



Mesh moving techniques in fluid-structure interaction: robustness, accumulated distortion and computational efficiency

Alexander Shamanskiy¹ · Bernd Simeon¹

Received: 19 June 2020 / Accepted: 16 November 2020 / Published online: 12 December 2020
© The Author(s) 2020

Abstract

An important ingredient of any moving-mesh method for fluid-structure interaction (FSI) problems is the mesh moving technique (MMT) used to adapt the computational mesh in the moving fluid domain. An ideal MMT is computationally inexpensive, can handle large mesh motions without inverting mesh elements and can sustain an FSI simulation for extensive periods of time without irreversibly distorting the mesh. Here we compare several commonly used MMTs which are based on the solution of elliptic partial differential equations, including harmonic extension, bi-harmonic extension and techniques based on the equations of linear elasticity. Moreover, we propose a novel MMT which utilizes ideas from continuation methods to efficiently solve the equations of nonlinear elasticity and proves to be robust even when the mesh undergoes extreme motions. In addition to that, we study how each MMT behaves when combined with the mesh-Jacobian-based stiffening. Finally, we evaluate the performance of different MMTs on a popular two-dimensional FSI benchmark reproduced by using an isogeometric partitioned solver with strong coupling.

Keywords Isogeometric analysis · Arbitrary Lagrangian–Eulerian methods · Mesh-Jacobian-based stiffening · Nonlinear elasticity · Continuation methods

1 Introduction

Fluid-structure interaction (FSI) constitutes a class of problems involving two-way dependence between structural objects and a fluid. As such, FSI is a vast topic with applications spanning a spectrum from aerospace and civil engineering [1] to biomechanical and cardiovascular simulations [9]. In FSI problems, the fluid exerts a force on the structure which deforms in response. As the structure moves, it changes the shape of the domain occupied by the fluid together with the fluid motion and, as a result, the force that the fluid exerts on the structure. This two-way coupling between the fluid and structure behavior as well as the necessity to accommodate the fluid domain motion in both the continuous and discrete formulations of the problem is what makes FSI so notoriously complex.

Since FSI problems rarely admit analytical solutions, computational methods are widely adopted in FSI research. Here, one can distinguish between the static-mesh and moving-mesh methods. While the former attempt to resolve the motion of the fluid domain implicitly, for example by means of a stationary background Cartesian mesh [21], the latter deal with the motion of the fluid domain by tracking its boundary and adapting the computational mesh correspondingly. In this work, we study and compare various mesh moving techniques (MMTs) which can be used to adapt the fluid mesh if the moving-mesh methods are used. The main focus here lies on the robustness of a given MMT, that is, how much mesh distortion it introduces and how much mesh motion it can handle without entangling or inverting mesh elements. Additionally, we pay attention to computational complexity of MMTs, which can significantly increase the overall FSI simulation time if left unchecked.

To study performance of MMTs in their natural habitat, we employ a popular two-dimensional FSI benchmark from [33]. In the benchmark, an unstable flow of a viscous fluid leads to development of the vortex shedding phenomenon which results in oscillations of a flexible beam structure. The oscillations grow in magnitude until they reach a stable

✉ Alexander Shamanskiy
shamansk@mathematik.uni-kl.de

¹ TU Kaiserslautern, Department of Mathematics,
Felix-Klein-Zentrum, Paul-Ehrlich-Str. 31, 67663
Kaiserslautern, Germany

periodic regime which lends itself well to studying possible long-term effects of MMTs on the fluid mesh. One of such long-term effects is accumulated mesh distortion where mesh elements increasingly become permanently distorted, deteriorating the simulation accuracy. In addition to the original benchmark, we employ its simplified version with no fluid mechanics involved to perform a large number of computationally inexpensive tests. In this way, we can concentrate on mesh motion and conduct a detailed analysis of the behavior of MMTs.

To reproduce the FSI benchmark, we use a partitioned solver with strong coupling and Aitken relaxation [17]. Although modern space-time (ST) methods are becoming increasingly common in FSI [1,25,32], we resort to classical arbitrary Lagrangian–Eulerian (ALE) methods [1,12,20] which are more straightforward in implementation. Our choice in favor of a basic partitioned ALE solver is justified since both ST and ALE methods make use of the same MMTs, so we can focus on mesh motion in this paper.

All MMTs we consider here are based on solution of elliptic partial differential equations. These include existing techniques such as harmonic extension [8,35], bi-harmonic extension [19] and a widely adopted technique based on linear elasticity theory [14,31]. Moreover, we propose an efficient MMT based on the equations of nonlinear elasticity and a logarithmic neo-Hookean material law which we refer to as tangential incremental nonlinear elasticity (TINE). Although techniques based on the equations of nonlinear elasticity have been proposed before [24,26,27], TINE is novel in that it uses the idea of a tangential continuation method [7,22] to efficiently solve the corresponding nonlinear equations. As a result, TINE is only slightly more computationally expensive than the linear-elasticity-based techniques which are linear in nature. On the other hand, it can handle as much mesh motion but does not suffer from the accumulated distortion effect.

Robustness of any MMT can be increased by additional augmentations. Probably the most popular one is the mesh-

Jacobian-based stiffening (MJBS) [14,23,30,31] which turns individual mesh elements stiffer or softer depending on their size and shape. In this work, we study how different MMTs react to the MJBS. Although not considered here, further possible MMT augmentations include solid layer extension [29] and element relaxation [26,27].

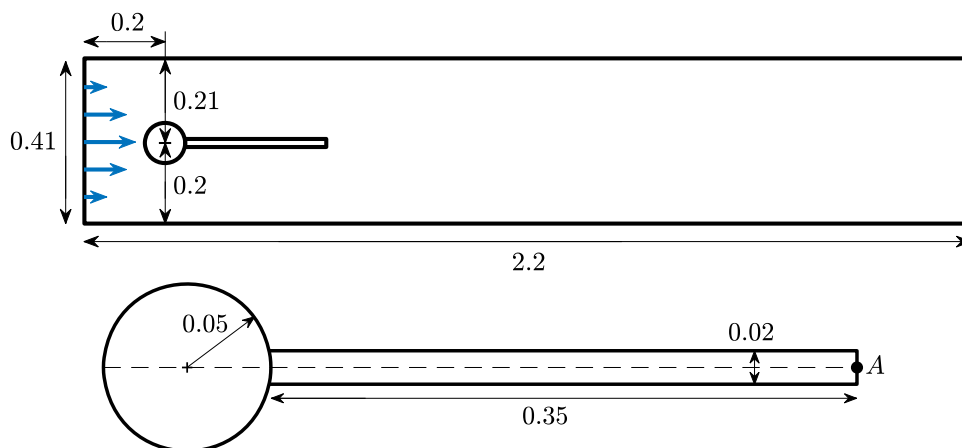
The research we present in this work has been conducted in the framework of isogeometric analysis (IGA) [6,11]. Despite that, the results are applicable to classical finite elements methods or any other mesh-based method for solving partial differential equations.

The rest of this paper is structured as follows. Section 2 outlines geometry and settings of the FSI benchmark and fixes the necessary notation. In Sect. 3, we describe various MMTs considered in this work as well as the MJBS. In Sect. 4, we consider the simplified benchmark and conduct a detailed analysis of the short-term and long-term behavior of different MMTs in artificial FSI-like conditions. After that, we proceed to performing a full FSI simulation of the benchmark in Sect. 5. We study the performance of the MMTs and check if the choice of a particular technique affects the simulation results. Finally, we discuss the results of the MMT analysis and FSI simulations, draw conclusion and outline further research directions in Sect. 6.

2 Benchmark description

In this section, we describe the geometry and mathematical model of the FSI benchmark from [33]. It studies the flow of an incompressible Newtonian fluid through a 2D channel as the fluid interacts with a submerged structure. The channel is a rectangle $[0, 2.2] \times [0, 0.41]$. The structure consists of a rigid disk $B_{0.05}(0.2, 0.2)$ and a flexible beam $[0.2, 0.6] \times [0.19, 0.21] \setminus B_{0.05}(0.2, 0.2)$ which is attached at its left end to the boundary of the disk. Figure 1 illustrates the setting; note that the geometry is intentionally non-symmetric.

Fig. 1 Top: the flow channel and the submerged structure in the initial configuration. Bottom: close-up on the structure



The top and bottom walls of the channel are impermeable to the fluid. The fluid enters the channel through the left wall with a prescribed velocity and exits through the right wall freely. The presence of the submerged structure changes the flow of the fluid and, in response, the fluid exerts a certain force on the structure. Thus, this is an FSI problem. Depending on the prescribed material parameters of the fluid and beam as well as on the inflow fluid velocity, this fluid force can result in a noticeable deformation of the beam. The beam deformation alters the shape of the flow channel, the flow itself and, as a consequence, the force exerted on the structure by the fluid. Such systems with a two-way interaction between components are called coupled systems.

In what follows, when we use the word “structure”, we refer to the flexible beam only. If the rigid disk is also considered, we explicitly mention it. Throughout this work, we use subscripts s and f to distinguish between objects related to the structure and fluid respectively. Moreover, we use subscript a when dealing with ALE mappings and related objects. Let $\Omega_f(t) \subset \mathbb{R}^2$ and $\Omega_s(t) \subset \mathbb{R}^2$ denote domains occupied by the fluid and structure respectively at time $t \in [0, T]$. Additionally, we use $\Omega_f^0 = \Omega_f(0)$ and $\Omega_s^0 = \Omega_s(0)$ to denote the fluid and structure domains at time $t = 0$. Furthermore, we use $\Gamma(t)$ to denote the FSI interface $\partial\Omega_f(t) \cap \partial\Omega_s(t)$ where the interaction between the fluid and structure takes place. Correspondingly, $\Gamma^0 = \Gamma(0)$ denotes the FSI interface at time $t = 0$. In the rest of this section, we briefly formulate the equations that we use to describe the motion of the beam and the fluid as well as their interaction. The main goal here is to fix the notation necessary for the ensuing description of MMTs. We encourage the reader to consult the following references for more information on: finite element discretization for the Navier–Stokes equations for incompressible flows [13]; finite element discretization for nonlinear elasticity problems [2,34]; ALE formulation of FSI problems [1,19]; partitioned solution approach to FSI problems [17].

2.1 Structure motion

We assume that the structure behavior can be characterized as elastic and compressible. Let the structure in its initial (undeformed) configuration occupy the domain Ω_s^0 . We can describe the structure motion in terms of a displacement field $\mathbf{u}_s : \Omega_s^0 \times [0, T] \rightarrow \mathbb{R}^2$. Note that displacement \mathbf{u}_s can describe a rigid body motion with no deformation involved. Information on whether actual deformation of Ω_s^0 takes place is contained in the deformation gradient $\mathbf{F}_s = \mathbf{I} + \nabla\mathbf{u}_s$. Two important objects derived from the deformation gradient are the Green–Lagrange strain tensor $\mathbf{E}_s = (\mathbf{F}_s^T \mathbf{F}_s - \mathbf{I})/2$ and the Jacobian determinant $J_s = \det \mathbf{F}_s$.

For the material behavior, we use the St. Venant–Kirchhoff constitutive law which links the second Piola–Kirchhoff stress tensor \mathbf{S}_s to the Green–Lagrange strain tensor \mathbf{E}_s :

$$\mathbf{S}_s = \lambda_s \operatorname{tr}(\mathbf{E}_s)\mathbf{I} + 2\mu_s \mathbf{E}_s. \tag{1}$$

The material law (1) includes the Lamé parameters λ_s and μ_s , which are constitutive parameters describing physical properties of the material. They can be computed from Young’s modulus E_s and Poisson’s ration ν_s as

$$\lambda_s = \frac{\nu_s E_s}{(1 + \nu_s)(1 - 2\nu_s)} \quad \text{and} \quad \mu_s = \frac{E_s}{2(1 + \nu_s)}. \tag{2}$$

Second Piola–Kirchhoff stress tensor \mathbf{S}_s measures forces appearing in the deformed structure with respect to its initial configuration Ω_s^0 . In FSI applications, it is important to have an ability to express these forces with respect to the deformed configuration $\Omega_s(t)$. This can be achieved by means of the Cauchy stress tensor $\boldsymbol{\sigma}_s$, which is related to \mathbf{S}_s by

$$\boldsymbol{\sigma}_s = \frac{1}{J_s} \mathbf{F}_s \mathbf{S}_s \mathbf{F}_s^T. \tag{3}$$

In the presence of a given external acceleration $\mathbf{g} : \Omega_s^0 \times [0, T] \rightarrow \mathbb{R}^2$, the displacement \mathbf{u}_s should satisfy the local conservation equations of linear momentum

$$\rho_s \ddot{\mathbf{u}}_s = \operatorname{div} \mathbf{P}_s + \rho_s \mathbf{g} \text{ in } \Omega_s^0. \tag{4}$$

Here, ρ_s is the structure density which we assume to be constant in Ω_s^0 , and $\mathbf{P}_s = \mathbf{F}_s \mathbf{S}_s$ is the first Piola–Kirchhoff stress tensor. Note that Eq. (4) are formulated in the stationary domain Ω_s^0 .

2.2 Fluid motion

To describe the fluid motion, we use the Navier–Stokes equations for incompressible flows. If a computational domain Ω_f does not change with time, the Navier–Stokes equations have the following form:

$$\rho_f \dot{\mathbf{v}}_f + \rho_f \nabla \mathbf{v}_f \cdot \mathbf{v}_f = \operatorname{div} \boldsymbol{\sigma}_f + \rho_f \mathbf{g}, \tag{5}$$

$$\operatorname{div} \mathbf{v}_f = \mathbf{0} \text{ in } \Omega_f. \tag{6}$$

Here, $\mathbf{v}_f : \Omega_f \times [0, T] \rightarrow \mathbb{R}^2$ is a vector field describing the fluid velocity at a given point in Ω , ρ_f is a constant fluid density, and $\boldsymbol{\sigma}_f$ denotes a Cauchy stress tensor. Behavior of an incompressible Newtonian fluid is characterized by the following constitutive law:

$$\boldsymbol{\sigma}_f = -p_f \mathbf{I} + \rho_f \nu_f (\nabla \mathbf{v}_f + \nabla \mathbf{v}_f^T), \tag{7}$$

where $p_f : \Omega_f \times [0, T] \rightarrow \mathbb{R}$ is a pressure field, and ν_f denotes the kinematic viscosity of the fluid.

In FSI applications, one has to account for the motion of the fluid domain. In this work, we consider a common strategy based on ALE mappings. Put simply, an ALE mapping is nothing else but a motion of the computational mesh in the fluid domain described in terms of an auxiliary displacement field $\mathbf{u}_a : \Omega_f^0 \times [0, T] \rightarrow \mathbb{R}^2$ such that $\Omega_f(t) = \Omega_f^0 + \mathbf{u}_a(\cdot, t)$. With an ALE mapping, the Navier–Stokes equations can be formulated in the moving domain $\Omega_f(t)$:

$$\rho_f \dot{\mathbf{v}}_f + \rho_f \nabla \mathbf{v}_f \cdot (\mathbf{v}_f - \dot{\mathbf{u}}_a) = \operatorname{div} \boldsymbol{\sigma}_f + \rho_f \mathbf{g}, \quad (8)$$

$$\operatorname{div} \mathbf{v}_f = \mathbf{0} \text{ in } \Omega_f(t). \quad (9)$$

Since the fluid domain motion is driven by the structure deformation, the ALE displacement \mathbf{u}_a has to comply with the structure displacement \mathbf{u}_s on the FSI interface:

$$\mathbf{u}_a = \mathbf{u}_s \text{ on } \Gamma(t). \quad (10)$$

Inside the fluid domain, \mathbf{u}_a can be chosen arbitrary, hence the term ALE. The only condition is that the ALE displacement should define an invertible deformation of the fluid domain. This means that the following condition has to hold:

$$J_a = \det \mathbf{F}_a = \det(\mathbf{I} + \nabla \mathbf{u}_a) > 0. \quad (11)$$

The scope of this work is to compare different options for defining the ALE displacement in the fluid domain provided the structure displacement on the FSI interface.

2.3 Interaction conditions

Physical interaction between the fluid and structure takes place on the FSI interface $\Gamma(t)$. It is characterized by the following two coupling conditions: the kinematic continuity

$$\mathbf{v}_f = \dot{\mathbf{u}}_s \text{ on } \Gamma(t), \quad (12)$$

which assures that the fluid stays attached to the structure; and the dynamic continuity

$$\begin{aligned} \boldsymbol{\sigma}_s \cdot \mathbf{n} &= \boldsymbol{\sigma}_f \cdot \mathbf{n} \text{ on } \Gamma(t) \Leftrightarrow \\ \mathbf{P}_s \cdot \mathbf{n} &= J_a \boldsymbol{\sigma}_f \mathbf{F}_a^{-T} \cdot \mathbf{n} \text{ on } \Gamma^0, \end{aligned} \quad (13)$$

which maintains the balance of forces on the FSI interface. If a partitioned approach to FSI is used, one has to enforce the coupling conditions by exchanging information between the fluid and structure solvers.

2.4 Initial and boundary conditions

To complete the definition of the benchmark, we provide suitable initial and boundary conditions. The system is initialized with zero initial conditions for the fluid velocity, structure displacement and structure velocity:

$$\begin{aligned} \mathbf{v}_f(\cdot, 0) &= \mathbf{0} \text{ in } \Omega_f^0, \\ \mathbf{u}_s(\cdot, 0) &= \mathbf{0} \text{ in } \Omega_s^0, \quad \dot{\mathbf{u}}_s(\cdot, 0) = \mathbf{0} \text{ in } \Omega_s^0. \end{aligned} \quad (14)$$

The main energy source of the system is an inflow boundary condition on the fluid velocity prescribed on the left end of the channel $\partial\Omega_f^{\text{in}}$. The condition prescribes a parabolic velocity profile

$$\mathbf{v}_{\text{par}} = \begin{pmatrix} v_{\text{max}} \left(\frac{2}{0.41} \right)^2 y(0.41 - y) \\ 0 \end{pmatrix} \quad (15)$$

with the maximum inflow velocity v_{max} serving as an adjustable parameter. In order to comply with the initial conditions, \mathbf{v}_{par} is made time-dependent by scaling it with time:

$$\mathbf{v}_{\text{in}}(t) = \begin{cases} \mathbf{v}_{\text{par}} \frac{1 - \cos(\pi t/2)}{2} & \text{if } t < 2s, \\ \mathbf{v}_{\text{par}} & \text{if } t \geq 2s. \end{cases} \quad (16)$$

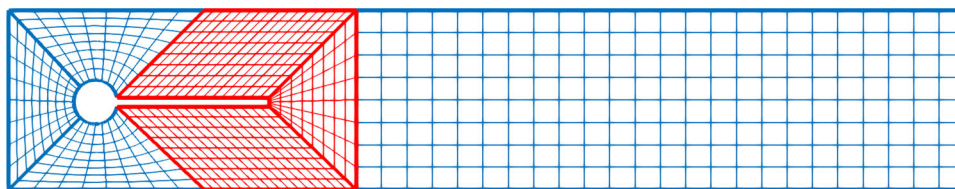
The resulting time-dependent inflow boundary condition $\mathbf{v}_f = \mathbf{v}_{\text{in}}$ provides a smooth warm-up phase for the simulation. On the right wall of the channel $\partial\Omega_f^{\text{out}}$, a do-nothing condition $\boldsymbol{\sigma}_f \cdot \mathbf{n} = \mathbf{0}$ is prescribed. Additionally, a no-slip boundary condition $\mathbf{v}_f = \mathbf{0}$ is set on the rest of the channel wall $\partial\Omega_f^{\text{ns}} = \partial\Omega_f^0 \setminus \partial\Omega_f^{\text{in}} \setminus \partial\Omega_f^{\text{out}} \setminus \Gamma^0$. Finally, the structure is fixed ($\mathbf{u}_s = \mathbf{0}$) on its left end $\partial\Omega_s^0 \setminus \Gamma^0$.

2.5 Geometry parametrization

In the spirit of IGA, we model computational domains as collections of tensor-product NURBS patches [18]. For the structure, we use a single quadratic NURBS patch, whereas the fluid domain is modeled with seven patches. Figure 2 illustrates the isoparametric mesh in the fluid domain after several applications of uniform h -refinement. For simplicity, we use matching parametrizations for neighboring fluid patches as well as for the structure patch and the surrounding fluid patches. This choice does not restrict the applicability of our results; however, it significantly simplifies the exchange of coupling information between the fluid and structure solvers.

Let us consider one of the NURBS patches forming the initial configuration of the fluid domain. Its parametrization $\mathbf{G}_f^0 : [0, 1]^2 \rightarrow \Omega_f^0$ can be written in terms of control points $\mathbf{c}_k \in \mathbb{R}^2$ and tensor-product NURBS basis functions

Fig. 2 Computational mesh in the fluid domain. Three patches adjacent to the beam are allowed to move



$N_k : [0, 1]^2 \rightarrow \mathbb{R}$ as

$$\mathbf{G}_f^0(\xi, \eta) = \sum_{k=1}^n \mathbf{c}_k N_k(\xi, \eta). \tag{17}$$

In IGA, we use the same NURBS basis functions to approximate solutions to PDEs. In our case, we use N_k to discretize the ALE displacement \mathbf{u}_a in space. Assuming that we use time moments $t_0 = 0, \dots, t_i, \dots, t_N = T$ for the time discretization, we denote the ALE displacement at time t_i by \mathbf{u}_a^i . Its NURBS representation can be written as

$$\mathbf{u}_a^i(\xi, \eta) = \sum_{k=1}^n \mathbf{d}_k^i N_k(\xi, \eta), \tag{18}$$

where \mathbf{d}_k^i are the corresponding control points. Given \mathbf{u}_a^i , we can easily construct a NURBS parametrization \mathbf{G}_f^i of the deformed fluid domain $\Omega_f^i = \Omega_f(t_i) = \Omega_f^0 + \mathbf{u}_a^i$ as

$$\mathbf{G}_f^i(\xi, \eta) = \sum_{k=1}^n (\mathbf{c}_k + \mathbf{d}_k^i) N_k(\xi, \eta). \tag{19}$$

Now \mathbf{G}_f^i can be used to discretize the Navier–Stokes Eqs. (8–9) in the deformed configuration of the fluid domain.

As a final comment, let us mention that it is possible to let only a portion of the fluid domain move while keeping the rest fixed. An obvious advantage of this approach is a considerable reduction in computational cost associated with construction of ALE mappings. In this work, only three patches of the fluid domain that are directly adjacent to the FSI interface are allowed to move. In Fig. 2, they are highlighted in red. In our experience, these three patches are more than capable of absorbing deformation of the structure that appears in this benchmark.

3 Mesh moving techniques

In this section, we introduce several commonly used MMTs as well as propose certain variations to them. All considered MMTs achieve the same goal: provided a displacement of the FSI interface at a given time, they extend it into the fluid domain. Note that although we present these techniques in

a 2D setting, one can readily apply them in 3D. When comparing different MMTs, we largely pay our attention to the maximum amount of mesh motion a particular technique can handle. That is, how much the mesh can move before the bijectivity condition (11) is violated. A secondary measure is of course the overall computational cost associated with computing ALE mappings using a given MMT.

After MMTs, we describe the MJBS which we use to augment each of the considered techniques. Finally, we discuss practical ways to check whether the bijectivity condition (11) is satisfied.

3.1 Harmonic extension (HE/IHE)

Probably the simplest way to extend displacement of the FSI interface into the fluid domain is by means of harmonic extension (HE) [8,19,35]. Given the interface displacement \mathbf{u}_s^i at time t_i , the ALE displacement \mathbf{u}_a^i is computed by solving Laplace’s equation in the initial configuration of the fluid domain Ω_f^0 for every displacement component:

$$\Delta \mathbf{u}_a^i = \mathbf{0} \quad \text{in } \Omega_f^0, \tag{20}$$

$$\mathbf{u}_a^i = \mathbf{u}_s^i \quad \text{on } \Gamma^0, \tag{21}$$

$$\mathbf{u}_a^i = \mathbf{0} \quad \text{on } \partial\Omega_f^0 \setminus \Gamma^0. \tag{22}$$

The interface displacement \mathbf{u}_s^i serves as a Dirichlet boundary condition on the FSI interface Γ^0 . At the rest of the boundary, the prescribed displacement is zero. Note that the method does not take into account information about the interface or ALE displacement from the previous time step.

The HE technique is the least computationally expensive way to construct ALE mappings. Let N be the number of inner control points in the fluid domain. Since all displacement components satisfy the same equation, they can be computed by solving a single linear system with an $N \times N$ matrix and an $N \times d$ right-hand side. Here, d is the problem dimension; in our case, $d = 2$. Moreover, once the matrix is assembled, it can be reused for all time steps, which drastically reduces the computational cost associated with matrix assembly in IGA. At each time step, one only has to update the right-hand side to take the current FSI interface displacement into account. This can be performed efficiently by storing a Dirichlet elimination matrix when assembling the main matrix.

Despite its computational efficiency, the HE technique has serious disadvantages. First of all, it treats displacement components as completely independent variables and does not promote bijectivity of the ALE mapping in any way. Second, solutions of Laplace's equation in the vicinity of corners behave like $r^{\pi/\omega}$, where r is the distance to the corner and ω is the corner angle. For reentrant corners, that is for $\omega > \pi$, solutions do not belong to $H^1(\Omega_f^0)$ since their derivatives tend to infinity. As a consequence, the corresponding ALE mapping may lose its bijectivity. Due to these two problems, the HE technique usually is only able to handle rather small mesh motions.

We propose a slight improvement to the HE technique achieved by turning it into an incremental algorithm. Assume that the ALE displacement \mathbf{u}_a^i at time t_i is known. We can use it to deform the initial configuration of the fluid domain Ω_f^0 and obtain the deformed configuration $\Omega_f^i = \Omega_f^0 + \mathbf{u}_a^i$ as Eq. (19) describes. We can then compute an ALE displacement increment $\delta\mathbf{u}_a^{i+1}$ by solving Laplace's equation in the deformed configuration:

$$\Delta\delta\mathbf{u}_a^{i+1} = \mathbf{0} \quad \text{in } \Omega_f^i, \quad (23)$$

$$\delta\mathbf{u}_a^{i+1} = \mathbf{u}_s^{i+1} - \mathbf{u}_s^i \quad \text{on } \Gamma^i, \quad (24)$$

$$\delta\mathbf{u}_a^{i+1} = \mathbf{0} \quad \text{on } \partial\Omega_f^i \setminus \Gamma^i. \quad (25)$$

And finally, we define the ALE displacement \mathbf{u}_a^{i+1} at time t_{i+1} as $\mathbf{u}_a^i + \delta\mathbf{u}_a^{i+1}$. Note that the resulting incremental harmonic extension (IHE) technique is not equivalent to the HE technique since ALE increments are computed in deformed configurations of the fluid domain.

One advantage of the IHE technique is that it uses the ALE displacement from the previous time step. Therefore, IHE can be expected to perform slightly better than the HE technique, meaning that it can handle larger mesh motions. On the other hand, each step of the IHE technique is formulated in a different configuration of the fluid domain than the previous one. As a result, the technique requires matrix assembly at every time step, which makes it more computationally expensive than HE.

In what follows, we apply the same ideas to other MMTs and consider both their non-incremental and incremental versions, which often share the same advantages and disadvantages as the HE and IHE techniques.

3.2 Bi-harmonic extension (BE/IBE)

The HE technique is weak when it comes to large mesh motions. To overcome this problem, one can search for the ALE displacement as a solution to the bi-harmonic equation:

$$\Delta^2\mathbf{u}_a = \mathbf{0} \quad \text{in } \Omega_f. \quad (26)$$

Solutions of the bi-harmonic equation are known to be more regular in comparison to Laplace's equation and do not have problems at reentrant corners [19].

However, this bi-harmonic extension (BE) technique is often dismissed as too computationally expensive. Indeed, in order to solve the bi-harmonic equation, one has two options: either use C^1 -conforming elements, which in IGA requires G^1 -continuity between patches [3,5]; or use mixed elements with an auxiliary variable \mathbf{q} to replace the bi-harmonic equation with two Laplace's equations [4]:

$$\Delta\mathbf{u}_a = \Delta\mathbf{q}, \quad \Delta\mathbf{q} = \mathbf{0} \quad \text{in } \Omega_f. \quad (27)$$

In this work, we consider only the latter option since it is easier to implement for multi-patch geometries.

In our interpretation, the BE technique has the following form: given the interface displacement \mathbf{u}_s^i at time t_i , the ALE displacement \mathbf{u}_a^i is computed by solving the following linear system in the initial configuration of the fluid domain Ω_f^0 :

$$\Delta\mathbf{u}_a^i = \Delta\mathbf{q}, \quad \Delta\mathbf{q} = \mathbf{0} \quad \text{in } \Omega_f^0, \quad (28)$$

$$\mathbf{u}_a^i = \mathbf{u}_s^i \quad \text{on } \Gamma^0, \quad (29)$$

$$\mathbf{u}_a^i = \mathbf{0} \quad \text{on } \partial\Omega_f^0 \setminus \Gamma^0, \quad (30)$$

$$\nabla\mathbf{q} \cdot \mathbf{n} = \mathbf{0} \quad \text{on } \partial\Omega_f^0. \quad (31)$$

The BE technique shares many similarities with HE which make both techniques very efficient: it does not use information from previous time steps; the linear system has to be assembled only once; the multiple-right-hand-sides approach can be used to compute all displacement components at once. However, the resulting linear system is twice the size of the HE linear system: the matrix is of size $2N \times 2N$, and the right-hand size is of size $2N \times d$. Moreover, the linear system has a saddle-point structure, so specialized linear solvers are necessary to solve it efficiently.

Just like with the HE technique, we propose an incremental variation of the bi-harmonic extension (IBE). An increment $\delta\mathbf{u}_a^{i+1}$ is computed by solving the following system in the deformed configuration Ω_f^i :

$$\Delta(\delta\mathbf{u}_a^{i+1}) = \Delta\mathbf{q}, \quad \Delta\mathbf{q} = \mathbf{0} \quad \text{in } \Omega_f^i, \quad (32)$$

$$\delta\mathbf{u}_a^{i+1} = \mathbf{u}_s^{i+1} - \mathbf{u}_s^i \quad \text{on } \Gamma^i, \quad (33)$$

$$\delta\mathbf{u}_a^{i+1} = \mathbf{0} \quad \text{on } \partial\Omega_f^i \setminus \Gamma^i, \quad (34)$$

$$\nabla\mathbf{q} \cdot \mathbf{n} = \mathbf{0} \quad \text{on } \partial\Omega_f^i. \quad (35)$$

After that, the ALE displacement \mathbf{u}_a^{i+1} at time t_{i+1} is defined as $\mathbf{u}_a^i + \delta\mathbf{u}_a^{i+1}$.

The IBE technique requires matrix assembly at each time step but can potentially handle larger mesh motions than the BE technique.

3.3 Linear elasticity (LE/ILE)

The next MMT we consider is based on structural mechanics. It is widely used in FSI applications and belongs to the state-of-the-art in the field [1,23,32]. The core idea is to treat the fluid domain as an elastic body and to construct ALE displacement as a solution to the equations of linear elasticity:

$$\text{div } \boldsymbol{\sigma}_a(\mathbf{u}_a) = \mathbf{0} \quad \text{in } \Omega_f. \tag{36}$$

Here, $\boldsymbol{\sigma}_a$ is the Cauchy stress tensor related to the linearized strain tensor $\boldsymbol{\varepsilon}_a = (\nabla \mathbf{u}_a + \nabla \mathbf{u}_a^T)/2$ by the Hooke’s law:

$$\boldsymbol{\sigma}_a = \lambda_a \text{tr}(\boldsymbol{\varepsilon}_a)\mathbf{I} + 2\mu_a \boldsymbol{\varepsilon}_a. \tag{37}$$

The Lamé parameters λ_a and μ_a depend on Young’s modulus E_a and Poisson’s ratio ν_a . Since we do not apply volumetric or surface force to the fluid domain, Young’s modulus does not affect the resulting ALE displacement. On the other hand, Poisson’s ratio is important because it regulates resistance of the fluid mesh to volumetric changes. A too high value (close to 0.5) would result in an almost incompressible behavior, which could lead to excessive distortion of the mesh elements and numerical instabilities. In contrast to that, a too low value (close to 0 or even negative) can reduce resistance of the fluid mesh to bijectivity violation. Therefore, we recommend choosing a value between 0.3 and 0.45. We refer to [22] for a more thorough analysis of the effects of Poisson’s ratio on the mesh motion.

Unlike the techniques based on harmonic and bi-harmonic extension, the linear-elasticity-based MMT is best known in its incremental version. That is, given the ALE displacement \mathbf{u}_a^i at time t_i , an increment $\delta \mathbf{u}_a^{i+1}$ is computed by solving the linear elasticity equations in the deformed configuration of the fluid domain Ω_f^i :

$$\text{div } \boldsymbol{\sigma}_a(\delta \mathbf{u}_a^{i+1}) = \mathbf{0} \quad \text{in } \Omega_f^i, \tag{38}$$

$$\delta \mathbf{u}_a^{i+1} = \mathbf{u}_s^{i+1} - \mathbf{u}_s^i \quad \text{on } \Gamma^i, \tag{39}$$

$$\delta \mathbf{u}_a^{i+1} = \mathbf{0} \quad \text{on } \partial \Omega_f^i \setminus \Gamma^i. \tag{40}$$

After that, the ALE displacement \mathbf{u}_a^{i+1} at time t_{i+1} is defined as $\mathbf{u}_a^i + \delta \mathbf{u}_a^{i+1}$. We refer to this technique as incremental linear elasticity (ILE). The ILE technique is known for its robustness and an ability to withstand large mesh motions. However, little to no research has been conducted to explain its superior behavior.

With respect to computational cost, the ILE technique involves solving a linear system with a matrix of size $dN \times dN$ and a right-hand side of size $dN \times 1$. The linear system has to be reassembled at each time step. Note that the size of the linear system depends on a dimension of the

problem. Therefore, it scales worse from 2D to 3D than the IHE and IBE techniques.

For the sake of completeness, let us also study a non-incremental version of the ILE technique. The ALE displacement \mathbf{u}_a^i at time t_i is computed by solving the equations of linear elasticity in the initial configuration of the fluid domain Ω_f^0 :

$$\text{div } \boldsymbol{\sigma}_a(\mathbf{u}_a^i) = \mathbf{0} \quad \text{in } \Omega_f^0, \tag{41}$$

$$\mathbf{u}_a^i = \mathbf{u}_s^i \quad \text{on } \Gamma^0, \tag{42}$$

$$\mathbf{u}_a^i = \mathbf{0} \quad \text{on } \partial \Omega_f^0 \setminus \Gamma^0. \tag{43}$$

We call this the linear elasticity (LE) technique. Although one can only expect it to perform well for small deformations, it is rather computationally inexpensive. Similarly to the HE and BE techniques, the LE technique requires matrix assembly only once and lets one reuse the matrix for every time step.

3.4 Nonlinear elasticity (TINE)

The last MMT we present in this paper is based on the equations of nonlinear elasticity. The idea is to construct the ALE displacement at each time step as an approximate solution to the local balance equations of linear momentum

$$\text{div } \mathbf{P}_a(\mathbf{u}_a^i) = \mathbf{0} \quad \text{in } \Omega_f^0, \tag{44}$$

where $\mathbf{P}_a = \mathbf{F}_a \mathbf{S}_a$. To ensure bijectivity of the ALE mapping, we use a logarithmic variation of the neo-Hookean material law

$$\mathbf{S}_a = \lambda_a \ln J_a \mathbf{C}_a^{-1} + \mu_a (\mathbf{I} - \mathbf{C}_a^{-1}), \tag{45}$$

where $\mathbf{C}_a = \mathbf{F}_a^T \mathbf{F}_a$ is the right Cauchy–Green strain tensor. Similarly to the LE and ILE techniques, the Lamé parameters λ_a and μ_a can be computed from Young’s modulus E_a and Poisson’s ratio ν_a , of which only Poisson’s ratio affects the solution of Eq. (44).

Due to the term $\ln J_a$ in the neo-Hookean law (45), any solution of Eq. (44) satisfies the bijectivity condition (11). This fact makes any MMT based on Eqs. (44–45) uniquely powerful since it explicitly enforces the bijectivity condition. Unfortunately, Eq. (44) are nonlinear, and an MMT attempting to fully solve them at each time step would be rather computationally expensive. However, since the ALE mapping should possess certain regularity in time, it is possible to use a solution of Eq. (44) at time t_i to efficiently construct an approximate solution at time t_{i+1} [22]. We refer to this technique as tangential incremental nonlinear elasticity (TINE). The TINE technique can be seen as pseudo time-stepping or an example of the continuation methods for nonlinear problems [7].

Let us look under the hood of TINE. It is based on a Newton-like linearization of Eq. (44). To define it, we need to transform Eq. (44) into a weak form, also known as variation formulation. To that end, let us define a solution space $\mathcal{V} = (H^1(\Omega_f^0))^d$ and a test space $\mathcal{V}_0 = \{\mathbf{w} \in (H^1(\Omega_f^0))^d \mid \mathbf{w} = \mathbf{0} \text{ on } \partial\Omega_f^0\}$. We can then write the weak form of Eq. (44) as

$$\begin{aligned} \text{find } \mathbf{u}_a \in \mathcal{V} \quad \text{such that } \quad \forall \mathbf{w} \in \mathcal{V}_0 \\ R(\mathbf{u}_a, \mathbf{w}) = \int_{\Omega} \mathbf{S}_a : \delta \mathbf{E}_a[\mathbf{w}] d\mathbf{x} = 0. \end{aligned} \quad (46)$$

Here, $\delta \mathbf{E}_a[\mathbf{w}] = \frac{1}{2}(\mathbf{F}_a^T \nabla \mathbf{w} + \nabla \mathbf{w}^T \mathbf{F}_a)$ is the variation of the Green–Lagrange strain tensor. The Taylor expansion at point $(\mathbf{u}_a, \mathbf{w})$ with an increment $\delta \mathbf{u}_a$ yields

$$\begin{aligned} R(\mathbf{u}_a + \delta \mathbf{u}_a, \mathbf{w}) = R(\mathbf{u}_a, \mathbf{w}) \\ + DR(\mathbf{u}_a, \mathbf{w}) \cdot \delta \mathbf{u}_a + o(\|\delta \mathbf{u}_a\|), \end{aligned} \quad (47)$$

where $DR(\mathbf{u}_a, \mathbf{w}) \cdot \delta \mathbf{u}_a$ is a directional derivative. We refer to [22,34] for details on computing $DR(\mathbf{u}_a, \mathbf{w}) \cdot \delta \mathbf{u}_a$.

The idea of the TINE technique is to use one Newton-like step

$$\begin{aligned} \text{find } \delta \mathbf{u}_a \in \mathcal{V} \quad \text{such that } \quad \forall \mathbf{w} \in \mathcal{V}_0 \\ DR(\mathbf{u}_a, \mathbf{w}) \cdot \delta \mathbf{u}_a = -R(\mathbf{u}_a, \mathbf{w}) \end{aligned} \quad (48)$$

per time step to compute an ALE increment and update the ALE displacement. Concretely, given the ALE displacement \mathbf{u}_a^i at time t_i , we find an ALE increment $\delta \mathbf{u}_a^{i+1}$ as a solution of the linear problem

$$DR(\mathbf{u}_a^i, \mathbf{w}) \cdot \delta \mathbf{u}_a^{i+1} = -R(\mathbf{u}_a^i, \mathbf{w}) \quad \forall \mathbf{w} \in \mathcal{V}_0, \quad (49)$$

$$\delta \mathbf{u}_a^{i+1} = \mathbf{u}_s^{i+1} - \mathbf{u}_s^i \quad \text{on } \Gamma^0, \quad (50)$$

$$\delta \mathbf{u}_a^{i+1} = \mathbf{0} \quad \text{on } \partial\Omega_f^0 \setminus \Gamma^0. \quad (51)$$

After that, we define the ALE displacement \mathbf{u}_a^{i+1} at time t_{i+1} as $\mathbf{u}_a^i + \delta \mathbf{u}_a^{i+1}$.

It is natural to compare the TINE and ILE techniques which are very similar at first glance. Both are incremental techniques based on the structural mechanics theory; both require solution of a linear system with a matrix of size $dN \times dN$ and a right-hand side of size $dN \times 1$; both require matrix assembly at each time step. In general, one can expect both techniques to be roughly equal in computational cost. Unlike ILE, however, the TINE technique explicitly enforces the bijectivity condition (11). Moreover, the TINE technique is based in the initial configuration of the fluid domain. As we show in Sects. 4 and 5, this last observation results in crucial differences in behavior of the ILE and TINE techniques when it comes to the accumulated distortion effect.

Let us also briefly compare TINE and the technique introduced in [26] which is based on fiber-reinforced hyperelasticity (nonlinear elasticity) and optimized zero-stress state (FRHE-OZSS). Although both rely on the equations of nonlinear elasticity to construct ALE displacement fields, FRHE-OZSS uses a smart anisotropic hyperelastic material law to achieve an optimal behavior of mesh elements. In comparison to FRHE-OZSS, TINE is rather basis and relies solely on a well-known logarithmic neo-Hookean material law (45). To the best of our understanding, however, the FRHE-OZSS technique solves the nonlinear Eq. (44) at every time step, which may be computationally expensive. Moreover, FRHE-OZSS requires a good initial guess for the nonlinear solver to converge, and in [26] this initial guess is constructed with a (slightly ad-hoc) ramp-up extrapolation approach. Here, TINE takes an alternative approach: it uses one Newton-like step to update the ALE displacement field accordingly to the motion of the FSI interface. The result is a good-quality initial guess for the ALE displacement at the next time step. What is more, this initial guess is so close to the true solution of Eq. (44) that it makes ensuing iterations of the nonlinear solver unnecessary. It would be interesting to apply the same ideas borrowed from the continuation methods in the context of FRHE-OZSS. In this work, however, we do not deal with the FRHE-OZSS technique any further.

3.5 Local stiffening

Most of the fluid mesh motion happens along the FSI interface, where the structure displacement is applied as a Dirichlet boundary conditions to the ALE displacement. On the other hand, mesh elements in the vicinity of the stationary boundary of the fluid domain remain largely intact. Therefore, their contribution into processing of the applied interface displacement is negligible. If the mesh motion could be redistributed away from the FSI interface towards the stationary boundary, the mesh could undergo larger motions without becoming invalid. This is the idea behind local stiffening, which changes the way different elements react to the motion-induced deformation.

Let $\mathbf{G} : [0, 1]^d \rightarrow \Omega$ be a parametrization of the computational domain Ω . Imagine that we have to compute integrals corresponding to matrix entries of the discretized linear system. One of the simplest ways to implement local stiffening is to drop the Jacobian determinant $\det \nabla \mathbf{G}$ when transforming the integrals from domain Ω to parametric domain $[0, 1]^d$:

$$\int_{\Omega} (\dots) d\mathbf{x} = \int_{[0,1]^d} (\dots) \det \nabla \mathbf{G} d\boldsymbol{\xi} \rightarrow \int_{[0,1]^d} (\dots) d\boldsymbol{\xi}. \quad (52)$$

This local stiffening method, initially called Jacobian-based stiffening (JBS) and later renamed to mesh-Jacobian-based

stiffening (MJBS), was first proposed in [31]. For elasticity problems, the MJBS can be interpreted as a local change of Young’s modulus

$$E \rightarrow \frac{E}{\det \nabla_{\xi} \mathbf{G}}, \tag{53}$$

which makes elements with small values of $\det \nabla \mathbf{G}$ stiffer and elements with large values softer. Therefore, the former elements undergo less motion and are less likely to become invalid.

A more advanced version of the MJBS introduced in [23] does not simply drop the Jacobian determinant but changes the degree with which it enters the integrals:

$$\begin{aligned} \int_{\Omega} (\dots) d\mathbf{x} &= \int_{[0,1]^d} (\dots) \det \nabla \mathbf{G} d\xi \\ &\rightarrow \int_{[0,1]^d} (\dots) (\det \nabla \mathbf{G})^{1-\chi} d\xi. \end{aligned} \tag{54}$$

Here, $\chi \geq 0$ is called the stiffening degree. The higher the stiffening degree is, the more local stiffening is achieved. $\chi = 0$ corresponds to no local stiffening, and $\chi = 1$ corresponds to Jacobian dropping (52). Too high stiffening degrees, however, may result in excessive mesh distortion.

Note that in its original form described in [31], the MJBS relies not on the Jacobian determinant of the mapping between the parametric space and the physical domain. Instead, it works with the mapping between the element domain and the physical domain [6]. That is, for each element Ω_k in the physical domain, consider an element-specific mapping $\mathbf{G}_k : \Omega_e \rightarrow \Omega_k$, where $\Omega_e = [-1, 1]^d$ is the element domain, also known as the parent element. Mapping \mathbf{G}_k and the domain parametrization \mathbf{G} are related via $\mathbf{G}_k = \mathbf{G} \circ \mathbf{S}_k$, where \mathbf{S}_k is a scaling mapping between the element domain and the image of the element Ω_k in the parametric space. Then the MJBS is defined in the following element-wise way:

$$\begin{aligned} \int_{\Omega_k} (\dots) d\mathbf{x} &= \int_{\Omega_e} (\dots) \det \nabla \mathbf{G}_k d\xi_e \\ &\rightarrow \int_{\Omega_e} (\dots) (\det \nabla \mathbf{G}_k)^{1-\chi} d\xi_e. \end{aligned} \tag{55}$$

In this formulation, the MJBS takes into account not only the size and shape of elements in the physical domain but also the size of the elements in the parametric space. As a result, it produces expected local stiffening when local refinement or non-uniform knot vectors are used, and formulation (55) is widely adopted in the IGA community [26]. However, if neither local refinement nor non-uniform knot vectors are used (like in this work), the MJBS formulations (54) and

(55) are equivalent. Since the IGA code that we use performs integration in the parametric space, the formulation (54) is easier to implement and we resort to it in this work.

The MJBS acts differently depending on whether a particular MMT is formulated in the initial or in the deformed configuration of the fluid domain. Namely, if the integrals for matrix entries are computed in the initial configuration Ω_f^0 , the MJBS is based only on the initial parametrization \mathbf{G}_f^0 . However, in the case of the deformed configuration Ω_f^i , the MJBS takes into account already applied motion since the parametrization \mathbf{G}_f^i of the deformed configuration is defined as $(\mathbf{I} + \mathbf{u}_a^i) \circ \mathbf{G}_f^0$, see Eq. (19). This effect has both advantages and disadvantages. From one point of view, if a particular mesh element becomes ill-shaped after the motion, its value of $\det \nabla \mathbf{G}_f^i$ decreases. As a result, this element receives more stiffening, which prevents it from becoming even more ill-shaped or invalid. On the other hand, in case of the ILE technique, this motion-aware stiffening essentially makes material properties of the mesh deformation-dependent, which can cause irreversible plastic deformation. For other MMTs based in the deformed configuration, namely IHE and IBE, the effect is similar. As we show in Sects. 4 and 5, this irreversible deformation accumulates over time and can significantly affect results of FSI simulations. We refer to this effect as accumulated distortion.

Regardless if the MJBS is motion-aware or not, the initial parametrization \mathbf{G}_f^0 of the fluid domain provides a major contribution to how much stiffening each element receives. Let us consider the moving part of the fluid domain, see Fig. 3. The top and bottom patches are perfect parallelograms, and $\det \nabla \mathbf{G}_f^0$ is constant. Therefore, elements of these patches receive no local stiffening with respect to each other. However, the right patch has a distinct tapered left side, where $\det \nabla \mathbf{G}_f^0$ becomes very small in comparison to the surrounding elements of the top and bottom patches. As a result, element on the left side of the right patch become much stiffer and maintain their shape. Consequently, angles of all three

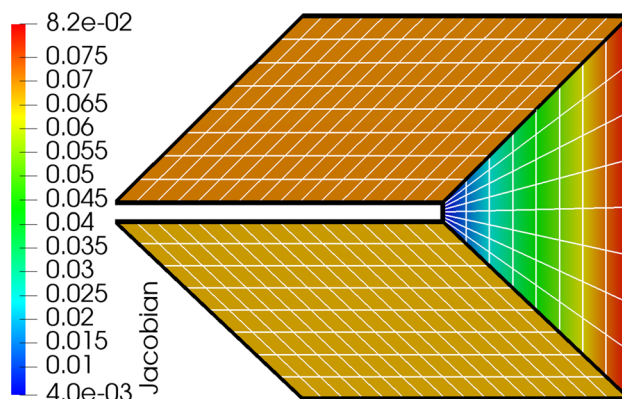


Fig. 3 MJBS potential of the motion fluid mesh part

mesh patches that are adjacent to the beam right end do not change much during the mesh motion. In particular, they do not exceed π , which would lead to $\det \nabla \mathbf{G}_f^i$ becoming negative, which means that the bijectivity condition (11) is not violated.

3.6 Bijectivity check

Let us briefly discuss ways to check the bijectivity condition (11) in practice. A solution which takes the NURBS nature of the ALE displacement \mathbf{u}_a into account is to express J_a as a NURBS function [10]. If all coefficients in a NURBS representation of J_a are positive, then the displacement \mathbf{u}_a satisfies the bijectivity condition. Unfortunately, this condition is only sufficient and not a necessary one. Therefore, it may often lead to false detection of bijectivity violation. In practice, we resort to a less elegant solution of sampling J_a at the Gaussian quadrature points associated with the NURBS basis of \mathbf{u}_a .

Note that whichever method is chosen, it introduces a certain computational overhead to construction of ALE mappings. Nevertheless, we recommend doing some bijectivity check at every time step, or at least with regular intervals. An ALE mapping that does not satisfy the bijectivity condition (11) quickly makes all ensuing computation results meaningless.

4 Benchmark ALE: mesh motion

In order to test and compare all MMTs, we first consider a simplified FSI-like test based on the benchmark introduced in Sect. 2. Instead of solving a fully coupled FSI problem, we ignore the fluid component and let the flexible beam oscillate freely in the presence of external acceleration $\mathbf{g} = (0, l)$. We use the beam motion to drive motion of the three fluid domain patches adjacent to the beam. By varying the parameter l , we can regulate magnitude of the mesh motion. Although this mesh motion test is artificial, it mimics real motion occurring in the original benchmark well enough. Moreover, it is significantly less computationally expensive, which allows us to conduct more tests and better assess properties of each MMT.

For this test, we refine the fluid domain parametrization by applying uniform h -refinement thrice. Figure 4 shows the corresponding computational mesh in the state of maximum beam deflection for loading levels $l = 1, 2, 3$. We use the following parameters for the structure motion: Young's modulus $E_s = 1.4 \times 10^6 \text{ kg m}^{-1} \text{ s}^{-2}$, Poisson's ratio $\nu_s = 0.4$, density $\rho_s = 10^3 \text{ kg m}^{-3}$. For the elasticity-based MMTs, we use Poisson's ratio $\nu_a = 0.3$. At each time step, we check whether the bijectivity condition (11) holds. We have implemented this test and the original FSI benchmark within G+Smo—an

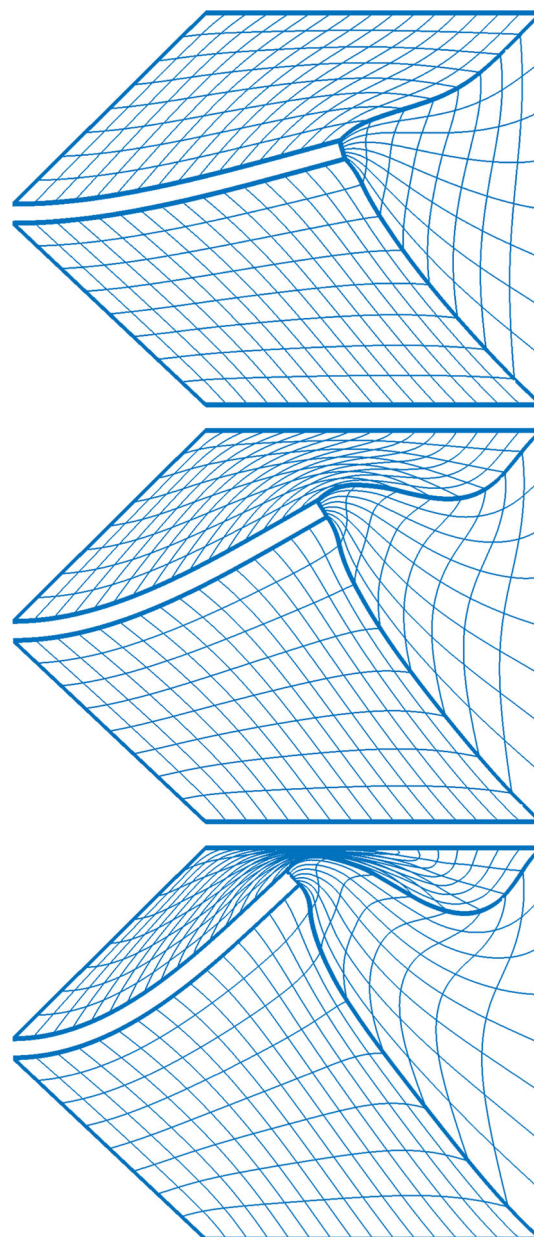


Fig. 4 Computational mesh with TINE in the state of maximum beam deflection for loading levels $l = 1, 2, 3$ (top to bottom)

open-source C++ library for isogeometric analysis [15]—using the `gsElasticity` submodule. As a linear system solver, we use Pardiso—an efficient parallel direct linear solver [16]. For reference, all simulations have been performed on a laptop with a 7th generation Intel Core i7 CPU using eight hyper-threads.

4.1 Single period test

First, we consider mesh motion over one period of beam oscillations. The goal is to study how much motion each

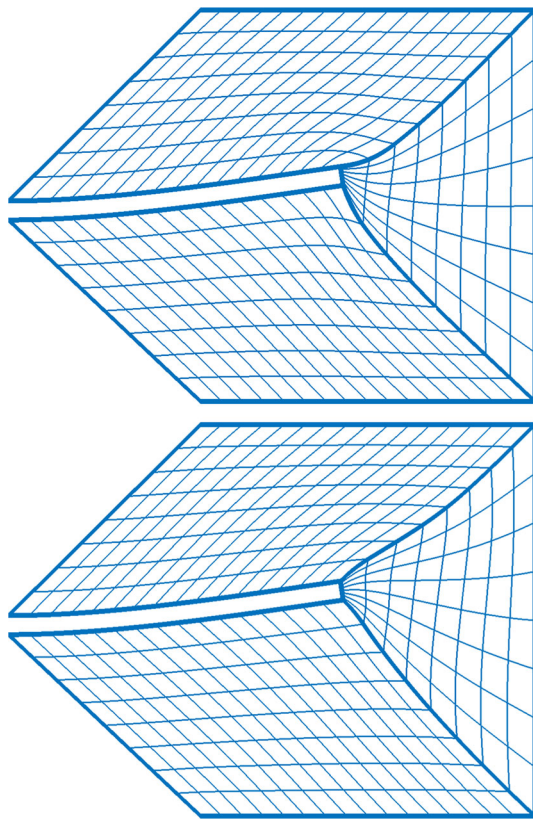


Fig. 5 Top: mesh motion with TINE for loading level $l = 0.5$ with stiffening degree $\chi = 0$. Bottom: mesh motion with TINE for $l = 0.5$ and $\chi = 2$

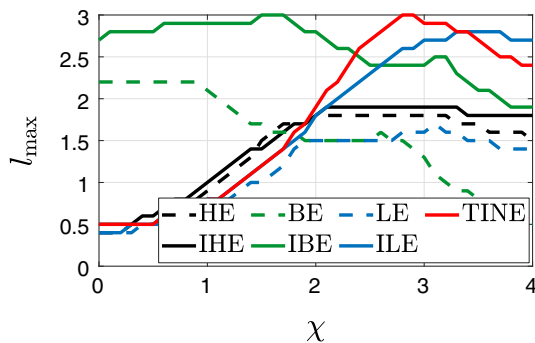


Fig. 6 Single oscillation period test: maximal achievable loading level l_{\max} vs the stiffening degree χ for different MMTs

MMT can handle. Here, the MJBS stiffening is of crucial importance. Without it, most MMTs can handle only small loading levels l . Usually, one of the patch corners adjacent to the right end of the beam becomes larger than π , which violates the bijectivity conditions (11). With the MJBS, all MMTs can keep these angles below π at least for moderate loading levels l , see Fig. 5.

Figure 6 shows a plot of the maximum achievable loading level l_{\max} versus the stiffening degree χ for each MMT.

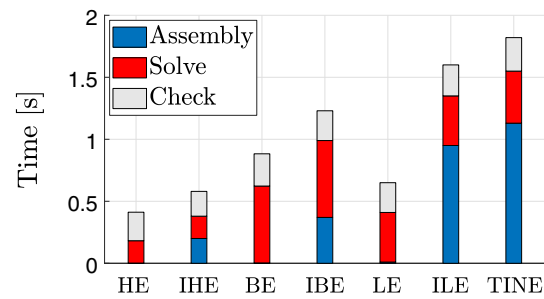


Fig. 7 Single oscillation period test: computational time for each MMT split into the assembly, solving and bijectivity check parts

Immediately, we can split all MMTs into two groups, with BE and IBE forming one group, and all other techniques belonging to the second group. The main difference between the two groups is that MMTs from the second group can handle almost no deformation without the MJBS. However, as the stiffening degree χ grows, these MMTs can handle increasingly larger loading levels with maximum loading levels achieved with $\chi \in [2, 3]$. With $\chi > 3$, we can observe some form of performance deterioration for all MMTs, which is likely caused by too much mesh distortion introduced by the MJBS.

For the BE and IBE techniques, the behavior is radically different. Already without the MJBS, they can handle larger loading levels than some MMTs from the second group can achieve even with high values of the stiffening degrees χ . However, as we increase the stiffening degree, BE and IBE show almost no response for $\chi < 1$ and start to slowly perform worse for $\chi > 1$.

Overall, MMTs can be ranged with respect to their capability to handle large mesh motion in the following way: IBE, ILE and TINE are the most powerful and can handle loading levels up to 2.8–3; BE occupies the second place with the maximum loading level of 2.2; and HE, IHE and LE are the least powerful with maximum loading levels of 1.6–1.9.

Figure 7 presents an analysis of computational complexity of each MMT. We have measured time required for linear system assembly, linear system solution and an ensuing check of the bijectivity condition (11). Note that this is real time and not CPU time. Although not a perfect measure of algorithm performance, real time still allows us to compare relative efficiency of different MMTs since we have implemented them in the same framework of G+Smo.

The time analysis shows that the HE, BE and LE techniques are significantly faster than their incremental versions. This result is not surprising because non-incremental techniques do not require matrix assembly at each time step. The BE and IBE techniques take the largest amount of time to solve the linear system due to the saddle-point structure

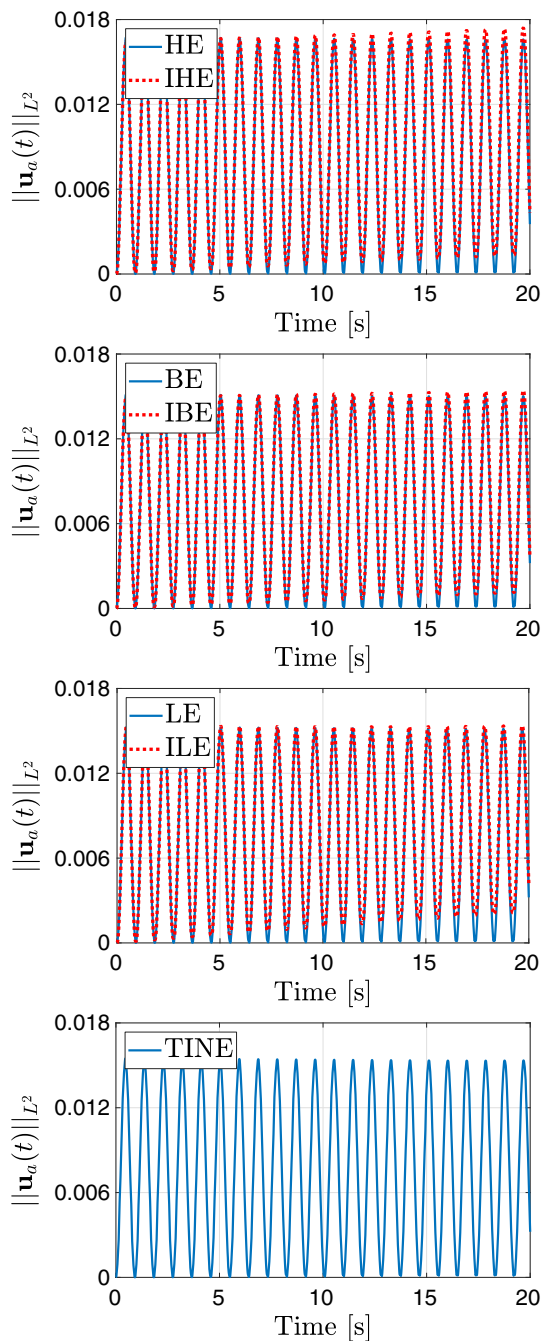


Fig. 8 Long-term behavior test. L^2 -norm of ALE displacement over time

of the system. At the same time, relative complexity of the elasticity equations makes the assembly time of the ILE and TINE techniques significantly larger than for other techniques. Overall, the HE, IHE and LE techniques are the most efficient. The ILE and TINE are the most computationally expensive, and the BE and IBE techniques fall in between. All techniques include a small overhead associated with the bijectivity check. This overhead can be reduced by not performing the check at every time step.

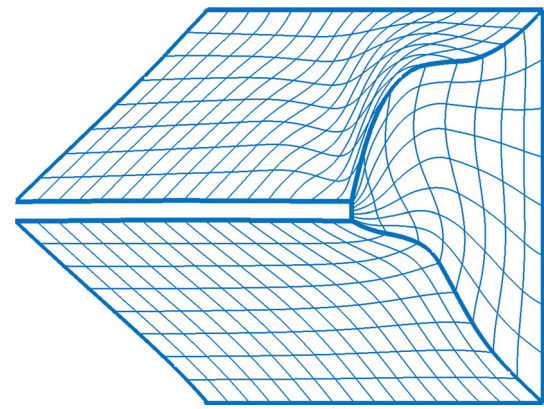


Fig. 9 Long-term behavior test. Accumulated mesh distortion with ILE and $\chi = 2$ after 20 oscillation periods with $l = 2$

4.2 Long-term behavior test

In the second test, we study the long-term behavior of different MMTs and their effect on the fluid mesh. To that end, we perform the simulation over a time period of 20s, which includes roughly 22 periods of beam oscillations. A quantity of interest is the L^2 -norm of the ALE displacement measured in the initial configuration of the fluid domain. A perfect MMT should return the fluid mesh to its initial state once the beam is not deformed. Therefore, the ALE norm $\|\mathbf{u}_a(t)\|_{L^2(\Omega_f^0)}$ should be close to zero at the end of each oscillation period. Figure 8 shows behavior of the ALE norm over time for each MMT with the loading level $l = 1.5$. We have used the MJBS with $\chi = 2$ for all MMTs with the exception of the BE and IBE techniques. These techniques are able to handle the loading level $l = 1.5$ without local stiffening.

As Fig. 8 shows, the ALE norm behaves periodically and returns to zero at the end of each oscillation period with the HE, BE, LE and TINE techniques. On the other hand, with the IHE, IBE and ILE techniques the ALE norm at the end of each oscillation period grows in a monotonous fashion. This effect has been previously reported in [28], and we refer to it as accumulated distortion. It appears only for MMTs which are based in the deformed configuration of the fluid domain. As a result, mesh motion becomes path-dependent, the fluid mesh does not return to its initial state, and its quality deteriorates over time. Figure 9 illustrates the state of the mesh at the end of the simulation with the ILE technique.

The accumulated distortion effect becomes even more prominent as the magnitude of mesh motion grows. To study it in more details, we perform the long-term behavior test for the IHE, IBE and ILE techniques with varying values of the loading level and stiffening degree. In Fig. 10, we plot values of the ALE norm at the end of each oscillation period. We can observe that both parameters seem to increase the rate of accumulated distortion.

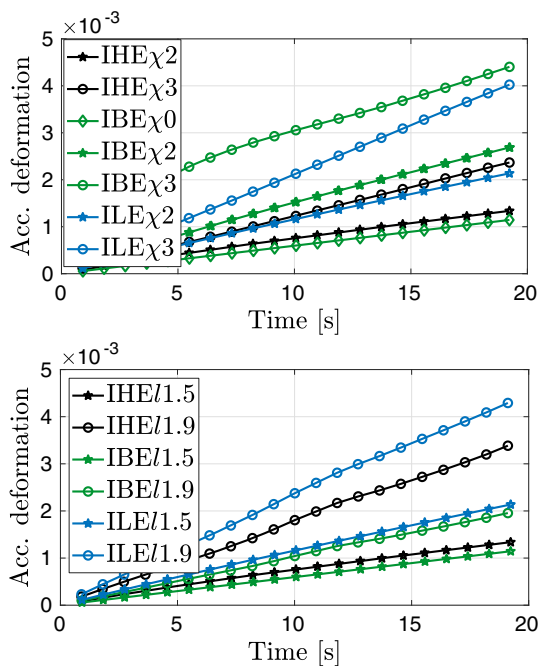


Fig. 10 Long-term behavior test. Accumulated distortion for incremental MMTs. Top: fixed loading level $l = 1.5$ and varying stiffening degree χ . Bottom: fixed stiffening degree $\chi = 2$ and varying loading level l

5 Benchmark FSI2: flow-induced vibrations

In this section, we perform the FSI simulation described in Sect. 2. From several simulation scenarios proposed in

[33], we choose a scenario titled FSI2 since it corresponds to the largest magnitude of beam displacement and mesh motion. Using this FSI simulation, we test and compare different MMTs introduced in Sect. 3. We use stiffening degree $\chi = 2.5$ for all MMTs but BE and IBE. To them, we apply no local stiffening.

For the analysis, we refine the geometry parametrization five times using uniform h -refinement and perform the simulation for 15 s with a time step $\Delta t = 0.0025$ s. For time integration, we use the Newmark method [34] with $\beta = 0.5$ and $\gamma = 1$ for the structure and the IMEX scheme [13] with $\theta = 0.5$ and no stabilization for the fluid. We achieve the coupling of fluid and structure by means of the partitioned Fluid–Dirichlet–Structure–Neumann algorithm [8].

The FSI2 scenario is characterized by the following parameters: fluid density $\rho_f = 10^3 \text{ kg} \cdot \text{m}^{-3}$, fluid kinetic viscosity $\nu_f = 10^{-3} \text{ m}^2 \cdot \text{s}^{-1}$, maximum inflow velocity $v_{\max} = 1.5 \text{ m} \cdot \text{s}^{-1}$, structure density $\rho_s = 10^4 \text{ kg} \cdot \text{m}^{-3}$, structure Young’s modulus $E = 1.4 \times 10^6 \text{ kg} \cdot \text{m}^{-1} \cdot \text{s}^{-2}$, structure Poisson’s ratio $\nu_s = 0.4$, gravitational acceleration $\mathbf{g} = (0, 0)^T \text{ m} \cdot \text{s}^{-2}$ and mesh Poisson’s ratio $\nu_a = 0.3$ (where applicable). The corresponding Reynolds number is $Re = 100$, which results in an unstable flow and development of vortex shedding. Alternating downward and upward forces exerted on the structure by the fluid lead to oscillations of the beam which grow in magnitude until they reach a fully periodic regime. Figure 11 illustrates typical fluid velocity field and beam deformation when the oscillations are fully developed.

Fig. 11 Benchmark FSI2: fully developed oscillation regime. Fluid velocity field for the maximal upward and downward beam deflection

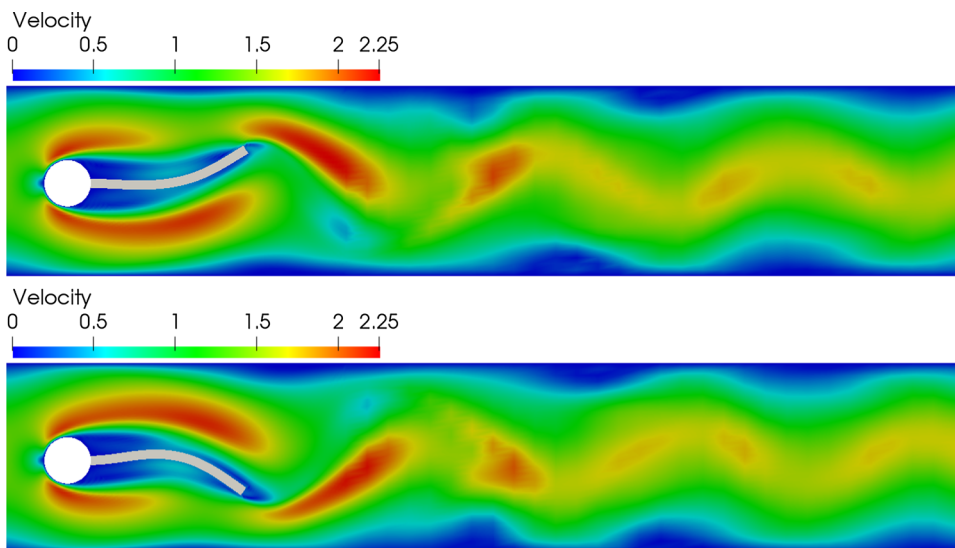


Table 1 Benchmark FSI2: simulation results with TINE ($\Delta t = 0.0025$ s, 44122 DoFs) vs reference results ($\Delta t = 0.001$ s, 304128 DoFs)

	F_D (N)	F_L (N)	u_s^x (A) (10^{-3} m)	u_s^y (A) (10^{-3} m)	Fr (1/s)
Results	203.19 ± 63.88	1.22 ± 223.35	-12.75 ± 11.37	1.24 ± 74.55	1.94
Reference	208.83 ± 73.75	0.88 ± 234.2	-14.58 ± 12.44	1.23 ± 80.6	2.0

Fig. 12 Benchmark FS12: drag and lift behavior with different MMTs

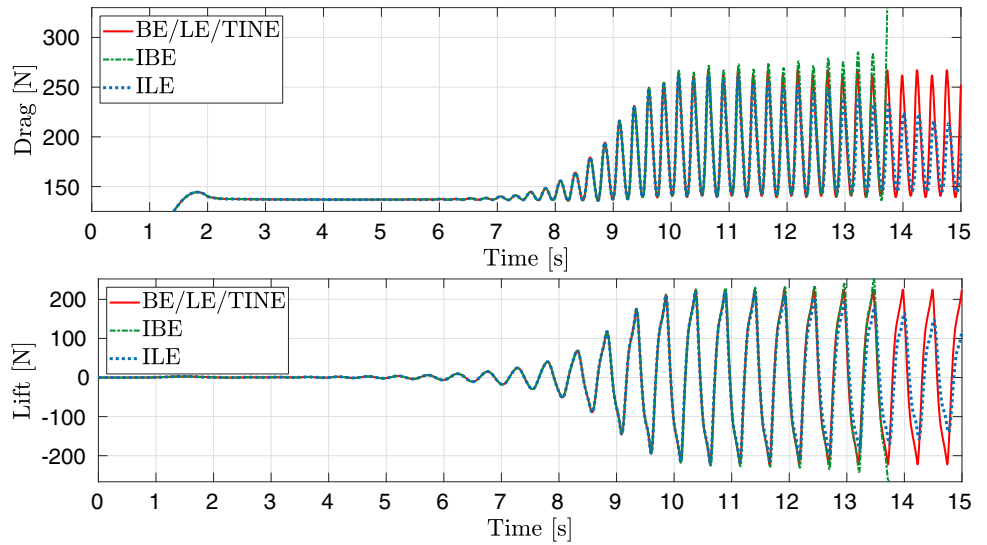


Fig. 13 Benchmark FS12: displacement of the middle point of the beam right end with different MMTs

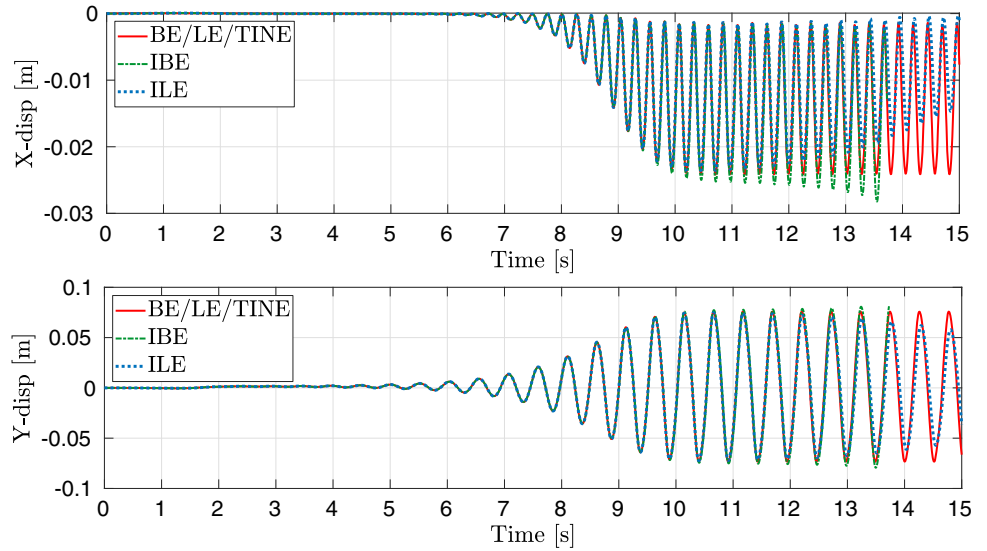


Fig. 14 Benchmark FS12: portion of simulation completed by different MMTs before the ALE mapping becomes invalid

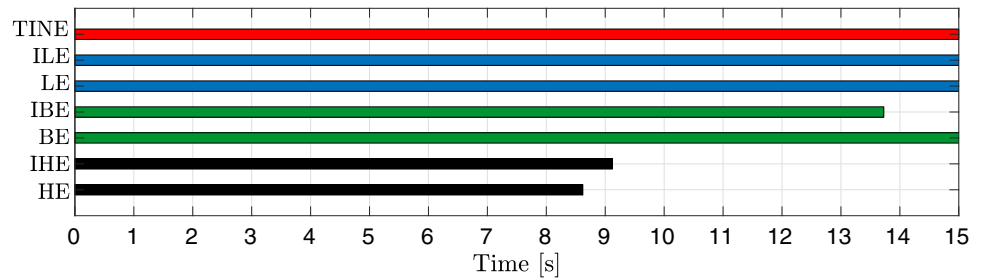
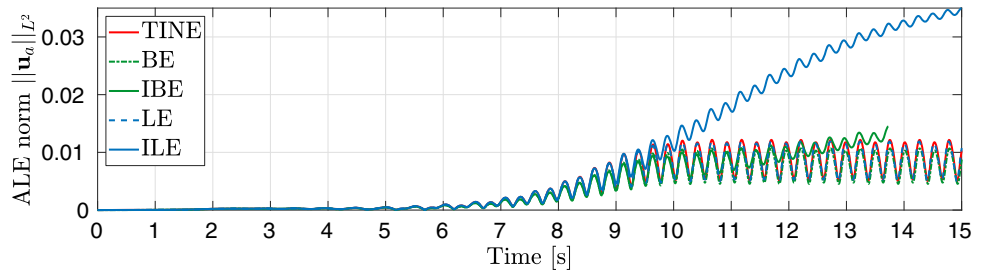


Fig. 15 Benchmark FS12: ALE displacement norm with different MMTs



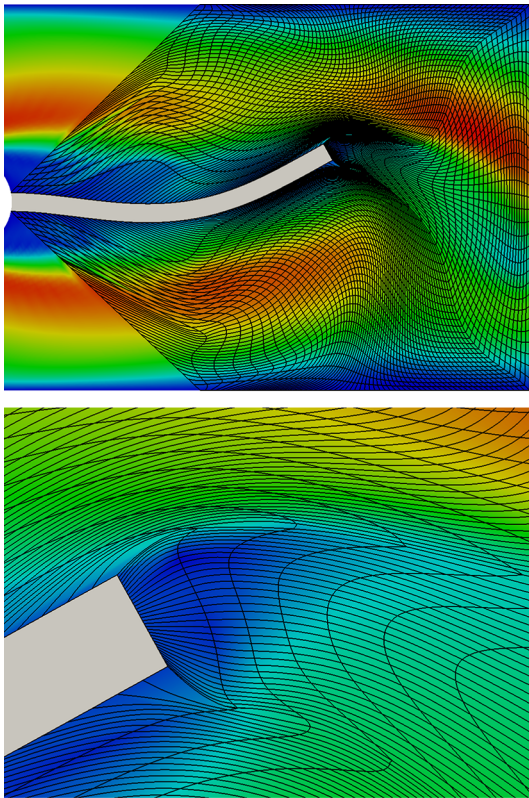


Fig. 16 Benchmark FSI2: accumulated distortion of the fluid mesh during the last oscillation period with the ILE technique

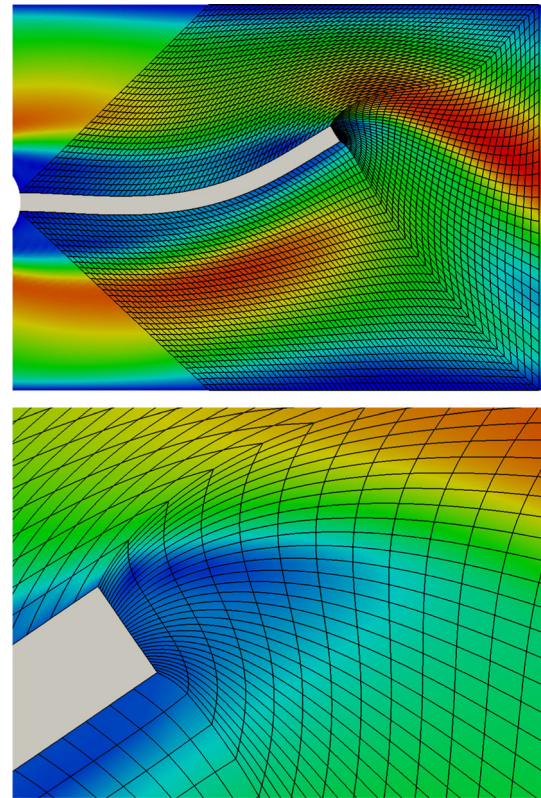


Fig. 17 Benchmark FSI2: fluid mesh with during the last oscillation period with the TINE technique

To assess the simulation accuracy, we study the following quantities of interest when the beam oscillations are fully developed : x - and y -displacement components $u_s^x(A)$ and $u_s^y(A)$ of the point A located in the middle of the beam right end; and drag and lift forces F_D and F_N exerted on the structure by the fluid which are defined as

$$(F_D, F_N) = \int_{\Sigma(t)} \sigma_f(\mathbf{v}_f, p_f) \cdot \mathbf{n} ds. \tag{56}$$

Here, $\Sigma(t)$ is the entire boundary of the submerged solid, including the rigid disk and the flexible beam, at time t . Since we expect the quantities of interest to behave periodically, we report them in terms of their mean $((*)_{\max} + (*)_{\min})/2$, amplitude $((*)_{\max} - (*)_{\min})/2$ and frequency. In Table 1, we compare the simulation results obtained with the TINE technique against the reference results from [33]. Overall, our results seem to undershoot the reference values by about 5%, which can be expected since we use much fewer degrees of freedom and a larger time step than the reference simulation. Despite this discrepancy, we are more than capable of reproducing a qualitatively correct behavior of the system and can use it to study the MMTs. Figures 12 and 13 illustrate behavior of the lift, drag and beam displacement.

Let us now focus on the fluid mesh motion. When performing the FSI simulation, we apply each of the seven MMTs (HE, IHE, BE, IBE, LE, ILE and TINE) and study how a particular MMT handles the mesh motion occurring during the simulation. Figure 14 illustrates what portion of the simulation can be completed using different MMTs. As we can observe, simulations with the HE, IHE and IBE techniques had to be terminated before they could reach the end. All three techniques have failed to maintain bijectivity of the ALE mapping; however, different reasons have led to the failure. In the case of HE and IHE, the simulations have stopped at the 9th second—when the oscillations in the system start to develop. As the applied mesh motion grows, the HE and IHE techniques fail due to their intrinsic inability to handle large motions.

On the other hand, the IBE technique is able to handle mesh motions occurring in the simulation but suffers from the accumulated deformation effect described in Sect. 4. As a result, the simulation fails at the 14th second. Figure 15 depicts behavior of the ALE norm for all MMTs. We can observe that the ILE technique suffers from even stronger accumulated deformation than IBE but still manages to maintain a bijective ALE mapping until the simulation end. Unfortunately, the highly distorted mesh produced by the ILE technique (see Fig. 16) affects the simulation results,

Table 2 Benchmark FSI2: computational time comparison for MMTs which successfully completed the simulation

	ALE time	Total time
BE	2 h 10 m (−48%)	23 h 55 m (−7.3%)
LE	1 h 39 m (−60%)	23 h 20 m (−9.6%)
ILE	4 h 7 m (+0%)	25 h 48 m (+0.0%)
TINE	4 h 52 m (+18%)	26 h 27 m (+2.5%)

The ILE technique time is used as a reference point for comparison

see Figs. 12 and 13. Instead of a stable periodic behavior in the fully developed oscillation regime, we observe signs of damping. We observe a similar effect when using the IBE technique: the accumulated mesh distortion results in spurious amplification of the oscillations.

Of the seven MMTs considered in this work, only three—BE, LE and TINE—were able to handle mesh motions occurring in the simulation and maintain high mesh quality until the simulation end. Most importantly, the BE, LE and TINE techniques have demonstrated no signs of accumulated distortion. Using these techniques, we were able to reproduce a stable periodic behavior of the system and correct simulation results. Figure 17 depicts the fluid mesh at the end of the simulation deformed using the TINE technique. Of course, the BE, LE and TINE techniques differ a lot in terms of their computational cost. However, since construction of the ALE mapping corresponds only to a small portion of a total computational effort required to perform an FSI simulation, the choice of MMT does not affect the total computational cost too much. Table 2 compares computational cost of BE, LE and TINE against the ILE technique which is often considered a default option in the FSI community.

6 Discussion and conclusion

In this work, we have described and compared several MMTs which can be used within moving-mesh methods for FSI problems. To evaluate each MMT, we have used a 2D FSI benchmark and its simplified version where the focus lies on mesh motion. Based on the tests performed in Sects. 4 and 5, we can make the following conclusions:

- Out of seven MMTs that we have considered, two most robust are the BE and TINE techniques. Both MMTs can handle large mesh motions and do not suffer from the accumulated distortion effect.
- BE is easier to implement, performs well even without the MJBS and is about two times less computationally expensive than TINE. Provided that the saddle-point structure of the linear system is accounted for, we recommend the

BE technique as the first method to try in many FSI applications.

- The TINE technique is the most computationally expensive of all considered MMTs. However, it can also handle the largest magnitude of mesh motion when combined with the MJBS. Given its high computational cost and implementation complexity, we recommend TINE for FSI applications where extreme mesh motions are expected.
- The IBE and ILE techniques can handle as much mesh motion as TINE and are slightly less computationally expensive. Unfortunately, both techniques suffer from the accumulated distortion effect which can affect the simulation results over long periods of time. We urge the reader to exercise caution when applying this techniques. Detrimental effects of accumulated distortion can be reset by a costly remeshing operation.
- Although not suitable for large mesh motions, the HE and LE techniques can be applied if only small mesh motions are expected. The exceptional implementation simplicity and low computational cost make HE and LE viable options in certain situations.
- Finally, the IHE technique can handle only small motions and suffers from accumulated distortion. We do not recommend using this technique.

We would like to emphasize that the performance of all MMTs is dependent on the chosen parametrization of the fluid domain. However, it is unlikely that the MMT behavior will be qualitatively different if a different geometry parametrization is used.

With respect to the further research directions, we see the following possibilities. One could study the effect of different geometry parametrizations on the MMT behavior. Moreover, isogeometric methods in FSI could benefit from alternative local stiffening approaches. The commonly used MJBS has no effect if a uniform geometry parametrization is used. It would also be interesting to combine BE and TINE with additional augmentation techniques such as the solid layer extension and mesh element relaxation. Finally, one could try to apply the continuation ideas used in the TINE technique to improve the efficiency of the FRHE-OZSS technique.

Acknowledgements We would like to express our gratitude to Dr. Michael Helmut Gfrerer and Dr. Matthias Möller for extensive and fruitful discussions about fluid-structure interaction. The support of this research by the German Research Council (DFG) under Grant No. SI 765/5-1 (project YASON) and by the German Federal Ministry of Education and Research (BMBF) under Grant No. 05M16UKD (project DYMARA) is greatly acknowledged.

Funding Open Access funding enabled and organized by Projekt DEAL.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Bazilevs Y, Takizawa K, Tezduyar TE (2013) Computational fluid-structure interaction: methods and applications. Wiley, Hoboken. <https://doi.org/10.1002/9781118483565>
- Bernal LM, Calo VM, Collier N, Espinosa GA, Fuentes F, Mahecha JC (2013) Isogeometric analysis of hyperelastic materials using PetIGA. *Proc Comput Sci* 18:1604–1613. <https://doi.org/10.1016/j.procs.2013.05.328>
- Birner K, Jüttler B, Mantzaflaris A (2018) Approximation power of G^1 -smooth isogeometric splines on volumetric two-patch domains. <https://hal.inria.fr/hal-02275654/document>. Accessed 19 June 2020
- Boffi D, Brezzi F, Fortin M et al (2013) Mixed finite element methods and applications, vol 44. Springer, Berlin. <https://doi.org/10.1007/978-3-642-36519-5>
- Collin A, Sangalli G, Takacs T (2016) Analysis-suitable G^1 multi-patch parametrizations for C^1 isogeometric spaces. *Comput Aided Geom Des* 47:93–113. <https://doi.org/10.1016/j.cagd.2016.05.009>
- Cottrell JA, Hughes TJR, Bazilevs Y (2009) Isogeometric analysis: toward integration of CAD and FEA. Wiley, Hoboken
- Deuffhard P (2011) Newton methods for nonlinear problems: affine invariance and adaptive algorithms. Springer, Berlin
- Dörfel MR (2011) Fluid-structure interaction: a differential-algebraic approach and acceleration techniques for strong coupling. VDI Verlag, Düsseldorf
- Formaggia L, Quarteroni A, Veneziani A (2010) Cardiovascular Mathematics: Modeling and simulation of the circulatory system. Springer, Berlin. <https://doi.org/10.1007/978-88-470-1152-6>
- Gravesen J, Evgrafov A, Nguyen DM, Nørtoft P (2014) Planar parametrization in isogeometric analysis. In: International conference on mathematical methods for curves and surfaces 2012. Springer, pp 189–212. <https://doi.org/10.1007/978-3-642-54382-1-11>
- Hughes TJR, Cottrell JA, Bazilevs Y (2005) Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Comput Methods Appl Mech Eng* 194:4135–4195. <https://doi.org/10.1016/j.cma.2004.10.008>
- Hughes TJR, Liu WK, Zimmermann TK (1981) Lagrangian–Eulerian finite element formulation for incompressible viscous flows. *Comput Methods Appl Mech Eng* 29(3):329–349. [https://doi.org/10.1016/0045-7825\(81\)90049-9](https://doi.org/10.1016/0045-7825(81)90049-9)
- John V (2016) Finite element methods for incompressible flow problems. Springer, Berlin. <https://doi.org/10.1007/978-3-319-45750-5>
- Johnson AA, Tezduyar TE (1994) Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces. *Comput Methods Appl Mech Eng* 119(1–2):73–94. [https://doi.org/10.1016/0045-7825\(94\)00077-8](https://doi.org/10.1016/0045-7825(94)00077-8)
- Jüttler B, Langer U, Mantzaflaris A, Moore S, Zulehner W (2014) Geometry + simulation modules: implementing isogeometric analysis. In: Proceedings in applied mathematics and mechanics. Wiley, pp 961–962. <https://doi.org/10.1002/pamm.201410461>
- Kourounis D, Fuchs A, Schenk O (2018) Towards the next generation of multiperiod optimal power flow solvers. *IEEE Trans Power Syst* 33:4005–4014. <https://doi.org/10.1109/TPWRS.2017.2789187>
- Küttler U, Wall WA (2008) Fixed-point fluid-structure interaction solvers with dynamic relaxation. *Comput Mech* 43(1):61–72. <https://doi.org/10.1007/s00466-008-0255-5>
- Piegl L, Tiller W (1997) The NURBS book. Springer, Berlin. <https://doi.org/10.1007/978-3-642-97385-7>
- Richter T (2010) Numerical methods for fluid-structure interaction problems. University of Heidelberg, Germany, Institute for Applied Mathematics
- Richter T, Wick T (2010) Finite elements for fluid-structure interaction in ALE and fully Eulerian coordinates. *Comput Methods Appl Mech Eng* 199(41–44):2633–2642. <https://doi.org/10.1016/j.cma.2010.04.016>
- Schilling D, Dede L, Scott MA, Evans JA, Borden MJ, Rank E, Hughes TJR (2012) An isogeometric design-through-analysis methodology based on adaptive hierarchical refinement of NURBS, immersed boundary methods, and T-spline CAD surfaces. *Comput Methods Appl Mech Eng* 249:116–150. <https://doi.org/10.1016/j.cma.2012.03.017>
- Shamanskiy A, Gfrerer MH, Hinz J, Simeon B (2020) Isogeometric parametrization inspired by large elastic deformation. *Comput Methods Appl Mech Eng* 363:112920. <https://doi.org/10.1016/j.cma.2020.112920>
- Stein K, Tezduyar TE, Benney R (2003) Mesh moving techniques for fluid-structure interactions with large displacements. *J Appl Mech* 70(1):58–63. <https://doi.org/10.1115/1.1530635>
- Suito H, Takizawa K, Huynh VQH, Sze D, Ueda T (2014) FSI analysis of the blood flow and geometrical characteristics in the thoracic aorta. *Comput Mech* 54(4):1035–1045. <https://doi.org/10.1007/s00466-014-1017-1>
- Takizawa K, Tezduyar TE (2012) Space-time fluid-structure interaction methods. *Math Models Methods Appl Sci* 22:1230001. <https://doi.org/10.1142/S0218202512300013>
- Takizawa K, Tezduyar TE, Avsar R (2020) A low-distortion mesh moving method based on fiber-reinforced hyperelasticity and optimized zero-stress state. *Comput Mech*. <https://doi.org/10.1007/s00466-020-01835-z>
- Takizawa K, Tezduyar TE, Boben J, Kostov N, Boswell C, Buscher A (2013) Fluid-structure interaction modeling of clusters of space-craft parachutes with modified geometric porosity. *Comput Mech* 52(6):1351–1364. <https://doi.org/10.1007/s00466-013-0880-5>
- Terahara T, Takizawa K, Tezduyar TE, Tsushima A, Shiozaki K (2020) Ventricle-valve-aorta flow analysis with the space-time isogeometric discretization and topology change. *Comput Mech*. <https://doi.org/10.1007/s00466-020-01822-4>
- Tezduyar T (2001) Finite element interface-tracking and interface-capturing techniques for flows with moving boundaries and interfaces. *ASME Publ HTD* 369:105–114
- Tezduyar TE, Aliabadi S, Behr M, Johnson AA, Mittal S (1993) Parallel finite-element computation of 3D flows. *Computer* 26(10):27–36. <https://doi.org/10.1109/2.237441>

31. Tezduyar TE, Behr M, Mittal S, Johnson AA (1992) Computation of unsteady incompressible flows with the stabilized finite element methods: space-time formulations, iterative strategies and massively parallel implementations. In: ASME pressure vessels and piping conference 1992. American Society of Mechanical Engineers Digital Collection, pp 7–24
32. Tezduyar TE, Sathe S (2007) Modelling of fluid-structure interactions with the space-time finite elements: solution techniques. *Int J Numer Methods Fluids* 54(6–8):855–900. <https://doi.org/10.1002/fluid.1430>
33. Turek S, Hron J (2006) Proposal for numerical benchmarking of fluid-structure interaction between an elastic object and laminar incompressible flow. In: *Fluid-structure interaction*. Springer, pp 371–385. <https://doi.org/10.1007/3-540-34596-5-15>
34. Wriggers P (2008) *Nonlinear finite element methods*. Springer, Berlin. <https://doi.org/10.1007/978-3-540-71001-1>
35. Wu Y, Cai XC (2014) A fully implicit domain decomposition based ALE framework for three-dimensional fluid-structure interaction with application in blood flow computation. *J Comput Phys* 258:524–537. <https://doi.org/10.1016/j.jcp.2013.10.046>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.