# A High-Order Discontinuous Galerkin Method for Fluid-Structure Interaction With Efficient Implicit-Explicit Time Stepping

Bradley Froehle<sup>a</sup>, Per-Olof Persson<sup>a,\*</sup>

<sup>a</sup>Department of Mathematics, University of California, Berkeley, Berkeley, CA 94720-3840, USA

#### Abstract

We present a high-order accurate scheme for coupled fluid-structure interaction problems. The fluid is discretized using a discontinuous Galerkin method on unstructured tetrahedral meshes, and the structure uses a high-order volumetric continuous Galerkin finite element method. Standard radial basis functions are used for the mesh deformation. The time integration is performed using a partitioned approach based on implicit-explicit Runge-Kutta methods. The resulting scheme fully decouples the implicit solution procedures for the fluid and the solid parts, which we perform using two separate efficient parallel solvers. We demonstrate up to fifth order accuracy in time on a non-trivial test problem, on which we also show that additional subiterations are not required. We solve a benchmark problem of a cantilever beam in a shedding flow, and show good agreement with other results in the literature. Finally, we solve for the flow around a thin membrane at a high angle of attack in both 2D and 3D, and compare with the results obtained with a rigid plate.

Keywords: Fluid-structure interaction, discontinuous Galerkin, implicit-explicit, Runge-Kutta

# 1. INTRODUCTION

Many important scientific and engineering problems require predictions of fluid-structure interaction (FSI). For example, oscillatory interactions in engineering systems (e.g. aircraft, turbines, and bridges) can lead to failure. The blood flow in arteries and artificial heart valves is highly dependent on structural interactions. These interactions often involve multiple scales and non-linear effects, which makes it challenging to solve even relatively simple problems accurately.

In this paper, we present a high-order discontinuous Galerkin (DG) formulation for the Navier-Stokes equations coupled to a rigid body or a finite element model of a non-linear hyperelastic structure. Many approaches have been suggested for the simulation of fluid-structure interaction [1, 2, 3], and a common way to treat the deformable domains is the use of Arbitrary Lagrangian Eulerian (ALE) methods [4, 5, 6, 7, 8]. In these efforts the discretization on the deformable domain is carried out on a deforming grid and thus the metric changes over time.

For the non-linear structure model we use a continuous Galerkin (CG) finite element discretization, integrated in time simultaneously with the DG discretization. The forces from the fluid are applied to the structure as a surface traction, and the structure displacements give a deformation of the fluid domain.

There are two main numerical approaches for the solution of the coupled fluid/structure system. In the fully coupled (monolithic) approach, the two equations are solved simultaneously. This leads to accurate results, but requires specialized codes and often leads to less efficient solvers. In the weakly coupled (partitioned) approach, standard solvers are applied for the fluid and structure separately. An appropriate coupling scheme is then used to account for the interaction between them, often together with repeated subiterations. This is an efficient and simple method, but can suffer from lower accuracy and instability.

<sup>\*</sup>Corresponding author. Tel.: +1-510-642-6947; Fax.: +1-510-642-8204.

Email addresses: bfroehle@berkeley.edu (Bradley Froehle), persson@berkeley.edu (Per-Olof Persson)

Preprint submitted to Journal of Computational Physics

In previous work [9], we demonstrated a high-order method for fluid-structure interaction problems using a fully monolithic explicit Runge-Kutta time integrator. While this approach was straightforward, the explicit time integrator may introduce undesirable timestep restrictions. However, using the same method with an implicit Runge-Kutta scheme would require forming not only the Jacobian matrices for the fluid and structure but also for the couplings between them.

In this work we use a partitioned scheme based on implicit-explicit Runge-Kutta [10] methods as presented in [11]. This scheme avoids solving the fully coupled system, yet still offers arbitrarily high orders of accuracy in time and the ability to use implicit solvers for both the fluid and the structure. The main idea is to use the coefficients of the implicit-explicit Runge-Kutta scheme to form a stage predictor for the fluid-to-structure coupling [12]. This decouples the system into two implicit problems—one for the fluid and one for the structure—each of which is solved using standard implicit solvers and without subiterations.

There are many other partitioned FSI schemes, many of which employ similar predictor-corrector frameworks to achieve first [13, 14] or second [3, 15] order accuracy. See [16] for a review of ideas used in partitioned schemes.

The paper is organized as follows. First we present the equations for both the fluid on the deforming domain and the non-linear structure. Next we describe a mesh deformation procedure based on radial basis functions, as well as the numerical solvers, the time integration procedure, and the force predictor. We verify the high-order accuracy of the scheme using a test problem of a heaving and pitching NACA airfoil in a laminar flow, subject to a simple smooth heaving motion. In addition we show a standard FSI test problem consisting of a flexible cantilever behind a square bluff body, showing good agreement with tip displacement and oscillation frequency to values found in the open literature. Lastly we show results from two and three dimensional simulations of thin membranes at a high angle of attack.

# 2. GOVERNING EQUATIONS

# 2.1. Compressible Navier-Stokes

The compressible Navier-Stokes equations are written as:

$$\frac{\partial}{\partial t}(\rho) + \frac{\partial}{\partial x_j}(\rho u_j) = 0 \tag{1}$$

$$\frac{\partial}{\partial t}(\rho u_i) + \frac{\partial}{\partial x_j}(\rho u_i u_j + p\delta_{ij}) = \frac{\partial}{\partial x_j}\tau_{ij} \qquad \text{for } i = 1, 2, 3$$
(2)

$$\frac{\partial}{\partial t}(\rho E) + \frac{\partial}{\partial x_j}(\rho u_j E + u_j p) = \frac{\partial}{\partial x_j}(-q_j + u_i \tau_{ij})$$
(3)

where  $\rho$  is the fluid density,  $u_1, u_2, u_3$  are the velocity components, and E is the total energy. The viscous stress tensor and heat flux are given by

$$\tau_{ij} = \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \frac{\partial u_k}{\partial x_j} \delta_{ij} \right) \quad \text{and} \quad q_j = -\frac{\mu}{\Pr} \frac{\partial}{\partial x_j} \left( E + \frac{p}{\rho} - \frac{1}{2} u_k u_k \right). \tag{4}$$

Here,  $\mu$  is the viscosity coefficient and Pr = 0.72 is the Prandtl number which we assume to be constant. For an ideal gas, the pressure p has the form

$$p = (\gamma - 1)\rho\left(E - \frac{1}{2}u_k u_k\right),\tag{5}$$

where  $\gamma$  is the adiabatic gas constant. We will also use the entropy  $s = p/\rho^{\gamma}$  for visualization. We impose two types of boundary conditions – free-stream at the far field, and adiabatic no-slip conditions at the boundaries of the structure.

# 2.2. Arbitrary Lagrangian Eulerian formulation

The deformable fluid domain is handled through an Arbitrary Lagrangian Eulerian (ALE) formulation. A point  $\boldsymbol{X}$  in a fixed reference domain V is mapped to  $\boldsymbol{x}(\boldsymbol{X},t)$  in a time-varying domain v(t). The deformation gradient  $\boldsymbol{G}$ , mapping velocity  $\boldsymbol{v}_{G}$ , and mapping Jacobian g are defined as

$$\boldsymbol{G} = \nabla_{\boldsymbol{X}} \boldsymbol{x}, \quad \boldsymbol{v}_{G} = \frac{\partial \boldsymbol{x}}{\partial t}, \quad \text{and} \quad \boldsymbol{g} = \det \boldsymbol{G}.$$
 (6)

We remind the readers that a system of conservation laws in the physical domain  $(\boldsymbol{x}, t)$ 

$$\frac{\partial \boldsymbol{u}}{\partial t} + \nabla \cdot \boldsymbol{f}(\boldsymbol{u}, \nabla \boldsymbol{u}) = 0$$
<sup>(7)</sup>

may be written as a system of conservation laws in the reference domain  $(\mathbf{X}, t)$ 

$$\frac{\partial \boldsymbol{U}}{\partial t} + \nabla_{\boldsymbol{X}} \cdot \boldsymbol{F}(\boldsymbol{U}, \nabla_{\boldsymbol{X}} \boldsymbol{U}) = 0$$
(8)

where the conserved quantities and fluxes in the reference domain are modified appropriately as

$$\boldsymbol{U} = g\boldsymbol{u}, \quad \boldsymbol{F} = g\boldsymbol{G}^{-1}\boldsymbol{f} - \boldsymbol{u}\boldsymbol{G}^{-1}\boldsymbol{v}_G. \tag{9}$$

Lastly, the gradient in the physical domain may be computed in the reference domain as

$$\nabla \boldsymbol{u} = (\nabla_{\boldsymbol{X}}(g^{-1}\boldsymbol{U}))\boldsymbol{G}^{-T} = (g^{-1}\nabla_{\boldsymbol{X}}\boldsymbol{U} - \boldsymbol{U}\nabla_{\boldsymbol{X}}(g^{-1}))\boldsymbol{G}^{-T}.$$
(10)

For more details on this mapping-based ALE formulation, including a discussion on how to enforce the geometric conservation law (GCL), see [17].

#### 2.3. Neo-Hookean Elasticity Model

We use a hyperelastic neo-Hookean formulation [18] for modeling deformable structures. Here, the structure position is given by a mapping  $\boldsymbol{x}(\boldsymbol{X},t)$ , which for each time t maps a point  $\boldsymbol{X}$  in the unstretched reference configuration to its location  $\boldsymbol{x}$  in the deformed configuration. From this we compute the mapping velocity and deformation gradient as

$$\boldsymbol{v}_G = \frac{\partial \boldsymbol{x}}{\partial t}, \quad \text{and} \quad \boldsymbol{G} = \nabla_{\boldsymbol{X}} \boldsymbol{x}(\boldsymbol{X}, t).$$
 (11)

We partition the boundary of the structure domain into regions of Dirichlet and Neumann boundary conditions,  $\partial V = \Gamma_D \cup \Gamma_N$ . On the Dirichlet boundary  $\Gamma_D$  we prescribe the material position  $\boldsymbol{x}_D$ , often corresponding to no displacement. On the Neumann boundary  $\Gamma_N$  we allow for a general surface traction (i.e., force per unit surface area) which we denote  $\boldsymbol{t}$ .

The governing equations for the structure are then given by

0

$$\frac{\partial \boldsymbol{p}}{\partial t} - \nabla \cdot \boldsymbol{P}(\boldsymbol{G}) = \boldsymbol{b} \qquad \text{in } \Omega \qquad (12)$$

$$\boldsymbol{P}(\boldsymbol{G}) \cdot \boldsymbol{N} = \boldsymbol{t} \qquad \qquad \text{on } \boldsymbol{\Gamma}_{N} \tag{13}$$

$$\boldsymbol{x} = \boldsymbol{x}_D$$
 on  $\Gamma_D$  (14)

where  $\mathbf{p} = \rho \mathbf{v}_G = \rho \partial \mathbf{x} / \partial t$  is the momentum,  $\mathbf{P}$  is the first Piola-Kirchhoff stress tensor,  $\mathbf{b}$  is an external body force per unit reference volume, and  $\mathbf{N}$  is a unit normal vector in the reference domain.

For a compressible neo-Hookean material the strain energy density is given by

$$W = \frac{\mu}{2}(\bar{I}_1 - 3) + \frac{\kappa}{2}(J - 1)^2$$
(15)

where  $\bar{I}_1$ , the first invariant of the deviatoric part of the left Cauchy-Green deformation tensor, and J, the determinant of the deformation gradient, are calculated as

 $\bar{I}_1 = J^{-2/3} I_1, \quad I_1 = \operatorname{tr} \boldsymbol{B} = \operatorname{tr}(\boldsymbol{G}\boldsymbol{G}^T), \quad J = \det \boldsymbol{G}.$ (16)

The constants  $\mu$  and  $\kappa$  are the shear and bulk modulus of the material.

The first Piola-Kirchhoff stress tensor is computed as

$$\boldsymbol{P}(\boldsymbol{G}) = \frac{\partial W}{\partial \boldsymbol{G}} = \mu J^{-2/3} \left( \boldsymbol{G} - \frac{1}{3} \operatorname{tr}(\boldsymbol{G}\boldsymbol{G}^T)\boldsymbol{G}^{-T} \right) + \kappa (J-1)J\boldsymbol{G}^{-T}.$$
(17)

For two dimensional problems we use a plane strain formulation in which we treat the stretch in the third dimension as constant [19], thinking of the problem as modeling the cross-section of an infinitely long prismatic structure. This results in similar equations, requiring only a slight modification to the Piola-Kirchhoff stress tensor.

# 3. SPATIAL DISCRETIZATION

#### 3.1. Fluid Spatial / DG

The fluid equations as described in Sec. 2.2 are discretized using a high-order discontinuous Galerkin formulation with tetrahedral mesh elements and nodal basis functions. The inviscid fluxes are computed using Roe's method [20], and the numerical fluxes for the viscous terms are chosen according to the compact DG method [21]. Below, we summarize this discretization for the ALE system of conservation laws (8). For simplicity, we change the notation and use lower-case symbols for the solution  $\boldsymbol{u}$ , and we omit the subscripts on the derivative operators. We also split the fluxes into an inviscid component  $F^i(\boldsymbol{u})$  and a viscous component  $F^v(\boldsymbol{u}, \nabla \boldsymbol{u})$ , corresponding to the second term in the left-hand side and the entire right-hand side of equations (1)-(3), respectively.

Following standard procedure for DG discretization of second-derivatives [22], we first introduce the auxiliary gradient variables  $q = \nabla u$ , and write (8) as the system of first order equations in the reference domain

$$\frac{\partial \boldsymbol{u}}{\partial t} + \nabla \cdot \boldsymbol{F}^{i}(\boldsymbol{u}) - \nabla \cdot \boldsymbol{F}^{v}(\boldsymbol{u}, \boldsymbol{q}) = 0, \qquad (18)$$

$$\nabla \boldsymbol{u} = \boldsymbol{q}.\tag{19}$$

We introduce a computational mesh of the reference fluid domain  $\Omega$  and denote its elements by  $\mathcal{T}_h = \{K\}$ . Furthermore, we introduce the finite element spaces  $\mathcal{V}_h^p$  and  $\Sigma_h^p$  as:

$$V_h^p = \{ \boldsymbol{v} \in [L^2(\Omega)]^5 \mid \boldsymbol{v}|_K \in [\mathcal{P}_p(K)]^5 \; \forall K \in \mathcal{T}_h \},$$

$$(20)$$

$$\Sigma_h^p = \{ \boldsymbol{\tau} \in [L^2(\Omega)]^{5 \times 3} \mid \boldsymbol{\tau}|_K \in [\mathcal{P}_p(K)]^{5 \times 3} \; \forall K \in \mathcal{T}_h \},$$
(21)

where  $\mathcal{P}_p(K)$  is the space of polynomial functions of degree at most  $p \geq 1$  on K, and 3 and 5 refer to the dimension and number of solution components of the Navier-Stokes equations in three dimensions. To obtain a form that is appropriate for discretization using the CDG method, we multiply the system of equations (18)-(19) by test functions  $\boldsymbol{v}, \boldsymbol{\tau}$  and integrate by parts. Our semi-discrete DG formulation is then expressed as: find  $\boldsymbol{u}_h \in V_h^p$  and  $\boldsymbol{q}_h \in \Sigma_h^p$  such that for all  $K \in \mathcal{T}_h$ , we have

$$\int_{K} \frac{\partial \boldsymbol{u}_{h}}{\partial t} \cdot \boldsymbol{v} \, dx + \int_{K} \left( \boldsymbol{F}^{i}(\boldsymbol{u}_{h}) - \boldsymbol{F}^{v}(\boldsymbol{u}_{h}, \boldsymbol{q}_{h}) \right) : \nabla \boldsymbol{v} \, dx$$
$$- \oint_{\partial K} \left( \widehat{\boldsymbol{F}^{i}(\boldsymbol{u}_{h})} - \boldsymbol{F}^{v}(\widehat{\boldsymbol{u}_{h}, \boldsymbol{q}_{h}}) \right) \cdot \boldsymbol{v} \, ds \qquad = \mathbf{0}, \qquad \forall \boldsymbol{v} \in [\mathcal{P}_{p}(K)]^{5}, \tag{22}$$

$$\int_{K} \boldsymbol{q}_{h} : \boldsymbol{\tau} \, d\boldsymbol{x} + \int_{K} \boldsymbol{u}_{h} \cdot (\nabla \cdot \boldsymbol{\tau}) \, d\boldsymbol{x} - \oint_{\partial K} (\hat{\boldsymbol{u}}_{h} \otimes \boldsymbol{n}) : \boldsymbol{\tau} \, d\boldsymbol{s} \qquad = \boldsymbol{0}, \qquad \forall \boldsymbol{\tau} \in [\mathcal{P}_{p}(K)]^{5 \times 3}.$$
(23)

To complete the description we need to specify the numerical fluxes for all element boundaries  $\partial K$ . The inviscid fluxes  $\widehat{F^i(u_h)}$  are computed using Roe's method [20], and the modification for our ALE formulation described in [17]. For the viscous fluxes  $\widehat{F_h^v}$ ,  $\hat{u}_h$ , we use a formulation based on the CDG method [21], which is a slight modification of the LDG method [23] to obtain a compact and sparser stencil with improved stability properties.

First, define a switch function  $S_K^{K'} \in \{-1, 1\}$  for each internal face e that element K shares with a neighboring element K'. We require that  $S_K^{K'} = -S_{K'}^K$ , but unlike the standard LDG method no other restrictions are imposed. Here we use the simple *natural switch*, which is positive if the global element number of K is greater than that of K', and negative otherwise. The numerical fluxes are then defined as follows:

• In (23),  $\hat{u}_h$ , is defined by standard "up-winding" according to the switch function:

$$\hat{\boldsymbol{u}}_{h} = \begin{cases} \boldsymbol{u}_{h}' & \text{if } S_{K}^{K'} = +1 \\ \boldsymbol{u}_{h} & \text{if } S_{K}^{K'} = -1, \end{cases}$$
(24)

where  $u'_h$  is the numerical solution defined by the neighboring element K' on the face. Since this expression does not depend on  $q_h$ , (23) can be solved element-wise for the gradients  $q_h$  in each element K, which therefore can be eliminated from the final discrete system.

• In (22), the numerical fluxes  $\widehat{F_h^v}$  are defined by first introducing the "face gradients"  $q_h^e$  for each face e of K, using a slight modification of (23):

$$\int_{K} \boldsymbol{q}_{h}^{e} : \boldsymbol{\tau} \, dx + \int_{K} \boldsymbol{u}_{h} \cdot (\nabla \cdot \boldsymbol{\tau}) \, dx - \oint_{\partial K} (\hat{\boldsymbol{u}}_{h}^{e} \otimes \boldsymbol{n}) : \boldsymbol{\tau} \, ds = \boldsymbol{0}, \qquad \forall \boldsymbol{\tau} \in [\mathcal{P}_{p}(K)]^{5 \times 3} \qquad (25)$$

with

$$\hat{\boldsymbol{u}}_{h}^{e} = \begin{cases} \hat{\boldsymbol{u}}_{h} & \text{on face } e, \text{ from equation (24)}, \\ \boldsymbol{u}_{h} & \text{otherwise.} \end{cases}$$
(26)

These are then used to define the numerical fluxes  $\widehat{F_h^v}$  on face e:

$$\widehat{\boldsymbol{F}_{h}^{v}}^{e} = C_{11}(\boldsymbol{u}_{h}^{\prime} - \boldsymbol{u}_{h}) + \begin{cases} \boldsymbol{F}^{v}(\boldsymbol{u}_{h}^{e}, \boldsymbol{q}_{h}^{e}) \cdot \boldsymbol{n} & \text{if } S_{K}^{K^{\prime}} = +1 \\ \boldsymbol{F}^{v}(\boldsymbol{u}_{h}^{e}, \boldsymbol{q}_{h}^{e}) \cdot \boldsymbol{n} & \text{if } S_{K}^{K^{\prime}} = -1 \end{cases}$$

$$\tag{27}$$

where  $u_h^{e'}, q_h^{e'}$  are the solutions / face gradients from the neighboring element K' on face e. Note that these fluxes can be seen as "down-winding" according to the switch function. The parameter  $C_{11}$  is used for additional stabilization, here we will use a value of  $C_{11} = 10/h_{\min}$  where  $h_{\min}$  is the height of the element with respect to face e.

For more details on the CDG scheme and its properties, including the compact sparsity pattern of the stencils, see [21]. At a boundary face, we impose either far field or no-slip conditions weakly through the fluxes, see [24].

We use standard finite element procedures for the discretization. We define a set of equidistributed nodes  $\boldsymbol{x}_j, j = 1, \ldots, N_p$ , within each element K, where for simplex elements  $N_p = \binom{p+D}{D}$  in D spatial dimensions. We then determine the shape functions as the Lagrange interpolation functions  $\phi_i(\boldsymbol{x}) \in \mathcal{P}_p(K)$  such that  $\phi_i(\boldsymbol{x}_j) = \delta_{ij}$ . Using these, the solution in each element can be written in terms of its discrete expansion coefficients  $\boldsymbol{u}_i$  as:

$$\boldsymbol{u}_h(\boldsymbol{x}) = \sum_{i=1}^n \boldsymbol{u}_i \phi_i(\boldsymbol{x}) \tag{28}$$

and similarly for the auxiliary variable  $q_h$ , the test functions  $v, \tau$ , and the time-derivatives  $\partial u_h/\partial t$ . We evaluate all integrals in (22),(23) using high-order Gaussian quadrature rules, and setting the test function expansion coefficients to the identity matrix and eliminating the local  $q_h$  variables, we obtain the semi-discrete form of our equations:

$$\boldsymbol{M}^{f} \frac{d\boldsymbol{u}^{f}}{dt} = \boldsymbol{r}^{f}(\boldsymbol{u}^{f}), \tag{29}$$

for solution vector  $\boldsymbol{u}^{f}$ , mass matrix  $\boldsymbol{M}^{f}$ , and residual function  $\boldsymbol{r}^{f}(\boldsymbol{u}^{f})$ .

# 3.2. Structure Spatial / CG

The structure equations as described in Sec. 2.3 are discretized as follows. The domain is represented using an unstructured simplicial mesh  $\mathcal{T}_h$ , and curved boundaries are fit using isoparametric elements. On this mesh, we define the space of continuous piecewise polynomials of degree p:

$$V_h^p = \{ \boldsymbol{v} \in [\mathcal{C}_0(\Omega)]^3 \mid \boldsymbol{v}|_K \in [\mathcal{P}_p(K)]^3 \; \forall K \in \mathcal{T}_h \},$$
(30)

We also define the subspaces of functions in  $V_h^p$  that satisfy the non-homogeneous Dirichlet boundary conditions:

$$V_{h,D}^{p} = \{ \boldsymbol{v} \in V_{h}^{p} \mid \boldsymbol{v}|_{\Gamma_{D}} = \boldsymbol{x}_{D} \},$$

$$(31)$$

as well as the homogeneous Dirichlet boundary conditions:

$$V_{h,0}^p = \{ \boldsymbol{v} \in V_h^p \mid \boldsymbol{v}|_{\Gamma_D} = 0 \}.$$

$$(32)$$

By multiplying (12) by an arbitrary test function  $\boldsymbol{z} \in V_{h,0}^p$ , integrating over the domain V, and applying Green's theorem, we obtain our finite element formulation: find  $\boldsymbol{x}_h \in V_{h,D}^p$  such that for all  $\boldsymbol{z} \in V_{h,0}^p$ ,

$$\int_{V} \rho \frac{\partial^{2} \boldsymbol{x}_{h}}{\partial t^{2}} \boldsymbol{z} \, dX = -\int_{V} \boldsymbol{P}(\boldsymbol{G}(\boldsymbol{x}_{h})) : \nabla \boldsymbol{z} \, dX + \oint_{\Gamma_{N}} \boldsymbol{t}(\boldsymbol{x}_{h}) \cdot \boldsymbol{z} \, dS + \int_{V} \boldsymbol{b} \cdot \boldsymbol{z} \, dX \,.$$
(33)

The system of equations (33) is implemented using standard finite element techniques. The discrete solution vector  $\mathbf{X}$  and the test functions are represented at the nodes using nodal basis functions. The integrals are computed using high-order Gauss integration rules. The computed elemental residuals are assembled into a global discrete residual vector  $\mathbf{R}(\mathbf{X})$ , to give the nonlinear ODE

$$\boldsymbol{M}\frac{d^2\boldsymbol{X}}{dt^2} = \boldsymbol{R}(\boldsymbol{X}) \tag{34}$$

which we immediately convert to a first-order system by introducing the velocity  $V_G = d\mathbf{X}/dt$ . The discrete positions and velocities are combined into a single solution vector  $\mathbf{u}^s = [\mathbf{X}; \mathbf{V}_G]$ , corresponding residual vector  $\mathbf{r}^s(\mathbf{u}^s) = [\mathbf{V}_G; \mathbf{R}(\mathbf{X})]$ , and mass matrix  $\mathbf{M}^s = \text{diag}(\mathbf{I}, \mathbf{M})$ , to obtain the semi-discrete form of our equations:

$$\boldsymbol{M}^{s} \frac{d\boldsymbol{u}^{s}}{dt} = \boldsymbol{r}^{s}(\boldsymbol{u}^{s}).$$
(35)

#### 3.3. Coupling / Radial Basis Functions

To simplify the coupling between the fluid and the structure, we insist on two requirements. First, we insist that the boundary faces of the two meshes are coincident. That is, along the fluid-structure boundary, each boundary face of the structure mesh directly matches a boundary face of the fluid mesh. Second, we require that on these interface elements the fluid and structure be discretized using the same polynomial order. In this work we use the same polynomial order for discretization throughout both domains. Because

in our ALE formulation we represent the fluid domain deformation using isoparametric elements, these two requirements ensure that the deformed fluid domain may exactly conform to the deformed structure.

To compute the fluid-to-structure coupling we numerically evaluate the momentum flux through the boundary faces and provide the quantities to the structure solver as the surface traction. To ensure a highly accurate coupling this transfer is done at the Gaussian integration nodes, not the solution nodes.

The structure-to-fluid coupling is a deformation of the fluid mesh in response to a change in the structure position. We represent the deformed fluid mesh and mapping velocity on each element of the fluid mesh using polynomials of the same order as the fluid discretization. Since we insist that the fluid and structure are discretized using the same order polynomials, the deformed fluid mesh will *exactly* conform to the deformed structure by setting the positions of the boundary fluid nodes to the deformed position of the structure. But, it is clear that this process alone is insufficient as the structure may undergo a large deformation, hence requiring interior nodes of the fluid mesh to move as well. There are a several ways to transfer the boundary displacement of the fluid mesh into a displacement of the interior. In this paper, we employ radial basis function interpolation [25, 26] which works well for small to moderate deformations.

Here we seek an interpolant giving the deformed fluid mesh position x as a function of the position X in the reference fluid mesh, and which is of the form

$$\boldsymbol{x}(\boldsymbol{X}) = \sum_{j=1}^{n} \boldsymbol{\alpha}_{j} \phi(\|\boldsymbol{X} - \boldsymbol{X}_{j}\|_{2}) + \boldsymbol{p}(\boldsymbol{X})$$
(36)

where  $\{X_j\}$  is a set of control points,  $\phi$  is a radial basis function, and p is a linear polynomial.

The coefficients  $\alpha_j$  and coefficients of the polynomial p are found by imposing the value of x at the control points  $X_j$ , i.e.,

$$\boldsymbol{x}_j = \boldsymbol{x}(\boldsymbol{X}_j) \quad \text{for } j = 1, \dots, N$$
 (37)

and additionally requiring

$$\sum_{j=1}^{N} \boldsymbol{\alpha}_{j} \boldsymbol{q}(\boldsymbol{X}_{j}) = 0$$
(38)

for all polynomials q of degree less than or equal to the degree of p, i.e. 1.

There are various options for the radial basis function  $\phi$ . Here we choose a compactly supported  $C^2$  function

$$\phi(r) = \begin{cases} (1-r)^4 (4r+1) & \text{if } 0 \le r \le 1\\ 0 & \text{if } 1 \le r. \end{cases}$$
(39)

In our case, the control points  $X_j$  are chosen to be all nodes of the reference fluid mesh which lie on the fluid-structure interface or any other boundary. For the interpolation, the deformed position of the nodes which lie on the fluid-structure interface have their values specified by the corresponding structure displacement. The user is free to choose the positions of the other control points to match the problem description, which in this paper always consists of no displacement.

Given the control values  $\boldsymbol{x}(\boldsymbol{X}_j)$ , the coefficients of the RBF interpolant may be found by solving a dense linear system of N + d + 1 equations in N + d + 1 variables, where N is the number of boundary nodes and d is the spatial dimension. For speed, we often precompute the LU factorization of this linear system, allowing the coefficients  $\alpha_j$  and  $\boldsymbol{p}$  to be efficiently solved using forward and backward substitution. The authors note that this is by no means the only way to solve for such coefficients, and other solution techniques may be more appropriate, especially in three dimensions.

We compute the mapping velocity  $v_G$  as the time derivative of the interpolant (eq. 36),

$$\boldsymbol{v}_G(\boldsymbol{X}) = \sum_{j=1}^N \frac{d\boldsymbol{\alpha}_j}{dt} \phi(\|\boldsymbol{X} - \boldsymbol{X}_j\|_2) + \frac{d\boldsymbol{p}}{dt}(\boldsymbol{X}).$$
(40)

Since the coefficients  $\alpha_j$  and p depend linearly on the displacements at the control nodes, it is clear that the quantities  $d\alpha_j/dt$  and dp/dt may be computed by the same procedure using boundary velocities instead of boundary positions.

# 4. IMPLICIT SOLVERS, TEMPORAL DISCRETIZATION, AND PARTITIONING

#### 4.1. Parallel Newton-Krylov Fluid Solvers

The systems of equations produced by the DG discretization are typically very large. We use polynomials of degree p = 3, which gives 20 degrees of freedom per tetrahedron and solution component. Since we have 5 solution components, a mesh of a hundred thousand tetrahedra (as in §5.4) has 10 million degrees of freedom. In addition, the Jacobian matrices tend to be less sparse than those typically obtained with low-order methods. Although we use an efficient compressed compact storage format [27], each Jacobian has about 3 billion entries, and requires 24GB of storage. It is clear that parallel computers are needed, both for storing these matrices and to perform the computations.

The parallel 3DG code [28] is based on the MPI interface. The domain is decomposed using the METIS software [29] and the discretization and matrix assembly are done in parallel. In order to solve the linear systems in the Newton method, we use an ILU-preconditioned GMRES solver. To maximize the performance of the preconditioner, we order the elements using the Minimum Discarded Fill (MDF) algorithm [27].

The simulations in this paper were done using an appropriate number of cores, from just a handful of cores for low fidelity 2D simulations to over 2048 cores for large 3D simulations. The simulation time depends on the problem, timestep, and tolerances in the Newton and Krylov solvers. In no case did a simulation last longer than 24 hours. For additional details on the performance of the parallel fluid solver, including its nearly perfect weak scaling, see [28].

### 4.2. Sparse Direct Structure Solvers

The system of equations produced by the CG discretization of the structure is generally much smaller than the that of the fluid, both because the structure domain is physically smaller and because the CG discretization avoids the duplicate nodes which would appear in a DG discretization. Nonetheless, we still find it expedient to use a parallel code, again based on MPI. The structure domain is decomposed using the METIS software [29], and the discretization and matrix assembly are each done in parallel. In order to solve the linear systems arising from Newton's method, we call the MUMPS [30, 31] parallel sparse direct solver, providing the matrix in distributed coordinate form.

The prescribed displacement at Dirichlet boundary nodes is enforced by elimination of the corresponding variables from the system of equations.

## 4.3. Implicit-Explicit Runge-Kutta Schemes

Our time integration is based on applying an Implicit-Explicit Runge-Kutta scheme to our system of differential equations. In general, an IMEX scheme is based on an additive splitting of a differential equation

$$M\frac{du}{dt} = f(u) + g(u) \tag{41}$$

into non-stiff terms f(u) and stiff terms g(u). The IMEX scheme consists of two paired Runge-Kutta schemes, an explicit Runge-Kutta scheme  $\hat{A}, \hat{b}, \hat{c}$  which is used to integrate f and a diagonally implicit Runge-Kutta scheme A, b, c which is used to integrate g. Most schemes satisfy  $\hat{c} = c$ , which in particular means that the first stage of the diagonally implicit Runge-Kutta scheme is explicit. In practice this allows for the following implicit stages to have higher stage order. Note that the stage times  $(b \text{ and } \hat{b})$  are not required to be equal, but will be equal in all cases we consider. The algorithm to advance a solution  $u_n$  at time  $t_n$  to a solution  $u_{n+1}$  at time  $t_n + \Delta t$  according to the IMEX scheme is presented in Alg. 1.

Note that in order to take one time step with the IMEX scheme, the algorithm involves s implicit solves of g and s explicit evaluations of f.

**Algorithm 1** Implicit-Explicit Runge-Kutta. Advance a numerical solution  $u_n$  of M(du/dt) = f(u) + g(u) from  $t_n$  to time  $t_n + \Delta t$ .

for stages $i = 1, \ldots, s$ do	
Define the stage solution as $u_n^{(i)} = u_n + \Delta t \sum_{i=1}^{i-1} \hat{a}_{ii} \hat{k}_i + \Delta t \sum_{i=1}^{i} a_{ii} k_i$ .	
$\sum_{j=1}^{n} \sum_{k=1}^{n} \sum_{k=1}^{n} \sum_{ij=1}^{n} \sum_{k=1}^{n} \sum_{ij=1}^{n} \sum_{k=1}^{n} \sum_{ij=1}^{n} \sum_{ij=1}^{n} \sum_{k=1}^{n} \sum_{ij=1}^{n} \sum_{ij=1}^$	
Solve for $k_i$ in $Mk_i = g(u_n^{(i)})$ .	$\triangleright$ Implicit solve of $g$ .
Solve for $\hat{k}_i$ in $M\hat{k}_i = f(u_n^{(i)})$ .	$\triangleright$ Explicit evaluation of $f$ .
end for	
Set $u_{n+1} = u_n + \Delta t \sum_{i=1}^{\infty} \hat{b}_i \hat{k}_i + \Delta t \sum_{i=1}^{\infty} b_i k_i.$	
<i>i</i> =1 <i>i</i> =1	

In this paper we consider the third, fourth, and fifth order IMEX schemes<sup>1</sup> presented by Kennedy & Carpenter [10]. These methods are L-stable, stiffly accurate, and singly diagonally implicit, and have 4, 6, and 8 stages respectively. We will later refer to these schemes by their order, calling them ARK3, ARK4, and ARK5.

# 4.4. Temporal Integrator

Consider for the moment our system of fluid  $u^f$  and structure  $u^s$  variables written as a coupled first order system of ordinary differential equations M du/dt = r(u) where

$$\boldsymbol{u} = \begin{bmatrix} \boldsymbol{u}^f \\ \boldsymbol{u}^s \end{bmatrix}, \quad \boldsymbol{r} = \begin{bmatrix} \boldsymbol{r}^f(\boldsymbol{u}^f; \, \boldsymbol{x}(\boldsymbol{u}^s)) \\ \boldsymbol{r}^s(\boldsymbol{u}^s; \, \boldsymbol{t}(\boldsymbol{u}^f)) \end{bmatrix}, \quad \boldsymbol{M} = \begin{bmatrix} \boldsymbol{M}^f & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{M}^s \end{bmatrix}.$$
(42)

Note that we have highlighted the dependence of the fluid on the structure arising from the ALE mesh motion x, and the structure on the fluid via the surface traction t.

In addition, observe that the discretized structure equation may be separated into two terms

$$\boldsymbol{r}^{s}(\boldsymbol{u}^{s};\boldsymbol{t}(\boldsymbol{u}^{f})) = \boldsymbol{r}^{ss}(\boldsymbol{u}^{s}) + \boldsymbol{r}^{sf}(\boldsymbol{t}(\boldsymbol{u}^{f}))$$

$$(43)$$

where the first gives the structure dynamics in the absence of an applied surface traction and the second accounts for the additional dynamics from the applied surface traction. Since the second term is linear in t, if  $\tilde{t}$  is any other surface traction, we may write the structure equation as

$$\boldsymbol{r}^{s}(\boldsymbol{u}^{s};\boldsymbol{t}(\boldsymbol{u}^{f})) = \boldsymbol{r}^{s}(\boldsymbol{u}^{s};\tilde{\boldsymbol{t}}) + \boldsymbol{r}^{sf}(\boldsymbol{t}(\boldsymbol{u}^{f}) - \tilde{\boldsymbol{t}})$$

$$(44)$$

Here we will generally think of  $\tilde{t}$  as a predicted value of  $t(u^f)$  and refer to it as a predicted fluid-to-structure coupling.

Using this formulation, we may split equation 42 as

$$\boldsymbol{M}\frac{d\boldsymbol{u}}{dt} = \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{r}^{sf}(\boldsymbol{t}(\boldsymbol{u}^{f}) - \tilde{\boldsymbol{t}}) \end{bmatrix} + \begin{bmatrix} \boldsymbol{r}^{f}(\boldsymbol{u}^{f}; \, \boldsymbol{x}(\boldsymbol{u}^{s})) \\ \boldsymbol{r}^{s}(\boldsymbol{u}^{s}; \, \tilde{\boldsymbol{t}}) \end{bmatrix}$$
(45)

where we intend to integrate the first term explicitly and the second term implicitly. Observe here how the predicted fluid-structure coupling allows us to complete the implicit solve in two phases, first calling a structure solver to compute the stage value of  $u^s$  and then calling a fluid solver to compute the stage value of  $u^f$ . We use the parallel solution techniques described in sections 4.1 and 4.2 where the compute cores are reused between each phase. There is no load balancing required between the two phases because they are performed sequentially.

<sup>&</sup>lt;sup>1</sup>Kennedy & Carpenter's paper calls these schemes ARK3(2)4L[2]SA, ARK4(3)6L[2]SA, and ARK5(4)8L[2]SA respectively.

Our scheme differs slightly from the standard IMEX formulation in that we avoid evaluating the explicit terms  $r^{sf}$  but instead update the stage flux for the structure equation using the corrected value of the coupling  $t(u^f)$ . Note that this naturally allows per stage Gauss-Seidel iterations where we treat the corrected fluid-to-structure coupling as the new predicted value and repeat the stage calculations. It has been reported that using one or more Gauss-Seidel iterations can improve the stability of the overall method [11], but we emphasize that these iterations are generally not required to achieve the design accuracy of the method.

Here we use the predictor suggested by van Zuijlen et al [12], namely the predicted value  $\tilde{t}$  at stage *i* is a linear combination of the corrected values t at previous stages

$$\tilde{\boldsymbol{t}}_i = \sum_{j=1}^{i-1} \frac{\hat{a}_{ij} - a_{ij}}{a_{ii}} \boldsymbol{t}_j \tag{46}$$

where  $a_{ij}$  (resp.  $\hat{a}_{ij}$ ) are the coefficients from the implicit (resp. explicit) Runge-Kutta integration scheme. The time integration scheme is written out fully in Alg. 2.

**Algorithm 2** Time integration scheme for the FSI system in equation 42. Given the structure  $(\boldsymbol{u}_n^s)$  and fluid  $(\boldsymbol{u}_n^f)$  values at time  $t_n$ , compute the values at  $t_{n+1} = t_n + \Delta t$ .

Set $\boldsymbol{t}_{n,1} = \boldsymbol{t}(\boldsymbol{u}_{n,1}^f)$	▷ Evaluate fluid-to-structure coupling (surface traction)
Set $\boldsymbol{k}_{n,1}^s = M_s^{-1} \boldsymbol{r}^s(\boldsymbol{u}_{n,1}^s, \boldsymbol{t}_{n,1})$	$\triangleright$ Evaluate structure residual
Set $\boldsymbol{k}_{n,1}^{f} = M_{f}^{-1} \boldsymbol{r}^{f}(\boldsymbol{u}_{n,1}^{f}, \boldsymbol{x}(\boldsymbol{u}_{n,1}^{s}))$	▷ Evaluate fluid residual
for stage $i = 2, \ldots, s$ do	
Set $ ilde{m{t}}_{n,i} = \sum_{j=1}^{i-1} rac{\hat{a}_{ij} - a_{ij}}{a_{ii}} m{t}_{n,j}$	$\triangleright$ Predict the fluid-to-structure coupling
Solve for $\boldsymbol{k}_{n,i}^s$ in $M_s \boldsymbol{k}_{n,i}^s = \boldsymbol{r}^s(\boldsymbol{u}_{n,i}^s, \tilde{\boldsymbol{t}}_{n,i})$	$\triangleright$ Implicit structure solve
where $oldsymbol{u}_{n,i}^s = oldsymbol{u}_n^s + \Delta t \sum_{j=1}^i a_{ij} oldsymbol{k}_{n,j}^s$	
Solve for $\boldsymbol{k}_{n,i}^f$ in $M_f \boldsymbol{k}_{n,i}^f = \boldsymbol{r}^f(\boldsymbol{u}_{n,i}^f, \boldsymbol{x}(\boldsymbol{u}_{n,i}^s))$	$\triangleright$ Implicit fluid solve
where $oldsymbol{u}_{n,i}^f = oldsymbol{u}_n^f + \Delta t \sum_{j=1}^i a_{ij} oldsymbol{k}_{n,j}^f$	
Set $\boldsymbol{t}_{n,i} = \boldsymbol{t}(\boldsymbol{u}_{n,i}^f)$	$\triangleright$ Correct the fluid-to-structure coupling
Set $oldsymbol{k}_{n,i}^s = M_s^{-1} oldsymbol{r}^s(oldsymbol{u}_{n,i}^s,oldsymbol{t}_{n,i})$	$\triangleright$ Re-evaluate structure residual
end for	
Set $\boldsymbol{u}_{n+1}^s = \boldsymbol{u}_n^s + \Delta t \sum_{i=1}^s b_i \boldsymbol{k}_{n,i}^s$	$\triangleright$ Advance structure
Set $\boldsymbol{u}_{n+1}^f = \boldsymbol{u}_n^f + \Delta t \sum_{i=1}^s b_i \boldsymbol{k}_{n,i}^f$	$\triangleright$ Advance fluid

### 5. RESULTS

# 5.1. Pitching and Heaving Airfoil

To validate the high-order convergence in time, we considered a simple test problem consisting of a pitching and heaving NACA 0012 airfoil. The airfoil is allowed rotate around a fixed pivot in the interior of the airfoil, as shown in Figure 1. Since the airfoil is treated as a rigid body, the structure variables consist only of the pitching angle  $\theta$  and the angular velocity  $\omega$ . The fluid is assigned no-slip boundary conditions on the interface with the airfoil, which contributes a torque  $\tau$  about the pivot of the airfoil.

The position of the pivot follows a prescribed vertical motion y(t) between t = 0 and t = 1 which is a Hermite polynomial satisfying y(0) = 0, y(1) = 1/4, and y'(0) = y'(1) = 0. In addition the airfoil is subjected to a torsional restoring force with torsional spring constant k. The equations of motion of the airfoil written as a first order system are

$$\frac{\partial \theta}{\partial t} = \omega \tag{47}$$

$$I\frac{\partial\omega}{\partial t} = -k\theta - \tau - lm\cos(\theta)y''(t) \tag{48}$$



Figure 1: Schematic for the pitching and heaving airfoil.

where I is the moment of inertia (around the pivot), l is the distance from the pivot to the center of mass, and m is the total mass of the airfoil.

The non-dimensionalized constants chosen for this problem were I = 1, k = 0.1, l = 0.2, and M = 1. The airfoil has chord length 1, and the pivot located along the midline a distance 1/3 from the leading edge. The far field fluid has velocity  $\boldsymbol{u} = [1, 0]^T$ , density 1, Mach 0.2, and a Reynolds number of 1000. To initialize the system at time t = 0 we let the structure be at rest and solve for the steady state solution of the fluid. The evolution of the system is shown in Figure 2.

To validate the temporal convergence of the scheme we measured the relative error in the angle of attack  $\theta(t)$  as compared to the solution of the same system using an explicit fourth order Runge-Kutta method with a suitably small timestep. A plot of the observed relative error as a function of timestep for the third, fourth, and fifth order ARK coefficients is shown in Figure 3. Note that in each case, the scheme exhibits convergence at the designed rate. For comparison we also solved the fully-coupled (monolithic) fluid-structure system using the implicit coefficients of the ARK scheme, by performing many Gauss-Seidel subiterations until achieving numerical convergence. This resulted in a negligible increase in accuracy despite a tremendous increase in computational cost. The plot demonstrates that this partitioned approach can attain up to 5th order accuracy in time, without subiterations or a specialized coupled solver.

#### 5.2. Cantilever Behind Rigid Square Body

Next we consider a variation on a standard fluid-structure interaction benchmark [32], which consists of a flexible cantilever behind a rigid square body as shown in Figure 4.

The cantilever and square body are assigned no-slip boundary conditions. We impose a far field boundary condition at the far walls, with the far field conditions corresponding to uniform flow to the right at 51.3 cm/s. To approximate incompressible flow, we assign a far field Mach number of 0.2. The flow has Reynolds number Re = 333 based on the dimension of the bluff body (1 cm).

The initial conditions are not consistently defined in the literature, and it has been observed that there is some sensitivity to the initial conditions in the results [33]. Here we follow [34] and start with imperfect flow conditions and the cantilever at rest.

We model the cantilever using the neo-Hookean formulation as described in Sec. 2.3, instead of the St. Venant-Kirchhoff model as the test problem describes. We are careful to assign the same elastic moduli, which are specified as Young's modulus  $E = 2.5 \times 10^5$  Pa and Poisson's ratio  $\nu = 0.35$ . The shear and bulk moduli are then calculated as  $\mu = E/(2(1 + \nu))$  and  $\kappa = E/(3(1 - 2\nu))$ .

The fluid domain was triangulated using 6576 degree 3 elements, for a total of 65,760 high-order nodes. The cantilever was triangulated using 64 degree 3 elements. The system was integrated in time using the ARK3 coefficients and a fixed time step of  $1 \times 10^{-3}$  s. One Gauss-Seidel iteration was performed at each integration stage to increase the stability of the coupling.

The Reynolds number considered is high enough that the flow separates at the bluff body and produces a von Kármán vortex street. This causes the cantilever to begin oscillating and after a period of a few seconds



Figure 2: The airfoil at various times (Mach number). The pivot location of the airfoil is smoothly moved upwards between time t = 0 and t = 1.

the fluid-cantilever system settles into a nearly periodic state. Figure 6 shows the vertical displacement of the tip as a function of time.

The observed tip vertical amplitude and oscillation frequency are compared to the existing literature in Table 1. Our observed a maximal tip amplitude of 1.12 cm and oscillation frequency of 3.18 Hz show good agreement with values obtained in the literature, which were computed using different fluid, structure, and temporal discretizations.

#### 5.3. Membrane, 2D

Next we considered a thin rectangular membrane with length 1 and height 0.01 in uniform incoming flow at a  $10^{\circ}$  angle of attack. This structure was modeled using the standard volumetric equations as described in Section 2.3, with highly anisotropic elements. We applied no-slip conditions on the membrane boundary and far field boundary conditions on the far fluid domain boundaries. The far field flow was set to unit density, unit velocity in the x direction, Mach 0.2, and Reynolds number of 1000. We assigned Dirichlet boundary conditions of no displacement to the front and rear faces of the membrane.

The membrane was set to a non-dimensionalized density of  $\rho = 40.0$ , Poisson's ratio of  $\nu = 0.3$ . We explored two different Young's moduli of  $E = 1 \times 10^3$  and  $E = 5 \times 10^3$ . For comparison we also investigated a fixed, rigid plate.

The membrane was discretized using 44 degree 3 triangular elements, and the fluid was discretized using 2575 degree 3 triangular elements. See Figure 7. The system was integrated in time using the ARK3 coefficients and a timestep of  $2 \times 10^{-3}$ .

A time history of the three cases is shown in Fig. 8. Here we see that the rigid plate is causing significant leading edge separation. This is avoided in the flexible membranes which are able to align with the incident flow, resulting in smaller vortices, at least for the higher stiffness membrane.

The lift and drag coefficients as a function of time for the rigid plate and two membranes are plotted in Fig. 9. In each case the coefficients were computed using a planform area of 1. The long term trend shows



Figure 3: The relative error in angle of attack,  $\|\theta(t) - \theta_{exact}(t)\|_{\infty}/\|\theta_{exact}(t)\|_{\infty}$ , as a function of timestep. The ARK3, ARK4, and ARK5 schemes achieve the expected order of accuracy. Solving the fully-coupled ("FC-") system using the implicit method from the IMEX scheme shows a negligible increase in accuracy despite a large increase in computational cost. A basic staggered weak coupling scheme is shown for comparison.



Figure 4: Flexible cantilever behind a rigid square body. All distances shown are in cm.



Figure 5: The the cantilever near maximal displacement (Entropy).



Figure 6: The vertical displacement of the cantilever tip as a function of time.

Author	Fluid	Structure	Coupling	f(Hz)	$d_{max}(\mathrm{cm})$
Kassiotis et al.[35]	FVM	FEM	P-BGS	2.98	1.05
Wood et al.[36]	FVM	FEM	P-BGS	2.94	1.15
Yvin[37]	FVM	FEM	P-BGS	3.16	1.20
Olivier et al. $[38]$	FVM	$\mathbf{FVM}$	P-BGS	3.17	0.95
Habchi et al.[39]	FVM	FVM	P-BGS	3.25	1.02
Walhorn et al.[34]	Stabilized FEM	FEM	P-BGS	3.14	1.02
Wall and $\operatorname{Ramm}[32]$	Stabilized FEM	FEM	P-BGS	2.99	1.22
Matthies and Steindorf[40]	FVM	FEM	P-BN	3.13	1.18
Dettmer and Perić[41]	Stabilized FEM	FEM	P-NR	3.03	1.25
Present study	DG FEM	FEM	IMEX	3.18	1.12

Table 1: A comparison of the oscillation frequency and maximal vertical tip displacement of the cantilever from the open literature, as reproduced from [39]. The coupling abbreviations stand for partitioned block Gauss-Seidel, partitioned block-Newton, and partitioned Newton-Raphson.



Figure 7: (a) The undeformed fluid (green) and structure (blue) meshes for the 2D Membrane at a  $10^{\circ}$  angle of attack. (b) The region near the structure is enlarged.

that the flexible membranes are able to increase the lift coefficient without a significant increase in drag, agreeing with the results in [9, 42].

#### 5.4. Membrane, 3D

In three dimensions we considered an extruded form of the 2D membrane module from Sec. 5.3. The membrane has length and width 1 and height 0.01 and is placed in uniform flow at an  $atan(5/12) \approx 22.6^{\circ}$  angle of attack. The fluid was assigned no-slip conditions at the membrane boundary and far field conditions at the far boundaries. The far field flow was set to unit density  $\rho = 1.0$ , unit velocity  $\boldsymbol{u} = [1.0, 0, 0.]^T$ , Mach 0.2 and Reynolds number of 2000. The membrane was assigned Dirichlet boundary conditions on the leading and trailing faces. The physical parameters of the membrane were chosen as density  $\rho = 100.0$ , Young's modulus  $E = 1 \times 10^3$ , and Poisson's ratio  $\nu = 0.35$ .

The membrane was discretized using 1317 highly anisotropic degree 3 elements. The fluid mesh had 108,358 degree 3 elements, for a total of about 2.17 million high-order nodes or almost 11 million degrees of freedom. A cross-section of the fluid mesh is shown in Fig. 10.

A timestep of  $1 \times 10^{-3}$  was used and the system was solved until T = 3.0. The Mach number is shown on iso-entropy surfaces for several time steps in Figure 11. Here we see that the leading edge of the membrane aligns with the incoming fluid and successfully prevents leading edge separation. In addition we see that the fluid curls around the sides of the membrane and exhibits a classic roll-up behavior. For comparison, in Figure 12 we show the behavior of a fluid when the membrane is replaced by a fixed rigid plate of the same dimensions.

#### 6. CONCLUSIONS

We have presented a high-order accurate scheme for fluid-structure interaction problems. By using a predictor for the fluid-to-structure coupling, the method allows the reuse of existing domain specific fluid and structure solvers while still maintaining a high-order of time accuracy. The accuracy in time was verified



Figure 8: The rigid plate and two membranes at time T = 1.0 (top), 2.0, 3.0, 4.0, 5.0, 6.0, and 7.0 (bottom). (Entropy).



Figure 9: Lift and drag coefficients as a function of time for a rigid plate and two flexible membranes at  $10^{\circ}$  angle of attack.



Figure 10: A cross section of the fluid mesh (blue) and entire structure mesh (green) in the reference configuration (left) and typical deformed configuration (right).



Figure 11: A three dimensional membrane at various times (Mach number on iso-entropy surfaces). The leading edge of the membrane aligns with the fluid and prevents separation.

using a grid convergence study. The overall implementation was validated by comparing results of a standard test FSI test problem to other values reported in the literature. Lastly, we demonstrated the applicability of this method to large three dimensional simulations.

In some cases one subiteration was used at each stage of the time integration scheme to improve the stability of the method. Further work will be required to fully understand the stability gained from one or more subiterations.

## 7. ACKNOWLEDGMENTS

The authors greatly appreciate the reviewers comments and suggestions that have undoubtedly improved this paper.

We would like to acknowledge the generous support from the AFOSR Computational Mathematics program under grant FA9550-10-1-0229, the Alfred P. Sloan foundation, and the Director, Office of Science, Computational and Technology Research, U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

#### References

- Z. Yosibash, R. M. Kirby, K. Myers, B. Szabó, G. Karniadakis, High-order finite elements for fluid-structure interaction problems, in: 44th AIAA/ASME/ASCE/AHS Structures, Structural Dynamics, and Materials Conference, Norfolk, Virginia. AIAA-2003-1729.
- [2] J. J. Reuther, J. J. Alonso, J. R. R. A. Martins, S. C. Smith, A coupled aero-structural optimization method for complete aircraft configurations, in: 37th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada. AIAA-99-0187.
- [3] P. Geuzaine, C. Grandmont, C. Farhat, Design and analysis of ALE schemes with provable second-order time-accuracy for inviscid and viscous flow simulations, J. Comput. Phys. 191 (2003) 206–227.
- [4] C. Farhat, P. Geuzaine, Design and analysis of robust ALE time-integrators for the solution of unsteady flow problems on moving grids, Comput. Methods Appl. Mech. Engrg. 193 (2004) 4073–4095.
- [5] C. S. Venkatasubban, A new finite element formulation for ALE (arbitrary Lagrangian Eulerian) compressible fluid mechanics, Internat. J. Engrg. Sci. 33 (1995) 1743–1762.
- [6] I. Lomtev, R. M. Kirby, G. E. Karniadakis, A discontinuous Galerkin ALE method for compressible viscous flows in moving domains, J. Comput. Phys. 155 (1999) 128–159.



Figure 12: A rigid plate at various times (Mach number on iso-entropy surfaces). Note the leading edge separation caused by the high angle of attack.

- [7] H. T. Ahn, Y. Kallinderis, Strongly coupled flow/structure interactions with a geometrically conservative ALE scheme on general hybrid meshes, J. Comput. Phys. 219 (2006) 671–696.
- [8] J. Cori, S. Etienne, D. Pelletier, A. Garon, Implicit Runge-Kutta time integrators for fluid-structure interactions, in: 48th AIAA Aerospace Sciences Meeting and Exhibit, Orlando, Florida. AIAA-2010-1445.
- [9] P.-O. Persson, J. Peraire, J. Bonet, A high order discontinuous Galerkin method for fluid-structure interaction, in: 18th AIAA Computational Fluid Dynamics Conference, Miami, Florida. AIAA-2007-4327.
- [10] C. A. Kennedy, M. H. Carpenter, Additive Runge-Kutta schemes for convection-diffusion-reaction equations, Appl. Numer. Math. 44 (2003) 139–181.
- [11] A. van Zuijlen, Fluid-Structure Interaction Simulations: Efficient Higher Order Time Integration of Partitioned Systems, Ph.D. thesis, TU Delft, 2006.
- [12] A. van Zuijlen, A. de Boer, H. Bijl, Higher-order time integration through smooth mesh deformation for 3D fluid-structure interaction simulations, J. Comput. Phys. 224 (2007) 414–430.
- S. Piperno, C. Farhat, B. Larrouturou, Partitioned procedures for the transient solution of coupled aeroelastic problems.
   I. Model problem, theory and two-dimensional application, Comput. Methods Appl. Mech. Engrg. 124 (1995) 79–112.
- [14] C. Farhat, M. Lesoinne, Two efficient staggered algorithms for the serial and parallel solution of three-dimensional nonlinear transient aeroelastic problems, Comput. Methods Appl. Mech. Engrg. 182 (2000) 499–515.
- [15] W. G. Dettmer, D. Perić, A new staggered scheme for fluid-structure interaction, Internat. J. Numer. Methods Engrg. 93 (2013) 1–22.
- [16] C. A. Felippa, K. Park, C. Farhat, Partitioned analysis of coupled mechanical systems, Computer Methods in Applied Mechanics and Engineering 190 (2001) 3247–3270.
- [17] P.-O. Persson, J. Bonet, J. Peraire, Discontinuous Galerkin solution of the Navier-Stokes equations on deformable domains, Comput. Methods Appl. Mech. Engrg. 198 (2009) 1585–1595.
- [18] G. A. Holzapfel, Nonlinear solid mechanics, John Wiley & Sons Ltd., Chichester, 2000. A continuum approach for engineering.
- [19] J. Bonet, R. D. Wood, Nonlinear continuum mechanics for finite element analysis, Cambridge University Press, Cambridge, 1997.
- [20] P. L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, J. Comput. Phys. 43 (1981) 357–372.
- [21] J. Peraire, P.-O. Persson, The compact discontinuous Galerkin (CDG) method for elliptic problems, SIAM J. Sci. Comput. 30 (2008) 1806–1824.
- [22] D. N. Arnold, F. Brezzi, B. Cockburn, L. D. Marini, Unified analysis of discontinuous Galerkin methods for elliptic problems, SIAM J. Numer. Anal. 39 (2001/02) 1749–1779.
- [23] B. Cockburn, C.-W. Shu, The local discontinuous Galerkin method for time-dependent convection-diffusion systems, SIAM J. Numer. Anal. 35 (1998) 2440–2463.
- [24] J. Peraire, P.-O. Persson, Adaptive High-Order Methods in Computational Fluid Dynamics, volume 2 of Advances in CFD, World Scientific Publishing Co.
- [25] A. Beckert, H. Wendland, Multivariate interpolation for fluid-structure-interaction problems using radial basis functions,

Aerosp. Sci. Technol. 5 (2001) 125-134.

- [26] A. de Boer, M. van der Schoot, H. Bijl, Mesh deformation based on radial basis function interpolation, Comput. Struct. 85 (2007) 784–795.
- [27] P.-O. Persson, J. Peraire, Newton-GMRES preconditioning for discontinuous Galerkin discretizations of the Navier-Stokes equations, SIAM J. Sci. Comput. 30 (2008) 2709–2733.
- [28] P.-O. Persson, Scalable parallel Newton-Krylov solvers for discontinuous Galerkin discretizations, in: 47th AIAA Aerospace Sciences Meeting and Exhibit, Orlando, Florida. AIAA-2009-606.
- [29] G. Karypis, V. Kumar, A fast and high quality multilevel scheme for partitioning irregular graphs, SIAM J. Sci. Comput. 20 (1998) 359–392.
- [30] P. R. Amestoy, I. S. Duff, J. Koster, J.-Y. L'Excellent, A fully asynchronous multifrontal solver using distributed dynamic scheduling, SIAM Journal on Matrix Analysis and Applications 23 (2001) 15–41.
- [31] P. R. Amestoy, A. Guermouche, J.-Y. L'Excellent, S. Pralet, Hybrid scheduling for the parallel solution of linear systems, Parallel Computing 32 (2006) 136–156.
- [32] W. A. Wall, E. Ramm, Fluid-structure interaction based upon a stabilized (ALE) finite element method, in: S. R. Idelsohn, E. Oñate, E. N. Dvorkin (Eds.), 4th World Congress on Computational Mechanics: New Trends and Applications, CIMNE, Barcelona, Spain, 1998.
- [33] B. Hübner, E. Walhorn, D. Dinkler, A monolithic approach to fluid-structure interaction using space-time finite elements, Comput. Methods Appl. Mech. Engrg. 193 (2004) 2087–2104.
- [34] E. Walhorn, B. Hübner, D. Dinkler, Space-time finite elements for fluid-structure interaction, PAMM 1 (2002) 81–82.
- [35] C. Kassiotis, A. Ibrahimbegovic, R. Niekamp, H. G. Matthies, Nonlinear fluid-structure interaction problem. Part I: implicit partitioned algorithm, nonlinear stability proof and validation examples, Comput. Mech. 47 (2011) 305–323.
- [36] C. Wood, A. Gil, O. Hassan, J. Bonet, Partitioned block-Gauss-Seidel coupling for dynamic fluid-structure interaction, Comput. Struct. 88 (2010) 1367–1382.
- [37] C. Yvin, Partitioned fluid-structure interaction with open-source tools, in: 12ème Journées de l'Hydrodynamique, Nantes, France.
- [38] M. Olivier, J.-F. Morissette, G. Dumas, A fluid-structure interaction solver for nano-air-vehicle flapping wings, in: 19th AIAA Computational Fluid Dynamics, San Antonio, Texas. AIAA-2009-3676.
- [39] C. Habchi, S. Russeil, D. Bougeard, J.-L. Harion, T. Lemenand, A. Ghanem, D. D. Valle, H. Peerhossaini, Partitioned solver for strongly coupled fluid-structure interaction, Comput. & Fluids 71 (2013) 306–319.
- [40] H. G. Matthies, J. Steindorf, Partitioned strong coupling algorithms for fluid-structure interaction, Comput. Struct. 81 (2003) 805–812.
- [41] W. Dettmer, D. Perić, A computational framework for fluid-structure interaction: Finite element formulation and applications, Comput. Methods Appl. Mech. Engrg. 195 (2006) 5754–5779.
- [42] D. J. Willis, E. R. Israeli, P.-O. Persson, M. Drela, J. Peraire, Multifidelity approaches for the computational analysis and design of effective flapping wing vehicles, in: 46th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada. AIAA-2008-518.