airfoil undergoing rigid body deformation in a steady fluid flow is analyzed to observe the effects of the position of the blending region on the output convergence. With the primary application of the paper being in aeroelasticity, where the structure is undergoing small deformation, the identified guidelines for blending radii could be applicable and useful. The proposed mitigation process is indeed not a complete solution procedure that will automatically reduce mesh-motion errors in all cases. The goal of this work is to identify trade-offs and general trends in the relationship between mesh motion parameters and output errors. A complete mitigation strategy would involve an optimization of these parameters, which is of interest in future work. The output convergence study verifies that the implementation of a GCL is not necessary for achieving high accuracy in high-order FSI simulations involving rigid body motions. Two different adaptive indicators are also studied and compared for mesh adaptation. While output-based mesh adaptation outperforms residual-based adaptation, the latter highlights some of the shortcomings of the output-based adaptation techniques in the inviscid flow regime.

A better understanding of the errors generated by the mesh motion algorithms is achieved from this work. The use of output-based mesh adaptation in efficiently reducing the spatial errors generated by the mesh distortion as well the spatial discretization has been demonstrated, thus, showing its applicability to FSI simulations. The effect of the distribution of elements in the initial mesh on the output convergence is also highlighted for high-order FSI simulations. Overall, the adaptive meshing procedure provides significant increase in the accuracy of unsteady outputs for aeroelastic problems compared to more common adaptation methods, such as uniform refinement.

VII. Acknowledgments

This work was supported by the U.S. Air Force Research Laboratory (AFRL) under the Michigan-AFRL Collaborative Center in Aerospace Vehicle Design (CCAVD), with Dr. Philip Beran as the task Technical Monitor.

References

[1] Guruswamy, G. P., “Interaction of fluids and structures for aircraft applications,” *Computational Structural Mechanics & Fluid Dynamics*, Elsevier, 1988, pp. 1–13.

[2] Hou, G., Wang, J., and Layton, A., “Numerical methods for fluid-structure interaction—a review,” *Communications in Computational Physics*, Vol. 12, No. 2, 2012, pp. 337–377. https://doi.org/10.4208/cicp.291210.290411s.

[3] Hübner, B., Walhorn, E., and Dinkler, D., “A monolithic approach to fluid–structure interaction using space–time finite elements,” *Computer methods in applied mechanics and engineering*, Vol. 193, No. 23-26, 2004, pp. 2087–2104.

[4] van Zuijlen, A. H., de Boer, A., and Bijl, H., “Higher-order Time Integration Through Smooth Mesh Deformation for 3D Fluid-structure Interaction Simulations,” *J. Comput. Phys.*, Vol. 224, No. 1, 2007, pp. 414–430. https://doi.org/10.1016/j.jcp.2007.03.024.

[5] Fidkowski, K. J., and Darmofal, D. L., “Review of Output-Based Error Estimation and Mesh Adaptation in Computational Fluid Dynamics,” *AIAA Journal*, Vol. 49, No. 4, 2011, pp. 673–694. https://doi.org/10.2514/1.J050073.

[6] Vanharen, J., Feuillet, R., and Alauzet, F., “Mesh adaptation for fluid-structure interaction problems,” *2018 Fluid Dynamics Conference*, 2018, p. 3244.

[7] Persson, P.-O., Bonet, J., and Peraire, J., “Discontinuous Galerkin solution of the Navier–Stokes equations on deformable domains,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 198, No. 17-20, 2009, pp. 1585–1595.

[8] Budd, C. J., Huang, W., and Russell, R. D., “Adaptivity with moving grids,” *Acta Numerica*, Vol. 18, 2009, pp. 111–241.

[9] Selim, M., Koomullil, R., et al., “Mesh deformation approaches–a survey,” *Journal of Physical Mathematics*, Vol. 7, No. 2, 2016.

[10] Batina, J. T., “Unsteady Euler airfoil solutions using unstructured dynamic meshes,” *AIAA Journal*, Vol. 28, No. 8, 1990, pp. 1381–1388.

[11] Kedward, L., Allen, C. B., and Rendall, T. C., “Efficient and exact mesh deformation using multiscale RBF interpolation,” *Journal of Computational Physics*, Vol. 345, 2017, pp. 732–751.

[12] Witteveen, J., “Explicit and robust inverse distance weighting mesh deformation for CFD,” *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, 2010, p. 165.

[13] Persson, P.-O., Peraire, J., and Bonet, J., “A high order discontinuous Galerkin method for fluid-structure interaction,” *18th AIAA Computational Fluid Dynamics Conference*, 2007, p. 4327.

[14] Lesoinne, M., and Farhat, C., “Geometric conservation laws for aeroelastic computations using unstructured dynamic meshes,” *12th Computational Fluid Dynamics Conference*, 1995, p. 1709.

[15] Lesoinne, M., and Farhat, C., "Geometric conservation laws for flow problems with moving boundaries and deformable meshes, and their impact on aeroelastic computations," *Computer Methods in Applied Mechanics and Engineering*, Vol. 134, No. 1, 1996, pp. 71 – 90. https://doi.org/https://doi.org/10.1016/0045-7825(96)01028-6.

[16] Kast, S. M., and Fidkowski, K. J., "Output-based Mesh Adaptation for High Order Navier-Stokes Simulations on Deformable Domains," *Journal of Computational Physics*, Vol. 252, No. 1, 2013, pp. 468–494. https://doi.org/10.1016/j.jcp.2013.06.007.

[17] Smith, M. J., Cesnik, C. E. S., and Hodges, D. H., "Evaluation of Some Data Transfer Algorithms for Noncontiguous Meshes," *Journal of Aerospace Engineering*, Vol. 13, No. 2, 2000, pp. 52–58. https://doi.org/10.1061/(ASCE)0893-1321(2000)13:2(52).

[18] Fidkowski, K. J., Oliver, T. A., Lu, J., and Darmofal, D. L., "*p*-Multigrid solution of high–order discontinuous Galerkin discretizations of the compressible Navier-Stokes equations," *Journal of Computational Physics*, Vol. 207, 2005, pp. 92–113. https://doi.org/10.1016/j.jcp.2005.01.005.

[19] van Zuijlen, A. H., and Bijl, H., "Implicit and explicit higher order time integration schemes for structural dynamics and fluid-structure interaction computations," *Computers & structures*, Vol. 83, No. 2-3, 2005, pp. 93–105.

[20] Kennedy, C. A., and Carpenter, M. H., "Additive Runge–Kutta schemes for convection–diffusion–reaction equations," *Applied numerical mathematics*, Vol. 44, No. 1-2, 2003, pp. 139–181.

[21] Froehle, B., and Persson, P.-O., "A high-order discontinuous Galerkin method for fluid–structure interaction with efficient implicit–explicit time stepping," *Journal of Computational Physics*, Vol. 272, 2014, pp. 455–470.

[22] Fidkowski, K. J., "Output-Based Space-Time Mesh Optimization for Unsteady Flows Using Continuous-in-Time Adjoints," *Journal of Computational Physics*, Vol. 341, No. 15, 2017, pp. 258–277. https://doi.org/10.1016/j.jcp.2017.04.005.

[23] Hecht, F., "BAMG: bidimensional anisotropic mesh generator," *User Guide. INRIA, Rocquencourt*, 1998.

[24] Barral, N., "Time-accurate anisotropic mesh adaptation for three-dimensional moving mesh problems," Ph.D. thesis, 2015.

[25] Barral, N., Olivier, G., and Alauzet, F., "Time-accurate anisotropic mesh adaptation for three-dimensional time-dependent problems with body-fitted moving geometries," *Journal of Computational Physics*, Vol. 331, 2017, pp. 157–187.

[26] Venditti, D. A., and Darmofal, D. L., "Grid adaptation for functional outputs: application to two-dimensional inviscid flows," *Journal of Computational Physics*, Vol. 176, No. 1, 2002, pp. 40–69.

[27] Bisplinghoff, R. L., Ashley, H., and Halfman, R. L., *Aeroelasticity*, Courier Corporation, 2013.

[28] Sanchez, R., L. Kline, H., Thomas, D., Variyar, A., Righi, M., Economon, T., Alonso, J., Palacios, R., Dimitriadis, G., and Terrapon, V., "Assessment of the fluid-structure interaction capabilities for aeronautical applications of the open-source solver SU2." 2016, pp. 1498–1529. https://doi.org/10.7712/100016.1903.6597.

[29] Ceze, M. A., and Fidkowski, K. J., "High-order output-based adaptive simulations of turbulent flow in two dimensions," *AIAA Journal*, Vol. 54, No. 9, 2016. https://doi.org/10.2514/1.J054517.

[30] Fung, Y. C., *An introduction to the theory of aeroelasticity*, Courier Dover Publications, 2008.