High-Order Methods for Low Reynolds Number Flows around Moving Obstacles Based on Universal Meshes

Evan S. Gawlik, Hardik Kabaria and Adrian J. Lew

SUMMARY

We propose a family of methods for simulating two-dimensional incompressible, low Reynolds number flow around a moving obstacle whose motion is prescribed. The methods make use of a universal mesh: a fixed background mesh that adapts to the geometry of the immersed obstacle at all times by adjusting a few elements in the neighborhood of the obstacle’s boundary. The resulting mesh provides a conforming triangulation of the fluid domain over which discretizations of any desired order of accuracy in space and time can be constructed using standard finite element spaces together with off-the-shelf time integrators. We demonstrate the approach by using Taylor-Hood elements to approximate the fluid velocity and pressure. To integrate in time, we consider implicit Runge-Kutta schemes as well as a fractional step scheme. We illustrate the methods and study their convergence numerically via examples that involve flow around obstacles that undergo prescribed deformations. Copyright © 0000 John Wiley & Sons, Ltd.

KEY WORDS: Fluid-structure interaction; viscous flow; moving interface

1. INTRODUCTION

One of the key challenges in numerical simulations of fluid flow around moving obstacles is the discretization of an evolving domain, namely, the domain occupied by the fluid. Commonly, this challenge is addressed using one of two tools: a deforming mesh, which deforms in concert with the moving fluid domain, or a fixed mesh, which triangulates or quadrangulates a larger domain in which the moving boundary is immersed for all times. In contrast, this paper presents a family of methods for simulating two-dimensional incompressible, low Reynolds number flow around moving obstacles with prescribed evolution using a universal mesh: a background triangulation that contains the fluid domain for all times and conforms to its geometry at all times by perturbing a small number of nodes in the neighborhood of the immersed fluid boundary.

The merits of this strategy are made most apparent when one considers the challenges associated with the construction of numerical methods for fluid flow around obstacles that are simultaneously robust and accurate to high order in space and time. The latter goal is particularly elusive for problems on moving domains, since errors in the discretization of the domain’s geometry (and in the discretization, if any, of its temporal evolution) can dictate the order of a method. This consideration renders deforming-mesh methods attractive, especially if implemented using curved elements along the boundary. In this light, it is perhaps surprising that many examples of deforming-mesh methods in the literature, with a few noteworthy exceptions [1–10], are often restricted to at
most second-order accuracy [11–19]. Moreover, robustness of the mesh motion poses a challenge to
deforming-mesh methods [20, 21], which often prescribe a motion for the mesh by solving systems
of equations (such as those of linear elasticity) for the positions of mesh nodes [22–26]. Regardless
of a deforming-mesh method’s mesh motion strategy, sufficiently large domain deformations may
lead to element distortions (or, in more severe cases, element inversions) that are detrimental both
to the accuracy of the spatial discretization and to the conditioning of the discrete governing
equations [27]. These considerations can ultimately mandate that the domain be remeshed from
scratch at various instants during a simulation [22, 28–30].

Fixed-mesh methods circumvent the difficulty of designing a robust mesh motion at the expense
of geometric conformity. As a consequence, fixed-mesh methods require special care in order to
account for the disagreement between the immersed obstacle boundary and element interfaces.
A variety of techniques aim to deal with this discrepancy, including adaptive refinement in
the neighborhood of the immersed boundary [31–33], cutting elements [34–38], enriching finite
element spaces [39, 40], cutting elements and enriching finite element spaces [41–43], Nitsche-
inspired methods [44, 45], smearing the interface [46, 47], modifying finite-difference stencils
near the boundary [33, 48–50], and introducing surrogate forcing terms in lieu of the boundary
conditions [51, 52]. Integration in time poses an additional challenge for fixed-mesh methods, since
nodes of the background mesh may occupy differing states (fluid vs. solid) over the course of a single
time step. This peculiarity is known to introduce numerical artifacts such as spurious oscillations
in the pressure field for some fixed-mesh methods [37, 53, 54]. Furthermore, even if a given spatial
discretization is known to deliver high-order spatial accuracy for steady flows around an embedded
obstacle, its incorporation into a numerical method for unsteady flow around moving obstacles with
high spatial and temporal accuracy is arguably a nontrivial task. These observations help to explain
why many fixed mesh methods, again with a few notable exceptions [55, 56], are often restricted to
first- or second-order accuracy [50, 52, 57–60].

The framework presented in this paper distinguishes itself from the preceding approaches by
exhibiting the following features simultaneously. First, a universal mesh delivers a conforming
representation of the evolving fluid domain at all times. This conforming mesh is obtained by
perturbing the nodes of a background mesh using a mapping which supplies not only an adaptation
of the background mesh, but also a mesh motion over short time intervals suitable for constructing
high-order discretizations of the governing equations. Second, the mesh motion strategy is robust, in
the sense that large domain deformations pose no threat to the quality of the conforming mesh, being
at all times derived from a small perturbation of the background mesh. Third, our approach provides
a systematic framework for constructing methods of a desired order of accuracy in space and in
time for low Reynolds number flows, simply by discretizing in space with a finite element space
of the appropriate order and choosing a time integrator of the appropriate order. We demonstrate
this by combining high-order Taylor-Hood elements with high-order implicit Runge Kutta schemes.
Finally, the framework is algorithmically simple. In its basic form, the alteration of the background
mesh requires adjustments to nodal coordinates only, not the mesh’s connectivity, and the nodal
motions are independent and explicitly defined.

To simplify the presentation and to emphasize the main contributions of the present work, we
restrict our attention to problems for which the flow has a low Reynolds number, the obstacle
boundary is smooth ($C^2$-regular), and the obstacle motion is prescribed. Needless to say, higher
Reynolds number flows pose additional challenges (the need for high resolution in boundary layers
and for stabilization of convective terms in the spatial discretization) that warrant enhancements to
the present strategy to ensure its viability. Likewise, the design of universal meshes for domains with
lower regularity, such as domains with corners, remains an area of active research. Unprescribed
obstacle motions would, of course, introduce additional complexity into the framework, but only in
the sense that additional unknowns would need to be solved for concurrently with the fluid variables.

**Organization.** This paper is organized as follows. In Section 2, we recall the governing equations
for incompressible, viscous flow around a moving obstacle with prescribed evolution, and we
recast the equations in weak form. In Section 3, we propose a discretization of the aforementioned
HIGH-ORDER METHODS FOR FLOWS AROUND MOVING OBSTACLES

Figure 1. Fluid domain $\Omega^t = D \setminus P^t$.

equations using a universal mesh in conjunction with Taylor-Hood finite elements [61]. To integrate in time, we propose the use of implicit Runge-Kutta schemes as well as a fractional step scheme. In Section 4, we apply the proposed methods to simulate flow around various obstacles with prescribed evolution: a rotating ellipse, an oscillating disk, and a rotating stirrer. We study numerically the convergence orders of the methods in the context of the rotating ellipse, where an analytical solution is readily manufactured. We close with some concluding remarks in Section 5.

2. PROBLEM

We study incompressible, viscous fluid flow around a moving obstacle immersed in a domain $D \subset \mathbb{R}^2$. We denote by $P^t \subset D$ the domain occupied by the obstacle at time $t$ and by $\Omega^t = D \setminus P^t$ the domain occupied by the fluid. Taking the fluid density to be everywhere unity, the governing equations for the velocity $u$ and pressure $p$ read

$$
\frac{\partial u}{\partial t} + u \cdot \nabla_x u - \nu \Delta_x u = -\nabla_x p \quad \text{in} \quad D \setminus P^t \tag{1}
$$

$$
\nabla_x \cdot u = 0 \quad \text{in} \quad D \setminus P^t, \tag{2}
$$

where $\nu > 0$ is the kinematic viscosity of the fluid. On the interface between the obstacle and the fluid, the no-slip condition holds:

$$
u(x,t) = v_P(x,t), \quad x \in \partial P^t \tag{3}
$$

where $v_P(x,t)$ is the prescribed velocity of the obstacle at $x \in \partial P^t$. On the remainder of the fluid boundary $\partial D$, and depending on the example under consideration, we impose either the natural boundary conditions

$$
\nu n - \nu \left( \nabla_x u + (\nabla_x u)^T \right) n = 0 \quad \text{on} \quad \partial D \tag{4}
$$

or the no-slip condition

$$
u = 0 \quad \text{on} \quad \partial D, \tag{5}
$$

and in this last case the pressure field is defined up to a constant.

**Weak Formulation.** For later use, it is convenient to record a weak formulation of (1-2). Let us introduce two collections of function spaces, one for each choice of boundary conditions discussed above. When the boundary conditions are given by (3-4), we denote

$$
\mathcal{V}^t = \{ u \in H^1(\Omega^t)^2 \mid u = 0 \text{ on } \partial P^t \}
$$

$$
\mathcal{Q}^t = L^2(\Omega^t).
$$
When the boundary conditions are given by (3) and (5), we denote
\[
\hat{\mathcal{V}}^t = H^1_0(\Omega^t)^2,
\mathcal{V}^t = \{ u \in H^1(\Omega^t)^2 \mid u = v_P(\cdot, t) \text{ on } \partial P^t, u = 0 \text{ on } \partial \mathcal{D} \},
\mathcal{Q}^t = L^2(\Omega^t)/\mathbb{R}.
\]

In either of these two settings, a weak formulation of (1-2) reads: Find \( u(\cdot, t) \in \mathcal{V}^t \) and \( p(\cdot, t) \in \mathcal{Q}^t \) such that
\[
m^t(\dot{u}, w) + a^t(u, w) + e^t(w, p) = 0 \quad \forall w \in \hat{\mathcal{V}}^t \tag{6}
\]
\[
e^t(u, q) = 0 \quad \forall q \in \mathcal{Q}^t \tag{7}
\]
for every \( t \in (0, T] \), where the meanings of \( \hat{\mathcal{V}}^t \), \( \mathcal{V}^t \), and \( \mathcal{Q}^t \) depend upon the boundary conditions under consideration, and
\[
m^t(u, w) = \int_{\Omega^t} u \cdot w \, dx
\]
\[
a^t(u, w) = \nu \int_{\Omega^t} (\nabla_x u + (\nabla_x u)^T) : \nabla_x w \, dx + \int_{\Omega^t} (u \cdot \nabla_x u) \cdot w \, dx
\]
\[
e^t(u, p) = -\int_{\Omega^t} (\nabla \cdot u)p \, dx.
\]

The well-posedness of this system in the case of a fixed domain without advection is proven in [62].

3. METHOD

In this section, we propose a discretization of (1-2) that is based upon the use of a universal mesh. Our approach follows that of [63], where a framework for constructing numerical methods for moving-boundary problems using universal meshes is introduced in the context of a parabolic model problem.

The discretization proceeds in several steps: (1) partitioning the temporal axis into short time intervals \( \bigcup_{n=1}^N (t^{n-1}, t^n) = (0, T] \); (2) constructing a conforming mesh \( S_h(t) \) for \( \Omega^t \), \( t \in (t^{n-1}, t^n] \), over each short time interval by adapting the universal mesh; (3) performing a Galerkin projection of the governing equations onto a finite element space associated with \( S_h(t) \) over each short time interval; and (4) choosing a time integrator to numerically integrate the resulting system of ODE’s over \( (t^{n-1}, t^n] \) for each \( n \). In the last step, the initial condition for numerical integration over \( (t^{n-1}, t^n] \) will come from projecting the discrete solution at time \( t = t^{n-1} \) onto the finite element space associated with the triangulation \( S_h(t^{n-1}) \), which generally differs from \( S_h(t^n) \).

A distinctive feature of our discretization is the manner in which the conforming triangulation \( S_h(t) \) of \( \Omega^t \) is constructed. As illustrated in Fig. 2, our method constructs \( S_h(t) \) by immersing \( \Omega^t \) in a background mesh \( T_h \) (the universal mesh), identifying a subtriangulation of \( T_h \) that approximates the immersed domain, and adjusting a few elements so that it conforms exactly to \( \Omega^t \). This approach differs markedly from classical deforming-mesh methods, where the task of triangulating \( \Omega^t \) is often handled by solving global systems of equations (such as those of linear elasticity) for nodal positions. We describe the construction of \( S_h(t) \) using a universal mesh in the following subsection and provide greater detail in Appendix A, which summarizes the presentation of [63].

The conditions under which a given background triangulation \( T_h \) can be so adjusted to conform to a family of domains \( \Omega^t \), \( t \in [0, T] \), are laid forth in [64, 65] and expanded in [66]. Briefly, the procedure is guaranteed to succeed if:

3.1. \( \Omega^t \) is \( C^2 \)-regular for every \( t \).

3.2. \( T_h \) is sufficiently refined in a neighborhood of \( \partial \Omega^t \) for every \( t \).
(3.iii) All triangles in $\mathcal{T}_h$ have angles bounded above by a constant $\theta < \pi/2$.

(3.iv) The intervals $(t^{n-1}, t^n]$ satisfy $\max_{1 \leq n \leq N} (t^n - t^{n-1}) \leq C h$ with a sufficiently small constant $C$.

### 3.1. Universal Mesh

Let $\mathcal{T}_h$ be a triangulation of $\mathcal{D}$ satisfying conditions (3.i-3.iv), with $h$ denoting the maximum diameter of an element $K \in \mathcal{T}_h$. For $i = 0, 1, 2, 3$, let $\mathcal{T}_{h,i}$ denote the collection of triangles $K \in \mathcal{T}_h$ for which exactly $i$ vertices of $K$ do not lie in the interior of $\Omega^t$.

Fix a partition $0 = t^0 < t^1 < \cdots < t^N = T$ of the temporal axis. Our approach for constructing a conforming mesh $S_h(t)$ for $\Omega^t$, $t \in (t^{n-1}, t^n]$, will consist of identifying a subtriangulation $S_h^n$ of the background triangulation $\mathcal{T}_h$ and defining a time-dependent bijection

$$\Phi^t : S_h^n \to \Omega^t, \quad t \in (t^{n-1}, t^n].$$

Here and in the sequel, we abuse notation by writing $S_h^n$ to denote both the triangulation (the list of vertices and their connectivities) as well as the region in $\mathbb{R}^2$ that it occupies. Our choice of $S_h^n$ is

$$S_h^n = \mathcal{T}_{h,0}^{n-1} \cup \mathcal{T}_{h,1}^{n-1} \cup \mathcal{T}_{h,2}^{n-1},$$

which is simply the set of triangles in the background triangulation with at least one vertex lying inside $\Omega^{t^{n-1}}$. Our choice of the map $\Phi^t : S_h^n \to \Omega^t$ is that detailed in [63]. We recapitulate the explicit formulas for $\Phi^t$ in Appendix A.

For each $t \in (t^{n-1}, t^n]$, the map $\Phi^t$ delivers a conforming mesh of $\Omega^t$ having the same connectivity as $S_h^n$ but consisting of triangles $\Phi^t(K)$, $K \in S_h^n$. We label this curvilinear mesh

---

Figure 2. Illustration of the manner in which a universal mesh provides a conforming triangulation of an immersed domain $\Omega^t$ for all times $t$. Over a short time interval $(t^{n-1}, t^n]$, an approximating subtriangulation $S_h^n$ is identified and adapted to the immersed domain using a map $\Phi^t : S_h^n \to \Omega^t$, $t \in (t^{n-1}, t^n]$. Over the next short time interval $(t^n, t^{n+1}]$, a new subtriangulation $S_h^{n+1}$ is identified and adapted to the immersed domain using a map $\Phi^{t+} : S_h^{n+1} \to \Omega^t$, $t \in (t^n, t^{n+1}]$. For visual clarity, the boundary of $\Omega^{t^{n-1}}$ has been juxtaposed in dashed lines onto the conforming mesh $\Phi^{t^n}(S_h^n)$ for $\Omega^{t^n}$. Likewise, the boundary of $\Omega^{t^n}$ has been juxtaposed in dashed lines onto the conforming mesh $\Phi^{t^{n+1}}(S_h^{n+1})$ for $\Omega^{t^{n+1}}$. 

---

\[ \Phi^t(S^n_h) \]

and set
\[ S_h(t) = \Phi^t(S^n_h), \quad t \in (t^{n-1}, t^n]. \]

The remainder of this section is devoted to a discretization of \((1-2)\) using finite element spaces over the evolving subtriangulation \(S_h(t)\). As is customary for readers familiar with ALE schemes, the resulting discretization (cf. \((17)\)) over each short time interval \((t^{n-1}, t^n]\) will resemble a discretization of
\[
\frac{Du}{Dt} + (u - v) \cdot \nabla_x u - v \Delta_x u = -\nabla_x p \quad \text{in } \mathcal{D} \setminus \mathcal{P} \quad \tag{8}
\]

\[
\nabla_x \cdot u = 0 \quad \text{in } \mathcal{D} \setminus \mathcal{P} \quad \tag{9}
\]

where
\[
\frac{Du}{Dt} = \frac{\partial u}{\partial t} + v \cdot \nabla_x u
\]

denotes the *material time derivative* of \(u\) along the path of a material particle that moves with the mesh \(S_h(t)\), whose velocity we denote by
\[
v(\Phi^t(X), t) = \dot{\Phi}^t(X) = \left. \frac{\partial}{\partial t} \right| \Phi^t(X). \quad \tag{10}
\]

Since the subtriangulation \(S_h(t)\) changes abruptly at each \(t^n, n = 1, 2, \ldots, N\), a projection will be used to transfer information between finite element spaces at such instants; cf. Section 3.3.

### 3.2. Galerkin Formulation over Short Time Intervals

We now describe a spatial discretization of \((1-2)\) that is obtained by performing a Galerkin projection of the weak equations \((6-7)\) onto finite element subspaces \(V^n_h \subset V^i, V^n_h \subset V^i\), and \(Q^n_h \subset Q^i\) over a short time interval \((t^{n-1}, t^n]\). We focus on the case in which the boundary conditions are given by \((3-4)\). The case in which the boundary conditions are given by \((3)\) and \((5)\) is handled similarly.

Here, we consider the use of Taylor-Hood \(P^k, P^{k-1}\) finite elements with an integer \(k \geq 2\) \([61]\).

Such elements approximate the velocity field \(u\) and the pressure field \(p\) with continuous functions that are elementwise polynomials of degree at most \(k\) and \(k - 1\), respectively, on \(S_h(t)\). These finite element spaces are easy to construct with the aid of the map \(\Phi^t : S^n_h \rightarrow \Omega^t\) introduced in Section 3.1.

Namely,
\[
V^t_h = \left\{ u_h \in C^0(\mathcal{T})^2 \mid u_h \circ \Phi^t|_K \in P^k(K)^2 \forall K \in S^n_h, u_h = 0 \text{ on } \partial \mathcal{P} \right\}
\]
\[
V^t_h = \left\{ u_h \in C^0(\mathcal{T})^2 \mid u_h \circ \Phi^t|_K \in P^k(K)^2 \forall K \in S^n_h, u_h = i_h^t v_P(\cdot, t) \text{ on } \partial \mathcal{P} \right\}
\]
\[
Q^t_h = \left\{ p_h \in C^0(\mathcal{T}) \mid p_h \circ \Phi^t|_K \in P^{k-1}(K) \forall K \in S^n_h \right\}.
\]

Here, \(i_h^t v_P(\cdot, t)\) denotes the nodal interpolant of \(v_P(\cdot, t)\) onto the space of continuous functions on \(\partial \mathcal{P}\) that are edgewise polynomials of degree \(k\), i.e. \(i_h^t v_P(\Phi^t(\cdot), t) \in P^k(e)\) for every edge \(e \subset \partial S^n_h\).

The Galerkin projection of \((6-7)\) over \((t^{n-1}, t^n]\) then reads: Find \(u_h(\cdot, t) \in V^t_h\) and \(p_h(\cdot, t) \in Q^t_h\) such that
\[
m^t(\bar{u}_h, w_h) + a^t(u_h, w_h) + c^t(w_h, p_h) = 0 \quad \forall w_h \in V^t_h \quad \tag{11}
\]
\[
c^t(u_h, q_h) = 0 \quad \forall q_h \in Q^t_h \quad \tag{12}
\]

for every \(t \in (t^{n-1}, t^n]\).

The system \((11-12)\) is equivalent to a system of differential-algebraic equations (DAEs). To deduce this, it is convenient to construct bases for \(V^t_h, V^t_h,\) and \(Q^t_h\) by composing a subset of shape functions on the background mesh \(T_h\) with the map \(\Phi^t|^{-1}\). Let
\[
\tilde{V}_h = \left\{ U_h \in C^0(\mathcal{T})^2 \mid U_h|_K \in P^k(K)^2 \forall K \in \mathcal{T}_h \right\}
\]
\[
\tilde{Q}_h = \left\{ P_h \in C^0(\mathcal{T}) \mid P_h|_K \in P^{k-1}(K) \forall K \in \mathcal{T}_h \right\}.
\]
In what follows, we will derive from (11-12) a system of DAEs of dimension $N_u + N_p$, where $N_u = \dim(\mathcal{V}_h)$ and $N_p = \dim(\mathcal{Q}_h)$.

Let $\{\tilde{N}_a\}_{a=1}^{N_u}$ and $\{\tilde{M}_k\}_{k=1}^{N_p}$ be the standard Lagrange bases for $\mathcal{V}_h$ and $\mathcal{Q}_h$, respectively, indexed by global degree of freedom numbers. Let $\{X_a\}_{a=1}^{N_u}$ and $\{Y_k\}_{k=1}^{N_p}$ denote the locations of the corresponding degrees of freedom in $\mathcal{T}_h$. Additionally, let

$$\tilde{I}_n^a = \{1 \leq a \leq N_u \mid \text{supp}(\tilde{N}_a) \cap \text{int}(S^0_h) \neq \emptyset, \tilde{N}_a = 0 \text{ on } \partial S^0_h \setminus \partial D\}$$
$$I_n^a = \{1 \leq a \leq N_u \mid \text{supp}(\tilde{N}_a) \cap \text{int}(S^0_h) \neq \emptyset\}$$
$$\tilde{I}_p^n = \{1 \leq k \leq N_p \mid \text{supp}(\tilde{M}_k) \cap \text{int}(S^0_h) \neq \emptyset\},$$

where $\text{supp}(f)$ denotes the support of a function $f$ and $\text{int}(S)$ denotes the interior of a set $S$. Bases for the spaces $\mathcal{V}_h$, $\mathcal{V}_h$, and $\mathcal{Q}_h$ are easily constructed with the aid of the functions $n_a^t : \Omega^t \to \mathbb{R}^2$ and $m_k^t : \Omega^t \to \mathbb{R}$ given by

$$n_a^t(\Phi^t(X)) = \tilde{N}_a(X), \quad a \in I_n^a \tag{13}$$
and

$$m_k^t(\Phi^t(X)) = \tilde{M}_k(X), \quad k \in I_p^n. \tag{14}$$

Namely,

$$\mathcal{V}_h = \text{span} \left\{ n_a^t \mid a \in I_n^a \right\}$$
$$\mathcal{V}_h = \mathcal{V}_h + \sum_{a \in \tilde{I}_n^a \setminus I_n^a} \nu_p(\Phi^t(X_a), t)n_a^t$$
$$\mathcal{Q}_h = \text{span} \left\{ m_k^t \mid k \in \tilde{I}_p^n \right\}.$$ 

If we adopt the convention that $n_a^t = 0$ in $\Omega^t$ for $a \notin \tilde{I}_n^a$ and $m_k^t = 0$ in $\Omega^t$ for $k \notin \tilde{I}_p^n$, we may expand

$$u_h(x, t) = \sum_{a=1}^{N_u} u_a(t)n_a^t(x) \tag{15}$$

and

$$p_h(x, t) = \sum_{k=1}^{N_p} p_k(t)m_k^t(x) \tag{16}$$

as linear combinations of the shape functions $n_a^t$ and $m_a^t$, bearing in mind that

$$u_h(\cdot, t) \in \mathcal{V}_h \implies u_a(t) = \nu_p(\Phi^t(X_a), t) \forall a \in I_n^a \setminus \tilde{I}_n^a.$$ 

In the expansions above, we adopt the convention that $u_a(t) = 0$ for $a \notin I_n^a$ and $p_k(t) = 0$ for $k \notin I_p^n$. Observe that by (13),

$$\dot{u}_h(x, t) = \sum_{a=1}^{N_u} \dot{u}_a(t)n_a^t(x) + \sum_{a=1}^{N_u} u_a(t)\frac{\partial n_a^t}{\partial t}(x)$$
$$= \sum_{a=1}^{N_u} \dot{u}_a(t)n_a^t(x) + \sum_{a=1}^{N_u} u_a(t)(-v(x, t) \cdot \nabla_x n_a^t(x))$$
$$= \sum_{a=1}^{N_u} \dot{u}_a(t)n_a^t(x) - v(x, t) \cdot \nabla_x u_h(x, t)$$

where $v$ is given by (10).
It follows that (11-12) is equivalent to the system of DAEs

$$\mathbf{M}(t) \begin{pmatrix} \dot{\mathbf{u}}(t) \\ 0 \end{pmatrix} + \mathbf{K}(t) \begin{pmatrix} \mathbf{u}(t) \\ \mathbf{p}(t) \end{pmatrix} + \begin{pmatrix} \mathbf{b}(\mathbf{u}(t), t) \\ 0 \end{pmatrix} = \begin{pmatrix} \mathbf{f}(t) \\ 0 \end{pmatrix}$$

(17)

where

$$\mathbf{M}(t) = \begin{pmatrix} \mathbf{M}_u(t) & 0 \\ 0 & 0 \end{pmatrix}$$

$$\mathbf{K}(t) = \begin{pmatrix} \mathbf{K}_u(t) & \mathbf{C}(t)^T \\ \mathbf{C}(t) & \mathbf{Z} \end{pmatrix}$$

and the entries of $\mathbf{M}_u(t), \mathbf{K}_u(t) \in \mathbb{R}^{N_u \times N_u}$, $\mathbf{C}(t), \mathbf{C}(t) \in \mathbb{R}^{N_p \times N_u}$, $\mathbf{Z} \in \mathbb{R}^{N_p \times N_p}$, and $\mathbf{b}(\mathbf{u}(t), t), \mathbf{f}(t) \in \mathbb{R}^{N_u}$ are given by

$$\mathbf{M}_{u, ab}(t) = \begin{cases} m^t(n^t_b, n^t_a) & \text{if } a \in \hat{\mathcal{I}}_u^n, b \in \mathcal{I}_u^n \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{K}_{u, ab}(t) = \begin{cases} a^t_v(n^t_b, n^t_a) & \text{if } a \in \hat{\mathcal{I}}_u^n, b \in \mathcal{I}_u^n \\ \delta_{ab} & \text{otherwise} \end{cases}$$

$$\mathbf{C}_{kk}(t) = \begin{cases} c^t(n^t_k, m^t_k) & \text{if } k \in \mathcal{I}_p^n, b \in \mathcal{I}_u^n \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{C}^*_{kk}(t) = \begin{cases} c^t(n^t_k, m^t_k) & \text{if } k \in \mathcal{I}_p^n, b \in \mathcal{I}_u^n \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{Z}_{kl} = \begin{cases} 0 & \text{if } k \in \mathcal{I}_p^n \\ \delta_{kl} & \text{otherwise} \end{cases}$$

$$\mathbf{b}_a(\mathbf{u}(t), t) = \begin{cases} b^t(u_h - v, u_h, n^t_k) & \text{if } a \in \hat{\mathcal{I}}_u^n \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{f}_a(t) = \begin{cases} \nu \varphi(\phi^t(X_a), t) & \text{if } a \in \mathcal{I}_u^n \setminus \hat{\mathcal{I}}_u^n \\ 0 & \text{otherwise} \end{cases}$$

Here, $\delta_{ab}$ denotes the Kronecker delta, and

$$a^t_v(u, w) = \nu \int_{\Omega} \left( \nabla_x u + (\nabla_x u)^T \right) : \nabla_x w \, dx$$

$$b^t(u_1, u_2, w) = \int_{\Omega} (u_1 \cdot \nabla_x u_2) \, w \, dx.$$

In hindsight, it is now evident that (17) is a discretization of the equations (8-9) that were alluded to earlier.

Remark. In the preceding paragraphs, we opted to construct a system of DAEs of dimension $N_u + N_p$ over each short time interval $(t^{n-1}, t^n]$, even though a portion of those DAEs correspond to degrees of freedom in the background mesh that do not belong to $\mathcal{S}_u^n$. This was accomplished by incorporating the set of trivial DAEs $u_a(t) = 0$ for $a \notin \mathcal{I}_u^n$ and $p_a(t) = 0$ for $k \notin \mathcal{I}_p^n$ into the system via the prescriptions $\mathbf{K}_{u, ab}(t) = \delta_{ab}$ for $a \notin \mathcal{I}_u^n$ and $\mathbf{Z}_{kl}(t) = \delta_{kl}$ for $k \notin \mathcal{I}_p^n$, respectively. The boundary conditions $u_a(t) = \nu \varphi(\Phi^t(X_a), t)$ for $a \in \mathcal{I}_u^n \setminus \hat{\mathcal{I}}_u^n$ were incorporated similarly via the prescriptions of $\mathbf{K}_u(t)$ and $\mathbf{f}(t)$.

We did this to highlight an important feature of the universal mesh: it permits the use of the same data structures (e.g., the matrices $\mathbf{M}(t)$ and $\mathbf{K}(t)$) over the complete duration of the simulation, not
merely over the intervals during which the mesh evolves continuously. The sparsity patterns of these data structures are invariant since the connectivity of the background mesh never changes.

Having said that, it is worth noting that one could, in principle, choose to replace the background mesh with a new one satisfying (3.i-3.iv) at any temporal node \( t^n \). Such a strategy may be useful if, for example, a local refinement or coarsening is desired at a particular stage of the simulation. Needless to say, the sizes and sparsity patterns of the data structures would generally change in this scenario. However, the theory presented in [66] suggests that the order of accuracy of the method is maintained as long as such replacements of the background mesh occur a number of times that remains bounded under refinement.

### 3.3. Initial Condition on each Short Time Interval

In order to complete the prescription of \((u(t), p(t))\) over a short time interval \([t^{n-1}, t^n]\), the system of DAEs (17) must be supplemented with an initial condition \(u(t_+^{n-1})\). Note that an initial condition for the pressure \(p\) is unnecessary.

Since the spaces \(V_h^{n-1}\) and \(V_h^{n+1}\) generally need not coincide, a projection is needed in order to transfer information between finite element spaces. To this end, we set

\[
u_h(\cdot, t_+^{n-1}) = \tilde{i}_h^{n+1} u_h(\cdot, t^{n-1}),
\]

where \(\tilde{i}_h^{n+1}\) is the nodal interpolant onto \(V_h^n\) [62, Chapter 1]. The corresponding vector \(u(t_+^{n-1})\) then consists of the coefficients \(u_i(t_+^{n-1})\) in the expansion (15).

We remark that more generally, one may consider the use of other surjective, linear projectors onto \(V_h^n\), such as the orthogonal projector onto \(V_h^n\) with respect to the \(L^2\)- or \(H^1\)-inner products. The theory presented in [66] supports the use of the \(L^2\)-projection, though the use of interpolation has always proven satisfactory in our numerical examples.

The influence of repeated projections such as (18) on the accuracy of the method is analyzed in [66, 67] and discussed in [63]. In brief, the projections introduce a half-order reduction in the method’s order of accuracy in the \(L^2\)-norm for linear parabolic problems when the \(L^2\)-projector is adopted. Conservation of total energy and momentum are of course also influenced, and it may be desirable in some situations to consider projectors designed with these considerations in mind; see, for example [68].

### 3.4. Temporal Discretization

The setup we have described thus far offers the freedom to employ a time integrator of one’s choosing to numerically integrate (17), a system of DAEs of index 2, from \(t = t^{n-1}\) to \(t = t^n\). Below we present two examples of integration schemes: a Singly Diagonally Implicit Runge-Kutta (SDIRK) scheme [69, 70], and a fractional step scheme [71–73]. In accordance with common guidelines for numerically solving DAEs, the SDIRK schemes we consider are stiffly accurate (and hence \(L\)-stable) methods [70, 74]. The same schemes are considered by, for instance, [75, 76], in their studies of high-order methods for the Navier-Stokes equations on fixed domains.

For the forthcoming discussion, we remind the reader that the temporal nodes \(t^n\) demarcate changes in the reference triangulation \(S_h^n\); hence, the time step \(\Delta t\) adopted during integration from \(t^{n-1}\) to \(t^n\) must be less than or equal to \(t^n - t^{n-1}\) for every \(n\). In practice, we often take \(\Delta t = t^n - t^{n-1}\), though this is by no means a necessity. Recall also that, in accordance with (3.iv), the time intervals \((t^n - t^{n-1})\) scale with the mesh spacing \(h\).

**Singly Diagonally Implicit Runge-Kutta.** Consider the use of a stiffly accurate \(s\)-stage Singly Diagonally Implicit Runge-Kutta (SDIRK) scheme of order \(\leq s\) with a time step \(\Delta t \leq (t^n - t^{n-1})\). At a given time \(\tau_0 \in [t^{n-1}, t^n]\), such an integrator advances the current numerical solution \((u_0, p_0) \approx (u(\tau_0^+), p(\tau_0^+))\) to time \(\tau = \tau_0 + \Delta t\) by solving a sequence of \(s\) systems of equations, as detailed below. The coefficients \(\gamma > 0\) and \(\beta_{ij} \in \mathbb{R}, i = 1, 2, \ldots, s, j = 0, 1, \ldots, i - 1\), for various SDIRK methods are tabulated in B, Tables I-IV.
Fractional Step Scheme. Our second example of a time integrator is a fractional step scheme with a time step \( \Delta t \leq (t^n - t^{n-1}) \). The scheme we propose is an adaptation of classical fractional step schemes [71–73] to the setting in which the fluid domain evolves with time.

At a given time \( \tau_0 \in [t^{n-1}, t^n] \), the fractional step scheme that we propose advances the current numerical solution \( u_0 \approx u(\tau_0^+) \) to time \( t = \tau_0 + \Delta t \) using a sequence of three steps. First, a preliminary approximation \( u_s \approx u(\tau_0 + \Delta t) \) that need not satisfy the incompressibility constraint is computed. Next, \( u_s \) is projected onto the space of divergence-free vector fields by solving a Neumann problem for an auxiliary variable \( \phi \), leading to a divergence-free quantity \( u_1 \approx u(\tau_0 + \Delta t) \) that serves as the time-\( \Delta t \) advancement of \( u_0 \). Finally, an approximation \( p_{1/2} \) to the pressure at \( t = \tau_0 + \Delta t/2 \) is computed.

To present the scheme in detail, we denote by \( M_p(t) \) and \( K_p(t) \) the \( N_p \times N_p \) matrices with entries

\[
M_{p,kl}(t) = \begin{cases} 
\int_{\Omega^t} m_k^l m_l^t \, dx & \text{if } k \in \mathcal{I}_p^n, l \in \mathcal{I}_p^n \\
0 & \text{otherwise}
\end{cases}
\]

\[
K_{p,kl}(t) = \begin{cases} 
\int_{\Omega^t} \nabla_x m_k^l \cdot \nabla_x m_l^t \, dx & \text{if } k \in \mathcal{I}_p^n, l \in \mathcal{I}_p^n \\
\delta_{kl} & \text{otherwise}
\end{cases}
\]

We denote \( \tau_{1/2} = \tau_0 + \Delta t/2 \), and we use \( p_{1/2} \) to denote a preliminary approximation to \( p(\tau_{1/2}) \), which will be specified shortly. The details of the algorithm follow.
Algorithm 3.2 Fractional step scheme for integration from $t = \tau_0 \in [t^{n-1}, t^n]$ to $t = \tau_0 + \Delta t \in [t^n, t^{n+1}]$.

Require: Initial condition $u_0 \approx u(\tau_0)$, and preliminary approximation $\bar{p}_{1/2} \approx p(\tau_{1/2})$.

1: Solve

\[
M_u(\tau_{1/2}) \left( \frac{u_* - u_0}{\Delta t} \right) + K_u(\tau_{1/2}) \left( \frac{u_0 + u_*}{2} \right) + \hat{C}(\tau_{1/2})T \bar{p}_{1/2} \\
+ b \left( \frac{u_0 + u_*}{2}, \tau_{1/2} \right) = f(\tau_0) + f(\tau_0 + \Delta t)
\]

for $u_*$.

2: With $\ell = \Delta t^{-1} C(\tau_{1/2}) u_*$, solve

\[
K_p(\tau_{1/2}) \phi = \ell
\]

for $\phi$.

3: Set

\[
\begin{align*}
  u_1 &= u_* - \Delta t M_u(\tau_{1/2})^{-1} \hat{C}(\tau_{1/2})T \phi \\
  p_{1/2} &= \bar{p}_{1/2} + \phi + \frac{\nu \Delta t}{2} M_p(\tau_{1/2})^{-1} \ell
\end{align*}
\]

4: Return $(u_1, p_{1/2}) \approx (u(\tau_0 + \Delta t), p(\tau_0 + \Delta t/2))$.

The precise choices that we made in the update formulas (the boundary conditions imposed on $u_*$, the boundary conditions imposed on $\phi$, and the update to the pressure) correspond to those made by the projection method “PmII” described in [73]. In particular, we prescribe the boundary values of $u_*$ with the known values of the velocity field at $\tau_0 + \Delta t$, we impose homogeneous Neumann boundary conditions on $\phi$, and we use a pressure update that is known to deliver second-order accuracy in time for both the velocity and pressure variables in the case of a fixed domain.

To understand the origin of the preceding scheme, it is instructive to consider its spatially continuous, temporally discrete counterpart on a fixed domain ($\Omega^t = \Omega^0 = D \setminus P^0 \forall t$). In this setting, Algorithm 3.2 reduces to the following scheme, where we denote by $u_0, u_*, u_1, \phi, p_{1/2}$, and $\bar{p}_{1/2}$ the spatially continuous counterparts of $u_0, u_*, u_1, \phi, p_{1/2}$, and $\bar{p}_{1/2}$, respectively:

1. Solve

\[
\begin{align*}
  \frac{u_* - u_0}{\Delta t} - \frac{\nu}{2} (\Delta x u_0 + \Delta x u_*) + \nabla_x \bar{p}_{1/2} + \frac{u_0 + u_*}{2} \cdot \nabla_x \frac{u_0 + u_*}{2} &= 0 \quad \text{in } \Omega^0 \\
  u_* &= 0 \quad \text{on } \partial D \\
  u_* &= v_p(\cdot, \tau_0 + \Delta t) \quad \text{on } \partial P^0
\end{align*}
\]

for $u_*$.

2. Solve

\[
\begin{align*}
  \Delta t \Delta x \phi &= \nabla_x \cdot u_* \quad \text{in } \Omega^0 \\
  \frac{\partial \phi}{\partial n} &= 0 \quad \text{on } \partial \Omega^0
\end{align*}
\]

for $\phi$.

3. Set

\[
\begin{align*}
  u_1 &= u_* - \Delta t \nabla_x \phi \\
  p_{1/2} &= \bar{p}_{1/2} + \phi - \frac{\nu \Delta t}{2} \Delta x \phi.
\end{align*}
\]
As mentioned earlier, the scheme above is precisely the second-order method “PmII” of [73]. We have numerical evidence (cf. Section 4.1) and heuristic reasoning to suggest that our extension of the method to moving domains is likewise second-order accurate in time, though a justification of this assertion warrants further analysis.

Note that in step 2 of the algorithm, the linear system to be solved for \( \phi \) is singular, as it corresponds to a Neumann problem whose solution is determined up to the addition of a constant. Defining \( \phi \) unambiguously requires, for example, imposing the value of one entry of the vector \( \phi \) arbitrarily. If a boundary condition of the form (4) were to be adopted on part of \( \partial D \), then \( \phi \) would need to satisfy homogeneous Dirichlet boundary conditions therein.

Finally, we describe our choice of \( \mathbf{p}_{1/2} \), which is set to be equal to the last computed value \( \mathbf{p}_{-1/2} \), interpolated onto the appropriate finite element space if necessary. More precisely, we set \( \mathbf{p}_{1/2} = \mathbf{p}_{-1/2} \) if \( \tau_0 \neq t^{n-1} \) and \( \mathbf{p}_{1/2} = 0 \) if \( \tau_0 = 0 \); otherwise, we set \( \mathbf{p}_{1/2} \) equal to the vector of coefficients in the expansion

\[
\sum_{k=1}^{N_p} \mathbf{p}_{1/2,k} m_{h,k}^{n-1} = t^1_h \left( \sum_{k=1}^{N_p} \mathbf{p}_{-1/2,k} m_{h,k}^{n-1} \right),
\]

where, abusing notation, \( i^p_h \) denotes the nodal interpolant onto \( \mathcal{Q}_h^p \). Note that choosing \( \mathbf{p}_{1/2} = 0 \) for \( \tau_0 = 0 \) reduces the accuracy of the very first time step to first order (cf. [73]), however, it is easy see that (by analogy with multi-step methods for ODEs [70]) this does not reduce the order of the scheme’s global truncation error.

3.5. Algorithm Summary

A summary of the proposed algorithm for integration over \([0, T]\) using a universal mesh \( \mathcal{T}_h \) and a temporal partition \( 0 = t^0 < t^1 < \cdots < t^N = T \) is as follows.

**Algorithm 3.3** Integration over \([0, T]\) using a universal mesh \( \mathcal{T}_h \) and a temporal partition \( 0 = t^0 < t^1 < \cdots < t^N = T \).

**Require:** Initial condition \( \mathbf{u}(0) \).

1. for \( n = 1, 2, \ldots, N \) do
2. Identify the subtriangulation \( \mathcal{S}_h^n \) of \( \mathcal{T}_h \) consisting of triangles with at least one vertex lying inside \( \Omega^{n-1} \).
3. Adapt \( \mathcal{S}_h^n \) to \( \Omega^{n-1} \) by computing \( \mathcal{S}_h(t^{n-1}) := \Phi^{n-1} \circ (\mathcal{S}_h^n) \), where \( \Phi^t : \mathcal{S}_h^n \to \Omega^t \) is the universal mesh map (29).
4. Project \( \mathbf{u}(t^{n-1}) \) onto the finite element space \( \mathcal{V}_h^{n-1} \) associated with \( \mathcal{S}_h(t^{n-1}) \), through (18), giving \( \mathbf{u}(t^{n-1}) \).
5. Numerically integrate (17) from \( t = t^{n-1} \) to \( t = t^n \) using a time integrator of one’s choosing with time step \( \Delta t \leq (t^n - t^{n-1}) \), giving \( (\mathbf{u}(t^n), \mathbf{p}(t^n)) \).
6. end for
7. return \( (\mathbf{u}(T), \mathbf{p}(T)) \)

**Remarks.**

1. In the last step of the algorithm, the numerical integration may require the evaluation of \( \mathcal{S}_h(t) = \Phi^t \circ (\mathcal{S}_h^n) \) at intermediate times \( t \in (t^{n-1}, t^n) \) in order to assemble the quantities \( \mathbf{M}(t), \mathbf{K}(t), \mathbf{b}(\mathbf{u}(t), t), \) and \( \mathbf{f}(t) \) at intermediate stages of integration.

2. When the fractional step scheme (3.2) is adopted for numerical integration, the output of step (4) in such scheme is \( (\mathbf{u}(t^n), \mathbf{p}(t^n - \Delta t/2)) \).
4. NUMERICAL EXAMPLES

In this section, we apply the proposed methods to simulate flow around various obstacles with prescribed evolution. We consider three examples of obstacles: a rotating ellipse, an oscillating disk, and a rotating stirrer. We consider the rotating ellipse in order to study numerically the convergence of the methods. The remaining examples serve to illustrate the features of the methodology.

4.1. Rotating Ellipse

To study numerically the convergence of the methods, we considered the case in which obstacle \( P_t \) is an ellipse with semi-major axis \( a = 1.0 \) and semi-minor axis \( b = 0.8 \), rotating at a fixed angular velocity \( \omega = 2.5 \), as depicted in Fig. 3. Using [77] for inspiration, we manufactured a solution by adding a forcing term to the right-hand side of (1) so that the exact solution is given by

\[
\begin{align*}
    u_1(x_1, x_2, t) &= -\frac{a^2 + b^2}{\sqrt{a^4 - b^4}} \omega e^{-\xi} (b \cos \omega t \sin \eta + a \sin \omega t \cos \eta) \\
    u_2(x_1, x_2, t) &= -\frac{a^2 + b^2}{\sqrt{a^4 - b^4}} \omega e^{-\xi} (b \sin \omega t \sin \eta - a \cos \omega t \cos \eta) \\
    p(x_1, x_2, t) &= \sin(x_1) \sin(x_2),
\end{align*}
\]

with \( \xi \geq 0 \) and \( \eta \in [0, 2\pi) \) related to the cartesian coordinates \( x_1 \) and \( x_2 \) via

\[
\begin{align*}
    x_1 \cos \omega t + x_2 \sin \omega t &= \frac{\sqrt{a^4 - b^4}}{a} \cosh \xi \cos \eta \\
    -x_1 \sin \omega t + x_2 \cos \omega t &= \frac{\sqrt{a^4 - b^4}}{b} \sinh \xi \sin \eta.
\end{align*}
\]

The velocity field so manufactured has the property that it is everywhere divergence-free and satisfies the no-slip condition (3) on \( \partial P_t \). On the remainder of the fluid boundary, we prescribed the known values of the velocity field. We took \( \nu = 1.0 \) so that the Reynolds number of the flow was \( Re = u_2(a, 0, 0) a / \nu = 2.5 \).

We studied the \( L^2 \)-error in \( u \) and \( p \) at time \( T = 0.05 \) on a sequence of uniform refinements of an equilateral triangle mesh with a lowest resolution mesh spacing of \( h_0 = 0.25 \), using a time step \( \Delta t = Th/h_0 \) and a temporal subdivision \( t^n = n\Delta t, n = 0, 1, 2, \ldots, T/\Delta t \). We considered three combinations of finite elements and time integrators: Taylor-Hood \( P^2-P^1 \) elements together with the fractional step scheme (3.2),† Taylor-Hood \( P^2-P^1 \) elements together with an SDIRK scheme of

\[ \text{ DOI: 10.1002/nme } \]
Figure 4. Convergence rates in the $L^2(\Omega^T)$-norm for the solution to incompressible, viscous flow around a rotating ellipse using three combinations of finite elements and time integrators with $\Delta t \propto h$: (1) Taylor-Hood $P_2$-$P_1$ elements together with the fractional step scheme (3.2), (2) Taylor-Hood $P_2$-$P_1$ elements together with a third-order implicit Runge-Kutta scheme, and (3) Taylor-Hood $P_3$-$P_2$ elements together with a fourth-order implicit Runge-Kutta scheme. Also shown in the tables are expected orders of convergence inferred from the theory presented in [63, 66].

<table>
<thead>
<tr>
<th>Velocity $h_0/h$</th>
<th>$P^2$-$P_1$ / Fractional step</th>
<th>$P^2$-$P_1$ / SDIRK(3)</th>
<th>$P^3$-$P_2$ / SDIRK(4)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Error</td>
<td>Order</td>
<td>Error</td>
</tr>
<tr>
<td>1</td>
<td>1.91e-02</td>
<td>-</td>
<td>3.75e-03</td>
</tr>
<tr>
<td>2</td>
<td>4.63e-03</td>
<td>2.04</td>
<td>5.75e-04</td>
</tr>
<tr>
<td>4</td>
<td>1.26e-03</td>
<td>1.88</td>
<td>8.98e-05</td>
</tr>
<tr>
<td>8</td>
<td>3.24e-04</td>
<td>1.96</td>
<td>1.23e-05</td>
</tr>
<tr>
<td>Expected Order</td>
<td>1.5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Pressure $h_0/h$</th>
<th>$P^2$-$P_1$ / Fractional step</th>
<th>$P^2$-$P_1$ / SDIRK(3)</th>
<th>$P^3$-$P_2$ / SDIRK(4)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Error</td>
<td>Order</td>
<td>Error</td>
</tr>
<tr>
<td>1</td>
<td>3.01e-01</td>
<td>-</td>
<td>1.52e-02</td>
</tr>
<tr>
<td>2</td>
<td>5.61e-02</td>
<td>2.42</td>
<td>2.81e-03</td>
</tr>
<tr>
<td>4</td>
<td>1.72e-02</td>
<td>1.71</td>
<td>6.63e-04</td>
</tr>
<tr>
<td>8</td>
<td>4.89e-03</td>
<td>1.81</td>
<td>1.83e-04</td>
</tr>
<tr>
<td>Expected Order</td>
<td>1.5</td>
<td></td>
<td>1.5</td>
</tr>
</tbody>
</table>

order 3 (cf. Table III), and Taylor-Hood $P^3$-$P_2$ elements together with an SDIRK scheme of order 4 (cf. Table IV). The resulting spatial discretizations for $h_0/h = 1, 2, 4$, and 8, respectively, had 1,851, 7,155, 28,131, and 111,555 degrees of freedom (for $P^2$-$P_1$ elements) and 4,419, 17,283, 68,355, and 271,875 degrees of freedom (for $P^3$-$P_2$ elements). For each of the combinations of finite elements and time integrators considered, we observed convergence rates that are at worst suboptimal by half
Figure 5. Universal mesh for a disk with unit diameter oscillating with amplitude $A = 0.1$ and frequency $\omega$. 

Further, these results are consistent with the predictions of [63, 66, 67], which derive a priori error estimates that are suboptimal by half an order in the $L^2$-norm for schemes that adopt a universal mesh in conjunction with piecewise polynomial finite element spaces to solve a parabolic model problem.

4.2. Oscillating Disk

As a second example, we considered the case in which the obstacle $P^t$ is a disk of radius $R = 1/2$ whose center oscillates vertically with amplitude $A$ and frequency $\omega$ about a point $(x_0, y_0) = (-3, 0)$:

$$P^t = \{(x, y) \mid (x - x_0)^2 + (y - y_0 + A \cos(\omega t))^2 < R^2\}.$$ 

We immersed the oscillating disk in a domain $D = [-6, 6] \times [-3, 3]$ and prescribed boundary conditions

$$u = (u_\infty(1 - e^{-t/2}), 0) \quad \text{on} \quad [-6, 6] \times \{-3\} \cup \{-6, 6\} \times \{3\} \cup \{-6\} \times [-3, 3]$$

$$pm - \nu (\nabla_x u + (\nabla_x u)^T) \quad \text{on} \quad \{6\} \times [-3, 3].$$

$$u = (0, \omega A \sin(\omega t)) \quad \text{on} \quad \partial P^t,$$

where $u_\infty = 1/2$.

Fig. 5 shows the universal mesh that we adopted for this simulation, as well as snapshots of the resulting conforming mesh for $\Omega^t = D \setminus P^t$ at a few representative instants in time when $A = 0.1$. The background mesh was constructed by inserting stencils of acute triangles into an adaptively refined quadtree; see [78] for details.

We first considered the cases in which $A = 0.1$ and $\omega = 0.8\omega_0$, $\omega_0$, and $1.2\omega_0$, where $\omega_0$ is the natural vortex shedding frequency for flow past a fixed disk of radius $R$, assuming a Strouhal number $St = (\omega_0/2\pi)(2R)/u_\infty = 0.195$. We took $\nu = 1/370$ so that $Re = (2R)u_\infty/\nu = 185$. We solved the problem using Taylor-Hood $P^2$-$P^1$ elements (leading to 18,701 degrees of freedom) together with the fractional step scheme (3.2), using $\Delta t = 0.2$ and $t^n = n\Delta t$. Figs. 6-7 show contours of the
vorticity $\nabla \times u$ and the pressure $p$ in each of the three cases at the largest time $t < 80$ for which the disk’s vertical displacement is $-A$. We observed a characteristic shift in the vortex shedding pattern’s phase relative to the disk’s oscillation as $\omega$ passed through $\omega_0$, which is consistent with past numerical and experimental studies of flow past an oscillating disk [79, 80].

Next, we fixed $\omega = 0.8\omega_0$ and $A = 0.2$ and studied the temporal evolution of the drag coefficient $C_D$ and the lift coefficient $C_L$ for the cases in which $\nu = 1/2$ (so that $Re = 1$) and $\nu = 1/370$.

Figure 6. Vorticity contours during flow past a disk with unit diameter oscillating with amplitude $A = 0.1$ and frequency (a) $\omega = 0.8\omega_0$, (b) $\omega = \omega_0$, and (c) $\omega = 1.2\omega_0$. The snapshot shown in each case corresponds to the largest time $t < 80$ for which the disk’s vertical displacement is $-A$. A characteristic shift in the vortex shedding pattern’s phase relative to the disk’s oscillation occurs as $\omega$ passes through $\omega_0$. 
Figure 7. Pressure contours during flow past a disk with unit diameter oscillating with amplitude $A = 0.1$ and frequency (a) $\omega = 0.8\omega_0$, (b) $\omega = \omega_0$, and (c) $\omega = 1.2\omega_0$. These snapshots correspond to the same instants in time as in Fig. 6.
Figure 8. Drag and lift coefficients during flow past an oscillating disk at $Re = 1$. The results of two simulations are plotted, one corresponding to the mesh in Fig. 5 and one corresponding to a refinement thereof.

Figure 9. Drag and lift coefficients during flow past an oscillating disk at $Re = 185$. The results of two simulations are plotted, one corresponding to the mesh in Fig. 5 and one corresponding to a refinement thereof.

For $Re = 185$ (Fig. 9), the simulation on the coarsest mesh exhibits spurious oscillations of the drag and lift coefficients, but these are significantly reduced upon refinement. We suspect that the
oscillations are attributable to the interpolation of the solution onto a new finite element space at each time $t^n$ (cf. Section 3.3), since identical numerical experiments with a fixed disk rendered drag and lift coefficient time series that were free of artificial oscillations. Diffusion seems to also play a role in mitigating the artificial oscillations, as evidenced by their absence in Fig. 8, where $Re = 1$. Based upon these observations, it may be worthwhile to explore the possibility of designing more sophisticated strategies for transferring information between finite element spaces, such as projecting the velocity onto the space of divergence-free vector fields after interpolating, in order to obtain more satisfactory results on coarse meshes at high Reynolds numbers.

Fig. 10 shows the convergence of the computed drag and lift coefficients under the aforementioned mesh refinement. To measure the errors in the time series, we computed a rectangle-rule approximation \( E \) to the integrated error \( \int_0^1 (C_i(t) - \bar{C}_i(t))^2 \, dt \) between the computed solution \( C_i(t) \) and a reference solution \( \bar{C}_i(t) \) obtained from a fine mesh with \( h = 0.145 \), and likewise for the lift coefficient. The errors in all cases converged to zero at rates approximately of the order $h^{1.5}$.

### 4.3. Stirring a Viscous Fluid

Our last example considers the case in which the obstacle boundary is a closed cubic spline in the shape of a propeller-like stirrer that rotates at a prescribed angular velocity

\[
\omega(t) = \omega_0 (1 - e^{-t/\tau})
\]

with $\omega_0 = 5.0$ and $\tau = 0.01$. The stirrer blades were of length $\approx 1.4$ and average width $\approx 0.3$. We took $\nu = 0.2$ in our simulations, so that the Reynolds number of the flow (treating the stirrer blade width as the characteristic length scale) was approximately 10.5. To approximate the velocity and pressure, we adopted Taylor-Hood $P^2$-$P^1$ elements. To integrate in time, we used an SDIRK scheme.
of order 3. We immersed the stirrer in hexagonal domain $D$ of diameter 4 and imposed Neumann boundary conditions (4) on $\partial D$. We adopted a uniform background mesh of equilateral triangles ($h = 0.0625, 18,701$ degrees of freedom).

Figs. 11 and 12 display snapshots of the mesh and velocity magnitude contours, respectively, at various times during the simulation. The robust nature of the methods introduced here is patent in this example, as traditional deforming-mesh methods could easily encounter difficulties with mesh entanglement upon rotation of the stirrer.

5. CONCLUDING REMARKS

We have presented a framework for computing incompressible, viscous flow around a moving obstacle with prescribed evolution using a universal mesh. By immersing the obstacle in a background mesh and adjusting a few elements in the neighborhood of obstacle’s boundary, the strategy provides a conforming triangulation of the fluid domain at all times over which a spatial discretization of the fluid velocity and pressure fields of any desired order may be constructed using standard finite elements. The resulting semidiscrete equations may be integrated in time using standard time integrators for ODEs. We illustrated the framework using Taylor-Hood finite elements together with Runge-Kutta time integrators and a fractional step scheme. Numerical convergence tests confirmed the theory presented in [63, 66], which predicts orders of convergence that are
suboptimal by half an order in the $L^2$-norm for a model parabolic problem. We demonstrated the method’s versatility on numerical examples that involve flow past an oscillating disk and flow around a rotating stirrer.

All examples in the manuscript involved flows with low-to-moderate Reynolds number. This enabled us to obtain accurate solutions with relatively coarse and isotropic meshes. For larger Reynolds numbers, we expect to have to modify the spatial discretization by including a stabilization of the advection term. More importantly, meshes will have to be anisotropic and refined around the boundary of the moving obstacle, to capture the boundary layer. Doing it with a universal mesh is an open problem.

ACKNOWLEDGEMENTS

This research was supported by the U.S. Department of Energy grant DE-FG02-97ER25308; Department of the Army Research Grant, grant number: W911NF-07-2-0027; NSF Career Award, grant number: CMMI-0747089; and NSF, grant number CMMI-1301396.

REFERENCES

2. Wang L, Persson PO. A discontinuous Galerkin method for the Navier-Stokes equations on deforming domains using unstructured moving space-time meshes 2013;
6. Montefusco F, Sousa F, Buscaglia G. High-order ALE schemes with applications in capillary flows. (Submitted) 2014;
A. THE UNIVERSAL MESH MAP

In this section, we detail the construction of the map $\Phi^t : S_h^n \to \Omega^t$ introduced in Section 3.1. We require the definition of three auxiliary maps: a boundary evolution map, a relaxation map, and a blend map. In what follows, we denote by $\phi : \mathcal{D} \to \mathbb{R}$ the signed distance function to $\partial \Omega^t$, taken to be positive outside $\Omega^t$ and negative inside $\Omega^t$. We denote by $\pi^t : \mathcal{D} \to \partial \Omega^t$ the closest point projection onto $\partial \Omega^t$.

**Auxiliary Maps.** The boundary evolution map $\gamma^t : \partial S_h^n \to \partial \Omega^t$ provides a correspondence between the piecewise linear boundary of $S_h^n$ and the boundary of $\Omega^t$ for $t \in (t^{n-1}, t^n]$. It is defined in terms of the closest point projection via

$$
\gamma^t = \pi^t \circ \pi^{n-1} \big|_{\partial S_h^n}.
$$

The relaxation map $p : S_h^n \to S_h^n$ identifies those vertices that lie both inside $\Omega^t$ and near $\partial \Omega^t$, and perturbs them in a direction away from $\partial \Omega^t$. It is defined in terms of the signed distance function via

$$
p(x) = \begin{cases} 
    x - \delta h \left( 1 + \frac{\phi^{n-1}(x)}{R h} \right) \nabla_x \phi^{n-1}(x) & \text{if } -R h < \phi^{n-1} < 0 \\
    x & \text{otherwise},
\end{cases}
$$

with $R > 1$ a small positive integer and $(1 + 1/R)^{-1} \leq \delta < 1$. We denote by $p(S_h^n)$ the triangulation obtained by applying the relaxation $p$ to the vertices of $S_h^n$ while preserving the mesh’s connectivity.

The blend map $\psi^t$ takes each straight triangle $K \in p(T_{h,2}^{n-1})$ and deforms it to a curved triangle that conforms exactly to the moving boundary. Letting $u, v, w$ denote the vertices of $K$, the blend map reads

$$
\psi^t(x) = \frac{1}{2(1 - \lambda_u)} \left[ \lambda_v \gamma^t((1 - \lambda_u)u + \lambda_u v) + \lambda_u \lambda_w \gamma^t(u) \right] 
\quad + \frac{1}{2(1 - \lambda_v)} \left[ \lambda_u \gamma^t((1 - \lambda_v)u + \lambda_v w) + \lambda_v \lambda_w \gamma^t(v) \right] + \lambda_u \lambda_v \lambda_w w, \quad (28)
$$

where $\lambda_u, \lambda_v, \lambda_w$ are the barycentric coordinates of $x \in K$. Here, we have employed the convention that the vertex $w$ is the unique vertex of $K$ lying inside $\Omega^{n-1}$.

**The Universal Mesh Map.** We now define $\Phi^t$ over each triangle $K \in S_h^n$ with vertices $u, v, w$ according to

$$
\Phi^t(x) = \begin{cases} 
    \lambda_u p(u) + \lambda_v p(v) + \lambda_w p(w) & \text{if } K \in T_{h,0}^{n-1} \\
    \lambda_u \gamma^t(u) + \lambda_v p(v) + \lambda_w p(w) & \text{if } K \in T_{h,1}^{n-1} \\
    \psi^t(\lambda_u u + \lambda_v v + \lambda_w w) & \text{if } K \in T_{h,2}^{n-1}
\end{cases}, \quad (29)
$$

where $\lambda_u, \lambda_v, \lambda_w$ are the barycentric coordinates of $x \in K$. Once again, we have employed the convention that for triangles $K \in T_{h,2}^{n-1}$, the vertex $w$ is the unique vertex of $K$ lying inside $\Omega^{n-1}$, and for triangles $K \in T_{h,1}^{n-1}$, the vertex $u$ is the unique vertex of $K$ lying outside $\Omega^{n-1}$.
Table I. SDIRK(1): Coefficients $\beta_{ij}$ for a $s = 1$-stage SDIRK scheme of order 1. ($\gamma = 1$)

<table>
<thead>
<tr>
<th>$i \setminus j$</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

Table II. SDIRK(2): Coefficients $\beta_{ij}$ for a $s = 2$-stage SDIRK scheme of order 2. ($\gamma = 1 - \sqrt{2}/2$)

<table>
<thead>
<tr>
<th>$i \setminus j$</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>$\sqrt{2}$</td>
<td>$1 + \sqrt{2}$</td>
</tr>
</tbody>
</table>

Table III. SDIRK(3): Coefficients $\beta_{ij}$ for a $s = 3$-stage SDIRK scheme of order 3. ($\gamma = 0.3586652150845899942$)

<table>
<thead>
<tr>
<th>$i \setminus j$</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0.352859819860479140</td>
<td>0.647140180139520860</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>$-1.25097989505606042$</td>
<td>3.72932966244456977</td>
<td>$-1.47834976738850935$</td>
</tr>
</tbody>
</table>

Table IV. SDIRK(4): Coefficients $\beta_{ij}$ for a $s = 5$-stage SDIRK scheme of order 4. ($\gamma = 1/4$)

<table>
<thead>
<tr>
<th>$i \setminus j$</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>$-1$</td>
<td>$2$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>$-\frac{13}{25}$</td>
<td>$\frac{42}{25}$</td>
<td>$-\frac{4}{25}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>$-\frac{4}{17}$</td>
<td>$\frac{89}{25}$</td>
<td>$-\frac{25}{156}$</td>
<td>$\frac{15}{136}$</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>$\frac{7}{3}$</td>
<td>$-\frac{37}{12}$</td>
<td>$-\frac{103}{24}$</td>
<td>$\frac{275}{8}$</td>
<td>$-\frac{85}{3}$</td>
</tr>
</tbody>
</table>

**Isoparametric Approximations.** In practical computations, it is convenient to approximate the map $\Phi^t$ (and hence the domain $\Omega^t$) with a polynomial interpolant

$$\Phi_{\text{approx}}^t(x) = \sum_a \tilde{N}_a(x) \Phi^t(\tilde{X}_a)$$

constructed from shape functions $\tilde{N}_a$ of a triangular Lagrange element with corresponding degrees of freedom $\tilde{X}_a$ on the reference triangulation $S^t_n$. Details are given in [63].

Assembly of the quantities $M(t)$, $K(t)$, $b(u(t), t)$, and $f(t)$ at a given time $t \in (t^{n-1}, t^n]$ can then be accomplished by computing the positions $\Phi^t(\tilde{X}_a)$ of the degrees of freedom in the (approximately) conforming mesh $\Phi_{\text{approx}}^t(S^t_n)$ and following standard practices to compute elementwise contributions over curved isoparametric elements [62]. We adopted this strategy in all of the numerical examples presented in this paper.

**B. SINGLY DIAGONALLY IMPLICIT RUNGE KUTTA TIME INTEGRATORS.**

Tables I-IV record the coefficients $\gamma > 0$ and $\beta_{ij} \in \mathbb{R}$, $i = 1, 2, \ldots, s$, $j = 0, 1, \ldots, i - 1$ for a collection of SDIRK schemes of orders 1 through 4. Note that the structure of the Runge-Kutta stages in Algorithm (3.1) differs from the structure that is most familiar to Runge-Kutta practitioners [70]; see [63, Appendix A] and [81] for details.