

# MESH MOVING METHODS IN FLOW COMPUTATIONS WITH THE SPACE–TIME AND ARBITRARY LAGRANGIAN–EULERIAN METHODS

Kenji TAKIZAWA<sup>1</sup>, Yuri BAZILEVS<sup>2</sup>, Tayfun E. TEZDUYAR<sup>1,3,\*</sup>

<sup>1</sup>Waseda University, Tokyo, Japan

<sup>2</sup>Brown University, Providence, Rhode Island, USA

<sup>3</sup>Rice University, Houston, Texas, USA

\*Corresponding Author: Tayfun E. TEZDUYAR (Email: tezdubar@tafsm.org)

(Received: 1-Apr-2022; accepted: 18-Apr-2022; published: 30-Jun-2022)

DOI: <http://dx.doi.org/10.55579/jaec.202262.377>

**Abstract.** A good mesh moving method is an important part of flow computations with moving-mesh methods like the space–time (ST) and Arbitrary Lagrangian–Eulerian (ALE) methods. With a good mesh moving method, we can decrease the remeshing frequency even when the fluid–solid and fluid–fluid interfaces undergo large displacements, decrease the element distortion in parts of the flow domain where we care about the solution accuracy more, and maintain the quality of the boundary layer meshes near the fluid–solid interfaces as the mesh moves to follow those interfaces. Since 1990, quite a few good mesh moving methods have been developed for use with the ST computational methods, from the mesh–Jacobian-based stiffening to a mesh moving method based on fiber-reinforced hyperelasticity to a linear-elasticity mesh moving method with no cycle-to-cycle accumulated distortion. These methods have been used in computation of many complex flow problems in the categories of fluid–particle interaction, fluid–structure interaction, and more generally, moving boundaries and interfaces. The computations were with both the ST and ALE methods. We provide an overview of these methods and present examples of the computations performed.

## Keywords

*Flow computation, fluid–structure interaction, mesh moving method, space–time method, ALE method, mesh–Jacobian-based stiffening, mesh moving based on fiber-reinforced hyperelasticity.*

## 1. Introduction

Many real-world fluid mechanics problems involve fluid–structure interaction (FSI), fluid–particle interaction, free-surface and two fluid–flows, and flows with moving mechanical components. These categories of problems can be seen as subcategories of flows with moving boundaries and interfaces (MBI) [1]–[3]. In FSI and MBI computations with moving-mesh methods [1]–[3], as the flow domain changes its shape, the mesh moves to facilitate that shape change, to follow the moving interfaces, and to control the mesh resolution near the moving solid surfaces. The moving-mesh methods are also called interface-tracking methods [1]–[3] since the mesh is moving to “track” the interface, as opposed to just “capturing” it over a nonmoving mesh. More discussion on the moving-mesh (interface-tracking)

and nonmoving-mesh (interface-capturing) methods, including the Mixed Interface-Tracking/Interface-Capturing Technique [4] and Fluid–Solid Interface-Tracking/Interface-Capturing Technique (FSITICT) [5] can be found in [1]–[3], [6].

The Deforming-Spatial-Domain/Stabilized Space–Time (DSD/SST) method [7]–[9], introduced in 1990, is a moving-mesh method. It is also called the “ST-SUPS” [2] because of its stabilization components SUPG and PSPG, and it is widely known that those abbreviations stand for the Streamline-Upwind/Petrov-Galerkin [10] and Pressure-Stabilizing/Petrov-Galerkin [7]. The Arbitrary Lagrangian–Eulerian (ALE) method is an older moving-mesh method, with its finite element version going at least as far back as 1981 [11]. It is far more commonly used than the ST computational methods. One of the earliest computations with the ALE-SUPS method was for parachute FSI [12]. The ALE-VMS method [2], [13]–[18] is the variational multiscale version of the ALE, with the VMS components coming from the residual-based VMS (RBVMS) method [19]–[22]. The VMS components of the ST-VMS method [23]–[25] also come from the RBVMS.

Over the years since their inception, the ST-SUPS, ALE-SUPS, RBVMS, ALE-VMS, and ST-VMS have become a powerful set of computational methods and have been applied to some of the most challenging classes of flow problems. The classes of problems computed with the ALE-SUPS, RBVMS, and ALE-VMS include parachutes [12], wind turbines [26]–[47], medical applications [13], [48]–[61], free-surface flows [62]–[66], aircraft applications [67, 68], turbomachinery [69]–[75], marine applications [76]–[78], bridges [79]–[83], stratified flows [84, 85], hypersonic flows [86], two-phase flows [87]–[93], additive manufacturing [94], immersogeometric FSI and flow analysis [95]–[99], and mixed ALE-VMS/Immersogeometric computations [57]–[59], [100]–[108] in the framework of the FSITICT. A comprehensive summary of the classes of flow problems computed with ST-SUPS and ST-VMS prior to July 2018 was provided in [109]. The classes of problems computed before and after July 2018 include parachutes [2, 45, 44], [110]–[120], wind turbines [2, 26, 33],

[40]–[43], [121]–[130], flapping-wing aerodynamics [2, 124, 125], [131]–[137], medical applications [60, 61, 124, 135], [138]–[152], spacecraft [112, 153], ground vehicles and tires [25, 45, 44, 144], [154]–[160], disk brakes [161], turbomachinery [42, 43], [162]–[169], fluid films [157, 160, 170], U-ducts [171], and Taylor–Couette flow [172, 173].

A good mesh moving method is an important part of flow computations with moving-mesh methods like the ST-SUPS, ALE-SUPS, ALE-VMS, and ST-VMS. The mesh moving method serves as a component of the mesh update method, with the other component being remeshing. Most of the time remeshing involves generating a new set of nodes, which was called “renoding” in [9, 174], but can also just consist of generating a new set of elements “reconnecting” [9, 174] the existing nodes. Remeshing can be full, or partial, as done in [175]. With a good mesh moving method, we can decrease the remeshing frequency even when the fluid–solid and fluid–fluid interfaces undergo large displacements, decrease the element distortion in parts of the flow domain where we care about the solution accuracy more, and maintain the quality of the boundary layer meshes near the fluid–solid interfaces as the mesh moves to follow those interfaces. A mesh moving method can be a special-purpose or general-purpose one. Normally, a special-purpose mesh moving method can be used for meshes generated with special-purpose mesh generation methods. A general-purpose mesh moving method, which requires solution of a set of equations governing the motion of the mesh, can be used for meshes generated with both special-purpose and general-purpose mesh generation methods. Since 1990 [7], quite a few good mesh moving methods have been developed for use with the ST-SUPS and ST-VMS, from the mesh-Jacobian-based stiffening [172, 176, 177] to a mesh moving method based on fiber-reinforced hyperelasticity [6] to a linear-elasticity-based mesh moving method with no cycle-to-cycle accumulated distortion [150]. These methods have been used in computation of many complex flow problems in the categories of fluid–particle interaction, FSI, and more generally, MBI. The computations were with both the ST and ALE methods, as reported in many of the publications cited in the

previous paragraph. In this article, we provide an overview of these mesh moving methods and present examples of the computations performed. The overview of the methods is in Sections 2-8, and the examples are in Sections 9-11. The concluding remarks are given in Section 12.

## 2. Special-purpose mesh moving methods

The first ST-SUPS computations reported in [7] were performed with special-purpose mesh moving methods. They included a “pulsating drop”, driven by surface tension, large-amplitude sloshing in a tank, and a cylinder drifting in a shear flow, translating in two directions. For the drifting cylinder, for example, the square-shaped inner mesh around the cylinder moves “glued” to the cylinder, with no deformation in the inner mesh. The mesh outside the square region absorbs all the mesh deformation. As another example, the ST-SUPS computation of a “freely falling NACA 0012 airfoil”, first reported in [176], was performed again with a square-shaped mesh translating with the airfoil in two directions. In that case, however, because the airfoil had also a pitching motion, the inner mesh around the airfoil was of circular shape, rotating with the airfoil, with no deformation inside that mesh. A ring-shaped mesh, with layers of elements, surrounded the inner mesh and absorbed all the rotation-driven deformation. The rest of the mesh inside the square-shaped region remained undeformed, only translating with the airfoil in two directions. Over the years since 1990, more special-purpose mesh moving methods were developed for ST-SUPS and ST-VMS computations, including those in 3D and with complex geometries. Examples of more recent computations are heart valve flow [149, 150] and tire aerodynamics with near-actual tire geometry, road contact, and tire deformation [156, 157].

## 3. Linear-elasticity mesh moving method and mesh-Jacobian-based stiffening

In the linear-elasticity mesh moving method introduced in [172, 176, 177], in computing the mesh motion from time level  $t_n$  to  $t_{n+1}$ , the nodal displacements are governed by the equations of linear elasticity. At the boundaries and interfaces, the required condition is for the normal component of the mesh velocity matching the normal component of the fluid velocity. However, typically, especially at curved boundaries and interfaces, all components of the mesh velocity matching all components of the fluid velocity is simpler to implement. It was also proposed in [172, 176, 177] that, in solving the elasticity equations with the finite element method, the Jacobian of the transformation from the element parent domain to physical domain be dropped. The purpose was, of course, to make the smaller elements, typically encountered near the interfaces and in regions where we care about the flow solution accuracy more, stiffer than the larger ones. That is how the mesh-Jacobian-based stiffening (MJBS) method was born in 1992. It did not have a name when it was born. The first time it was given a name was in [178]; it was called “Jacobian-based stiffening”. Also in [178], the method was generalized by introducing a “stiffening power”, denoted by the symbol  $\chi$ . Defined as being always positive,  $\chi$  governs to what degree the smaller elements become stiffer than the larger ones. When  $\chi = 1$ , the method is, of course, the same as the original method, and typically  $\chi = 1$ . Realizing that the method name should be clearer on which Jacobian is implied, the method was renamed “mesh-Jacobian-based stiffening”. in [150].

Since its inception in 1992, the MJBS has been used in computation of many complex flow problems in the categories of fluid–particle interaction, FSI, and more generally, MBI. The computations were with both the ST and ALE methods, as reported in many of the publications cited in the third paragraph of Section 1.

## 4. Solid-extension mesh moving technique

In computing flow problems with fluid–solid interfaces, if the solid part is not deforming, maintaining the quality of the boundary layer meshes near the interfaces can be achieved by simply having some layers of elements around the object moving glued to the object. This can be done by using a special-purpose mesh moving method or by activating the elasticity-based mesh moving method outside the layers of elements around the object. The inner-boundary conditions for the elasticity equations come from the outer surfaces of the layers of elements. Either way, we get full control of the mesh resolution in these layers throughout the computation. The earliest examples of ST-SUPS computations with layers of elements moving glued to solid objects can be found in [7] in the context of special-purpose mesh moving methods, and in [172, 177] in combination with an elasticity-based mesh moving method activated outside the layers.

Moving the layers of elements glued to the solid object is not applicable in an FSI computation where the solid part is deforming. The Solid-Extension Mesh Moving Technique (SEMMT) was introduced in 2001 [179] to maintain the quality of the boundary layer meshes near the fluid–solid interfaces also in FSI computations. In the SEMMT [2], [179]–[182], in solving the elasticity equations, the layered inner meshes around the solid surfaces are dealt with like extensions of the structure mesh. The SEMMT has two versions: “SEMMT – Single Domain” (SEMMT-SD) and “SEMMT – Multiple Domain” (SEMMT-MD). In the SEMMT-SD, the elements in the inner mesh are assigned higher stiffness, and the elasticity equations governing the motion of the inner and outer meshes are solved together, over a single domain. In the SEMMT-MD, the equations governing the motion of the inner and outer meshes are solved separately, over multiple domains. In Step 1, the motion of the inner mesh is solved for, with zero-stress boundary condition on the outer surfaces. In Step 2, the motion of the outer mesh is solved for, with displacement boundary condition on the inner surfaces. Test computations

with both versions were presented in [2], [181]–[183]. The tests included simple but revealing 2D deformation cases in [2], [181]–[183] and a 2D FSI model problem in [181, 182]. The SEMMT was also used, in combination with the MJBS, in computational flow analysis of wind turbine blade strip subjected to environmental erosion in [184] and particle-laden-airflow erosion in [185].

## 5. ST NURBS mesh update method

The isogeometric analysis (IGA), with the convincing results obtained by using IGA basis functions in space [13, 48, 186, 187], offered a new direction in computational flow analysis. It offered higher accuracy in representing the geometry and in the flow solution. The ST-IGA [23], in addition to offering an ST framework in using IGA basis functions in space [118, 145, 146, 163] with even higher accuracy in the flow solution, offered the option of using IGA basis functions also in time [23, 24, 131, 132]. That enabled higher accuracy in representing the path or trajectory and creation of new methods that are possible in an ST framework (see Section 1 in [159] for the most recent summary of those methods). The ST/NURBS Mesh Update Method (STNMUM), introduced in [131, 132] and named in [123], is one of those methods.

With the ST-IGA and IGA basis functions in time, motion of the solid surfaces can be represented more accurately, and the representation of the corresponding mesh motion would be consistent with that. These features of the ST-IGA were pointed out in [23, 24] and shown at work in [131, 132]. Also with the ST-IGA and IGA basis functions in time, we can have better efficiency in time representation of the mesh motion and deformation and in remeshing. These features made the STNMUM what it is. The classes of problems with spinning surfaces, for example, can be handled very effectively with the STNMUM. With quadratic NURBS basis function in time representing the spinning motion, we can represent the circular paths exactly, provided that we have sufficient number of patches for the full rotation. It is of course also desirable

to be able to specify a constant angular velocity for speeds invariant along the circular paths. That is achieved by the secondary mapping introduced in [23, 24, 131]

The STNMUM has been used in a large number of 3D computations with complex geometries. The classes of problems computed include flapping-wing aerodynamics [2, 124, 125], [131]-[137], spacecraft [112], wind turbines [33], [40]-[43], [123]-[127], ground vehicles and tires [25, 44, 45, 144], [154]-[160], disk brakes [161], and turbomachinery [42, 43], [162]-[169].

## 6. Nonlinear-elasticity mesh moving method

In the linear-elasticity mesh moving methods, a common practice in computing the mesh motion from  $t_n$  to  $t_{n+1}$  is to compute the nodal displacements from the configuration at  $t_n$ . Although many flow computations were successfully performed this way, it is known that (see, for example, [6, 150]), the method is path-dependent. Because of that, in flow computations where we expect cyclic or near-cyclic results, the method leads to non-cyclic results and cycle-to-cycle accumulated mesh distortion. It is hard to reverse the process if the nodes accumulate in some region of the mesh. Moving the mesh as governed by the nonlinear-elasticity equations of large-deformation mechanics is path-independent. The nonlinear-elasticity mesh moving method was used in a good number of ST-SUPS and ST-VMS computations of flow problems with FSI and other MBI (see, for example, [25, 113, 125, 135, 142]). It was commented in [113] that one would have a good selection of constitutive models and can define the zero-stress state (ZSS) of the large-deformation mechanics locally in arbitrary orientations. In the computations reported in [113], the constitutive model was St. Venant–Kirchhoff, in [25, 125, 135, 142], neo-Hookean. The MJBS can of course be used also in the nonlinear-elasticity mesh moving method, as was the case in [25, 142]. In the remainder of this section, we will cover two closely related meth-

ods: element-based mesh relaxation (EBMR) and locally-defined ZSS.

### 6.1. EBMR

The EBMR, introduced in [113], restores the mesh integrity lost during the mesh motion, but does that without remeshing. The loss of mesh integrity in regions that we care more about does not happen so often because of the mesh moving methods discussed so far, but could happen in computations with high complexity. The FSI computations reported in [111, 113, 114, 115, 116, 124] for spacecraft parachute clusters had that type of complexity. It was proposed in [113] that, when we see a loss of mesh integrity, with the EBMR, the mesh is relaxed, without changing the mesh at the fluid–structure interface, and the mesh integrity is restored to some extent. This is, as commented in [113], is less disruptive and less time-consuming than remeshing. The EBMR does not change the number of elements or nodes. It only moves slightly some of the nodes and thus improves the element quality in parts of the mesh. The motion of the nodes is determined from the nonlinear-elasticity equations of large-deformation mechanics and an element-based ZSS (EBZSS). The EBZSS is essentially a shape generated for each element, and by design, the undeformed shape consists of “target elements” and is the shape we want to reach by solving the nonlinear-elasticity equations. The options for constructing the target element shapes can be found in [113]. The FSI computations reported in [113] for spacecraft parachute clusters were performed with the EBMR.

### 6.2. Locally-defined ZSS

The locally-defined ZSS was created as an arterial ZSS estimation [60], [188]-[193]. It was formulated, at the beginning, as the EBZSS in the context of finite element discretization [188, 189], then as the EBZSS in the context of isogeometric discretization [190, 191], and then as the integration-point-based ZSS (IPBZSS) in the context of isogeometric discretization [192, 193]. In the EBZSS, for each element,

the ZSS is defined by a set of positions. When nodes (or control points) from different elements map to the same node in the mesh, their ZSS-defining positions do not have to be the same. In the reference configuration, however, all elements are connected by nodes, with the displacements measured from that configuration. Formulating the structural mechanics problem in this fashion was called “element-based total Lagrangian” (EBTL) method in [188]. The EBTL is a key part of the EBMR [113]. In the IPBZSS, how we define the EBZSS is extended to its integration-point counterpart, with the ZSS represented in terms of the metric tensor. The IPBZSS has more parameters than the EBZSS. Consequently, while the conversion from the EBZSS representation to IPBZSS representation is straightforward and will be exact, the conversion, in general, will not be exact. Formulating the structural mechanics problem in this fashion was called “integration-point-based total Lagrangian” (IPBTL) method in [6].

## 7. Mesh relaxation and mesh moving based on fiber-reinforced hyperelasticity and optimized ZSS

Methods for mesh relaxation and mesh moving based on fiber-reinforced hyperelasticity and optimized ZSS were introduced in [6]. The methods were introduced targeting isogeometric discretization, but they are, of course, also applicable to finite element discretization. To reduce distortion during the mesh deformation, the element is stiffened in multiple directions by placing reinforcement fibers in those directions. The ZSS is optimized by seeking, with mesh relaxation, orthogonality of the parametric directions and making the ZSS time-dependent as needed. With the mesh relaxation, after the initial creation of a mesh, we improve its quality and have an equilibrium state with the optimized ZSS, boundary conditions, and constitutive law. The NURBS mesh used in the computational flow analysis reported in [127] for a tsunami-shelter

vertical-axis wind turbine was obtained with the mesh relaxation method.

## 8. Back-cycle-based mesh moving method

As mentioned in Section 6, in computing the mesh motion from  $t_n$  to  $t_{n+1}$  with a linear-elasticity mesh moving method, if the nodal displacements are computed from the configuration at  $t_n$ , the mesh moving method is path-dependent. As also mentioned in that section, in flow computations where we expect cyclic or near-cyclic results, a path-dependent mesh moving method leads to non-cyclic results and cycle-to-cycle accumulated mesh distortion. The methods described in that section for moving the mesh as governed by the nonlinear-elasticity equations of large-deformation mechanics give us the path-independence we want. Nevertheless, linear-elasticity mesh moving methods where the displacements are not computed from the configuration at  $t_0$ , and other methods where the displacements are not computed from the configuration at  $t_0$ , are still widely used.

The back-cycle-based mesh moving (BCBMM) method, introduced in [150] as a linear-elasticity mesh moving method, has no cycle-to-cycle accumulated distortion. In the BCBMM, in computing the mesh motion from  $t_n$  and  $t_{n+1}$  with the linear-elasticity equations, in any cycle, the nodal displacements are computed from the configurations in the first cycle. It was pointed out in [150] that in later cycles, as needed, the configurations the nodal displacements are computed from can be changed from the first cycle to a higher cycle and the need may arise if, prior to the solution becoming cyclic or near-cyclic, there are significant differences between the solutions at different cycles.

The half-cycle-based mesh moving (HCBMM) method was introduced in [194] as a special-case version of the BCBMM. It is for special cases where the boundary or interface motion in the second half of the cycle consists of reversing the steps in the first half and we prefer the mesh motion to have the same reversal pattern. This

is accomplished by modifying how the displacements are computed in the second half of the first cycle. In the second half, at a given time level, the displacements are computed from the configuration at the corresponding time level in the first half. In the second and higher cycles, the displacements are computed as in the BCBMM.

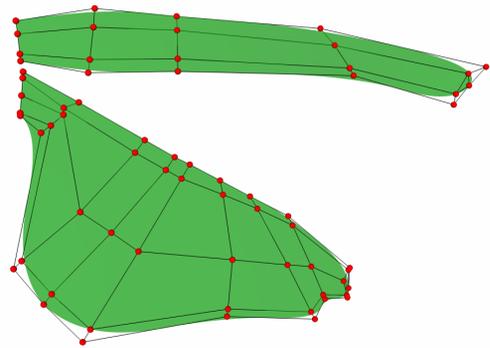
Detailed 2D and 3D test computations with finite element meshes were presented in [194], using the mesh motion associated with wing pitching as the test case, to demonstrate how the BCBMM and HCBMM function as mesh moving methods with no cycle-to-cycle accumulated mesh distortion. The BCBMM and HCBMM were introduced and tested in the context of a linear-elasticity mesh moving method. However, it is clear that they are applicable also to other mesh moving methods, such as those based on the Poisson's equation, where the displacements are not computed from the configuration at  $t_0$ .

## 9. Flapping aerodynamics of actual locust wings

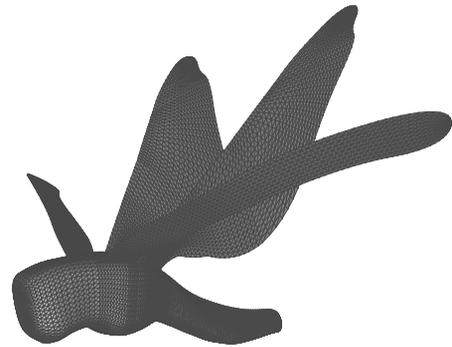
This is an example of the classes of computations performed with the STNMUM. The details of this class of computations can be found in [2], [131]-[134]. The motion and deformation of the wings come from video recordings a locust while in flight in a wind tunnel. The video data was obtained by our research collaborators at Baylor College of Medicine in Houston. Figure 1 shows the wing geometries, as represented by quadratic NURBS.

The results presented here come from a finite element computation with tetrahedral elements. The triangular meshes over the wing and body surfaces were generated from the NURBS-represented surface geometries. The wings have a finite thickness, 1% of the forewing root chord. Figure 2 shows the surface meshes.

The volume mesh has one layer of manually-generated refined mesh near the wing surfaces. The locust is placed in a cylindrical region to have increased refinement around the locust. The region between the forewing and hindwing



**Fig. 1:** Flapping aerodynamics of actual locust wings. Forewing and hindwing geometries as represented by NURBS and the control mesh. The forewing is made of a single patch, and the hindwing two patches.

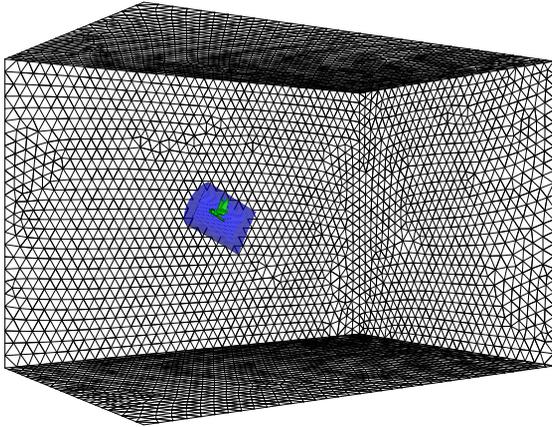


**Fig. 2:** Flapping aerodynamics of actual locust wings. Surface meshes.

has even more refinement. The cylindrical region is tilted to the body angle of the locust during the flight. The meshes inside and outside the cylindrical region were both generated by an automatic mesh generator. Figure 3 shows the boundaries of the volume mesh and cylindrical refinement region.

The mesh deformation and remeshing, when needed, occur only inside the cylindrical region. The mesh motion in each flapping cycle is represented by four cubic NURBS patches in time. This can be seen in Fig. 4.

The patches have different number of control points in time, and the volume mesh has different number of nodes and elements in each patch. The numbers are shown in Tab. 1.



**Fig. 3:** Flapping aerodynamics of actual locust wings. Boundaries of the volume mesh and cylindrical refinement region.

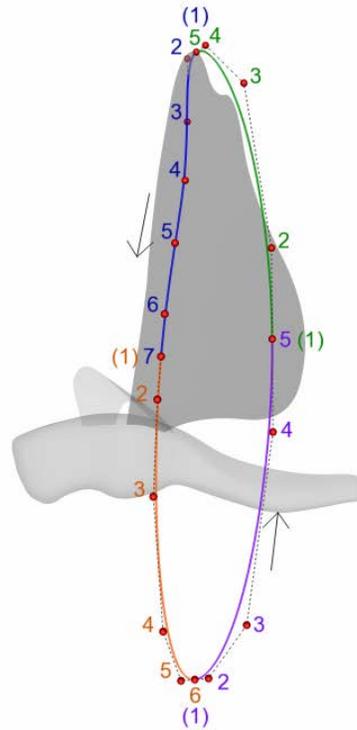
**Tab. 1:** Flapping aerodynamics of actual locust wings. Number of temporal control points and knot spans in each NURBS patch. Number of nodes and elements in the volume mesh in each patch and the control point where the mesh is generated.

Temporal Patch	Fig. 4 Color	Control Points	Knot Spans
1	Blue	7	4
2	Orange	6	3
3	Purple	5	2
4	Green	5	2

Temporal Patch	Meshing Point	Nodes	Elements
1	4	355,229	2,115,916
2	3	389,981	2,323,144
3	2	346,993	2,066,797
4	3	380,034	2,264,324

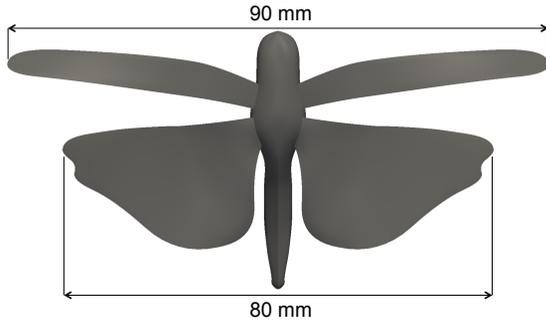
In each patch, the volume mesh is generated at an interior control point and moved to the other control points with the linear-elasticity mesh moving method and MJBS. This means that the remeshing occurs at the patch boundaries. The control points where the volume mesh is generated are shown in the caption of Fig. 4 and in Tab. 1. With all those control points, the motion of the volume mesh now has the cubic NURBS representation in time. The representation is built only once, in the first flapping cycle, with the same representation used in the subsequent cycles. The volume mesh is generated only



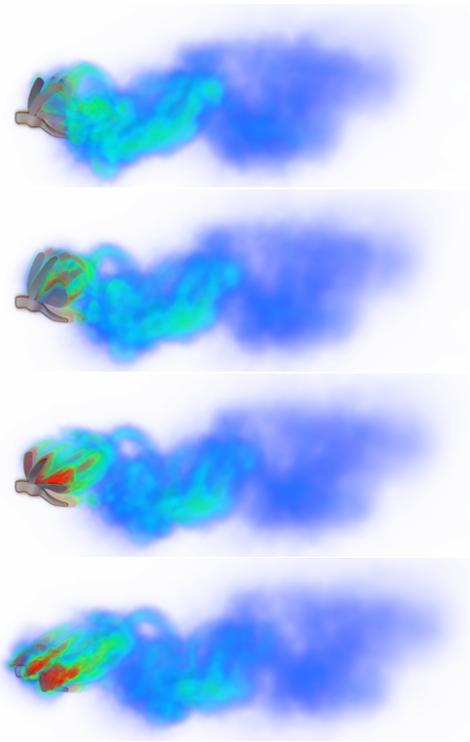
**Fig. 4:** Flapping aerodynamics of actual locust wings. Hindwing tip trajectory represented by four cubic NURBS patches in time. The figure also shows the temporal control points for each patch. The numbers in parentheses indicate the control points that are at the end of one patch and start of the next. The volume mesh is generated at the control points 4, 3, 2, and 3 of the blue, orange, purple, and green patches. Then, in each patch, the mesh is moved to the other control points with the linear-elasticity mesh moving method and MJBS. This means that the remeshing occurs at the patch boundaries. With all those control points, the motion of the volume mesh now has the cubic NURBS representation in time. The representation is built only once, in the first flapping cycle, with the same representation used in the subsequent cycles. The volume mesh is generated only four times for the entire flow computation with multiple flapping cycles.

four times for the entire flow computation with multiple flapping cycles.

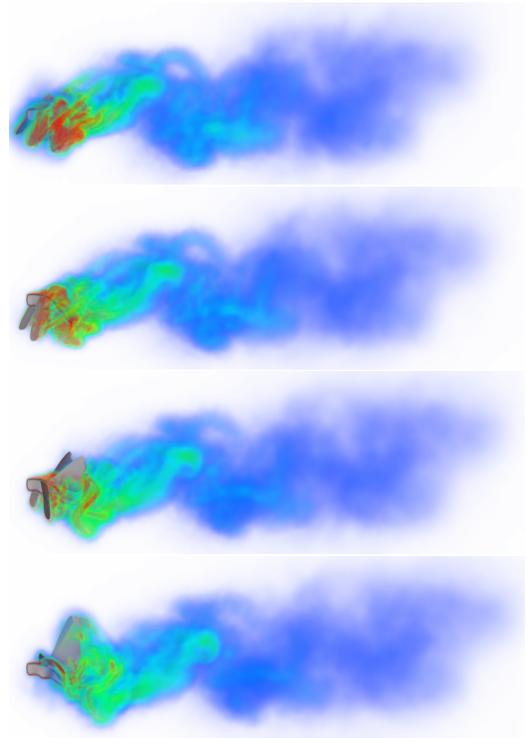
Figure 5 shows length scales. The flight speed is 2.4 m/s. The flapping cycle is  $T = 0.047$  s. Each knot span has 25 time steps, resulting in 275 time steps per flapping cycle. There are 4 nonlinear iterations per time step. The compu-



**Fig. 5:** Flapping aerodynamics of actual locust wings. Boundaries of the volume mesh and cylindrical refinement region.



**Fig. 6:** Flapping aerodynamics of actual locust wings. Vorticity magnitude for the first four of eight equally-spaced instants during the second flapping cycle.



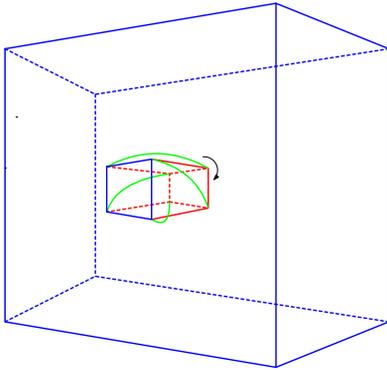
**Fig. 7:** Flapping aerodynamics of actual locust wings. Vorticity magnitude for the last four of eight equally-spaced instants during the second flapping cycle.

## 10. A square-section block undergoing torsion

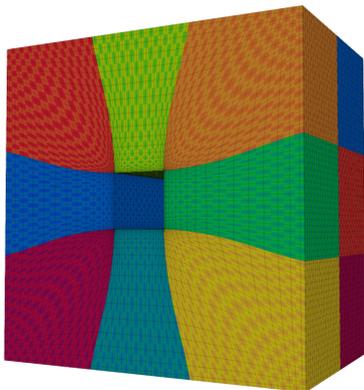
This is an example of mesh relaxation and mesh moving based on fiber-reinforced hyperelasticity and optimized ZSS. For comparison purposes, we also show the results from mesh moving based on the MJBS and mesh relaxation and mesh moving based on neo-Hookean hyperelasticity. The details of the mesh relaxation and mesh moving computations can be found in [6]. Table 2 shows the material properties for the MJBS, and Tab. 3 shows the material properties for the neo-Hookean (“NH”) and fiber-reinforced (“FR”) hyperelasticity. We are using nondimensional values. In the case of Tab. 3, the Poisson’s ratio can be written as  $\nu = \frac{3\kappa_B - 2\mu}{2(3\kappa_B + \mu)}$ .

Figure 8 shows the box-shaped domain used in the mesh relaxation and mesh moving computations, with the square-section block in its initial and twisted configurations. The deformation of

tation was performed with the ST-SUPS in the first two nonlinear iterations, and the ST-VMS in the last two. Figures. 6 and 7 show the vorticity magnitude at equally-spaced instants during the second flapping cycle.



**Fig. 8:** A square-section block undergoing torsion. Box-shaped domain used in the mesh relaxation and mesh moving computations, with the square-section block in its initial (red) and twisted (green) configurations. One end of the block is fixed centrally on the front plane of the box. The other end is at the midpoint between the front and back planes and undergoes a  $90^\circ$  rotation about its own center.

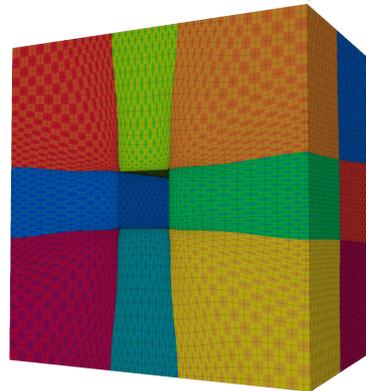
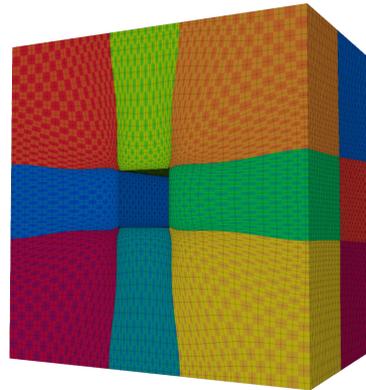


**Fig. 9:** A square-section block undergoing torsion. The initial mesh  $\mathbf{x}_0$ . The checkerboard pattern is for differentiating between the NURBS elements. The colors are for differentiating between the NURBS patches.

**Tab. 2:** A square-section block undergoing torsion. Material properties and  $\chi$  for the MJBS. For the definitions of the symbols, see [6].

	$\nu$	$E$	$\chi$
MJBS1	0.3	1	2
MJBS2	0.3	1	1

the block, as given in Fig. 8, defines the specified boundary positions in computing the positions of the rest of the points in the box-shaped domain. We use different material properties for

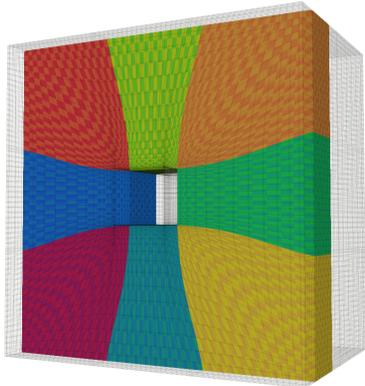


**Fig. 10:** A square-section block undergoing torsion. Mesh after the relaxation. Neo-Hookean (top) and fiber-reinforced (bottom) hyperelasticity. The checkerboard pattern is for differentiating between the NURBS elements. The colors are for differentiating between the NURBS patches.

**Tab. 3:** A square-section block undergoing torsion. Material properties for the neo-Hookean (“NH”) and fiber-reinforced (“FR”) hyperelasticity. For the definitions of the symbols, see [6].

	$\kappa_B$	$\beta_B$	$\mu$	$C_1$	$C_2$
NH1	$10^{-1}$	4	$2 \times 10^2$	0	-
NH2	$10^{-3}$	0	1	0	-
FR1	$10^{-1}$	4	$2 \times 10^2$	$10^2$	$10^1$
FR2	$10^{-3}$	0	1	0.5	1.0

the first two layers of elements around the block and the rest of the elements, which we define as the “inner” and “outer” elements. In the test computations with the MJBS, NH, and FR, we use the material properties of MJBS1, NH1, and FR1 for the inner elements, and MJBS2, NH2, and FR2 for the outer elements.



**Fig. 11:** A square-section block undergoing torsion. The portion of  $\mathbf{x}_0$  used in visualizing the deformed-mesh configurations. The checkerboard pattern is for differentiating between the NURBS elements. The colors are for differentiating between the NURBS patches.

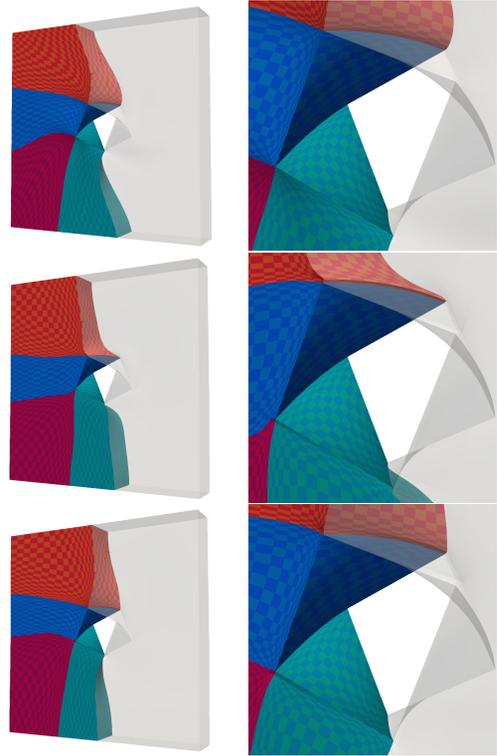
**Tab. 4:** A square-section block undergoing torsion. Mesh orthogonality measures for the initial and post-relaxation meshes, with the neo-Hookean (“NH”) and fiber-reinforced (“FR”) hyperelasticity. Deviation of the mesh-line angles from  $\pi/2$ , represented in terms of the  $L_2$  and max norms.

Method	$L_2$ norm	Max norm
$\mathbf{x}_0$	4.8°	43.1°
NH	0.4°	75.8°
FR	0.1°	35.6°

**Tab. 5:** A square-section block undergoing torsion. Mesh orthogonality measures for the initial and post-relaxation meshes, with the neo-Hookean (“NH”) and fiber-reinforced (“FR”) hyperelasticity. Deviation of the mesh-line angles from  $\pi/2$ , represented in terms of the  $L_2$  and max norms.

Method	$L_2$ norm	Max norm
MJBS	8.4°	78.1°
NH	1.2°	88.5°
FR	1.1°	64.2°

We use a quadratic NURBS mesh with 136,000 elements and 167,728 control points. The initial mesh,  $\mathbf{x}_0$ , is shown in Fig. 9. Figure 10 shows the mesh after the relaxation. No mesh relaxation is used with the MJBS. Table 4 shows the mesh orthogonality measures for the initial and post-relaxation meshes. The torsion takes place in 500 steps. In visualizing the deformed-mesh configurations, we use the portion of  $\mathbf{x}_0$  shown in Fig. 11. Of that portion, be-



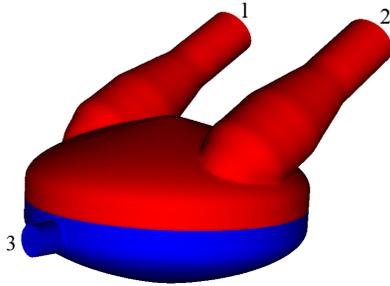
**Fig. 12:** A square-section block undergoing torsion. Post-torsion meshes. MJBS (top) and neo-Hookean (middle) and fiber-reinforced (bottom) hyperelasticity. The right frame is the zoomed view. The checkerboard pattern is for differentiating between the NURBS elements. The colors are for differentiating between the NURBS patches.

cause of the symmetry, we visualize only half of the NURBS patches. Figure 12 shows the post-torsion meshes. Table 5 shows the mesh orthogonality measures for the post-torsion meshes.

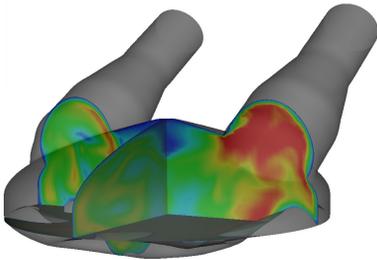
## 11. Ventricular assist device

This is an example of the classes of computations performed with the MJBS. It is FSI simulation of a ventricular assist device, coming from [54]. The computational model has a width of 7.7 cm, and an apex-to-apex height of 4.5 cm. The main blood chamber has two arms, with one assigned exclusively as the inlet, and the other as the outlet, each with a diameter of 1.5 cm. The air cham-

ber has one small inlet/outlet port with a diameter of 0.8 cm. These are labeled in Fig. 13. The device is driven by the prescribed flow in and out of the air chamber. The fill period is 0.45 s and the ejection period is 0.3 s, with the stroke volume of 73 mL.



**Fig. 13:** Ventricular assist device. The blood domain is in (red), and the air domain is in (blue). The inlet and outlet faces of the blood chamber are labeled 1 and 2, respectively. The air-side inlet/outlet face is labeled 3.



**Fig. 14:** Ventricular assist device. Flow speed in the deformed blood chamber configuration at  $t = 0.15$  s.

The ALE-VMS method employing linear tetrahedral elements was used for the flow simulation. At the start of the simulation, the number of elements is 238,322 in the air chamber and 497,160 in the blood chamber. Isogeometric Kirchhoff–Love shell [195]–[198] was used in modeling the thin membrane, with a thickness of 0.25 mm, separating the blood and air chambers. The shell was discretized using 1,024  $C^1$ -continuous quadratic NURBS elements. The simulations were run for two time cycles of 0.75 s each, with 750 time steps per cycle.

Due to very large deformations of the membrane, an occasional remeshing was necessary

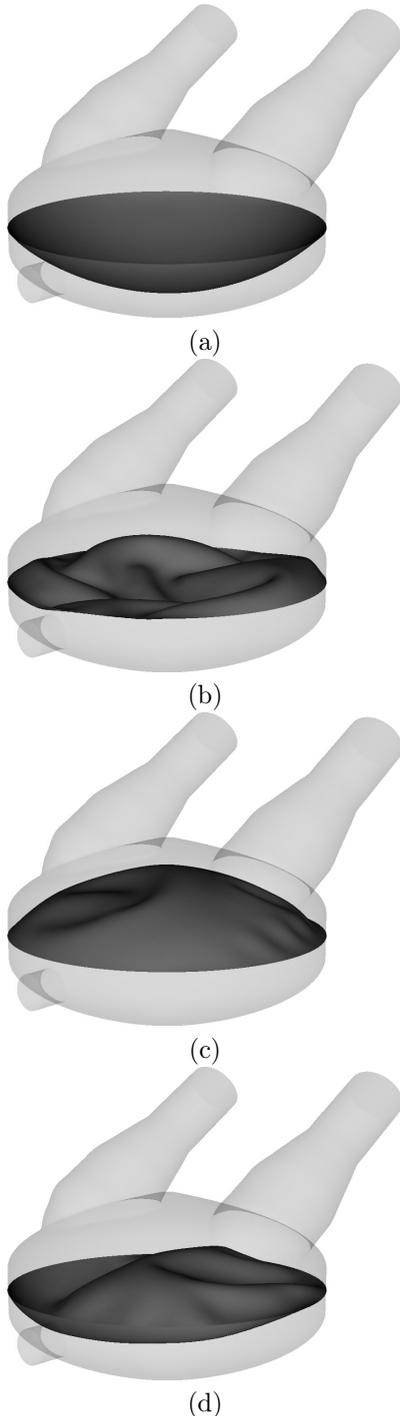


**Fig. 15:** Ventricular assist device. Top view of the membrane deformed configuration at  $t = 0.15$  s. Note the smoothness of the wrinkles.

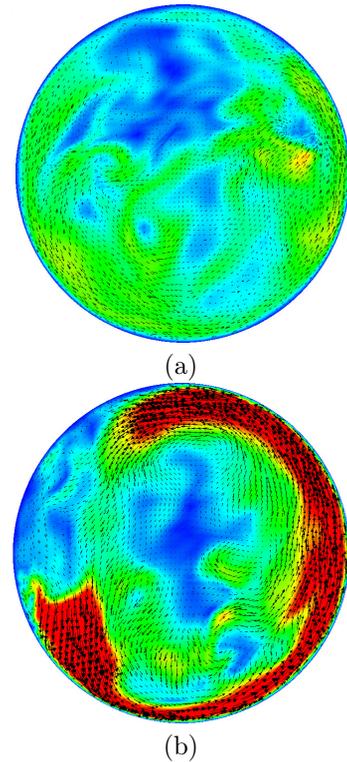
to keep the fluid mesh quality under control. Remeshing was triggered once an element volume was reduced to 70% or increased to 170%. During the remesh, the surface meshes of the blood and air chambers, and the fluid–structure interface mesh, were preserved, and a new tetrahedral mesh was generated in the interiors. The solution data at the current step (fluid velocity, acceleration, pressure, and mesh velocity and displacement) were transferred to the new mesh by means of nodal interpolation. No special procedures for transferring the pressure field (e.g., pressure clipping [9, 23, 199]) were employed.

The results reported are for the second cycle. Figures 14–16 show snapshots of the computed flow speed and membrane deformation. The simulation captures complex membrane motion, with many folds, clearly seen in Figs. 15 and 16. The deformed membrane surface is notably smooth, with no sharp kinks on the mesh edges, which is due to the presence of the bending terms in the Kirchhoff–Love shell formulation and the underlying smoothness of the NURBS discretization. The smooth buckling motion of the membrane combined with the MJBS retains the quality of the fluid mechanics mesh near the fluid–structure interface and in the interiors.

During the fill stage, a strong vortex is created in the blood chamber. The vortex is then destroyed early in the eject phase, as seen in Fig. 17.



**Fig. 16:** Ventricular assist device. The membrane deformed configuration at time (a)  $t = 0$  s, (b)  $t = 0.15$  s, (c)  $t = 0.3$  s, and (d)  $t = 0.525$  s.



**Fig. 17:** Ventricular assist device. Blood flow speed above the plane separating the blood and air chambers. In-plane vectors shown during (a) expel stage ( $t = 0.14$  s) and (b) fill stage ( $t = 0.665$  s).

## 12. Concluding remarks

We have provided an overview of some of the mesh moving methods developed since 1990 in connection with the ST-SUPS and ST-VMS and used also with the ALE-SUPS and ALE-VMS. A good mesh moving method can decrease the element distortion in parts of the flow domain where we care about the solution accuracy more, maintain the quality of the boundary layer meshes near the fluid–solid interfaces as the mesh moves to follow those interfaces, and decrease the remeshing frequency even when the fluid–solid and fluid–fluid interfaces undergo large displacements. The mesh moving methods we provided an overview for range from special-purpose mesh moving methods to the MJBS to a mesh moving method based on fiber-reinforced hyperelasticity. These methods have been used in computation of many complex flow problems

in the categories of fluid–particle interaction, FSI, and more generally, MBI. The classes of applications range from flapping aerodynamics of actual locust wings to ventricular assist devices, and we also presented in this article examples of the computations performed.

## Acknowledgement

This work was supported in part by JST-CREST (first author); Grant-in-Aid for Scientific Research (A) 18H04100 from Japan Society for the Promotion of Science (first author); and Rice–Waseda research agreement (first author). The mathematical model and computational method parts of the work were also supported in part by ARO Grant W911NF-17-1-0046 (third author) and Top Global University Project of Waseda University (third author).

## References

- [1] Tezduyar, T.E. (2004). Finite Element Methods for Fluid Dynamics with Moving Boundaries and Interfaces. In Stein, E., Borst, R.D., & Hughes, T.J.R. (editors), *Encyclopedia of Computational Mechanics*, Volume 3: Fluids, chapter 17, Wiley.
- [2] Bazilevs, Y., Takizawa, K., & Tezduyar, T.E. (February 2013). *Computational Fluid–Structure Interaction: Methods and Applications*. Wiley.
- [3] Tezduyar, T.E., Takizawa, K., & Bazilevs, Y. (December 2017). Fluid–Structure Interaction and Flows with Moving Boundaries and Interfaces. In Stein, E., Borst, R.D., & Hughes, T.J.R. (editors), *Encyclopedia of Computational Mechanics Second Edition*, Part 2 Fluids, Wiley, published online.
- [4] Tezduyar, T.E. (2001). Finite Element Methods for Flow Problems with Moving Boundaries and Interfaces. *Archives of Computational Methods in Engineering*, 8, 83–130.
- [5] Tezduyar, T.E., Takizawa, K., Moorman, C., Wright, S., & Christopher, J. (2010). Space–Time Finite Element Computation of Complex Fluid–Structure Interactions. *International Journal for Numerical Methods in Fluids*, 64, 1201–1218.
- [6] Takizawa, K., Tezduyar, T.E., & Avsar, R. (2020). A Low-Distortion Mesh Moving Method Based on Fiber-Reinforced Hyperelasticity and Optimized Zero-Stress State. *Computational Mechanics*, 65, 1567–1591.
- [7] Tezduyar, T.E. (1992). Stabilized Finite Element Formulations for Incompressible Flow Computations. *Advances in Applied Mechanics*, 28, 1–44.
- [8] Tezduyar, T.E. (2003). Computation of Moving Boundaries and Interfaces and Stabilization Parameters. *International Journal for Numerical Methods in Fluids*, 43, 555–575.
- [9] Tezduyar, T.E. & Sathe, S. (2007). Modeling of Fluid–Structure Interactions with the Space–Time Finite Elements: Solution Techniques. *International Journal for Numerical Methods in Fluids*, 54, 855–900.
- [10] Brooks, A.N. & Hughes, T.J.R. (1982). Streamline Upwind/Petrov-Galerkin Formulations for Convection Dominated Flows with Particular Emphasis on the Incompressible Navier-Stokes Equations. *Computer Methods in Applied Mechanics and Engineering*, 32, 199–259.
- [11] Hughes, T.J.R., Liu, W.K., & Zimmermann, T.K. (1981). Lagrangian–Eulerian finite element formulation for incompressible viscous flows. *Computer Methods in Applied Mechanics and Engineering*, 29, 329–349.
- [12] Kalro, V. & Tezduyar, T.E. (2000). A Parallel 3D Computational Method for Fluid–Structure Interactions in Parachute Systems. *Computer Methods in Applied Mechanics and Engineering*, 190, 321–332.

- [13] Bazilevs, Y., Calo, V.M., Hughes, T.J.R., & Zhang, Y. (2008). Isogeometric fluid–structure interaction: theory, algorithms, and computations. *Computational Mechanics*, 43, 3–37.
- [14] Takizawa, K., Bazilevs, Y., & Tezduyar, T.E. (2012). Space–Time and ALE–VMS Techniques for Patient-Specific Cardiovascular Fluid–Structure Interaction Modeling. *Archives of Computational Methods in Engineering*, 19, 171–225.
- [15] Bazilevs, Y., Hsu, M.C., Takizawa, K., & Tezduyar, T.E. (2012). ALE–VMS and ST–VMS Methods for Computer Modeling of Wind-Turbine Rotor Aerodynamics and Fluid–Structure Interaction. *Mathematical Models and Methods in Applied Sciences*, 22(supp02), 1230002.
- [16] Bazilevs, Y., Takizawa, K., & Tezduyar, T.E. (2013). Challenges and Directions in Computational Fluid–Structure Interaction. *Mathematical Models and Methods in Applied Sciences*, 23, 215–221.
- [17] Bazilevs, Y., Takizawa, K., & Tezduyar, T.E. (2015). New Directions and Challenging Computations in Fluid Dynamics Modeling with Stabilized and Multiscale Methods. *Mathematical Models and Methods in Applied Sciences*, 25, 2217–2226.
- [18] Bazilevs, Y., Takizawa, K., & Tezduyar, T.E. (2019). Computational Analysis Methods for Complex Unsteady Flow Problems. *Mathematical Models and Methods in Applied Sciences*, 29, 825–838.
- [19] Hughes, T.J.R. (1995). Multiscale Phenomena: Green’s Functions, The Dirichlet-to-Neumann Formulation, Subgrid Scale Models, Bubbles, and the Origins of Stabilized Methods. *Computer Methods in Applied Mechanics and Engineering*, 127, 387–401.
- [20] Hughes, T.J.R., Oberai, A.A., & Mazzei, L. (2001). Large Eddy Simulation of Turbulent Channel Flows by the Variational Multiscale Method. *Physics of Fluids*, 13, 1784–1799.
- [21] Bazilevs, Y., Calo, V.M., Cottrell, J.A., Hughes, T.J.R., Reali, A., & Scovazzi, G. (2007). Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 197, 173–201.
- [22] Bazilevs, Y. & Akkerman, I. (2010). Large eddy simulation of turbulent Taylor–Couette flow using isogeometric analysis and the residual-based variational multiscale method. *Journal of Computational Physics*, 229, 3402–3414.
- [23] Takizawa, K. & Tezduyar, T.E. (2011). Multiscale Space–Time Fluid–Structure Interaction Techniques. *Computational Mechanics*, 48, 247–267.
- [24] Takizawa, K. & Tezduyar, T.E. (2012). Space–Time Fluid–Structure Interaction Methods. *Mathematical Models and Methods in Applied Sciences*, 22(supp02), 1230001.
- [25] Takizawa, K., Tezduyar, T.E., & Kuraishi, T. (2015). Multiscale ST Methods for Thermo-Fluid Analysis of a Ground Vehicle and its Tires. *Mathematical Models and Methods in Applied Sciences*, 25, 2227–2255.
- [26] Bazilevs, Y., Hsu, M.C., Akkerman, I., Wright, S., Takizawa, K., Henicke, B., Spielman, T., & Tezduyar, T.E. (2011). 3D Simulation of Wind Turbine Rotors at Full Scale. Part I: Geometry Modeling and Aerodynamics. *International Journal for Numerical Methods in Fluids*, 65, 207–235.
- [27] Bazilevs, Y., Hsu, M.C., Kiendl, J., Wüchener, R., & Bletzinger, K.U. (2011). 3D simulation of wind turbine rotors at full scale. Part II: Fluid–structure interaction modeling with composite blades. *International Journal for Numerical Methods in Fluids*, 65, 236–253.
- [28] Hsu, M.C., Akkerman, I., & Bazilevs, Y. (2011). High-performance computing of wind turbine aerodynamics using isogeometric analysis. *Computers & Fluids*, 49, 93–100.

- [29] Bazilevs, Y., Hsu, M.C., & Scott, M.A. (2012). Isogeometric Fluid–Structure Interaction Analysis with Emphasis on Non-Matching Discretizations, and with Application to Wind Turbines. *Computer Methods in Applied Mechanics and Engineering*, 249–252, 28–41.
- [30] Hsu, M.C., Akkerman, I., & Bazilevs, Y. (2014). Finite element simulation of wind turbine aerodynamics: Validation study using NREL Phase VI experiment. *Wind Energy*, 17, 461–481.
- [31] Korobenko, A., Hsu, M.C., Akkerman, I., Tippmann, J., & Bazilevs, Y. (2013). Structural mechanics modeling and FSI simulation of wind turbines. *Mathematical Models and Methods in Applied Sciences*, 23, 249–272.
- [32] Korobenko, A., Hsu, M.C., Akkerman, I., & Bazilevs, Y. (2013). Aerodynamic simulation of vertical-axis wind turbines. *Journal of Applied Mechanics*, 81, 021011.
- [33] Bazilevs, Y., Takizawa, K., Tezduyar, T.E., Hsu, M.C., Kostov, N., & McIntyre, S. (2014). Aerodynamic and FSI Analysis of Wind Turbines with the ALE-VMS and ST-VMS Methods. *Archives of Computational Methods in Engineering*, 21, 359–398.
- [34] Bazilevs, Y., Korobenko, A., Deng, X., Yan, J., Kinzel, M., & Dabiri, J.O. (2014). FSI modeling of vertical-axis wind turbines. *Journal of Applied Mechanics*, 81, 081006.
- [35] Bazilevs, Y., Korobenko, A., Deng, X., & Yan, J. (2015). Novel structural modeling and mesh moving techniques for advanced FSI simulation of wind turbines. *International Journal for Numerical Methods in Engineering*, 102, 766–783.
- [36] Bazilevs, Y., Korobenko, A., Yan, J., Pal, A., Gohari, S.M.I., & Sarkar, S. (2015). ALE-VMS Formulation for Stratified Turbulent Incompressible Flows with Applications. *Mathematical Models and Methods in Applied Sciences*, 25, 2349–2375.
- [37] Bazilevs, Y., Korobenko, A., Deng, X., & Yan, J. (2016). FSI modeling for fatigue-damage prediction in full-scale wind-turbine blades. *Journal of Applied Mechanics*, 83(6), 061010.
- [38] Yan, J., Korobenko, A., Deng, X., & Bazilevs, Y. (2016). Computational free-surface fluid–structure interaction with application to floating offshore wind turbines. *Computers and Fluids*, 141, 155–174.
- [39] Korobenko, A., Yan, J., Gohari, S.M.I., Sarkar, S., & Bazilevs, Y. (2017). FSI simulation of two back-to-back wind turbines in atmospheric boundary layer flow. *Computers & Fluids*, 158, 167–175.
- [40] Korobenko, A., Bazilevs, Y., Takizawa, K., & Tezduyar, T.E. (2018). Recent Advances in ALE-VMS and ST-VMS Computational Aerodynamic and FSI Analysis of Wind Turbines. In Tezduyar, T.E. (editor), *Frontiers in Computational Fluid–Structure Interaction and Flow Simulation: Research from Lead Investigators under Forty – 2018*, Modeling and Simulation in Science, Engineering and Technology, Springer, 253–336.
- [41] Korobenko, A., Bazilevs, Y., Takizawa, K., & Tezduyar, T.E. (2019). Computer modeling of wind turbines: 1. ALE-VMS and ST-VMS aerodynamic and FSI analysis. *Archives of Computational Methods in Engineering*, 26, 1059–1099.
- [42] Bazilevs, Y., Takizawa, K., Tezduyar, T.E., Hsu, M.C., Otoguro, Y., Mochizuki, H., & Wu, M.C.H. (2020). Wind Turbine and Turbomachinery Computational Analysis with the ALE and Space–Time Variational Multiscale Methods and Isogeometric Discretization. *Journal of Advanced Engineering and Computation*, 4, 1–32.
- [43] Bazilevs, Y., Takizawa, K., Tezduyar, T.E., Hsu, M.C., Otoguro, Y., Mochizuki, H., & Wu, M.C.H. (2020). ALE and Space–Time Variational Multiscale Isogeometric Analysis of Wind Turbines and

- Turbomachinery. In Grama, A. & Sameh, A. (editors), *Parallel Algorithms in Computational Science and Engineering*, Modeling and Simulation in Science, Engineering and Technology, Springer, 195–233.
- [44] Takizawa, K., Bazilevs, Y., Tezduyar, T.E., & Korobenko, A. (2020). Variational Multiscale Flow Analysis in Aerospace, Energy and Transportation Technologies. In Grama, A. & Sameh, A. (editors), *Parallel Algorithms in Computational Science and Engineering*, Modeling and Simulation in Science, Engineering and Technology, Springer, 235–280.
- [45] Takizawa, K., Bazilevs, Y., Tezduyar, T.E., & Korobenko, A. (2020). Computational Flow Analysis in Aerospace, Energy and Transportation Technologies with the Variational Multiscale Methods. *Journal of Advanced Engineering and Computation*, 4, 83–117.
- [46] Bayram, A.M., Bear, C., Bear, M., & Korobenko, A. (2020). Performance analysis of two vertical-axis hydrokinetic turbines using variational multiscale method. *Computers & Fluids*, 200, 104432, available online.
- [47] Ravensbergen, M., Bayram, A.M., & Korobenko, A. (2020). The Actuator Line Method for Wind Turbine Modelling Applied in a Variational Multiscale Framework. *Computers & Fluids*, 201, 104465, available online.
- [48] Bazilevs, Y., Calo, V.M., Zhang, Y., & Hughes, T.J.R. (2006). Isogeometric fluid–structure interaction analysis with applications to arterial blood flow. *Computational Mechanics*, 38, 310–322.
- [49] Bazilevs, Y., Gohean, J.R., Hughes, T.J.R., Moser, R.D., & Zhang, Y. (2009). Patient-specific isogeometric fluid–structure interaction analysis of thoracic aortic blood flow due to implantation of the Jarvik 2000 left ventricular assist device. *Computer Methods in Applied Mechanics and Engineering*, 198, 3534–3550.
- [50] Bazilevs, Y., Hsu, M.C., Benson, D., Sankaran, S., & Marsden, A. (2009). Computational Fluid–Structure Interaction: Methods and Application to a Total Cavopulmonary Connection. *Computational Mechanics*, 45, 77–89.
- [51] Bazilevs, Y., Hsu, M.C., Zhang, Y., Wang, W., Liang, X., Kvamsdal, T., Brekken, R., & Isaksen, J. (2010). A Fully-Coupled Fluid–Structure Interaction Simulation of Cerebral Aneurysms. *Computational Mechanics*, 46, 3–16.
- [52] Bazilevs, Y., Hsu, M.C., Zhang, Y., Wang, W., Kvamsdal, T., Hentschel, S., & Isaksen, J. (2010). Computational Fluid–Structure Interaction: Methods and Application to Cerebral Aneurysms. *Biomechanics and Modeling in Mechanobiology*, 9, 481–498.
- [53] Hsu, M.C. & Bazilevs, Y. (2011). Blood vessel tissue prestress modeling for vascular fluid–structure interaction simulations. *Finite Elements in Analysis and Design*, 47, 593–599.
- [54] Long, C.C., Marsden, A.L., & Bazilevs, Y. (2013). Fluid–structure interaction simulation of pulsatile ventricular assist devices. *Computational Mechanics*, 52, 971–981.
- [55] Long, C.C., Esmaily-Moghadam, M., Marsden, A.L., & Bazilevs, Y. (2014). Computation of residence time in the simulation of pulsatile ventricular assist devices. *Computational Mechanics*, 54, 911–919.
- [56] Long, C.C., Marsden, A.L., & Bazilevs, Y. (2014). Shape optimization of pulsatile ventricular assist devices using FSI to minimize thrombotic risk. *Computational Mechanics*, 54, 921–932.
- [57] Hsu, M.C., Kamensky, D., Bazilevs, Y., Sacks, M.S., & Hughes, T.J.R. (2014). Fluid–structure interaction analysis of bio-prosthetic heart valves: significance of arterial wall deformation. *Computational Mechanics*, 54, 1055–1071.

- [58] Hsu, M.C., Kamensky, D., Xu, F., Kiendl, J., Wang, C., Wu, M.C.H., Mineroff, J., Reali, A., Bazilevs, Y., & Sacks, M.S. (2015). Dynamic and fluid–structure interaction simulations of bioprosthetic heart valves using parametric design with T-splines and Fung-type material models. *Computational Mechanics*, *55*, 1211–1225.
- [59] Kamensky, D., Hsu, M.C., Schillinger, D., Evans, J.A., Aggarwal, A., Bazilevs, Y., Sacks, M.S., & Hughes, T.J.R. (2015). An immersogeometric variational framework for fluid–structure interaction: Application to bioprosthetic heart valves. *Computer Methods in Applied Mechanics and Engineering*, *284*, 1005–1053.
- [60] Takizawa, K., Bazilevs, Y., Tezduyar, T.E., & Hsu, M.C. (2019). Computational Cardiovascular Flow Analysis with the Variational Multiscale Methods. *Journal of Advanced Engineering and Computation*, *3*, 366–405.
- [61] Hughes, T.J.R., Takizawa, K., Bazilevs, Y., Tezduyar, T.E., & Hsu, M.C. (2020). Computational Cardiovascular Analysis with the Variational Multiscale Methods and Isogeometric Discretization. In Grama, A. & Sameh, A. (editors), *Parallel Algorithms in Computational Science and Engineering*, Modeling and Simulation in Science, Engineering and Technology, Springer, 151–193.
- [62] Akkerman, I., Bazilevs, Y., Benson, D.J., Farthing, M.W., & Kees, C.E. (2012). Free-Surface Flow and Fluid–Object Interaction Modeling with Emphasis on Ship Hydrodynamics. *Journal of Applied Mechanics*, *79*, 010905.
- [63] Akkerman, I., Dunaway, J., Kvandal, J., Spinks, J., & Bazilevs, Y. (2012). Toward free-surface modeling of planing vessels: simulation of the Fridsma hull using ALE-VMS. *Computational Mechanics*, *50*, 719–727.
- [64] Yan, J., Deng, X., Korobenko, A., & Bazilevs, Y. (2017). Free-surface flow modeling and simulation of horizontal-axis tidal-stream turbines. *Computers and Fluids*, *158*, 157–166.
- [65] Yan, J., Deng, X., Xu, F., Xu, S., & Zhu, Q. (2020). Numerical simulations of two back-to-back horizontal axis tidal stream turbines in free-surface flows. *Journal of Applied Mechanics*, *87*(6).
- [66] Zhu, Q., Yan, J., Tejada-Martínez, A., & Bazilevs, Y. (2020). Variational multiscale modeling of Langmuir turbulent boundary layers in shallow water using Isogeometric Analysis. *Mechanics Research Communications*, *108*, 103570.
- [67] Wang, C., Wu, M.C.H., Xu, F., Hsu, M.C., & Bazilevs, Y. (2017). Modeling of a hydraulic arresting gear using fluid–structure interaction and isogeometric analysis. *Computers and Fluids*, *142*, 3–14.
- [68] Wu, M.C.H., Kamensky, D., Wang, C., Herrema, A.J., Xu, F., Pigazzini, M.S., Verma, A., Marsden, A.L., Bazilevs, Y., & Hsu, M.C. (2017). Optimizing fluid–structure interaction systems with immersogeometric analysis and surrogate modeling: Application to a hydraulic arresting gear. *Computer Methods in Applied Mechanics and Engineering*, *316*, 668–693.
- [69] Xu, F., Moutsanidis, G., Kamensky, D., Hsu, M.C., Murugan, M., Ghoshal, A., & Bazilevs, Y. (2017). Compressible flows on moving domains: Stabilized methods, weakly enforced essential boundary conditions, sliding interfaces, and application to gas-turbine modeling. *Computers & Fluids*, *158*, 201–220.
- [70] Murugan, M., Ghoshal, A., Xu, F., Hsu, M.C., Bazilevs, Y., Bravo, L., & Kerner, K. (2017). Analytical Study of Articulating Turbine Rotor Blade Concept for Improved Off-Design Performance of Gas Turbine Engines. *Journal of Engineering for Gas Turbines and Power*, *139*, 102601–6.
- [71] Castorrini, A., Corsini, A., Rispoli, F., Takizawa, K., & Tezduyar, T.E. (2019). A

- stabilized ALE method for computational fluid–structure interaction analysis of passive morphing in turbomachinery. *Mathematical Models and Methods in Applied Sciences*, 29, 967–994.
- [72] Kozak, N., Xu, F., Rajanna, M.R., Bravo, L., Murugan, M., Ghoshal, A., Bazilevs, Y., & Hsu, M.C. (2020). High-Fidelity Finite Element Modeling and Analysis of Adaptive Gas Turbine Stator–Rotor Flow Interaction at Off-Design Conditions. *Journal of Mechanics*, 36, 595–606.
- [73] Kozak, N., Rajanna, M.R., Wu, M.C.H., Murugan, M., Bravo, L., Ghoshal, A., Hsu, M.C., & Bazilevs, Y. (2020). Optimizing Gas Turbine Performance Using the Surrogate Management Framework and High-Fidelity Flow Modeling. *Energies*, 13, 4283.
- [74] Bazilevs, Y., Takizawa, K., Wu, M.C.H., Kuraishi, T., Avsar, R., Xu, Z., & Tezduyar, T.E. (2021). Gas turbine computational flow and structure analysis with isogeometric discretization and a complex-geometry mesh generation method. *Computational Mechanics*, 67, 57–84.
- [75] Zhu, Q. & Yan, J. (2021). A moving-domain CFD solver in FEniCS with applications to tidal turbine simulations in turbulent flows. *Computers & Mathematics with Applications*, 81, 532–546.
- [76] Augier, B., Yan, J., Korobenko, A., Czarnowski, J., Ketterman, G., & Bazilevs, Y. (2015). Experimental and numerical FSI study of compliant hydrofoils. *Computational Mechanics*, 55, 1079–1090.
- [77] Yan, J., Augier, B., Korobenko, A., Czarnowski, J., Ketterman, G., & Bazilevs, Y. (2016). FSI modeling of a propulsion system based on compliant hydrofoils in a tandem configuration. *Computers and Fluids*, 141, 201–211.
- [78] Zhu, Q., Xu, F., Xu, S., Hsu, M.C., & Yan, J. (2020). An immersogeometric formulation for free-surface flows with application to marine engineering problems. *Computer Methods in Applied Mechanics and Engineering*, 361, 112748.
- [79] Helgedagsrud, T.A., Bazilevs, Y., Mathisen, K.M., & Oiseth, O.A. (2019). Computational and experimental investigation of free vibration and flutter of bridge decks. *Computational Mechanics*, 63, 121–136.
- [80] Helgedagsrud, T.A., Bazilevs, Y., Korobenko, A., Mathisen, K.M., & Oiseth, O.A. (2019). Using ALE-VMS to compute aerodynamic derivatives of bridge sections. *Computers & Fluids*, 179, 820–832.
- [81] Helgedagsrud, T.A., Akkerman, I., Bazilevs, Y., Mathisen, K.M., & Oiseth, O.A. (2019). Isogeometric modeling and experimental investigation of moving-domain bridge aerodynamics. *ASCE Journal of Engineering Mechanics*, 145, 04019026.
- [82] Helgedagsrud, T.A., Bazilevs, Y., Mathisen, K.M., Yan, J., & Oiseth, O.A. (2019). Modeling and simulation of bridge-section buffeting response in turbulent flow. *Mathematical Models and Methods in Applied Sciences*, 29, 939–966.
- [83] Helgedagsrud, T.A., Bazilevs, Y., Mathisen, K.M., & Oiseth, O.A. (2019). ALE-VMS methods for wind-resistant design of long-span bridges. *Journal of Wind Engineering and Industrial Aerodynamics*, 191, 143–153.
- [84] Yan, J., Korobenko, A., Tejada-Martinez, A.E., Golshan, R., & Bazilevs, Y. (2017). A new variational multiscale formulation for stratified incompressible turbulent flows. *Computers & Fluids*, 158, 150–156.
- [85] Ravensbergen, M., Helgedagsrud, T.A., Bazilevs, Y., & Korobenko, A. (2020). A variational multiscale framework for atmospheric turbulent flows over complex environmental terrains. *Computer Methods in Applied Mechanics and Engineering*, 368, 113182.

- [86] Codoni, D., Moutsanidis, G., Hsu, M.C., Bazilevs, Y., Johansen, C., & Korobenko, A. (2021). Stabilized methods for high-speed compressible flows: toward hypersonic simulations. *Computational Mechanics*, *67*, 785–809.
- [87] Yan, J., Yan, W., Lin, S., & Wagner, G. (2018). A fully coupled finite element formulation for liquid–solid–gas thermo–fluid flow with melting and solidification. *Computer Methods in Applied Mechanics and Engineering*, *336*, 444–470.
- [88] Yan, J., S. Lin, S., Bazilevs, Y., & Wagner, G. (2019). Isogeometric analysis of multi-phase flows with surface tension and with application to dynamics of rising bubbles. *Computers & Fluids*, *179*, 777–789.
- [89] Xu, S., Liu, N., & Yan, J. (2019). Residual-based variational multi-scale modeling for particle-laden gravity currents over flat and triangular wavy terrains. *Computers & Fluids*, *188*, 114–124.
- [90] Bayram, A.M. & Korobenko, A. (2020). Variational Multiscale Framework for Cavitating Flows. *Computational Mechanics*, *66*, 49–67.
- [91] Zhao, Z. & Yan, J. (2020). Variational multi-scale modeling of interfacial flows with a balanced-force surface tension model. *Mechanics Research Communications*, 103608.
- [92] Cen, H., Zhou, Q., & Korobenko, A. (2021). Variational Multiscale Framework for Cavitating Flows. *Computers & Fluids*, *214*, 104765.
- [93] Zhao, Z., Zhu, Q., & Yan, J. (2021). A thermal multi-phase flow model for directed energy deposition processes via a moving signed distance function. *Computer Methods in Applied Mechanics and Engineering*, *373*, 113518.
- [94] Zhu, Q., Liu, Z., & Yan, J. (2021). Machine learning for metal additive manufacturing: predicting temperature and melt pool fluid dynamics using physics-informed neural networks. *Computational Mechanics*, *67*, 619–635.
- [95] Hsu, M.C., Wang, C., Xu, F., Herrema, A.J., & Krishnamurthy, A. (2016). Direct immersogeometric fluid flow analysis using B-rep CAD models. *Computer Aided Geometric Design*, *43*, 143–158.
- [96] Xu, F., Schillinger, D., Kamensky, D., Varduhn, V., Wang, C., & Hsu, M.C. (2016). The tetrahedral finite cell method for fluids: Immersogeometric analysis of turbulent flow around complex geometries. *Computers & Fluids*, *141*, 135–154.
- [97] Wang, C., Xu, F., Hsu, M.C., & Krishnamurthy, A. (2017). Rapid B-rep model pre-processing for immersogeometric analysis using analytic surfaces. *Computer Aided Geometric Design*, *52-53*, 190–204.
- [98] Xu, S., Xu, F., Kommajosula, A., Hsu, M.C., & Ganapathysubramanian, B. (2019). Immersogeometric analysis of moving objects in incompressible flows. *Computers & Fluids*, *189*, 24–33.
- [99] Xu, S., Gao, B., Lofquist, A., Fernando, M., Hsu, M.C., Sundar, H., & Ganapathysubramanian, B. (2020). An octree-based immersogeometric approach for modeling inertial migration of particles in channels. *Computers & Fluids*, *214*, 104764.
- [100] Kamensky, D., Evans, J.A., & Hsu, M.C. (2015). Stability and Conservation Properties of Collocated Constraints in Immersogeometric Fluid–Thin Structure Interaction Analysis. *Communications in Computational Physics*, *18*, 1147–1180.
- [101] Kamensky, D., Evans, J.A., Hsu, M.C., & Bazilevs, Y. (2017). Projection-based stabilization of interface Lagrange multipliers in immersogeometric fluid–thin structure interaction analysis, with application to heart valve modeling. *Computers and Mathematics with Applications*, *74*, 2068–2088.

- [102] Kamensky, D., Hsu, M.C., Yu, Y., Evans, J.A., Sacks, M.S., & Hughes, T.J.R. (2017). Immersogeometric cardiovascular fluid–structure interaction analysis with divergence-conforming B-splines. *Computer Methods in Applied Mechanics and Engineering*, 314, 408–472.
- [103] Xu, F., Morganti, S., Zakerzadeh, R., Kamensky, D., Auricchio, F., Reali, A., Hughes, T.J.R., Sacks, M.S., & Hsu, M.C. (2018). A framework for designing patient-specific bioprosthetic heart valves using immersogeometric fluid–structure interaction analysis. *International Journal for Numerical Methods in Biomedical Engineering*, 34, e2938.
- [104] Yu, Y., Kamensky, D., Hsu, M.C., Lu, X.Y., Bazilevs, Y., & Hughes, T.J.R. (2018). Error estimates for projection-based dynamic augmented Lagrangian boundary condition enforcement, with application to fluid–structure interaction. *Mathematical Models and Methods in Applied Science*, 28, 2457–2509.
- [105] Wu, M.C.H., Zakerzadeh, R., Kamensky, D., Kiendl, J., Sacks, M.S., & Hsu, M.C. (2018). An anisotropic constitutive model for immersogeometric fluid–structure interaction analysis of bioprosthetic heart valves. *Journal of Biomechanics*, 74, 23–31.
- [106] Wu, M.C.H., Muchowski, H.M., Johnson, E.L., Rajanna, M.R., & Hsu, M.C. (2019). Immersogeometric fluid-structure interaction modeling and simulation of transcatheter aortic valve replacement. *Computer Methods in Applied Mechanics and Engineering*, 357, 112556.
- [107] Johnson, E.L., Wu, M.C.H., Xu, F., Wiese, N.M., Rajanna, M.R., Herrema, A.J., Ganapathysubramanian, B., Hughes, T.J.R., Sacks, M.S., & Hsu, M.C. (2020). Thinner biological tissues induce leaflet flutter in aortic heart valve replacements. *Proceedings of the National Academy of Sciences*, 117, 19007–19016.
- [108] Xu, F., Johnson, E.L., Wang, C., Jafari, A., Yang, C.H., Sacks, M.S., Krishnamurthy, A., & Hsu, M.C. (2021). Computational investigation of left ventricular hemodynamics following bioprosthetic aortic and mitral valve replacement. *Mechanics Research Communications*, <https://doi.org/10.1016/j.mechrescom.2020.103604>.
- [109] Tezduyar, T.E. & Takizawa, K. (2019). Space–time computations in practical engineering applications: A summary of the 25-year history. *Computational Mechanics*, 63, 747–753.
- [110] Takizawa, K. & Tezduyar, T.E. (2012). Computational Methods for Parachute Fluid–Structure Interactions. *Archives of Computational Methods in Engineering*, 19, 125–169.
- [111] Takizawa, K., Fritze, M., Montes, D., Spielman, T., & Tezduyar, T.E. (2012). Fluid–structure interaction modeling of ringsail parachutes with disreefing and modified geometric porosity. *Computational Mechanics*, 50, 835–854.
- [112] Takizawa, K., Montes, D., Fritze, M., McIntyre, S., Boben, J., & Tezduyar, T.E. (2013). Methods for FSI modeling of spacecraft parachute dynamics and cover separation. *Mathematical Models and Methods in Applied Sciences*, 23, 307–338.
- [113] Takizawa, K., Tezduyar, T.E., Boben, J., Kostov, N., Boswell, C., & Buscher, A. (2013). Fluid–structure interaction modeling of clusters of spacecraft parachutes with modified geometric porosity. *Computational Mechanics*, 52, 1351–1364.
- [114] Takizawa, K., Tezduyar, T.E., Kolesar, R., Boswell, C., Kanai, T., & Montel, K. (2014). Multiscale Methods for Gore Curvature Calculations from FSI Modeling of Spacecraft Parachutes. *Computational Mechanics*, 54, 1461–1476.
- [115] Takizawa, K., Tezduyar, T.E., Boswell, C., Kolesar, R., & Montel, K. (2014). FSI

- Modeling of the Reefed Stages and Dismantling of the Orion Spacecraft Parachutes. *Computational Mechanics*, 54, 1203–1220.
- [116] Takizawa, K., Tezduyar, T.E., Boswell, C., Tsutsui, Y., & Montel, K. (2015). Special Methods for Aerodynamic-Moment Calculations from Parachute FSI Modeling. *Computational Mechanics*, 55, 1059–1069.
- [117] Takizawa, K., Tezduyar, T.E., & Kolesar, R. (2015). FSI Modeling of the Orion Spacecraft Drogue Parachutes. *Computational Mechanics*, 55, 1167–1179.
- [118] Takizawa, K., Tezduyar, T.E., & Terahara, T. (2016). Ram-Air Parachute Structural and Fluid Mechanics Computations with the Space-Time Isogeometric Analysis (ST-IGA). *Computers & Fluids*, 141, 191–200.
- [119] Takizawa, K., Tezduyar, T.E., & Kanai, T. (2017). Porosity models and computational methods for compressible-flow aerodynamics of parachutes with geometric porosity. *Mathematical Models and Methods in Applied Sciences*, 27, 771–806.
- [120] Kanai, T., Takizawa, K., Tezduyar, T.E., Tanaka, T., & Hartmann, A. (2019). Compressible-Flow Geometric-Porosity Modeling and Spacecraft Parachute Computation with Isogeometric Discretization. *Computational Mechanics*, 63, 301–321.
- [121] Takizawa, K., Henicke, B., Tezduyar, T.E., Hsu, M.C., & Bazilevs, Y. (2011). Stabilized Space-Time Computation of Wind-Turbine Rotor Aerodynamics. *Computational Mechanics*, 48, 333–344.
- [122] Takizawa, K., Henicke, B., Montes, D., Tezduyar, T.E., Hsu, M.C., & Bazilevs, Y. (2011). Numerical-Performance Studies for the Stabilized Space-Time Computation of Wind-Turbine Rotor Aerodynamics. *Computational Mechanics*, 48, 647–657.
- [123] Takizawa, K., Tezduyar, T.E., McIntyre, S., Kostov, N., Kolesar, R., & Habluetzel, C. (2014). Space-time VMS computation of wind-turbine rotor and tower aerodynamics. *Computational Mechanics*, 53, 1–15.
- [124] Takizawa, K., Bazilevs, Y., Tezduyar, T.E., Hsu, M.C., Øiseth, O., Mathisen, K.M., Kostov, N., & McIntyre, S. (2014). Engineering Analysis and Design with ALE-VMS and Space-Time Methods. *Archives of Computational Methods in Engineering*, 21, 481–508.
- [125] Takizawa, K. (2014). Computational Engineering Analysis with the New-Generation Space-Time Methods. *Computational Mechanics*, 54, 193–211.
- [126] Takizawa, K., Tezduyar, T.E., Mochizuki, H., Hattori, H., Mei, S., Pan, L., & Montel, K. (2015). Space-time VMS method for flow computations with slip interfaces (ST-SI). *Mathematical Models and Methods in Applied Sciences*, 25, 2377–2406.
- [127] Otoguro, Y., Mochizuki, H., Takizawa, K., & Tezduyar, T.E. (2020). Space-Time Variational Multiscale Isogeometric Analysis of a tsunami-shelter vertical-axis wind turbine. *Computational Mechanics*, 66, 1443–1460.
- [128] Kuraishi, T., Zhang, F., Takizawa, K., & Tezduyar, T.E. (2021). Wind turbine wake computation with the ST-VMS method, isogeometric discretization and multidomain method: I. Computational framework. *Computational Mechanics*, 68, 113–130.
- [129] Kuraishi, T., Zhang, F., Takizawa, K., & Tezduyar, T.E. (2021). Wind turbine wake computation with the ST-VMS method, isogeometric discretization and multidomain method: II. Spatial and temporal resolution. *Computational Mechanics*, 68, 175–184.
- [130] Zhang, F., Kuraishi, T., Takizawa, K., & Tezduyar, T.E. (2022). Wind turbine wake computation with the ST-VMS method and isogeometric discretization: Directional preference in spatial refinement. *Computational Mechanics*, 69, 1031–1040.

- [131] Takizawa, K., Henicke, B., Puntel, A., Spielman, T., & Tezduyar, T.E. (2012). Space-time computational techniques for the aerodynamics of flapping wings. *Journal of Applied Mechanics*, 79, 010903.
- [132] Takizawa, K., Henicke, B., Puntel, A., Kostov, N., & Tezduyar, T.E. (2012). Space-Time Techniques for Computational Aerodynamics Modeling of Flapping Wings of an Actual Locust. *Computational Mechanics*, 50, 743–760.
- [133] Takizawa, K., Kostov, N., Puntel, A., Henicke, B., & Tezduyar, T.E. (2012). Space-time computational analysis of bio-inspired flapping-wing aerodynamics of a micro aerial vehicle. *Computational Mechanics*, 50, 761–778.
- [134] Takizawa, K., Henicke, B., Puntel, A., Kostov, N., & Tezduyar, T.E. (2013). Computer Modeling Techniques for Flapping-Wing Aerodynamics of a Locust. *Computers & Fluids*, 85, 125–134.
- [135] Takizawa, K., Tezduyar, T.E., Buscher, A., & Asada, S. (2014). Space-Time Interface-Tracking with Topology Change (ST-TC). *Computational Mechanics*, 54, 955–971.
- [136] Takizawa, K., Tezduyar, T.E., & Kostov, N. (2014). Sequentially-coupled space-time FSI analysis of bio-inspired flapping-wing aerodynamics of an MAV. *Computational Mechanics*, 54, 213–233.
- [137] Takizawa, K., Tezduyar, T.E., & Buscher, A. (2015). Space-Time Computational Analysis of MAV Flapping-Wing Aerodynamics with Wing Clapping. *Computational Mechanics*, 55, 1131–1141.
- [138] Takizawa, K., Schjodt, K., Puntel, A., Kostov, N., & Tezduyar, T.E. (2012). Patient-specific computer modeling of blood flow in cerebral arteries with aneurysm and stent. *Computational Mechanics*, 50, 675–686.
- [139] Takizawa, K., Schjodt, K., Puntel, A., Kostov, N., & Tezduyar, T.E. (2013). Patient-Specific Computational Analysis of the Influence of a Stent on the Unsteady Flow in Cerebral Aneurysms. *Computational Mechanics*, 51, 1061–1073.
- [140] Takizawa, K., Bazilevs, Y., Tezduyar, T.E., Long, C.C., Marsden, A.L., & Schjodt, K. (2014). ST and ALE-VMS Methods for Patient-Specific Cardiovascular Fluid Mechanics Modeling. *Mathematical Models and Methods in Applied Sciences*, 24, 2437–2486.
- [141] Takizawa, K., Tezduyar, T.E., Buscher, A., & Asada, S. (2014). Space-Time Fluid Mechanics Computation of Heart Valve Models. *Computational Mechanics*, 54, 973–986.
- [142] Suito, H., Takizawa, K., Huynh, V.Q.H., Sze, D., & Ueda, T. (2014). FSI analysis of the blood flow and geometrical characteristics in the thoracic aorta. *Computational Mechanics*, 54, 1035–1045.
- [143] Suito, H., Takizawa, K., Huynh, V.Q.H., Sze, D., Ueda, T., & Tezduyar, T.E. (2016). A geometrical-characteristics study in patient-specific FSI analysis of blood flow in the thoracic aorta. In Bazilevs, Y. & Takizawa, K. (editors), *Advances in Computational Fluid-Structure Interaction and Flow Simulation: New Methods and Challenging Computations*, Modeling and Simulation in Science, Engineering and Technology, Springer, 379–386.
- [144] Takizawa, K. & Tezduyar, T.E. (2016). New directions in space-time computational methods. In Bazilevs, Y. & Takizawa, K. (editors), *Advances in Computational Fluid-Structure Interaction and Flow Simulation: New Methods and Challenging Computations*, Modeling and Simulation in Science, Engineering and Technology, Springer, 159–178.
- [145] Takizawa, K., Tezduyar, T.E., Terahara, T., & Sasaki, T. (2018). Heart valve flow computation with the Space-Time Slip Interface Topology Change (ST-SI-TC) method and Isogeometric Analysis (IGA). In Wriggers, P. & Lenarz, T. (editors),

- Biomedical Technology: Modeling, Experiments and Simulation*, Lecture Notes in Applied and Computational Mechanics, Springer, 77–99.
- [146] Takizawa, K., Tezduyar, T.E., Terahara, T., & Sasaki, T. (2017). Heart valve flow computation with the integrated Space–Time VMS, Slip Interface, Topology Change and Isogeometric Discretization methods. *Computers & Fluids*, *158*, 176–188.
- [147] Takizawa, K., Tezduyar, T.E., Uchikawa, H., Terahara, T., Sasaki, T., Shiozaki, K., Yoshida, A., Komiya, K., & Inoue, G. (2018). Aorta Flow Analysis and Heart Valve Flow and Structure Analysis. In Tezduyar, T.E. (editor), *Frontiers in Computational Fluid–Structure Interaction and Flow Simulation: Research from Lead Investigators under Forty – 2018*, Modeling and Simulation in Science, Engineering and Technology, Springer, 29–89.
- [148] Takizawa, K., Tezduyar, T.E., Uchikawa, H., Terahara, T., Sasaki, T., & Yoshida, A. (2019). Mesh refinement influence and cardiac-cycle flow periodicity in aorta flow analysis with isogeometric discretization. *Computers & Fluids*, *179*, 790–798.
- [149] Terahara, T., Takizawa, K., Tezduyar, T.E., Bazilevs, Y., & Hsu, M.C. (2020). Heart Valve Isogeometric Sequentially-Coupled FSI Analysis with the Space–Time Topology Change Method. *Computational Mechanics*, *65*, 1167–1187.
- [150] Terahara, T., Takizawa, K., Tezduyar, T.E., Tsushima, A., & Shiozaki, K. (2020). Ventricle-valve-aorta flow analysis with the Space–Time Isogeometric Discretization and Topology Change. *Computational Mechanics*, *65*, 1343–1363.
- [151] Yu, Y., Zhang, Y.J., Takizawa, K., Tezduyar, T.E., & Sasaki, T. (2020). Anatomically Realistic Lumen Motion Representation in Patient-Specific Space–Time Isogeometric Flow Analysis of Coronary Arteries with Time-Dependent Medical-Image Data. *Computational Mechanics*, *65*, 395–404.
- [152] Takizawa, K., Terahara, T., & Tezduyar, T.E. (2022). Space–Time Flow Computation with Contact Between the Moving Solid Surfaces. In Aldakheel, F., Hudobivnik, B., Soleimani, M., Wessels, H., Weissenfels, C., & Marino, M. (editors), *Current Trends and Open Problems in Computational Mechanics*, Springer, 517–525.
- [153] Takizawa, K., Montes, D., McIntyre, S., & Tezduyar, T.E. (2013). Space–Time VMS Methods for Modeling of Incompressible Flows at High Reynolds Numbers. *Mathematical Models and Methods in Applied Sciences*, *23*, 223–248.
- [154] Takizawa, K., Tezduyar, T.E., Asada, S., & Kuraishi, T. (2016). Space–Time Method for Flow Computations with Slip Interfaces and Topology Changes (ST-SITC). *Computers & Fluids*, *141*, 124–134.
- [155] Kuraishi, T., Takizawa, K., & Tezduyar, T.E. (2018). Space–Time Computational Analysis of Tire Aerodynamics with Actual Geometry, Road Contact and Tire Deformation. In Tezduyar, T.E. (editor), *Frontiers in Computational Fluid–Structure Interaction and Flow Simulation: Research from Lead Investigators under Forty – 2018*, Modeling and Simulation in Science, Engineering and Technology, Springer, 337–376.
- [156] Kuraishi, T., Takizawa, K., & Tezduyar, T.E. (2019). Tire Aerodynamics with Actual Tire Geometry, Road Contact and Tire Deformation. *Computational Mechanics*, *63*, 1165–1185.
- [157] Kuraishi, T., Takizawa, K., & Tezduyar, T.E. (2019). Space–Time Computational Analysis of Tire Aerodynamics with Actual Geometry, Road Contact, Tire Deformation, Road Roughness and Fluid Film. *Computational Mechanics*, *64*, 1699–1718.
- [158] Tezduyar, T.E., Takizawa, K., & Kuraishi, T. (2022). Space–Time Computational FSI and Flow Analysis: 2004 and Beyond. In Aldakheel, F., Hudobivnik,

- B., Soleimani, M., Wessels, H., Weissenfels, C., & Marino, M. (editors), *Current Trends and Open Problems in Computational Mechanics*, Springer, 537–544.
- [159] Kuraishi, T., Yamasaki, S., Takizawa, K., Tezduyar, T.E., Xu, Z., & Kaneko, R. (2022). Space–time isogeometric analysis of car and tire aerodynamics with road contact and tire deformation and rotation, *Computational Mechanics*, published online, DOI: 10.1007/s00466-022-02155-0.
- [160] Kuraishi, T., Terahara, T., Takizawa, K., & Tezduyar, T.E. (2022). Computational flow analysis with boundary layer and contact representation: I. Tire aerodynamics with road contact. *Journal of Mechanics*, 38, 77–87.
- [161] Takizawa, K., Tezduyar, T.E., Kuraishi, T., Tabata, S., & Takagi, H. (2016). Computational thermo-fluid analysis of a disk brake. *Computational Mechanics*, 57, 965–977.
- [162] Takizawa, K., Tezduyar, T.E., & Hattori, H. (2017). Computational Analysis of Flow-Driven String Dynamics in Turbomachinery. *Computers & Fluids*, 142, 109–117.
- [163] Takizawa, K., Tezduyar, T.E., Otoguro, Y., Terahara, T., Kuraishi, T., & Hattori, H. (2017). Turbocharger Flow Computations with the Space–Time Isogeometric Analysis (ST-IGA). *Computers & Fluids*, 142, 15–20.
- [164] Otoguro, Y., Takizawa, K., & Tezduyar, T.E. (2017). Space–time VMS computational flow analysis with isogeometric discretization and a general-purpose NURBS mesh generation method. *Computers & Fluids*, 158, 189–200.
- [165] Otoguro, Y., Takizawa, K., & Tezduyar, T.E. (2018). A General-Purpose NURBS Mesh Generation Method for Complex Geometries. In Tezduyar, T.E. (editor), *Frontiers in Computational Fluid–Structure Interaction and Flow Simulation: Research from Lead Investigators under Forty – 2018*, Modeling and Simulation in Science, Engineering and Technology, Springer, 399–434.
- [166] Otoguro, Y., Takizawa, K., Tezduyar, T.E., Nagaoka, K., & Mei, S. (2019). Turbocharger turbine and exhaust manifold flow computation with the Space–Time Variational Multiscale Method and Isogeometric Analysis. *Computers & Fluids*, 179, 764–776.
- [167] Komiya, K., Kanai, T., Otoguro, Y., Kaneko, M., Hirota, K., Zhang, Y., Takizawa, K., Tezduyar, T.E., Nohmi, M., Tsuneda, T., Kawai, M., & Isono, M. (2019). Computational analysis of flow-driven string dynamics in a pump and residence time calculation. *IOP conference series earth and environmental science*, 240, 062014.
- [168] Kanai, T., Takizawa, K., Tezduyar, T.E., Komiya, K., Kaneko, M., Hirota, K., Nohmi, M., Tsuneda, T., Kawai, M., & Isono, M. (2019). Methods for Computation of Flow-Driven String Dynamics in a Pump and Residence Time. *Mathematical Models and Methods in Applied Sciences*, 29, 839–870.
- [169] Otoguro, Y., Takizawa, K., Tezduyar, T.E., Nagaoka, K., Avsar, R., & Zhang, Y. (2019). Space–Time VMS Flow Analysis of a Turbocharger Turbine with Isogeometric Discretization: Computations with Time-Dependent and Steady-Inflow Representations of the Intake/Exhaust Cycle. *Computational Mechanics*, 64, 1403–1419.
- [170] Kuraishi, T., Takizawa, K., & Tezduyar, T.E. (2019). Space–Time Isogeometric Flow Analysis with Built-in Reynolds-Equation Limit. *Mathematical Models and Methods in Applied Sciences*, 29, 871–904.
- [171] Aydinbakar, L., Takizawa, K., Tezduyar, T.E., & Matsuda, D. (2021). U-duct turbulent-flow computation with the ST-VMS method and isogeometric discretization. *Computational Mechanics*, 67, 823–843.

- [172] Tezduyar, T., Aliabadi, S., Behr, M., Johnson, A., & Mittal, S. (1993). Parallel Finite-Element Computation of 3D Flows. *Computer*, 26(10), 27–36.
- [173] Aydinbakar, L., Takizawa, K., Tezduyar, T.E., & Kuraishi, T. (2021). Space–Time VMS Isogeometric Analysis of the Taylor–Couette Flow. *Computational Mechanics*, 67, 1515–1541.
- [174] Tezduyar, T.E., Sathe, S., Senga, M., Aureli, L., Stein, K., & Griffin, B. (2005). Finite Element Modeling of Fluid–Structure Interactions with Space–Time and Advanced Mesh Update Techniques. In *Proceedings of the 10th International Conference on Numerical Methods in Continuum Mechanics (CD-ROM)*, Zilina, Slovakia.
- [175] Johnson, A.A. & Tezduyar, T.E. (1999). Advanced Mesh Generation and Update Methods for 3D Flow Simulations. *Computational Mechanics*, 23, 130–143.
- [176] Tezduyar, T.E., Behr, M., Mittal, S., & Johnson, A.A. (1992). Computation of Unsteady Incompressible Flows with the Finite Element Methods: Space–Time Formulations, Iterative Strategies and Massively Parallel Implementations. In *New Methods in Transient Analysis*, PVP-Vol.246/AMD-Vol.143, New York: ASME, 7–24.
- [177] Johnson, A.A. & Tezduyar, T.E. (1994). Mesh Update Strategies in Parallel Finite Element Computations of Flow Problems with Moving Boundaries and Interfaces. *Computer Methods in Applied Mechanics and Engineering*, 119, 73–94.
- [178] Stein, K., Tezduyar, T., & Benney, R. (2003). Mesh Moving Techniques for Fluid–Structure Interactions with Large Displacements. *Journal of Applied Mechanics*, 70, 58–63.
- [179] Tezduyar, T. (2001). Finite Element Interface-Tracking and Interface-Capturing Techniques for Flows with Moving Boundaries and Interfaces. In *Proceedings of the ASME Symposium on Fluid-Physics and Heat Transfer for Macro- and Micro-Scale Gas-Liquid and Phase-Change Flows (CD-ROM)*, ASME Paper IMECE2001/HTD-24206, New York, New York: ASME.
- [180] Tezduyar, T.E. (2003). Stabilized Finite Element Formulations and Interface-Tracking and Interface-Capturing Techniques for Incompressible Flows. In Hafez, M.M. (editor), *Numerical Simulations of Incompressible Flows*, New Jersey: World Scientific, 221–239.
- [181] Stein, K., Tezduyar, T.E., & Benney, R. (2004). Automatic Mesh Update with the Solid-Extension Mesh Moving Technique. *Computer Methods in Applied Mechanics and Engineering*, 193, 2019–2032.
- [182] Tezduyar, T.E., Sathe, S., Keedy, R., & Stein, K. (2006). Space–Time Finite Element Techniques for Computation of Fluid–Structure Interactions. *Computer Methods in Applied Mechanics and Engineering*, 195, 2002–2027.
- [183] Stein, K. & Tezduyar, T. (2002). Advanced Mesh Update Techniques for Problems Involving Large Displacements. In *Proceedings of the Fifth World Congress on Computational Mechanics*, On-line publication: Paper-ID: 81489, <http://www.researchgate.net/publication/303737884/>, Vienna, Austria.
- [184] Castorrini, A., Corsini, A., Rispoli, F., Venturini, P., Takizawa, K., & Tezduyar, T.E. (2019). Computational analysis of performance deterioration of a wind turbine blade strip subjected to environmental erosion. *Computational Mechanics*, 64, 1133–1153.
- [185] Castorrini, A., Venturini, P., Corsini, A., Rispoli, F., Takizawa, K., & Tezduyar, T.E. (2020). Computational Analysis of Particle-Laden-Airflow Erosion and Experimental Verification. *Computational Mechanics*, 65, 1549–1565.
- [186] Hughes, T.J.R., Cottrell, J.A., & Bazilevs, Y. (2005). Isogeometric analysis: CAD,

- finite elements, NURBS, exact geometry, and mesh refinement. *Computer Methods in Applied Mechanics and Engineering*, 194, 4135–4195.
- [187] Bazilevs, Y. & Hughes, T.J.R. (2008). NURBS-based isogeometric analysis for the computation of flows about rotating components. *Computational Mechanics*, 43, 143–150.
- [188] Takizawa, K., Takagi, H., Tezduyar, T.E., & Torii, R. (2014). Estimation of Element-Based Zero-Stress State for Arterial FSI Computations. *Computational Mechanics*, 54, 895–910.
- [189] Takizawa, K., Torii, R., Takagi, H., Tezduyar, T.E., & Xu, X.Y. (2014). Coronary arterial dynamics computation with medical-image-based time-dependent anatomical models and element-based zero-stress state estimates. *Computational Mechanics*, 54, 1047–1053.
- [190] Takizawa, K., Tezduyar, T.E., & Sasaki, T. (2018). Estimation of element-based zero-stress state in arterial FSI computations with isogeometric wall discretization. In Wriggers, P. & Lenarz, T. (editors), *Biomedical Technology: Modeling, Experiments and Simulation*, Lecture Notes in Applied and Computational Mechanics, Springer, 101–122.
- [191] Takizawa, K., Tezduyar, T.E., & Sasaki, T. (2017). Aorta modeling with the element-based zero-stress state and isogeometric discretization. *Computational Mechanics*, 59, 265–280.
- [192] Sasaki, T., Takizawa, K., & Tezduyar, T.E. (2019). Aorta Zero-Stress State Modeling with T-Spline Discretization. *Computational Mechanics*, 63, 1315–1331.
- [193] Sasaki, T., Takizawa, K., & Tezduyar, T.E. (2019). Medical-Image-Based Aorta Modeling with Zero-Stress-State Estimation. *Computational Mechanics*, 64, 249–271.
- [194] Tonon, P., Sanches, R.A.K., Takizawa, K., & Tezduyar, T.E. (2021). A linear-elasticity-based mesh moving method with no cycle-to-cycle accumulated distortion. *Computational Mechanics*, 67, 413–434.
- [195] Kiendl, J., Bazilevs, Y., Hsu, M.C., Wüchner, R., & Bletzinger, K.U. (2010). The bending strip method for isogeometric analysis of Kirchhoff–Love shell structures comprised of multiple patches. *Computer Methods in Applied Mechanics and Engineering*, 199, 2403–2416.
- [196] Kiendl, J., Hsu, M.C., Wu, M.C.H., & Reali, A. (2015). Isogeometric Kirchhoff–Love shell formulations for general hyperelastic materials. *Computer Methods in Applied Mechanics and Engineering*, 291, 280–303.
- [197] Takizawa, K., Tezduyar, T.E., & Sasaki, T. (2019). Isogeometric hyperelastic shell analysis with out-of-plane deformation mapping. *Computational Mechanics*, 63, 681–700.
- [198] Taniguchi, Y., Takizawa, K., Otaguro, Y., & Tezduyar, T.E. (2022). A hyperelastic extended Kirchhoff–Love shell model with out-of-plane normal stress: I. Out-of-plane deformation, *Computational Mechanics*, published online, DOI: 10.1007/s00466-022-02166-x.
- [199] Johnson, A.A. & Tezduyar, T.E. (1996). Simulation of Multiple Spheres Falling in a Liquid-Filled Tube. *Computer Methods in Applied Mechanics and Engineering*, 134, 351–373.

## About Authors

**Kenji TAKIZAWA** received his PhD from Tokyo Institute of Technology in 2005, and he is currently a Professor in Department of Modern Mechanical Engineering at Waseda University. He has been conducting computational fluid mechanics research since 2000,

teaching classes on that subject since 2010, and has been conducting computational FSI research since 2003. He has published over 110 Web-of-Science-indexed journal articles on computational fluid and structural mechanics and FSI. He is a Web of Science Highly Cited Researcher. He coauthored a textbook titled *Computational Fluid–Structure Interaction: Methods and Applications*, published by Wiley, with the Japanese translation published by Morikita Publishing Company. He served an Associate Editor of *ASME Journal of Applied Mechanics* and was responsible for the manuscripts on computational fluid mechanics and FSI. More information on Takizawa can be found at <http://www.jp.tafsm.org>.

**Yuri BAZILEVS** received his PhD from University of Texas at Austin in 2006, and he is currently the E. Paul Sorensen Professor of Engineering at Brown University. He has been conducting computational fluid mechanics research since 2000, teaching classes on that subject since 2008, and has been conducting computational FSI research since 2005. He has published over 170 Web-of-Science-indexed journal articles on computational fluid and structural mechanics and FSI. He is a Web of Science Highly Cited Researcher. He coauthored a book on isogeometric analysis, a technique widely used in computational mechanics, and FSI. He coauthored a textbook titled *Computational Fluid–Structure Interaction: Methods and Applications*, published by Wiley, with the Japanese translation published by Morikita Publishing Company. He is an Associate Editor of Elsevier journal *Computers & Fluids* and is responsible for the manuscripts on computational fluid mechanics and FSI. More information on Bazilevs can be found at <https://vivo.brown.edu/display/ybazilev>

**Tayfun E. TEZDUYAR** received his PhD from Caltech in 1982, and he is currently the James F. Barbour Professor of Mechanical Engineering at Rice University and Professor in Faculty of Science and Engineering at Waseda University. He has been conducting computational fluid mechanics research since 1979, teaching classes on that subject since 1987, and has been conducting computational FSI research since 1991. He has published over 260 Web-of-Science-indexed journal articles on computational fluid and structural mechanics and FSI. He is a Web of Science Highly Cited Researcher. He coauthored a textbook titled *Computational Fluid–Structure Interaction: Methods and Applications*, published by Wiley, with the Japanese translation published by Morikita Publishing Company. He is an Editor of Springer journal *Computational Mechanics* and is responsible for the manuscripts on computational fluid mechanics and FSI. More information on Tezduyar can be found at <http://www.tafsm.org/tezduyar/>.