



Ventricle-valve-aorta flow analysis with the Space–Time Isogeometric Discretization and Topology Change

Takuya Terahara¹ · Kenji Takizawa¹ · Tayfun E. Tezduyar^{2,3} · Atsushi Tsushima¹ · Kensuke Shiozaki¹

Received: 9 December 2019 / Accepted: 18 January 2020 / Published online: 20 February 2020
© The Author(s) 2020

Abstract

We address the computational challenges of and presents results from ventricle-valve-aorta flow analysis. Including the left ventricle (LV) in the model makes the flow into the valve, and consequently the flow into the aorta, anatomically more realistic. The challenges include accurate representation of the boundary layers near moving solid surfaces even when the valve leaflets come into contact, computation with high geometric complexity, anatomically realistic representation of the LV motion, and flow stability at the inflow boundary, which has a traction condition. The challenges are mainly addressed with a Space–Time (ST) method that integrates three special ST methods around the core, ST Variational Multiscale (ST-VMS) method. The three special methods are the ST Slip Interface (ST-SI) and ST Topology Change (ST-TC) methods and ST Isogeometric Analysis (ST-IGA). The ST-discretization feature of the integrated method, ST-SI-TC-IGA, provides higher-order accuracy compared to standard discretization methods. The VMS feature addresses the computational challenges associated with the multiscale nature of the unsteady flow in the LV, valve and aorta. The moving-mesh feature of the ST framework enables high-resolution computation near the leaflets. The ST-TC enables moving-mesh computation even with the TC created by the contact between the leaflets, dealing with the contact while maintaining high-resolution representation near the leaflets. The ST-IGA provides smoother representation of the LV, valve and aorta surfaces and increased accuracy in the flow solution. The ST-SI connects the separately generated LV, valve and aorta NURBS meshes, enabling easier mesh generation, connects the mesh zones containing the leaflets, enabling a more effective mesh moving, helps the ST-TC deal with leaflet–leaflet contact location change and contact sliding, and helps the ST-TC and ST-IGA keep the element density in the narrow spaces near the contact areas at a reasonable level. The ST-SI-TC-IGA is supplemented with two other special methods in this article. A structural mechanics computation method generates the LV motion from the CT scans of the LV and anatomically realistic values for the LV volume ratio. The Constrained-Flow-Profile (CFP) Traction provides flow stability at the inflow boundary. Test computation with the CFP Traction shows its effectiveness as an inflow stabilization method, and computation with the LV-valve-aorta model shows the effectiveness of the ST-SI-TC-IGA and the two supplemental methods.

Keywords Ventricle · Heart valve · Aorta · Medical image · Space–Time method · Isogeometric discretization · Inflow stabilization · T-spline discretization

1 Introduction

We address the computational challenges encountered in and present results from ventricle-valve-aorta flow analysis. We include the left ventricle (LV) in the model because that makes the flow into the valve anatomically more realistic, which, in turn, makes the flow into the aorta more realistic. The challenges include (i) accurate representation of the boundary layers near the valve leaflets as they come into contact, (ii) computation with high geometric complexity, (iii) anatomically realistic representation of the LV motion, and

✉ Kenji Takizawa
Kenji.Takizawa@tafsm.org

Tayfun E. Tezduyar
tezduyar@tafsm.org

¹ Department of Modern Mechanical Engineering, Waseda University, 3-4-1 Ookubo, Shinjuku-ku, Tokyo 169-8555, Japan

² Mechanical Engineering, Rice University – MS 321, 6100 Main Street, Houston, TX 77005, USA

³ Faculty of Science and Engineering, Waseda University, 3-4-1 Ookubo, Shinjuku-ku, Tokyo 169-8555, Japan

(iv) flow stability at the inflow boundary, where we have a traction condition.

The challenges are mainly addressed with the space–time (ST) computational method “ST-SI-TC-IGA” [1,2]. The ST-SI-TC-IGA integrates three special ST methods around the core, ST Variational Multiscale (ST-VMS) method [3–5], which subsumes its precursor “ST-SUPS” (see Sect. 2). The three special methods are the ST Slip Interface (ST-SI) [6,7] and ST Topology Change (ST-TC) [8,9] methods and the ST Isogeometric Analysis (ST-IGA) [3,10,11].

The ST-discretization feature of the ST-SI-TC-IGA provides higher-order accuracy compared to standard discretization methods. The VMS feature addresses the computational challenges associated with the multiscale nature of the unsteady flow in the LV, valve and aorta. The moving-mesh feature of the ST framework enables high-resolution computation near the leaflets. The ST-TC enables moving-mesh computation even with the TC created by the contact between the leaflets, dealing with the contact while maintaining high-resolution representation near the leaflets. The ST-IGA provides smoother representation of the LV, valve and aorta surfaces and increased accuracy in the flow solution. The ST-SI connects the separately generated LV, valve and aorta NURBS meshes, enabling easier mesh generation, and connects the mesh zones containing the leaflets, enabling a more effective mesh moving. It also helps the ST-TC deal with leaflet–leaflet contact location change and contact sliding and helps the ST-TC and ST-IGA keep the element density in the narrow spaces near the contact areas at a reasonable level.

The ST-SI-TC-IGA is supplemented with two other special methods. a) A structural mechanics computation method generates the LV motion from the CT scans of the LV and anatomically realistic values for the LV volume ratio. In this method, the structural mechanics computations, performed in different ways for the diastole and systole, generate a “table” of LV volumes and shapes. From that and the volume ratio given, we obtain the cardiac-cycle representation of the LV motion by using cubic B-splines in time and the ST-C [12]. b) The Constrained-Flow-Profile (CFP) Traction provides flow stability at the inflow boundary. This is done by placing adjacent to the inflow boundary a special-purpose element consisting of 27 basis functions. An SI connects the flow solutions over that element and the rest of the mesh. The special-purpose element, with only one unspecified control-point velocity at the inflow, results in a constrained flow profile, which is quadratic. The solution obtained for the unspecified velocity, together with the quadratic profile, represents the flow rate generated by the traction conditions specified at the inflow and outflow boundaries.

We first present a 2D-channel-flow test computation with the CFP to show how it performs compared to the analytical solution. We then present a computation with an LV-valve-

aorta model created based on the CT scans of the LV and aorta.

In Sect. 2, we provide an overview of the ST-VMS and ST-SUPS. The overviews of the ST-SI, ST-TC, ST-SI-TC, ST-IGA and ST-SI-TC-IGA are provided in Sects. 3–7. The CFP Traction is presented Sect. 8, and the 2D-channel-flow test computation with it in Sect. 9. The LV-aorta-valve flow analysis is presented in Sect. 10, with the special structural mechanics computation method that generates the LV motion described in the first subsection. The concluding remarks are given in Sect. 11. The mesh moving method used in the LV-aorta-valve flow computation is described in “Appendix A”.

2 ST-VMS and ST-SUPS

This section, included for completeness, is mostly from [13]. The Deforming-Spatial-Domain/Stabilized ST (DSD/SST) method [14–16] was introduced for computation of flows with moving boundaries and interfaces (MBI), including fluid–structure interaction (FSI). In MBI computations the DSD/SST functions as a moving-mesh method. Moving the fluid mechanics mesh to follow an interface enables mesh-resolution control near the interface and, consequently, high-resolution boundary-layer representation near fluid–solid interfaces. Because the stabilization components of the original DSD/SST are the Streamline-Upwind/Petrov-Galerkin (SUPG) [17] and Pressure-Stabilizing/Petrov-Galerkin (PSPG) [14] stabilizations, it is now called “ST-SUPS”. The ST-VMS is the VMS version of the DSD/SST. The VMS components of the ST-VMS are from the residual-based VMS (RBVMS) method [18–21]. The ST-VMS has two more stabilization terms beyond those in the ST-SUPS, and the additional terms give the method better turbulence modeling features. The ST-SUPS and ST-VMS, because of the higher-order accuracy of the ST framework (see [3,4]), are desirable also in computations without MBI.

As a moving-mesh method, the DSD/SST is an alternative to the Arbitrary Lagrangian–Eulerian (ALE) method, which is older (see, for example, [22]) and more commonly used. The ALE-VMS method [23–29] is the VMS version of the ALE. It succeeded the ST-SUPS and ALE-SUPS [30] and preceded the ST-VMS. To increase their scope and accuracy, the ALE-VMS and RBVMS are often supplemented with special methods, such as those for weakly-enforced Dirichlet boundary conditions [31–33], “sliding interfaces” [34,35], and backflow stabilization [36]. The ALE-SUPS, RBVMS and ALE-VMS have been applied to many classes of FSI, MBI and fluid mechanics problems. The classes of problems include ram-air parachute FSI [30], wind-turbine aerodynamics and FSI [37–47], more specifically, vertical-axis wind turbines [46–49], floating wind turbines [50], wind turbines in atmospheric boundary layers [45–47,51], and fatigue

damage in wind-turbine blades [52], patient-specific cardiovascular fluid mechanics and FSI [23,53–58], biomedical-device FSI [59–64], ship hydrodynamics with free-surface flow and fluid–object interaction [65,66], hydrodynamics and FSI of a hydraulic arresting gear [67,68], hydrodynamics of tidal-stream turbines with free-surface flow [69], passive-morphing FSI in turbomachinery [70], bioinspired FSI for marine propulsion [71,72], bridge aerodynamics and fluid–object interaction [73–75], and mixed ALE-VMS/Immersogeometric computations [62–64,76,77] in the framework of the Fluid–Solid Interface-Tracking/Interface-Capturing Technique [78]. Recent advances in stabilized and multiscale methods may be found for stratified incompressible flows in [79], for divergence-conforming discretizations of incompressible flows in [80], and for compressible flows with emphasis on gas-turbine modeling in [81].

The ST-SUPS and ST-VMS have also been applied to many classes of FSI, MBI and fluid mechanics problems (see [82] for a comprehensive summary). The classes of problems include spacecraft parachute analysis for the landing-stage parachutes [26,83–86], cover-separation parachutes [87] and the drogue parachutes [88–90], wind-turbine aerodynamics for horizontal-axis wind-turbine rotors [26,37,91,92], full horizontal-axis wind-turbines [43,93–95] and vertical-axis wind-turbines [6,46,47], flapping-wing aerodynamics for an actual locust [10,26,96,97], bioinspired MAVs [94,95,98,99] and wing-clapping [8,100], blood flow analysis of cerebral aneurysms [94,101], stent-blocked aneurysms [101–103], aortas [104–108], heart valves [1,8,9,95,106,108–111] and coronary arteries in motion [112], spacecraft aerodynamics [87,113], thermo-fluid analysis of ground vehicles and their tires [5,109], thermo-fluid analysis of disk brakes [7], flow-driven string dynamics in turbomachinery [114–116], flow analysis of turbocharger turbines [11,117–120], flow around tires with road contact and deformation [2,13,109,121,122], fluid films [13,123], ram-air parachutes [124], and compressible-flow spacecraft parachute aerodynamics [125,126].

The ST-SUPS, ALE-SUPS, RBVMS, ALE-VMS and ST-VMS all have some embedded stabilization parameters that play a significant role (see [26]). These parameters involve a measure of the local length scale (also known as “element length”) and other parameters such as the element Reynolds and Courant numbers. There are many ways of defining the stabilization parameters. Some of the newer options for the stabilization parameters used with the SUPS and VMS can be found in [2,5,6,10,92,93,127–130]. Some of the earlier stabilization parameters used with the SUPS and VMS were also used in computations with other SUPG-like methods, such as the computations reported in [70,131–142]. We will specify which ones we use here when we describe the computations in Sects. 9 and 10.

For more on the ST-VMS and ST-SUPS, see [26]. In the flow analyses presented here, the ST discretization provides higher-order accuracy compared to standard discretization methods. The VMS and SUPS features of the ST-VMS and ST-SUPS address the computational challenges associated with the multiscale nature of the unsteady flow in the LV, valve and aorta. The moving-mesh feature of the ST framework enables high-resolution computation near the leaflets.

3 ST-SI

This section, included for completeness, is mostly from [13]. The ST-SI was introduced in [6], in the context of incompressible-flow equations, to retain the desirable moving-mesh features of the ST-VMS and ST-SUPS in computations involving spinning solid surfaces, such as a turbine rotor. The mesh covering the spinning surface spins with it, retaining the high-resolution representation of the boundary layers, while the mesh on the other side of the SI remains unaffected. This is accomplished by adding to the ST-VMS formulation interface terms similar to those in the version of the ALE-VMS for computations with sliding interfaces [34,35]. The interface terms account for the compatibility conditions for the velocity and stress at the SI, accurately connecting the two sides of the solution. An ST-SI version where the SI is between fluid and solid domains was also presented in [6]. The SI in that case is a “fluid–solid SI” rather than a standard “fluid–fluid SI” and enables weak enforcement of the Dirichlet boundary conditions for the fluid. The ST-SI introduced in [7] for the coupled incompressible-flow and thermal-transport equations retains the high-resolution representation of the thermo-fluid boundary layers near spinning solid surfaces. These ST-SI methods have been applied to aerodynamic analysis of vertical-axis wind turbines [6,46,47], thermo-fluid analysis of disk brakes [7], flow-driven string dynamics in turbomachinery [114–116], flow analysis of turbocharger turbines [11,117–120], flow around tires with road contact and deformation [2,13,109,121,122], fluid films [13,123], aerodynamic analysis of ram-air parachutes [124], and flow analysis of heart valves [1,106,108,110,111].

In the ST-SI version presented in [6] the SI is between a thin porous structure and the fluid on its two sides. This enables dealing with the porosity in a fashion consistent with how the standard fluid–fluid SIs are dealt with and how the Dirichlet conditions are enforced weakly with fluid–solid SIs. This version also enables handling thin structures that have T-junctions. This method has been applied to incompressible-flow aerodynamic analysis of ram-air parachutes with fabric porosity [124]. The compressible-flow ST-SI methods were introduced in [125], including the version where the SI is between a thin porous structure and the fluid on its two sides.

Compressible-flow porosity models were also introduced in [125]. These, together with the compressible-flow ST SUPG method [143], extended the ST computational analysis range to compressible-flow aerodynamics of parachutes with fabric and geometric porosities. That enabled ST computational flow analysis of the Orion spacecraft drogue parachute in the compressible-flow regime [125,126].

For more on the ST-SI, see [6,7]. In the CFP Traction, the ST-SI connects the special-purpose element placed adjacent to the inflow boundary to the rest of the mesh, and we will describe the CFP Traction in Sect. 8. In the LV-aorta-valve flow analysis, the ST-SI connects the separately generated LV, valve and aorta NURBS meshes, enabling easier mesh generation, connects the three mesh zones of the valve, enabling a more effective mesh moving, and connects the two separately generated parts of the LV mesh, again enabling easier mesh generation. By integrating it with the ST-TC, we deal with leaflet–leaflet contact location change and contact sliding, and we will describe the ST-SI-TC in Sect. 5. By integrating it with the ST-TC and ST-IGA, we keep the element density in the narrow spaces near the contact areas at a reasonable level, and we will describe the ST-IGA and ST-SI-TC-IGA in Sects. 6 and 7.

4 ST-TC

This section, included for completeness, is mostly from [13]. The ST-TC [8,9] was introduced for moving-mesh computation of flow problems with TC, such as contact between solid surfaces. Even before the ST-TC, the ST-SUPS and ST-VMS, when used with robust mesh update methods, have proven effective in flow computations where the solid surfaces are in near contact or create other near TC, if the nearness is sufficiently near for the purpose of solving the problem. Many classes of problems can be solved that way with sufficient accuracy. For examples of such computations, see the references mentioned in [8]. The ST-TC made moving-mesh computations possible even when there is an actual contact between solid surfaces or other TC. By collapsing elements as needed, without changing the connectivity of the “parent” mesh, the ST-TC can handle an actual TC while maintaining high-resolution boundary layer representation near solid surfaces. This enabled successful moving-mesh computation of heart valve flows [1,8,9,95,106,108–111], wing clapping [8,100], flow around a rotating tire with road contact and prescribed deformation [2,13,109,121,122], and fluid films [13,123].

For more on the ST-TC, see [8,9]. In the LV-aorta-valve flow analysis, the ST-TC enables moving-mesh computation even with the TC created by the contact between the leaflets, dealing with the contact while maintaining high-resolution representation near the leaflets.

5 ST-SI-TC

This section, included for completeness, is mostly from [13]. The ST-SI-TC is the integration of the ST-SI and ST-TC. A fluid–fluid SI requires elements on both sides of the SI. When part of an SI needs to coincide with a solid surface, which happens for example when the solid surfaces on two sides of an SI come into contact or when an SI reaches a solid surface, the elements between the coinciding SI part and the solid surface need to collapse with the ST-TC mechanism. The collapse switches the SI from fluid–fluid SI to fluid–solid SI. With that, an SI can be a mixture of fluid–fluid and fluid–solid SIs. With the ST-SI-TC, the elements collapse and are reborn independent of the nodes representing a solid surface. The ST-SI-TC enables high-resolution flow representation even when parts of the SI are coinciding with a solid surface. It also enables dealing with contact location change and contact sliding. This was applied to heart valve flow analysis [1,106,108,110,111], tire aerodynamics with road contact and deformation [2,13,109,121,122], and fluid films [13,123].

For more on the ST-SI-TC, see [1,121]. In the computational analyses presented here, the ST-SI-TC enables contact location change and contact sliding between the leaflets.

6 ST-IGA

This section, included for completeness, is mostly from [13]. The success with IGA basis functions in space [23,34,53,144] motivated the integration of the ST methods with isogeometric discretization, which we broadly call “ST-IGA.” The ST-IGA was introduced in [3]. Computations with the ST-VMS and ST-IGA were first reported in [3] in a 2D context, with IGA basis functions in space for flow past an airfoil, and in both space and time for the advection equation. Using higher-order basis functions in time enables deriving full benefit from using higher-order basis functions in space. This was demonstrated with the stability and accuracy analysis given in [3] for the advection equation.

The ST-IGA with IGA basis functions in time enables a more accurate representation of the motion of the solid surfaces and a mesh motion consistent with that. This was pointed out in [3,4] and demonstrated in [10,96,98]. It also enables more efficient temporal representation of the motion and deformation of the volume meshes, and more efficient remeshing. These motivated the development of the ST/NURBS Mesh Update Method (STNMUM) [10,96,98], with the name coined in [93]. The STNMUM has a wide scope that includes spinning solid surfaces. With the spinning motion represented by quadratic NURBS in time, and with sufficient number of temporal patches for a full rotation, the circular paths are represented exactly. A “secondary

mapping” [3,4,10,26] enables also specifying a constant angular velocity for invariant speeds along the circular paths. The ST framework and NURBS in time also enable, with the “ST-C” method, extracting a continuous representation from the computed data and, in large-scale computations, efficient data compression [5,7,12,109,114–116]. The STN-MUM and the ST-IGA with IGA basis functions in time have been used in many 3D computations. The classes of problems solved are flapping-wing aerodynamics for an actual locust [10,26,96,97], bioinspired MAVs [94,95,98,99] and wing-clapping [8,100], separation aerodynamics of spacecraft [87], aerodynamics of horizontal-axis [43,93–95] and vertical-axis [6,46,47] wind-turbines, thermo-fluid analysis of ground vehicles and their tires [5,109], thermo-fluid analysis of disk brakes [7], flow-driven string dynamics in turbomachinery [114–116], flow analysis of turbocharger turbines [11,117–120], and flow analysis of coronary arteries in motion [112].

The ST-IGA with IGA basis functions in space enables more accurate representation of the geometry and increased accuracy in the flow solution. It accomplishes that with fewer control points, and consequently with larger effective element sizes. That in turn enables using larger time-step sizes while keeping the Courant number at a desirable level for good accuracy. It has been used in ST computational flow analysis of turbocharger turbines [11,117–120], flow-driven string dynamics in turbomachinery [115,116], ram-air parachutes [124], spacecraft parachutes [126], aortas [106–108], heart valves [1,106,108,110,111], coronary arteries in motion [112], tires with road contact and deformation [2,13,122], and fluid films [13,123]. Using IGA basis functions in space is now a key part of some of the newest arterial zero-stress-state (ZSS) estimation methods [108,145–150] and related shell analysis [151].

For more on the ST-IGA, see [11,26,96,124]. In the flow analyses presented here, the ST-IGA provides smoother representation of the LV, valve and aorta surfaces and increased accuracy in the flow solution.

7 ST-SI-IGA and ST-SI-TC-IGA

This section, included for completeness, is mostly from [13]. The ST-SI-IGA is the integration of the ST-SI and ST-IGA, and the ST-SI-TC-IGA is the integration of the ST-SI, ST-TC and ST-IGA. The turbocharger turbine flow [11,117–120] and flow-driven string dynamics in turbomachinery [115,116] were computed with the ST-SI-IGA. The IGA basis functions were used in the spatial discretization of the fluid mechanics equations and also in the temporal representation of the rotor and spinning-mesh motion. That enabled accurate representation of the turbine geometry and rotor motion and increased accuracy in the flow solution. The

IGA basis functions were used also in the spatial discretization of the string structural dynamics equations. That enabled increased accuracy in the structural dynamics solution, as well as smoothness in the string shape and fluid dynamics forces computed on the string.

The ram-air parachute analysis [124] and spacecraft parachute compressible-flow analysis [126] were conducted with the ST-SI-IGA, based on the ST-SI version that weakly enforces the Dirichlet conditions and the ST-SI version that accounts for the porosity of a thin structure. The ST-IGA with IGA basis functions in space enabled, with relatively few number of unknowns, accurate representation of the parafoil and parachute geometries and increased accuracy in the flow solution. The volume mesh needed to be generated both inside and outside the parafoil. Mesh generation inside was challenging near the trailing edge because of the narrowing space. The spacecraft parachute has a very complex geometry, including gores and gaps. Using IGA basis functions addressed those challenges and still kept the element density near the trailing edge of the parafoil and around the spacecraft parachute at a reasonable level.

The heart valve flow analysis [1,106,108,110,111] was conducted with the ST-SI-TC-IGA. The method, beyond enabling a more accurate representation of the geometry and increased accuracy in the flow solution, kept the element density in the narrow spaces near the contact areas at a reasonable level. When solid surfaces come into contact, the elements between the surface and the SI collapse. Before the elements collapse, the boundaries could be curved and rather complex, and the narrow spaces might have high-aspect-ratio elements. With NURBS elements, it was possible to deal with such adverse conditions rather effectively.

In computational analysis of flow around tires with road contact and deformation [2,13], the ST-SI-TC-IGA enabled a more accurate representation of the geometry and motion of the tire surfaces, a mesh motion consistent with that, and increased accuracy in the flow solution. It also kept the element density in the tire grooves and in the narrow spaces near the contact areas at a reasonable level. In addition, we benefit from the mesh generation flexibility provided by using SIs.

In computational analysis of fluid films [13,123], the ST-SI-TC-IGA enabled solution with a computational cost comparable to that of the Reynolds-equation model for the comparable solution quality [123]. With that, narrow gaps associated with the road roughness [13] can be accounted for in the flow analysis around tires.

An SI provides mesh generation flexibility in a general context by accurately connecting the two sides of the solution computed over nonmatching meshes. This type of mesh generation flexibility is especially valuable in complex-geometry flow computations with isogeometric discretization, removing the matching requirement between the NURBS patches without loss of accuracy. This feature was used in the

flow analysis of heart valves [1,106,108,110,111], turbocharger turbines [11,117–120], and spacecraft parachute compressible-flow analysis [126].

For more on the ST-SI-TC-IGA, see [1,2]. In the computations presented here, the ST-SI-TC-IGA is used for the reasons given and as described in the earlier paragraphs of this section.

8 Constrained-Flow-Profile traction

In computational flow analysis, typically, we specify the velocity at the inflow boundary, and the traction at the outflow boundary. Sometimes, however, we may want to specify the traction at both the inflow and outflow. To discuss how that would work, we first write the expressions for the flow rate, influx energy at the inflow, and the outflux energy at the outflow:

$$Q = - \int_{\Gamma_{\text{in}}} \mathbf{n} \cdot \mathbf{u} d\Gamma, \quad (1)$$

$$P_{\text{in}} = - \int_{\Gamma_{\text{in}}} \mathbf{n} \cdot \mathbf{u} \left(\frac{1}{2} \rho \|\mathbf{u}\|^2 + p \right) d\Gamma, \quad (2)$$

$$P_{\text{out}} = \int_{\Gamma_{\text{out}}} \mathbf{n} \cdot \mathbf{u} \left(\frac{1}{2} \rho \|\mathbf{u}\|^2 + p \right) d\Gamma. \quad (3)$$

Here \mathbf{u} is the velocity, \mathbf{n} is the unit normal vector, Γ_{in} and Γ_{out} are the inflow and outflow boundaries, ρ is the density, and p is the pressure. We note that a spatially-constant pressure value of p_0 can be added to or subtracted from the pressure field without changing the energy balance.

Even though the flow rate would correspond to the pressure difference between the inflow and outflow, without specifying the full velocity profile at the inflow, the energy coming into the domain would not be under control. With the energy influx not being under control, for the same flow rate we could get different flow fields. An uncontrolled energy influx can also occur at the outflow if the flow reverses, and an outflow stabilization method was introduced [36] to remedy that. The method basically removes the incoming portion of the energy flux.

Targeting the inflow boundary, we introduce a simple method: Constrained-Flow-Profile (CFP) traction. Figure 1 illustrates the concept. We place adjacent to the inflow boundary a special-purpose element consisting of $3^{n_{\text{sd}}}$ basis functions, where n_{sd} is the number of space dimensions. An SI connects the flow solutions over that element and the rest of the mesh. The special-purpose element, with only one unspecified control-point velocity at the inflow, results in a constrained flow profile, which is quadratic. The solution obtained for the unspecified velocity, together with the quadratic profile, represents the flow rate generated by the traction conditions specified at the inflow and outflow bound-

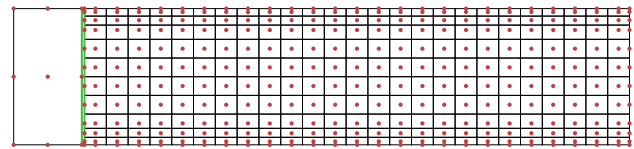


Fig. 1 A quadratic NURBS mesh illustrating the Constrained-Flow-Profile (CFP) Traction. The left edge is the inflow boundary. The dots are the control points and the thin lines are the element boundaries. The pair of thicker lines represent the SI connecting the flow solutions over a special-purpose element consisting of 9 basis functions and the rest of the mesh. The special-purpose element, with only one unspecified control-point velocity at the inflow, results in a constrained flow profile, which is quadratic. The solution obtained for the unspecified velocity, together with the quadratic profile, represents the flow rate corresponding to the traction conditions specified at the inflow and outflow boundaries

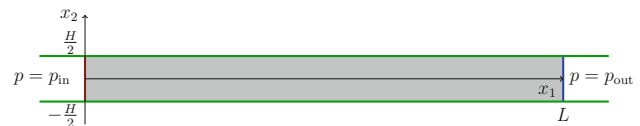


Fig. 2 A 2D straight channel. The problem setup

aries. As shown in [123], having a single quadratic NURBS element across a narrow channel is enough to represent the flow, which will be a low-Reynolds-number flow, and what we do here was inspired by that.

9 Test computation with the CFP

Using a 2D channel flow as the test case, we compare the solution obtained with the CFP to the analytical solution and the solution obtained with the outflow stabilization method [36] applied to the inflow, which we will identify with the acronym “OS.” For completeness, we provide from [36] the outflow stabilization term, written in the ST framework:

$$-\beta \int_{P_{\text{OS}}} \mathbf{w}^h \cdot \left\{ \mathbf{n} \cdot (\mathbf{u}^h - \mathbf{v}^h) \right\}_- \rho \mathbf{u}^h dP. \quad (4)$$

Here P_{OS} is the part of the ST-slice lateral boundary where we apply the outflow stabilization method, \mathbf{v} is the velocity of the lateral boundary, \mathbf{w} is the test function, and a superscript “ h ” indicates that the function is coming from a finite-dimensional space. The function $\{A\}_- = \frac{A-|A|}{2}$. The parameter $\beta \geq \frac{1}{2}$ for stability [27].

9.1 Problem setup and analytical solution

The symbols H and L represent the channel height and length, and p_{in} and p_{out} the inflow and outflow pressures (see Fig. 2). At low Reynolds number, the solution is known

to be of the form

$$u_1(x_2) = -\frac{1}{2\mu} \frac{dp}{dx_1} \left(\left(\frac{H}{2} \right)^2 - x_2^2 \right), \tag{5}$$

where μ is the viscosity and $-\frac{H}{2} \leq x_2 \leq \frac{H}{2}$, and the pressure gradient can be expressed as

$$\frac{dp}{dx_1} = -\frac{p_{in} - p_{out}}{L}. \tag{6}$$

From that, the flow rate is

$$Q = \frac{H^3 (p_{in} - p_{out})}{12\mu L}, \tag{7}$$

and the energy influx

$$P_{in} = \rho \frac{H^7}{280} \left(\frac{p_{in} - p_{out}}{2\mu L} \right)^3 + p_{in} Q, \tag{8}$$

$$= \alpha \frac{1}{2} \rho \left(\frac{Q}{H} \right)^3 H + p_{in} Q, \tag{9}$$

where $\alpha = \frac{54}{35}$, which is the kinetic energy factor. Depending on the flow conditions, the factor will change as a function of the Reynolds number

$$Re = \frac{\rho Q}{\mu}. \tag{10}$$

The total energy in the domain is

$$\int_{\Omega} p_{total} d\Omega = \int_{\Omega} \frac{1}{2} \rho \|\mathbf{u}\|^2 d\Omega + \int_{\Omega} p d\Omega \tag{11}$$

$$= \frac{1}{60} \rho L H^5 \left(\frac{1}{2\mu} \frac{dp}{dx_1} \right)^2 + \frac{p_{in} + p_{out}}{2} L H \tag{12}$$

$$= \frac{6}{5} \frac{1}{2} \rho \left(\frac{Q}{H} \right)^2 L H + \frac{p_{in} + p_{out}}{2} L H, \tag{13}$$

and the average over the domain

$$\bar{p}_{total} = \frac{6}{5} \frac{1}{2} \rho \left(\frac{Q}{H} \right)^2 + \frac{p_{in} + p_{out}}{2}. \tag{14}$$

9.2 Meshes and boundary conditions

We set $H = 1.3 \times 10^{-2}$ m, $L = 1.365 \times 10^{-1}$ m, $\rho = 1050$ kg/m³, $\mu = 4.2 \times 10^{-3}$ Pa s, $p_{in} = 1.31$ Pa, and $p_{out} = 0$. The boundary conditions are no-slip on the channel walls and traction conditions corresponding to p_{in} and p_{out} at the inflow and outflow. We use two kinds of meshes, a regular mesh for the OS, and the kind shown in Fig. 1 for the CFP.



Fig. 3 Meshes for the OS (top) and CFP (bottom). The lines are the element boundaries. Except for the part of the domain where the CFP mesh has only one element, the two meshes have identical refinements

Table 1 Number of control points (nc) and elements (ne) for the two quadratic NURBS meshes

	nc	ne
OS	9472	8820
CFP	9033	8401

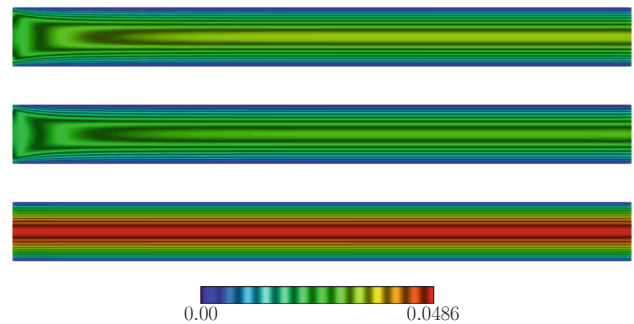


Fig. 4 Velocity magnitude (m/s) for the OS ($\beta = 0.5$), OS ($\beta = 1.0$), and CFP

Except for the part of the domain where the CFP mesh has only one element, the two meshes have identical refinements, with increasing mesh refinement in the normal direction as we get closer and closer to the walls. Figure 3 shows the meshes. The number of control points and elements for the two meshes are given in Table 1.

9.3 Computational conditions

We start the computations with zero velocity, and suddenly apply the traction boundary conditions corresponding to p_{in} and p_{out} . We use the ST-VMS, with the stabilization parameters given by Eqs. (4)–(9) in [2]. The time-step size is 8.6×10^{-3} s. The number of nonlinear iterations per time step is 3, and the number of GMRES iterations per nonlinear iteration is 300.

9.4 Results

Without the CFP or OS method, the solution quickly diverges. Figure 4 shows the velocity magnitude for the OS and CFP. We note that in both the CFP and OS computations, at the outflow boundary we use the outflow stabilization method [36] with $\beta = 0.5$. The maximum velocity from the analytical solution is 4.84×10^{-2} m/s. With the CFP, we achieve, as

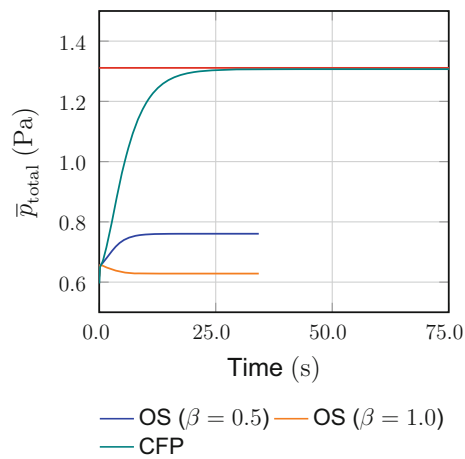


Fig. 5 Time evolution of \bar{p}_{total} as it reaches its steady value. The straight red line represents the analytical solution. (Color figure online)

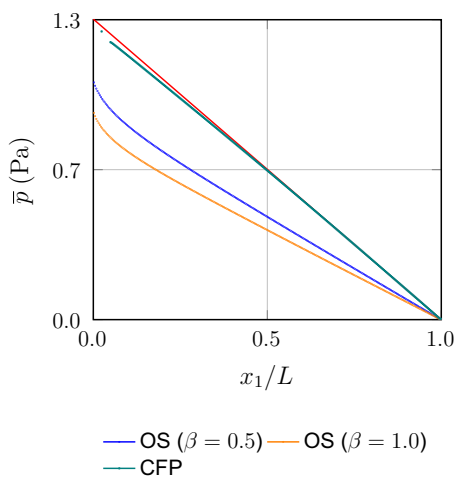


Fig. 6 Cross-sectional average pressure, evaluated at the element centers. The straight red line represents the analytical solution. (Color figure online)

expected, a parabolic profile at the inflow boundary, which is the form in the analytical solution.

Figure 5 shows \bar{p}_{total} . Figure 6 shows the cross-sectional average pressure. In the OS computation the outflow stabilization term cancels the kinetic energy influx, and that causes the substantial discrepancy between the computed pressure and p_{in} . In the CFP computation the discrepancy is very small. The discrepancy could be because of the sudden transition from a mesh with a single element to a mesh with many elements, creating a large discrepancy between the integration accuracies on the two sides of the SI. This can be remedied by having additional SIs and thus having a milder transition.

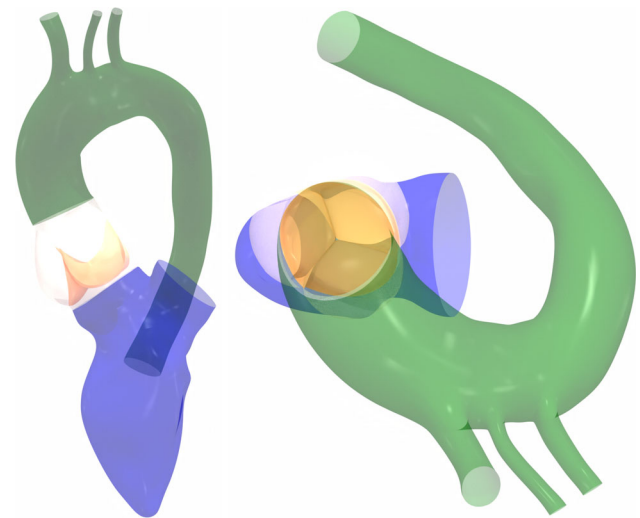


Fig. 7 LV-valve-aorta model. LV (blue), leaflets (orange), and aorta (green). The left picture shows the front view, and the right picture shows the view along the valve axis. (Color figure online)

10 Ventricle-valve-aorta flow analysis

The flow computation model consists of the LV, aortic valve with sinuses, and the aorta. We do not include the mitral valve in the model. The boundary between the LV and the left atrium becomes our inflow boundary. We use the CFP Traction there when the mitral valve is open, and zero-velocity when it is closed. The aorta main outlet is treated as a regular outflow boundary with traction condition, and the three smaller outlets are treated as prescribed-velocity outflow boundaries.

10.1 Geometry

The entire model is shown in Fig. 7. The quadratic NURBS meshes for the three parts are generated separately and the SIs connect the three solution parts. The LV shrinks and expands with a cardiac cycle of $T = 0.9$ s, the valve opens and closes, and the aorta remains stationary. We describe the three parts more in the next three subsections.

10.1.1 LV geometry and motion

We build the LV geometry and motion based on CT scans of the LV at an instant in the cardiac cycle and anatomically realistic values for the volume ratio defined as

$$\frac{V - V_{\min}}{V_{\max} - V_{\min}} \quad (15)$$

and ejection fraction (EF) defined as

$$\text{EF} = \frac{V_{\max} - V_{\min}}{V_{\max}}. \quad (16)$$

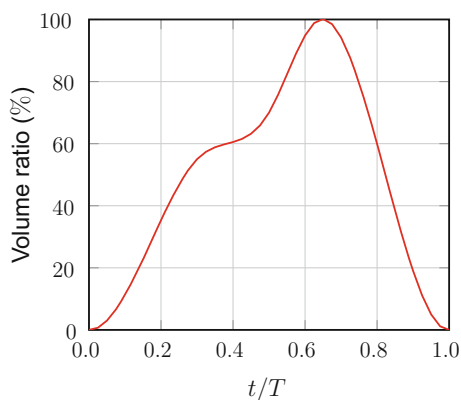


Fig. 8 Volume ratio from [152], represented by cubic B-splines in time

The volume ratio is given in Fig. 8 (from [152]), EF = 70 % (from [153]), and we set $V_{\max} = 107 \text{ ml}$.

By starting from the CT geometry and using a shell model, we perform a sequence of structural mechanics computations to obtain the cardiac-cycle LV shapes used in building the motion. In the diastole, using the CT geometry as the ZSS configuration, we apply a gradually-increasing spatially-uniform pressure until the LV volume reaches V_{\max} . In the structural mechanics computations performed in the systole, we start from the CT geometry with zero-pressure load. The ZSS configurations are defined by applying to the CT geometry a factor expressed as λ^{-2} , where λ is the uniform stretch, with gradually-decreasing values of λ . This is done by applying the factor to the metric tensor of the ZSS (see Section 2.2 in [149]). We decrease the stretch until the LV volume comes down to V_{\min} .

The structural mechanics computations generate a “table” of LV volumes and shapes. From that and the volume ratio given in Fig. 8, we obtain the cardiac-cycle representation of the LV motion by using cubic B-splines in time and the ST-C [12]. The cycle periodicity is assured with the procedure followed for the same purpose in [98,112].

The shell formulation is from [151], and the thickness is variable, as shown in Fig. 9. Fung material model is used with the exponent factor $D_2 = 8.365$, and the pressure applied is scaled as $p^* = \frac{p}{2D_1D_2}$. In the computations, we fix some of the control points, shown in Fig. 10, to obtain anatomically realistic shapes.

Figure 11 shows the LV at its most expanded stage. Figure 12 shows how the LV volume varies with the applied pressure. Figure 13 shows the LV at its most shrunk stage. Figure 14 shows how the LV volume varies as the stretch decreases. We truncate the LV structure, roughly where the arrows in Fig. 10 indicate, to create the actual LV part used in the fluid mechanics computation. The truncated segment still provides us some guidance in building the valve part and the segment of the aorta part connecting to the valve.

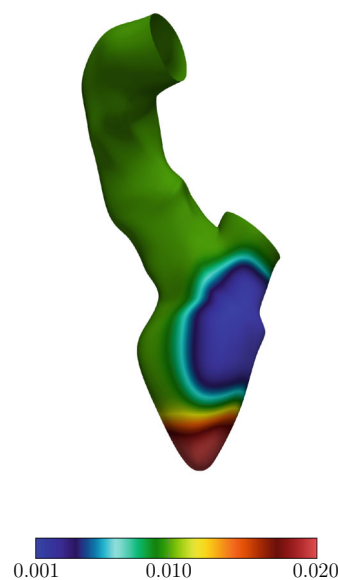


Fig. 9 Shell thickness distribution (mm)

Fig. 10 LV geometry and the control points. The lines are the element boundaries. Red points remain fixed during the structural mechanics computations. The arrows indicate roughly where we truncate the LV structure to create the actual LV part used in the fluid mechanics computation

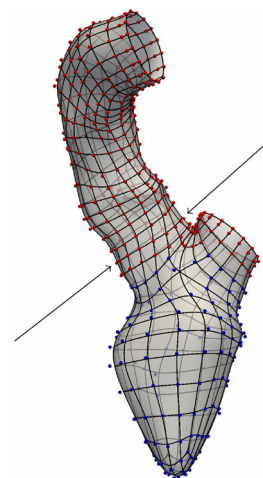


Fig. 11 LV at its most expanded stage



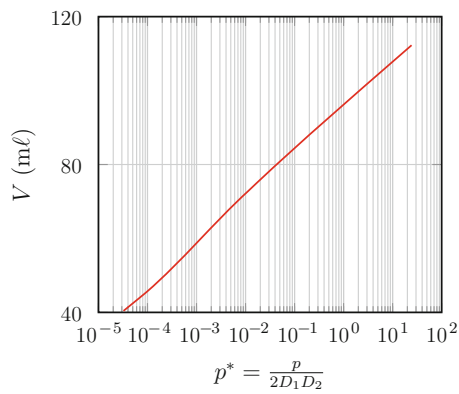


Fig. 12 LV volume variation with the applied pressure

Fig. 13 LV at its most shrunk stage

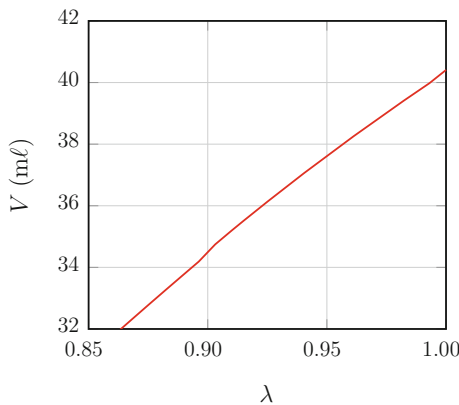


Fig. 14 LV volume variation with the stretch

The starting fluid mechanics volume mesh is generated using the method described in [117]. Then the mesh is moved to conform to the LV shapes obtained in the structural mechanics computations. The mesh moving method is the one described in “Appendix A”, with $(J_M)_0 = 1$ and $\chi = 1$. The mesh moving generates a “table” of LV volumes and fluid mechanics volume meshes. From that and

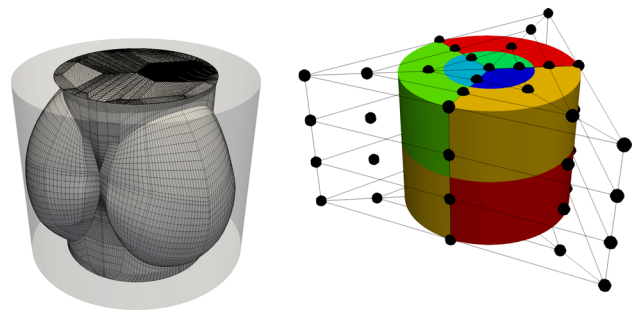


Fig. 15 Valve-mesh motion with the SSDM. Template mesh (left) and simple shape (right) prior to the transformation. The lines are the element boundaries in the template mesh and control lines in the simple shape

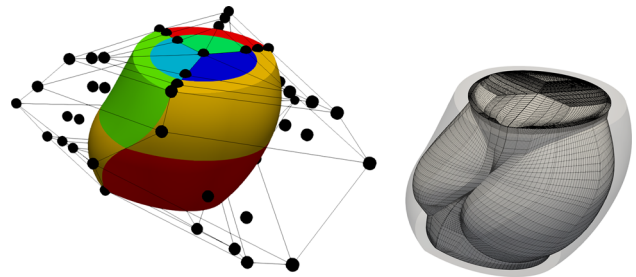


Fig. 16 Valve-mesh motion with the SSDM. Simple shape (left) and template mesh (right) after the transformation. The lines are the element boundaries in the template mesh and control lines in the simple shape

the volume ratio given in Fig. 8, we obtain the cardiac-cycle representation of the fluid mechanics mesh motion by using cubic B-splines in time and the ST-C.

10.1.2 Valve motion

The valve-mesh motion is obtained by transformation from the mesh motion used in the valve computation reported in [106]. We note that the mesh motion in [106] is based on the ST-SI-TC-IGA, which deals with the TC created by the contact between the leaflets while maintaining high-resolution representation near them. The transformation is achieved with the simple-shape deformation model (SSDM) [96], and in that context the mesh motion from [106] serves as a template. The process, illustrated in Figs. 15 and 16, starts with the template mesh. A simple shape, represented by far fewer control points, is created to enclose the template mesh. Then the simple shape transforms to have its upper and lower surfaces match the aorta and LV geometries. From that, the transformation of the template mesh is obtained by projection. The transformation with the SSDM is applied also to the motion of the valve leaflets, which are modeled in the template mesh, generating the cardiac-cycle representation of the leaflet motion by linear functions in time. The

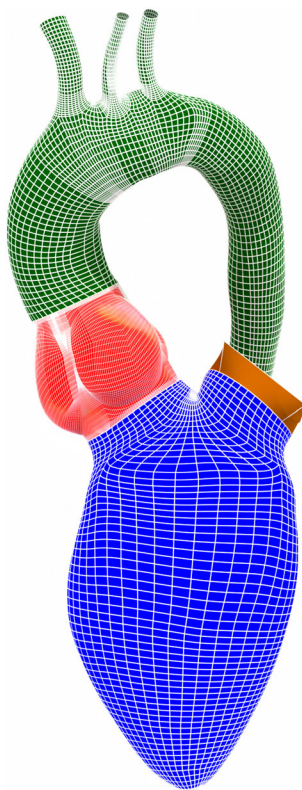


Fig. 17 The three mesh parts and the special-purpose element (brown) added for the CFP Traction. The lines are the element boundaries

projection takes place as many times as the number of flow-computation time steps in the cardiac cycle.

10.1.3 Aorta

The aorta geometry is based on a different set of CT scans and is represented by cubic T-splines. From that, the fluid mechanics volume mesh is generated using the method described in [117]. The aorta part remains rigid during the fluid mechanics computation.

10.1.4 Mesh, boundary conditions, and blood properties

To the mesh composed of the three parts, we add, for the CFP Traction, an SI and a special-purpose element consisting of 27 basis functions. Figure 17 shows the entire mesh. The number of control points and elements are given in Table 2. Figure 18 shows the SIs and boundary conditions. There are altogether 7 SIs. Two connect the three main parts of the mesh, one connects the special-purpose element to the rest of the mesh, three connect the three zones of the valve mesh, and one is for mesh generation convenience in the LV. The boundary condition on the valve surfaces and arterial walls is no-slip. The tractions specified at the inflow and outflow boundaries correspond to $p_{in} = 10.5$ kPa and

Table 2 Number of control points (nc) and elements (ne) for the mesh parts

	nc	ne
LV	51,920	36,707
Valve	558,318	420,000
Aorta	166,736	115,848
CFP	27	1
Total	777,001	572,556

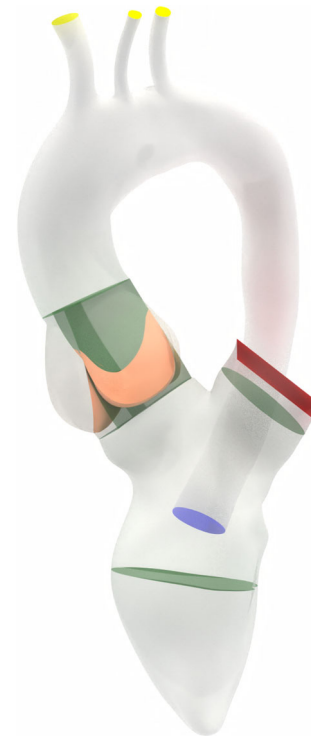


Fig. 18 SIs (green) and boundary conditions. CFP Traction inflow (red), traction outflow (blue), and prescribed-velocity outflow (yellow). (Color figure online)

$p_{out} = 0$. The estimated flow rates at the prescribed-velocity outflow boundaries come from the Murray’s law [154], a flow rate distribution proportional to D^3 , where D is the average diameter of the outflow cross-section (see [107]). We set $\rho = 1050$ kg/m³ and $\mu = 4.2 \times 10^{-3}$ Pa s. Figure 19 shows the mesh at different instants in the cardiac cycle. Figures 20 and 21 show the mesh in the valve at the same instants.

10.2 Computational conditions

We use the ST-SUPS, with the stabilization parameters given by Eqs. (4)–(9) in [2], and the ST-SI-TC-IGA. The time-step size is 2.81×10^{-3} s. The number of nonlinear iterations per time step is 3, and the number of GMRES iterations per nonlinear iteration is 300.

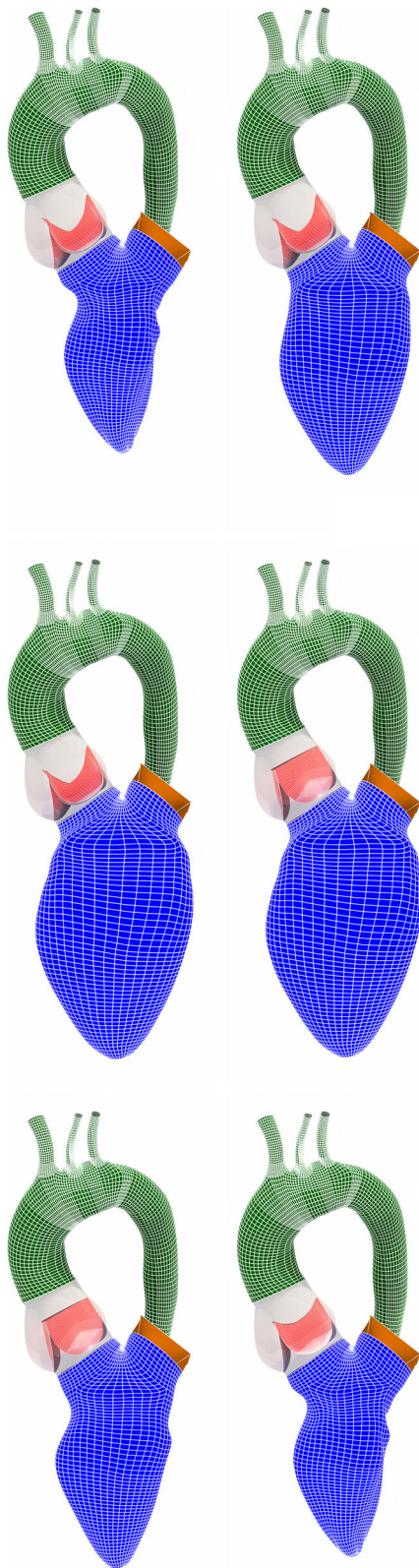


Fig. 19 Mesh at different instants in the cardiac cycle. The frames (from left to right and top to bottom) are for $t/T = 0.003, 0.250, 0.625, 0.684, 0.894, 0.981$. For the significance of time, see Fig. 8. The lines are the element boundaries. The sinuses are transparent to make the valve motion visible

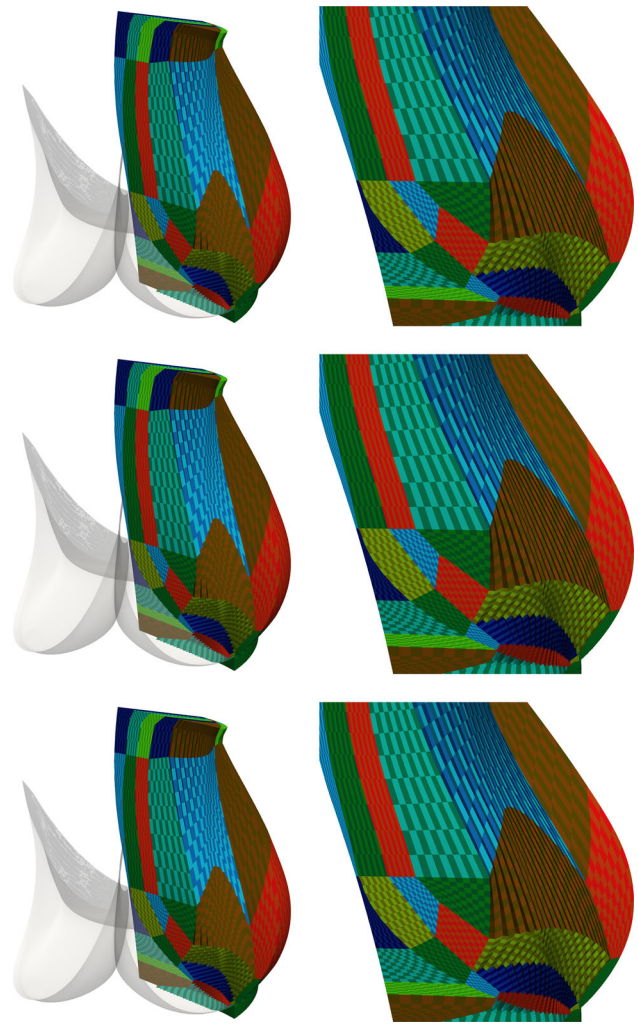


Fig. 20 Mesh in the valve at the cardiac-cycle instants of the first three pictures in Fig. 19. The right pictures are the zoomed views. The checkerboard pattern is for differentiating between the NURBS elements, and the colors are for differentiating between the NURBS patches

10.3 Results

Figure 22 shows the flow patterns. The flow patterns demonstrate that we are able to capture the spiral flow through the valve and in the aorta and we have a reasonable flow field even when the leaflets come into contact. Figure 23 and 24 show the magnitude of the wall shear stress (WSS) on the leaflet lower and upper surfaces. The WSS is, as expected, high on the lower surface.

11 Concluding remarks

We have addressed the computational challenges encountered in LV-valve-aorta flow analysis and presented results from the computation performed. We included the LV in the

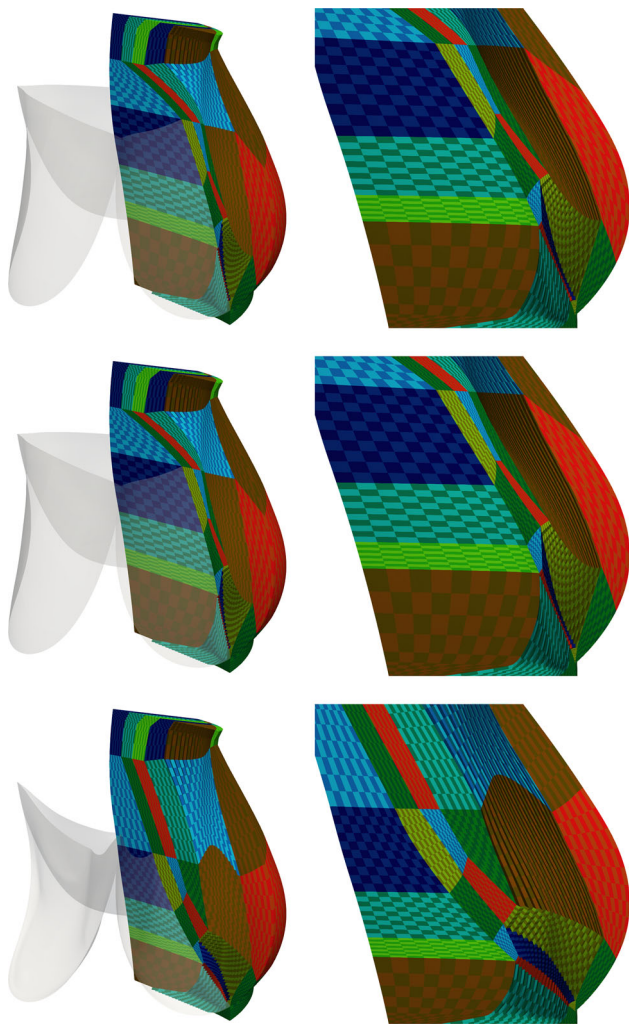


Fig. 21 Mesh in the valve at the cardiac-cycle instants of the last three pictures in Fig. 19. The right pictures are the zoomed views. The checkerboard pattern is for differentiating between the NURBS elements, and the colors are for differentiating between the NURBS patches

model to make the flow into the valve anatomically more realistic, which, in turn, makes the flow into the aorta more realistic. The challenges include (i) accurate representation of the boundary layers near the valve leaflets as they come into contact during the cardiac cycle, (ii) computation over a flow domain with high geometric complexity, (iii) anatomically realistic representation of the LV motion, and (iv) flow stability at the inflow boundary, where we have a traction condition corresponding to the inflow pressure.

The challenges were mainly addressed with the ST-SI-TC-IGA, which integrates the special ST methods ST-SI, ST-TC and ST-IGA around the core method ST-VMS, which subsumes its precursor ST-SUPS. The ST-SI-TC-IGA was supplemented with two other special methods: a structural mechanics computation method generating the LV motion



Fig. 22 Isosurfaces corresponding to a positive value of the second invariant of the velocity gradient tensor, colored by the velocity magnitude (m/s). The frames are for the same instants as in Fig. 19

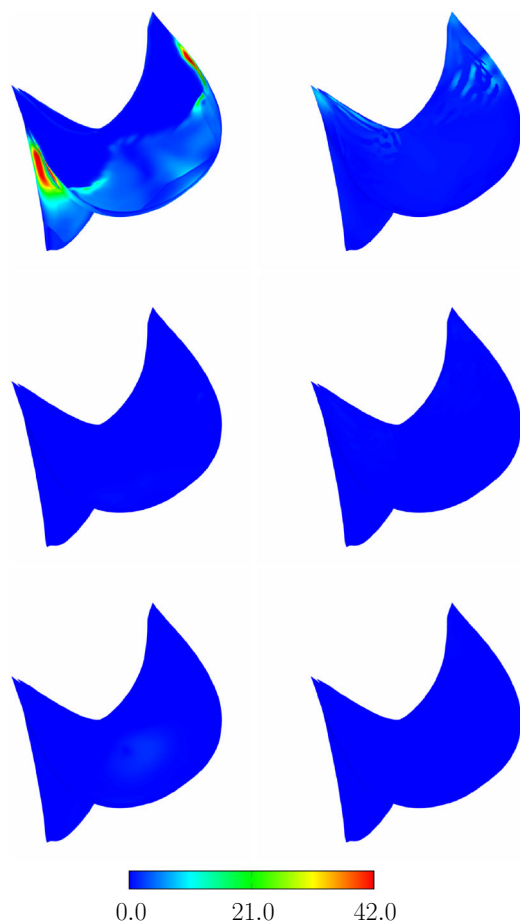


Fig. 23 Magnitude of the WSS (Pa) on the leaflet lower (left) and upper (right) surfaces. The frames are for the same instants as in the first three pictures of Fig. 19

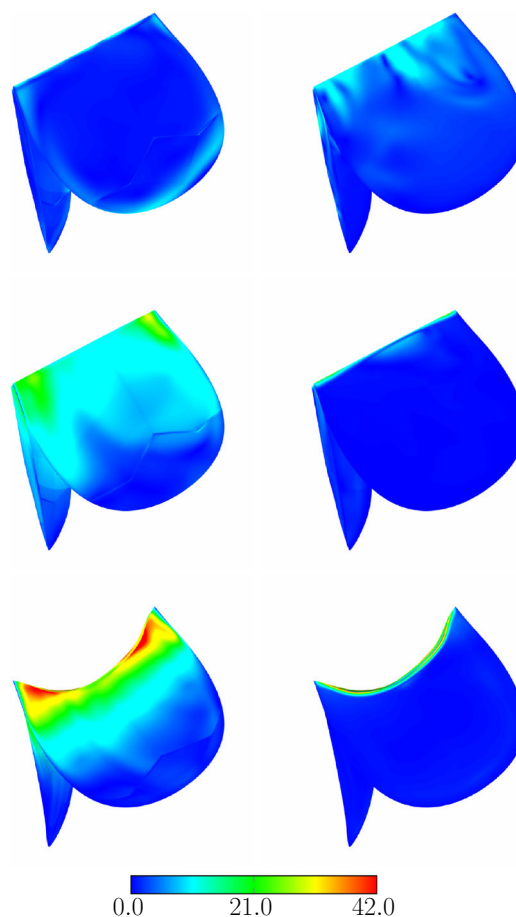


Fig. 24 Magnitude of the WSS (Pa) on the leaflet lower (left) and upper (right) surfaces. The frames are for the same instants as in the last three pictures of Fig. 19

and the CFP Traction. We summarize how these methods address the challenges.

1. The ST-discretization feature of the ST-SI-TC-IGA provides higher-order accuracy compared to standard discretization methods.
2. The VMS feature addresses the computational challenges associated with the multiscale nature of the unsteady flow in the LV, valve and aorta.
3. The moving-mesh feature of the ST framework enables high-resolution computation near the leaflets.
4. The ST-TC enables moving-mesh computation even with the TC created by the contact between the leaflets, dealing with the contact while maintaining high-resolution representation near the leaflets.
5. The ST-IGA provides smoother representation of the LV, valve and aorta surfaces and increased accuracy in the flow solution.
6. The ST-SI connects the separately generated LV, valve and aorta NURBS meshes, enabling easier mesh genera-

- tion, and connects the mesh zones containing the leaflets, enabling a more effective mesh moving.
7. The ST-SI also helps ST-TC deal with leaflet–leaflet contact location change and contact sliding and helps ST-TC and ST-IGA keep the element density in the narrow spaces near the contact areas at a reasonable level.
8. The special structural mechanics computation method generates the LV motion from the CT scans of the LV and anatomically realistic values for the LV volume ratio.
9. The CFP Traction provides flow stability at the inflow boundary. This is done by placing adjacent to the inflow boundary a special-purpose element, and an SI connects the flow solutions over that element and the rest of the mesh. The special-purpose element, with only one unspecified control-point velocity at the inflow that we solve for, results in a constrained flow profile that represents the flow rate generated by the traction

conditions specified at the inflow and outflow boundaries.

We first presented a low-Reynolds-number 2D-channel-flow test computation with the CFP Traction and showed that the results were very close to the analytical solution. This demonstrated the effectiveness of the CFP Traction as an inflow stabilization method. We then presented a computation with an LV-valve-aorta model created based on the CT scans of the LV and aorta. We were successful in capturing the spiral flow through the valve and in the aorta and we had a reasonable flow field even when the leaflets came into contact. This demonstrated the effectiveness of the ST-SI-TC-IGA and the two supplemental methods in a highly challenging computational cardiovascular flow analysis.

Acknowledgements This work was supported in part by JST-CREST; Grant-in-Aid for Scientific Research (A) 18H04100 from Japan Society for the Promotion of Science; and Rice–Waseda research agreement. This work was also supported (first author) in part by Grant-in-Aid for JSPS Research Fellow 17J11096. The mathematical model and computational method parts of the work were also supported (third author) in part by ARO Grant W911NF-17-1-0046 and Top Global University Project of Waseda University.

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

A Mesh moving method and Mesh-Jacobian-based stiffening

In the mesh moving method introduced in 1992 [155–157], the motion of the internal nodes is determined by solving the equations of linear elasticity. As the boundary condition, the normal velocity of the mesh at the boundaries and interfaces is specified to match the velocity of the fluid. With the notation of [26], the mesh moving formulation can be written as

$$\int_{\Omega_i} \boldsymbol{\varepsilon}(\mathbf{w}^h) : \boldsymbol{\sigma}(\hat{\mathbf{y}}^h - \hat{\mathbf{y}}_i^h) d\Omega = 0. \tag{17}$$

Here $\boldsymbol{\varepsilon}$ is the strain tensor, $\boldsymbol{\sigma}$ is the stress tensor, $\hat{\mathbf{y}}^h$ is the displacement from the reference configuration \mathbf{X}^h to the current configuration $\mathbf{x}^h = \mathbf{X}^h + \hat{\mathbf{y}}^h$, and $\hat{\mathbf{y}}_i^h$ is the displacement

from \mathbf{X}^h to the \tilde{t} configuration $\mathbf{x}_i^h = \mathbf{X}^h + \hat{\mathbf{y}}_i^h$ we compute the mesh motion from. The most common selection is $\tilde{t} = t_n$ when we are computing the mesh at t_{n+1} .

In [155–157] the mesh deformation is dealt with selectively based on the sizes of the elements. Selective treatment based on element sizes is attained by altering the way we account for the Jacobian of the transformation from the element domain to the physical domain:

$$J_M = \det \left(\frac{\partial \mathbf{x}_i^h}{\partial \boldsymbol{\xi}} \right), \tag{18}$$

where $\boldsymbol{\xi}$ is the parametric coordinate, and the subscript “M” is to clarify that this is the mesh Jacobian. The objective is to stiffen the smaller elements, which are typically placed near solid surfaces, more than the larger ones. When this method was first introduced in [155–157], it consisted of simply dropping J_M from Eq. (17). That is equivalent to modifying Eq. (17) as

$$\int_{\Omega_i} \boldsymbol{\varepsilon}(\mathbf{w}^h) : \boldsymbol{\sigma}(\hat{\mathbf{y}}^h - \hat{\mathbf{y}}_i^h) \left(\frac{J_M}{(J_M)_0} \right)^{-1} d\Omega = 0, \tag{19}$$

where $(J_M)_0$ was originally intended to be a global scaling value and is now seen as a free parameter. The modification results in the smaller elements being stiffened more than the larger ones, and the element stiffening factor is $\frac{(J_M)_0}{J_M}$. The method was given the name “Jacobian-based stiffening” in [158]. It was also augmented in [158] to a more extensive kind by introducing a stiffening power χ that determines the degree by which the smaller elements are rendered stiffer than the larger ones:

$$\int_{\Omega_i} \boldsymbol{\varepsilon}(\mathbf{w}^h) : \boldsymbol{\sigma}(\hat{\mathbf{y}}^h - \hat{\mathbf{y}}_i^h) \left(\frac{J_M}{(J_M)_0} \right)^{-\chi} d\Omega = 0. \tag{20}$$

This approach, when $\chi = 1$, would be identical to the method first introduced in [155], and most of the time $\chi = 1$.

Remark 1 To also clarify that the “Jacobian” in the name of the method is the mesh Jacobian, we are renaming the method “Mesh-Jacobian-based stiffening” (MJBS).

From earlier tests with different \tilde{t} settings, we know that setting $\tilde{t} = t_n$ gives reasonably good performance for most problems. We believe that is partly because when we compute the mesh motion from a configuration close to the current configuration, J_M provides the method feedback on which elements to protect.

Remark 2 While computing from the $\tilde{t} = t_n$ configuration works well for most problems, there are two closely related drawbacks: (i) the method is path-dependent, (ii) once elements accumulate in some region, it is hard to undo that. The

path-dependence leads to non-cyclic results in FSI computations that we expect to have cyclic or near-cyclic results. Using the large-deformation elasticity equations (see, for example, [85]) is one way of addressing this issue. For FSI problems with cyclic or near-cyclic solutions, as an alternative, we introduce here the “Back-cycle-based mesh moving” (BCBMM) method. With T being the cycle period, in the k th cycle, as we move from t_n to t_{n+1} , we compute the mesh motion from the configuration at $\tilde{t} = t_n - (k - k_{\text{FORW}} - 1)T$ or $\tilde{t} = t_{n+1} - (k - k_{\text{FORW}} - 1)T$, where $k \geq 2$. We start with $k_{\text{FORW}} = 0$. With that, in any cycle, we compute the mesh motion based on the first cycle. In later cycles, as needed, we can set k_{FORW} to higher values to compute the mesh motion based on the recent past cycles. That would be a good way if there are significant differences between the FSI solutions at different cycles before the solution becomes cyclic or near-cyclic.

We note that in practical computations the element stiffening method proposed in [159] is approximately equivalent to the MJBS method with $\chi = 1$. To show that, we first approximate $\frac{(J_M)_0}{J_M}$ as

$$\frac{(J_M)_0}{J_M} \approx \frac{V_0}{V}, \quad (21)$$

where V is the element volume and V_0 is a global scaling value. The approximation becomes exact for simplex elements. In [159], the stiffening factor, represented by “ $1 + \tau_m$ ”, is given as

$$1 + \tau_m = 1 + \frac{1 - \frac{V_{\min}}{V_{\max}}}{\frac{V}{V_{\max}}}, \quad (22)$$

where V_{\min} and V_{\max} are the minimum and maximum element volumes. This can also be written as

$$1 + \tau_m = 1 + \frac{V_{\max} - V_{\min}}{V}. \quad (23)$$

In practical computations, $V_{\max} \gg V_{\min}$, and for most of the elements that matter,

$$\frac{V_{\max} - V_{\min}}{V} \gg 1. \quad (24)$$

Therefore

$$1 + \tau_m \approx \frac{V_{\max} - V_{\min}}{V}. \quad (25)$$

Because $V_{\max} - V_{\min}$ is a global scaling value just like $(J_M)_0$ is, the stiffening factor given by Eq. (25) is equivalent to the one given by Eq. (21).

Remark 3 We note that a stiffening-factor expression in terms of the mesh Jacobian gives us the option of evaluating the factor at the integration points, without the need for any pre-calculation. We also note that such an expression is naturally applicable to isogeometric discretization.

References

1. Takizawa K, Tezduyar TE, Terahara T, Sasaki T (2017) Heart valve flow computation with the integrated space–time VMS, slip interface, topology change and isogeometric discretization methods. *Comput Fluids* 158:176–188. <https://doi.org/10.1016/j.compfluid.2016.11.012>
2. Kuraishi T, Takizawa K, Tezduyar TE (2019) Tire aerodynamics with actual tire geometry, road contact and tire deformation. *Comput Mech* 63:1165–1185. <https://doi.org/10.1007/s00466-018-1642-1>
3. Takizawa K, Tezduyar TE (2011) Multiscale space–time fluid–structure interaction techniques. *Comput Mech* 48:247–267. <https://doi.org/10.1007/s00466-011-0571-z>
4. Takizawa K, Tezduyar TE (2012) Space–time fluid–structure interaction methods. *Math Models Methods Appl Sci* 22(supp02):1230001. <https://doi.org/10.1142/S0218202512300013>
5. Takizawa K, Tezduyar TE, Kuraishi T (2015) Multiscale ST methods for thermo–fluid analysis of a ground vehicle and its tires. *Math Models Methods Appl Sci* 25:2227–2255. <https://doi.org/10.1142/S0218202515400072>
6. Takizawa K, Tezduyar TE, Mochizuki H, Hattori H, Mei S, Pan L, Montel K (2015) Space–time VMS method for flow computations with slip interfaces (ST-SI). *Math Models Methods Appl Sci* 25:2377–2406. <https://doi.org/10.1142/S0218202515400126>
7. Takizawa K, Tezduyar TE, Kuraishi T, Tabata S, Takagi H (2016) Computational thermo–fluid analysis of a disk brake. *Comput Mech* 57:965–977. <https://doi.org/10.1007/s00466-016-1272-4>
8. Takizawa K, Tezduyar TE, Buscher A, Asada S (2014) Space–time interface-tracking with topology change (ST-TC). *Comput Mech* 54:955–971. <https://doi.org/10.1007/s00466-013-0935-7>
9. Takizawa K, Tezduyar TE, Buscher A, Asada S (2014) Space–time fluid mechanics computation of heart valve models. *Comput Mech* 54:973–986. <https://doi.org/10.1007/s00466-014-1046-9>
10. Takizawa K, Henicke B, Puntel A, Spielman T, Tezduyar TE (2012) Space–time computational techniques for the aerodynamics of flapping wings. *J Appl Mech* 79:010903. <https://doi.org/10.1115/1.4005073>
11. Takizawa K, Tezduyar TE, Otoguro Y, Terahara T, Kuraishi T, Hattori H (2017) Turbocharger flow computations with the space–time isogeometric analysis (ST-IGA). *Comput Fluids* 142:15–20. <https://doi.org/10.1016/j.compfluid.2016.02.021>
12. Takizawa K, Tezduyar TE (2014) Space–time computation techniques with continuous representation in time (ST-C). *Comput Mech* 53:91–99. <https://doi.org/10.1007/s00466-013-0895-y>
13. Kuraishi T, Takizawa K, Tezduyar TE (2019) Space–time computational analysis of tire aerodynamics with actual geometry, road contact, tire deformation, road roughness and fluid film. *Comput Mech* 64:1699–1718. <https://doi.org/10.1007/s00466-019-01746-8>
14. Tezduyar TE (1992) Stabilized finite element formulations for incompressible flow computations. *Adv Appl Mech* 28:1–44. [https://doi.org/10.1016/S0065-2156\(08\)70153-4](https://doi.org/10.1016/S0065-2156(08)70153-4)
15. Tezduyar TE (2003) Computation of moving boundaries and interfaces and stabilization parameters. *Int J Numer Methods Fluids* 43:555–575. <https://doi.org/10.1002/flid.505>

16. Tezduyar TE, Sathe S (2007) Modeling of fluid–structure interactions with the space–time finite elements: solution techniques. *Int J Numer Methods Fluids* 54:855–900. <https://doi.org/10.1002/flid.1430>
17. Brooks AN, Hughes TJR (1982) Streamline upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equations. *Comput Methods Appl Mech Eng* 32:199–259
18. Hughes TJR (1995) Multiscale phenomena: Green’s functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles, and the origins of stabilized methods. *Comput Methods Appl Mech Eng* 127:387–401
19. Hughes TJR, Oberai AA, Mazzei L (2001) Large eddy simulation of turbulent channel flows by the variational multiscale method. *Phys Fluids* 13:1784–1799
20. Bazilevs Y, Calo VM, Cottrell JA, Hughes TJR, Reali A, Scovazzi G (2007) Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows. *Comput Methods Appl Mech Eng* 197:173–201
21. Bazilevs Y, Akkerman I (2010) Large eddy simulation of turbulent Taylor–Couette flow using isogeometric analysis and the residual-based variational multiscale method. *J Comput Phys* 229:3402–3414
22. Hughes TJR, Liu WK, Zimmermann TK (1981) Lagrangian–Eulerian finite element formulation for incompressible viscous flows. *Comput Methods Appl Mech Eng* 29:329–349
23. Bazilevs Y, Calo VM, Hughes TJR, Zhang Y (2008) Isogeometric fluid–structure interaction: theory, algorithms, and computations. *Comput Mech* 43:3–37
24. Takizawa K, Bazilevs Y, Tezduyar TE (2012) Space–time and ALE-VMS techniques for patient-specific cardiovascular fluid–structure interaction modeling. *Arch Comput Methods Eng* 19:171–225. <https://doi.org/10.1007/s11831-012-9071-3>
25. Bazilevs Y, Hsu M-C, Takizawa K, Tezduyar TE (2012) ALE-VMS and ST-VMS methods for computer modeling of wind-turbine rotor aerodynamics and fluid–structure interaction. *Math Models Methods Appl Sci* 22(supp02):1230002. <https://doi.org/10.1142/S0218202512300025>
26. Bazilevs Y, Takizawa K, Tezduyar TE (2013) Computational fluid–structure interaction: methods and applications. Wiley, Hoboken ISBN 978-0470978771
27. Bazilevs Y, Takizawa K, Tezduyar TE (2013) Challenges and directions in computational fluid–structure interaction. *Math Models Methods Appl Sci* 23:215–221. <https://doi.org/10.1142/S0218202513400010>
28. Bazilevs Y, Takizawa K, Tezduyar TE (2015) New directions and challenging computations in fluid dynamics modeling with stabilized and multiscale methods. *Math Models Methods Appl Sci* 25:2217–2226. <https://doi.org/10.1142/S0218202515020029>
29. Bazilevs Y, Takizawa K, Tezduyar TE (2019) Computational analysis methods for complex unsteady flow problems. *Math Models Methods Appl Sci* 29:825–838. <https://doi.org/10.1142/S0218202519020020>
30. Kalro V, Tezduyar TE (2000) A parallel 3D computational method for fluid–structure interactions in parachute systems. *Comput Methods Appl Mech Eng* 190:321–332. [https://doi.org/10.1016/S0045-7825\(00\)00204-8](https://doi.org/10.1016/S0045-7825(00)00204-8)
31. Bazilevs Y, Hughes TJR (2007) Weak imposition of Dirichlet boundary conditions in fluid mechanics. *Comput Fluids* 36:12–26
32. Bazilevs Y, Michler C, Calo VM, Hughes TJR (2010) Isogeometric variational multiscale modeling of wall-bounded turbulent flows with weakly enforced boundary conditions on unstretched meshes. *Comput Methods Appl Mech Eng* 199:780–790
33. Hsu M-C, Akkerman I, Bazilevs Y (2012) Wind turbine aerodynamics using ALE-VMS: validation and role of weakly enforced boundary conditions. *Comput Mech* 50:499–511
34. Bazilevs Y, Hughes TJR (2008) NURBS-based isogeometric analysis for the computation of flows about rotating components. *Comput Mech* 43:143–150
35. Hsu M-C, Bazilevs Y (2012) Fluid–structure interaction modeling of wind turbines: simulating the full machine. *Comput Mech* 50:821–833
36. Moghadam ME, Bazilevs Y, Hsia T-Y, Vignon-Clementel IE, Marsden AL (2011) A comparison of outlet boundary treatments for prevention of backflow divergence with relevance to blood flow simulations. *Comput Mech* 48:277–291. <https://doi.org/10.1007/s00466-011-0599-0>
37. Bazilevs Y, Hsu M-C, Akkerman I, Wright S, Takizawa K, Henicke B, Spielman T, Tezduyar TE (2011) 3D simulation of wind turbine rotors at full scale. Part I: geometry modeling and aerodynamics. *Int J Numer Methods Fluids* 65:207–235. <https://doi.org/10.1002/flid.2400>
38. Bazilevs Y, Hsu M-C, Kiendl J, Wüchner R, Bletzinger K-U (2011) 3D simulation of wind turbine rotors at full scale. Part II: fluid–structure interaction modeling with composite blades. *Int J Numer Methods Fluids* 65:236–253
39. Hsu M-C, Akkerman I, Bazilevs Y (2011) High-performance computing of wind turbine aerodynamics using isogeometric analysis. *Comput Fluids* 49:93–100
40. Bazilevs Y, Hsu M-C, Scott MA (2012) Isogeometric fluid–structure interaction analysis with emphasis on non-matching discretizations, and with application to wind turbines. *Comput Methods Appl Mech Eng* 249–252:28–41
41. 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
42. Korobenko A, Hsu M-C, Akkerman I, Tippmann J, Bazilevs Y (2013) Structural mechanics modeling and FSI simulation of wind turbines. *Math Models Methods Appl Sci* 23:249–272
43. Bazilevs Y, Takizawa K, Tezduyar TE, Hsu M-C, Kostov N, McIntyre S (2014) Aerodynamic and FSI analysis of wind turbines with the ALE-VMS and ST-VMS methods. *Arch Comput Methods Eng* 21:359–398. <https://doi.org/10.1007/s11831-014-9119-7>
44. Bazilevs Y, Korobenko A, Deng X, Yan J (2015) Novel structural modeling and mesh moving techniques for advanced FSI simulation of wind turbines. *Int J Numer Methods Eng* 102:766–783. <https://doi.org/10.1002/nme.4738>
45. Korobenko A, Yan J, Gohari SMI, Sarkar S, Bazilevs Y (2017) FSI simulation of two back-to-back wind turbines in atmospheric boundary layer flow. *Comput Fluids* 158:167–175. <https://doi.org/10.1016/j.compfluid.2017.05.010>
46. Korobenko A, Bazilevs Y, Takizawa K, Tezduyar TE (2018) Recent advances in ALE-VMS and ST-VMS computational aerodynamic and FSI analysis of wind turbines. In: Tezduyar TE (ed) *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, Berlin, pp 253–336. https://doi.org/10.1007/978-3-319-96469-0_7 (ISBN 978-3-319-96468-3)
47. Korobenko A, Bazilevs Y, Takizawa K, Tezduyar TE (2019) Computer modeling of wind turbines: 1. ALE-VMS and ST-VMS aerodynamic and FSI analysis. *Arch Comput Methods Eng* 26:1059–1099. <https://doi.org/10.1007/s11831-018-9292-1>
48. Korobenko A, Hsu M-C, Akkerman I, Bazilevs Y (2013) Aerodynamic simulation of vertical-axis wind turbines. *J Appl Mech* 81:021011. <https://doi.org/10.1115/1.4024415>
49. Bazilevs Y, Korobenko A, Deng X, Yan J, Kinzel M, Dabiri JO (2014) FSI modeling of vertical-axis wind turbines. *J Appl Mech* 81:081006. <https://doi.org/10.1115/1.4027466>

50. Yan J, Korobenko A, Deng X, Bazilevs Y (2016) Computational free-surface fluid–structure interaction with application to floating offshore wind turbines. *Comput Fluids* 141:155–174. <https://doi.org/10.1016/j.compfluid.2016.03.008>
51. Bazilevs Y, Korobenko A, Yan J, Pal A, Gohari SMI, Sarkar S (2015) ALE-VMS formulation for stratified turbulent incompressible flows with applications. *Math Models Methods Appl Sci* 25:2349–2375. <https://doi.org/10.1142/S0218202515400114>
52. Bazilevs Y, Korobenko A, Deng X, Yan J (2016) FSI modeling for fatigue-damage prediction in full-scale wind-turbine blades. *J Appl Mech* 83(6):061010
53. Bazilevs Y, Calo VM, Zhang Y, Hughes TJR (2006) Isogeometric fluid–structure interaction analysis with applications to arterial blood flow. *Comput Mech* 38:310–322
54. Bazilevs Y, Gohean JR, Hughes TJR, Moser RD, 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. *Comput Methods Appl Mech Eng* 198:3534–3550
55. Bazilevs Y, Hsu M-C, Benson D, Sankaran S, Marsden A (2009) Computational fluid–structure interaction: methods and application to a total cavopulmonary connection. *Comput Mech* 45:77–89
56. 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. *Comput Mech* 46:3–16
57. 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. *Biomech Model Mechanobiol* 9:481–498
58. Hsu M-C, Bazilevs Y (2011) Blood vessel tissue prestress modeling for vascular fluid–structure interaction simulations. *Finite Elem Anal Des* 47:593–599
59. Long CC, Marsden AL, Bazilevs Y (2013) Fluid–structure interaction simulation of pulsatile ventricular assist devices. *Comput Mech* 52:971–981. <https://doi.org/10.1007/s00466-013-0858-3>
60. Long CC, Esmaily-Moghadam M, Marsden AL, Bazilevs Y (2014) Computation of residence time in the simulation of pulsatile ventricular assist devices. *Comput Mech* 54:911–919. <https://doi.org/10.1007/s00466-013-0931-y>
61. Long CC, Marsden AL, Bazilevs Y (2014) Shape optimization of pulsatile ventricular assist devices using FSI to minimize thrombotic risk. *Comput Mech* 54:921–932. <https://doi.org/10.1007/s00466-013-0967-z>
62. Hsu M-C, Kamensky D, Bazilevs Y, Sacks MS, Hughes TJR (2014) Fluid–structure interaction analysis of bioprosthetic heart valves: significance of arterial wall deformation. *Comput Mech* 54:1055–1071. <https://doi.org/10.1007/s00466-014-1059-4>
63. Hsu M-C, Kamensky D, Xu F, Kiendl J, Wang C, Wu MCH, Mineroff J, Reali A, Bazilevs Y, Sacks MS (2015) Dynamic and fluid–structure interaction simulations of bioprosthetic heart valves using parametric design with T-splines and Fung-type material models. *Comput Mech* 55:1211–1225. <https://doi.org/10.1007/s00466-015-1166-x>
64. Kamensky D, Hsu M-C, Schillinger D, Evans JA, Aggarwal A, Bazilevs Y, Sacks MS, Hughes TJR (2015) An immersogeometric variational framework for fluid–structure interaction: application to bioprosthetic heart valves. *Comput Methods Appl Mech Eng* 284:1005–1053
65. Akkerman I, Bazilevs Y, Benson DJ, Farthing MW, Kees CE (2012) Free-surface flow and fluid–object interaction modeling with emphasis on ship hydrodynamics. *J Appl Mech* 79:010905
66. 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. *Comput Mech* 50:719–727
67. Wang C, Wu MCH, Xu F, Hsu M-C, Bazilevs Y (2017) Modeling of a hydraulic arresting gear using fluid–structure interaction and isogeometric analysis. *Comput Fluids* 142:3–14. <https://doi.org/10.1016/j.compfluid.2015.12.004>
68. Wu MCH, Kamensky D, Wang C, Herrema AJ, Xu F, Pigazzini MS, Verma A, Marsden AL, Bazilevs Y, Hsu M-C (2017) Optimizing fluid–structure interaction systems with immersogeometric analysis and surrogate modeling: application to a hydraulic arresting gear. *Comput Methods Appl Mech Eng* 316:668–693
69. Yan J, Deng X, Korobenko A, Bazilevs Y (2017) Free-surface flow modeling and simulation of horizontal-axis tidal-stream turbines. *Comput Fluids* 158:157–166. <https://doi.org/10.1016/j.compfluid.2016.06.016>
70. Castorriani A, Corsini A, Rispoli F, Takizawa K, Tezduyar TE (2019) A stabilized ALE method for computational fluid–structure interaction analysis of passive morphing in turbomachinery. *Math Models Methods Appl Sci* 29:967–994. <https://doi.org/10.1142/S0218202519410057>
71. Augier B, Yan J, Korobenko A, Czarnowski J, Ketterman G, Bazilevs Y (2015) Experimental and numerical FSI study of compliant hydrofoils. *Comput Mech* 55:1079–1090. <https://doi.org/10.1007/s00466-014-1090-5>
72. 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. *Comput Fluids* 141:201–211. <https://doi.org/10.1016/j.compfluid.2015.07.013>
73. Helgedagsrud TA, Bazilevs Y, Mathisen KM, Oiseth OA (2018) Computational and experimental investigation of free vibration and flutter of bridge decks. *Comput Mech*. <https://doi.org/10.1007/s00466-018-1587-4>
74. Helgedagsrud TA, Bazilevs Y, Korobenko A, Mathisen KM, Oiseth OA (2018) Using ALE-VMS to compute aerodynamic derivatives of bridge sections. *Comput Fluids*. <https://doi.org/10.1016/j.compfluid.2018.04.037>
75. Helgedagsrud TA, Akkerman I, Bazilevs Y, Mathisen KM, Oiseth OA (2019) Isogeometric modeling and experimental investigation of moving-domain bridge aerodynamics. *ASCE J Eng Mech* 145:04019026. [https://doi.org/10.1061/\(ASCE\)EM.1943-7889.0001601](https://doi.org/10.1061/(ASCE)EM.1943-7889.0001601)
76. Kamensky D, Evans JA, 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. *Comput Math Appl* 74:2068–2088. <https://doi.org/10.1016/j.camwa.2017.07.006>
77. Yu Y, Kamensky D, Hsu M-C, Lu XY, Bazilevs Y, Hughes TJR (2018) Error estimates for projection-based dynamic augmented Lagrangian boundary condition enforcement, with application to fluid–structure interaction. *Math Models Methods Appl Sci* 28:2457–2509. <https://doi.org/10.1142/S0218202518500537>
78. Tezduyar TE, Takizawa K, Moorman C, Wright S, Christopher J (2010) Space–time finite element computation of complex fluid–structure interactions. *Int J Numer Methods Fluids* 64:1201–1218. <https://doi.org/10.1002/flid.2221>
79. Yan J, Korobenko A, Tejada-Martinez AE, Golshan R, Bazilevs Y (2017) A new variational multiscale formulation for stratified incompressible turbulent flows. *Comput Fluids* 158:150–156. <https://doi.org/10.1016/j.compfluid.2016.12.004>
80. van Opstal TM, Yan J, Coley C, Evans JA, Kvamsdal T, Bazilevs Y (2017) Isogeometric divergence-conforming variational multiscale formulation of incompressible turbulent flows. *Comput Methods Appl Mech Eng* 316:859–879. <https://doi.org/10.1016/j.cma.2016.10.015>
81. 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. *Comput Fluids* 158:201–220. <https://doi.org/10.1016/j.compfluid.2017.02.006>

82. Tezduyar TE, Takizawa K (2019) Space–time computations in practical engineering applications: a summary of the 25-year history. *Comput Mech* 63:747–753. <https://doi.org/10.1007/s00466-018-1620-7>
83. Takizawa K, Tezduyar TE (2012) Computational methods for parachute fluid–structure interactions. *Arch Comput Methods Eng* 19:125–169. <https://doi.org/10.1007/s11831-012-9070-4>
84. Takizawa K, Fritze M, Montes D, Spielman T, Tezduyar TE (2012) Fluid–structure interaction modeling of ringsail parachutes with disreefing and modified geometric porosity. *Comput Mech* 50:835–854. <https://doi.org/10.1007/s00466-012-0761-3>
85. Takizawa K, Tezduyar TE, Boben J, Kostov N, Boswell C, Buscher A (2013) Fluid–structure interaction modeling of clusters of spacecraft parachutes with modified geometric porosity. *Comput Mech* 52:1351–1364. <https://doi.org/10.1007/s00466-013-0880-5>
86. Takizawa K, Tezduyar TE, Boswell C, Tsutsui Y, Montel K (2015) Special methods for aerodynamic-moment calculations from parachute FSI modeling. *Comput Mech* 55:1059–1069. <https://doi.org/10.1007/s00466-014-1074-5>
87. Takizawa K, Montes D, Fritze M, McIntyre S, Boben J, Tezduyar TE (2013) Methods for FSI modeling of spacecraft parachute dynamics and cover separation. *Math Models Methods Appl Sci* 23:307–338. <https://doi.org/10.1142/S0218202513400058>
88. Takizawa K, Tezduyar TE, Boswell C, Kolesar R, Montel K (2014) FSI modeling of the reefed stages and disreefing of the Orion spacecraft drogue parachutes. *Comput Mech* 54:1203–1220. <https://doi.org/10.1007/s00466-014-1052-y>
89. Takizawa K, Tezduyar TE, Kolesar R, Boswell C, Kanai T, Montel K (2014) Multiscale methods for gore curvature calculations from FSI modeling of spacecraft parachutes. *Comput Mech* 54:1461–1476. <https://doi.org/10.1007/s00466-014-1069-2>
90. Takizawa K, Tezduyar TE, Kolesar R (2015) FSI modeling of the Orion spacecraft drogue parachutes. *Comput Mech* 55:1167–1179. <https://doi.org/10.1007/s00466-014-1108-z>
91. Takizawa K, Henicke B, Tezduyar TE, Hsu M-C, Bazilevs Y (2011) Stabilized space–time computation of wind-turbine rotor aerodynamics. *Comput Mech* 48:333–344. <https://doi.org/10.1007/s00466-011-0589-2>
92. Takizawa K, Henicke B, Montes D, Tezduyar TE, Hsu M-C, Bazilevs Y (2011) Numerical-performance studies for the stabilized space–time computation of wind-turbine rotor aerodynamics. *Comput Mech* 48:647–657. <https://doi.org/10.1007/s00466-011-0614-5>
93. Takizawa K, Tezduyar TE, McIntyre S, Kostov N, Kolesar R, Habluetzel C (2014) Space–time VMS computation of wind-turbine rotor and tower aerodynamics. *Comput Mech* 53:1–15. <https://doi.org/10.1007/s00466-013-0888-x>
94. Takizawa K, Bazilevs Y, Tezduyar TE, Hsu M-C, Øiseth O, Mathisen KM, Kostov N, McIntyre S (2014) Engineering analysis and design with ALE-VMS and space–time methods. *Arch Comput Methods Eng* 21:481–508. <https://doi.org/10.1007/s11831-014-9113-0>
95. Takizawa K (2014) Computational engineering analysis with the new-generation space–time methods. *Comput Mech* 54:193–211. <https://doi.org/10.1007/s00466-014-0999-z>
96. Takizawa K, Henicke B, Puntel A, Kostov N, Tezduyar TE (2012) Space–time techniques for computational aerodynamics modeling of flapping wings of an actual locust. *Comput Mech* 50:743–760. <https://doi.org/10.1007/s00466-012-0759-x>
97. Takizawa K, Henicke B, Puntel A, Kostov N, Tezduyar TE (2013) Computer modeling techniques for flapping-wing aerodynamics of a locust. *Comput Fluids* 85:125–134. <https://doi.org/10.1016/j.compfluid.2012.11.008>
98. Takizawa K, Kostov N, Puntel A, Henicke B, Tezduyar TE (2012) Space–time computational analysis of bio-inspired flapping-wing aerodynamics of a micro aerial vehicle. *Comput Mech* 50:761–778. <https://doi.org/10.1007/s00466-012-0758-y>
99. Takizawa K, Tezduyar TE, Kostov N (2014) Sequentially-coupled space–time FSI analysis of bio-inspired flapping-wing aerodynamics of an MAV. *Comput Mech* 54:213–233. <https://doi.org/10.1007/s00466-014-0980-x>
100. Takizawa K, Tezduyar TE, Buscher A (2015) Space–time computational analysis of MAV flapping-wing aerodynamics with wing clapping. *Comput Mech* 55:1131–1141. <https://doi.org/10.1007/s00466-014-1095-0>
101. Takizawa K, Bazilevs Y, Tezduyar TE, Long CC, Marsden AL, Schjodt K (2014) ST and ALE-VMS methods for patient-specific cardiovascular fluid mechanics modeling. *Math Models Methods Appl Sci* 24:2437–2486. <https://doi.org/10.1142/S0218202514500250>
102. Takizawa K, Schjodt K, Puntel A, Kostov N, Tezduyar TE (2012) Patient-specific computer modeling of blood flow in cerebral arteries with aneurysm and stent. *Comput Mech* 50:675–686. <https://doi.org/10.1007/s00466-012-0760-4>
103. Takizawa K, Schjodt K, Puntel A, Kostov N, Tezduyar TE (2013) Patient-specific computational analysis of the influence of a stent on the unsteady flow in cerebral aneurysms. *Comput Mech* 51:1061–1073. <https://doi.org/10.1007/s00466-012-0790-y>
104. Suito H, Takizawa K, Huynh VQH, Sze D, Ueda T (2014) FSI analysis of the blood flow and geometrical characteristics in the thoracic aorta. *Comput Mech* 54:1035–1045. <https://doi.org/10.1007/s00466-014-1017-1>
105. Suito H, Takizawa K, Huynh VQH, Sze D, Ueda T, Tezduyar TE (2016) A geometrical-characteristics study in patient-specific FSI analysis of blood flow in the thoracic aorta. In: Bazilevs Y, Takizawa K (eds) *Advances in computational fluid–structure interaction and flow simulation: new methods and challenging computations, modeling and simulation in science, engineering and technology*. Springer, Berlin, pp 379–386. https://doi.org/10.1007/978-3-319-40827-9_29 (ISBN 978-3-319-40825-5)
106. Takizawa K, Tezduyar TE, 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 TE (ed) *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, Berlin, pp 29–89. https://doi.org/10.1007/978-3-319-96469-0_2 (ISBN 978-3-319-96468-3)
107. Takizawa K, Tezduyar TE, Uchikawa H, Terahara T, Sasaki T, Yoshida A (2019) Mesh refinement influence and cardiac-cycle flow periodicity in aorta flow analysis with isogeometric discretization. *Comput Fluids* 179:790–798. <https://doi.org/10.1016/j.compfluid.2018.05.025>
108. Takizawa K, Bazilevs Y, Tezduyar TE, Hsu M-C (2019) Computational cardiovascular flow analysis with the variational multiscale methods. *J Adv Eng Comput* 3:366–405. <https://doi.org/10.25073/jaec.201932.245>
109. Takizawa K, Tezduyar TE (2016) New directions in space–time computational methods. In: Bazilevs Y, Takizawa K (eds) *Advances in computational fluid–structure interaction and flow simulation: new methods and challenging computations, modeling and simulation in science, engineering and technology*. Springer, Berlin, pp 159–178. https://doi.org/10.1007/978-3-319-40827-9_13 (ISBN 978-3-319-40825-5)
110. Takizawa K, Tezduyar TE, 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 (eds) *Biomedical technology: modeling, experiments and simulation, lecture notes in applied and compu-*

- tational mechanics. Springer, Berlin, pp 77–99. https://doi.org/10.1007/978-3-319-59548-1_6 (ISBN 978-3-319-59547-4)
111. Terahara T, Takizawa K, Tezduyar TE, Bazilevs Y, Hsu M-C (2020) Heart valve isogeometric sequentially-coupled FSI analysis with the space–time topology change method. *Comput Mech*. <https://doi.org/10.1007/s00466-019-01813-0>
 112. Yu Y, Zhang YJ, Takizawa K, Tezduyar TE, Sasaki T (2019) Anatomically realistic lumen motion representation in patient-specific space–time isogeometric flow analysis of coronary arteries with time-dependent medical-image data. *Comput Mech*. <https://doi.org/10.1007/s00466-019-01774-4>
 113. Takizawa K, Montes D, McIntyre S, Tezduyar TE (2013) Space-time VMS methods for modeling of incompressible flows at high Reynolds numbers. *Math Models Methods Appl Sci* 23:223–248. <https://doi.org/10.1142/S0218202513400022>
 114. Takizawa K, Tezduyar TE, Hattori H (2017) Computational analysis of flow-driven string dynamics in turbomachinery. *Comput Fluids* 142:109–117. <https://doi.org/10.1016/j.compfluid.2016.02.019>
 115. Komiya K, Kanai T, Otaguro Y, Kaneko M, Hirota K, Zhang Y, Takizawa K, Tezduyar TE, Nohmi M, Tsuneda T, Kawai M, Isono M (2019) Computational analysis of flow-driven string dynamics in a pump and residence time calculation. *IOP Conf Ser Earth Environ Sci* 240:062014. <https://doi.org/10.1088/1755-1315/240/6/062014>
 116. Kanai T, Takizawa K, Tezduyar TE, 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. *Math Models Methods Appl Sci* 29:839–870. <https://doi.org/10.1142/S021820251941001X>
 117. Otaguro Y, Takizawa K, Tezduyar TE (2017) Space–time VMS computational flow analysis with isogeometric discretization and a general-purpose NURBS mesh generation method. *Comput Fluids* 158:189–200. <https://doi.org/10.1016/j.compfluid.2017.04.017>
 118. Otaguro Y, Takizawa K, Tezduyar TE (2018) A general-purpose NURBS mesh generation method for complex geometries. In: Tezduyar TE (ed) *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, Berlin, pp 399–434. https://doi.org/10.1007/978-3-319-96469-0_10 (ISBN 978-3-319-96468-3)
 119. Otaguro Y, Takizawa K, Tezduyar TE, Nagaoka K, Mei S (2019) Turbocharger turbine and exhaust manifold flow computation with the space–time variational multiscale method and isogeometric analysis. *Comput Fluids* 179:764–776. <https://doi.org/10.1016/j.compfluid.2018.05.019>
 120. Otaguro Y, Takizawa K, Tezduyar TE, 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. *Comput Mech* 64:1403–1419. <https://doi.org/10.1007/s00466-019-01722-2>
 121. Takizawa K, Tezduyar TE, Asada S, Kuraishi T (2016) Space–time method for flow computations with slip interfaces and topology changes (ST-SI–TC). *Comput Fluids* 141:124–134. <https://doi.org/10.1016/j.compfluid.2016.05.006>
 122. Kuraishi T, Takizawa K, Tezduyar TE (2018) Space–time computational analysis of tire aerodynamics with actual geometry, road contact and tire deformation. In: Tezduyar TE (ed) *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, Berlin, pp 337–376. https://doi.org/10.1007/978-3-319-96469-0_8 (ISBN 978-3-319-96468-3)
 123. Kuraishi T, Takizawa K, Tezduyar TE (2019) Space–time Isogeometric flow analysis with built-in Reynolds-equation limit. *Math Models Methods Appl Sci* 29:871–904. <https://doi.org/10.1142/S0218202519410021>
 124. Takizawa K, Tezduyar TE, Terahara T (2016) Ram-air parachute structural and fluid mechanics computations with the space–time isogeometric analysis (ST-IGA). *Comput Fluids* 141:191–200. <https://doi.org/10.1016/j.compfluid.2016.05.027>
 125. Takizawa K, Tezduyar TE, Kanai T (2017) Porosity models and computational methods for compressible-flow aerodynamics of parachutes with geometric porosity. *Math Models Methods Appl Sci* 27:771–806. <https://doi.org/10.1142/S0218202517500166>
 126. Kanai T, Takizawa K, Tezduyar TE, Tanaka T, Hartmann A (2019) Compressible-flow geometric-porosity modeling and spacecraft parachute computation with isogeometric discretization. *Comput Mech* 63:301–321. <https://doi.org/10.1007/s00466-018-1595-4>
 127. Hsu M-C, Bazilevs Y, Calo VM, Tezduyar TE, Hughes TJR (2010) Improving stability of stabilized and multiscale formulations in flow simulations at small time steps. *Comput Methods Appl Mech Eng* 199:828–840. <https://doi.org/10.1016/j.cma.2009.06.019>
 128. Takizawa K, Tezduyar TE, Otaguro Y (2018) Stabilization and discontinuity-capturing parameters for space–time flow computations with finite element and isogeometric discretizations. *Comput Mech* 62:1169–1186. <https://doi.org/10.1007/s00466-018-1557-x>
 129. Takizawa K, Ueda Y, Tezduyar TE (2019) A node-numbering-invariant directional length scale for simplex elements. *Math Models Methods Appl Sci*. <https://doi.org/10.1142/S0218202519500581>
 130. Otaguro Y, Takizawa K, Tezduyar TE (2020) Element length calculation in B-spline meshes for complex geometries. *Comput Mech*. <https://doi.org/10.1007/s00466-019-01809-w>
 131. Corsini A, Menichini C, Rispoli F, Santoriello A, Tezduyar TE (2009) A multiscale finite element formulation with discontinuity capturing for turbulence models with dominant reaction like terms. *J Appl Mech* 76:021211. <https://doi.org/10.1115/1.3062967>
 132. Rispoli F, Saavedra R, Menichini F, Tezduyar TE (2009) Computation of inviscid supersonic flows around cylinders and spheres with the V-SGS stabilization and $YZ\beta$ shock-capturing. *J Appl Mech* 76:021209. <https://doi.org/10.1115/1.3057496>
 133. Corsini A, Iossa C, Rispoli F, Tezduyar TE (2010) A DRD finite element formulation for computing turbulent reacting flows in gas turbine combustors. *Comput Mech* 46:159–167. <https://doi.org/10.1007/s00466-009-0441-0>
 134. Corsini A, Rispoli F, Tezduyar TE (2011) Stabilized finite element computation of NOx emission in aero-engine combustors. *Int J Numer Methods Fluids* 65:254–270. <https://doi.org/10.1002/fld.2451>
 135. Corsini A, Rispoli F, Tezduyar TE (2012) Computer modeling of wave-energy air turbines with the SUPG/PSPG formulation and discontinuity-capturing technique. *J Appl Mech* 79:010910. <https://doi.org/10.1115/1.4005060>
 136. Corsini A, Rispoli F, Sheard AG, Tezduyar TE (2012) Computational analysis of noise reduction devices in axial fans with stabilized finite element formulations. *Comput Mech* 50:695–705. <https://doi.org/10.1007/s00466-012-0789-4>
 137. Kler PA, Dalcin LD, Paz RR, Tezduyar TE (2013) SUPG and discontinuity-capturing methods for coupled fluid mechanics and electrochemical transport problems. *Comput Mech* 51:171–185. <https://doi.org/10.1007/s00466-012-0712-z>
 138. Corsini A, Rispoli F, Sheard AG, Takizawa K, Tezduyar TE, Venturini P (2014) A variational multiscale method for particle-cloud tracking in turbomachinery flows. *Comput Mech* 54:1191–1202. <https://doi.org/10.1007/s00466-014-1050-0>

139. Rispoli F, Delibra G, Venturini P, Corsini A, Saavedra R, Tezduyar TE (2015) Particle tracking and particle-shock interaction in compressible-flow computations with the V-SGS stabilization and $YZ\beta$ shock-capturing. *Comput Mech* 55:1201–1209. <https://doi.org/10.1007/s00466-015-1160-3>
140. Cardillo L, Corsini A, Delibra G, Rispoli F, Tezduyar TE (2016) Flow analysis of a wave-energy air turbine with the SUPG/PSPG stabilization and discontinuity-capturing directional dissipation. *Comput Fluids* 141:184–190. <https://doi.org/10.1016/j.compfluid.2016.07.011>
141. Castorrini A, Corsini A, Rispoli F, Venturini P, Takizawa K, Tezduyar TE (2016) Computational analysis of wind-turbine blade rain erosion. *Comput Fluids* 141:175–183. <https://doi.org/10.1016/j.compfluid.2016.08.013>
142. Castorrini A, Corsini A, Rispoli F, Venturini P, Takizawa K, Tezduyar TE (2019) Computational analysis of performance deterioration of a wind turbine blade strip subjected to environmental erosion. *Comput Mech* 64:1133–1153. <https://doi.org/10.1007/s00466-019-01697-0>
143. Tezduyar TE, Aliabadi SK, Behr M, Mittal S (1994) Massively parallel finite element simulation of compressible and incompressible flows. *Comput Methods Appl Mech Eng* 119:157–177. [https://doi.org/10.1016/0045-7825\(94\)00082-4](https://doi.org/10.1016/0045-7825(94)00082-4)
144. Hughes TJR, Cottrell JA, Bazilevs Y (2005) Isogeometric analysis: CAD, finite elements, NURBS, exact geometry, and mesh refinement. *Comput Methods Appl Mech Eng* 194:4135–4195
145. Takizawa K, Takagi H, Tezduyar TE, Torii R (2014) Estimation of element-based zero-stress state for arterial FSI computations. *Comput Mech* 54:895–910. <https://doi.org/10.1007/s00466-013-0919-7>
146. Takizawa K, Torii R, Takagi H, Tezduyar TE, Xu XY (2014) Coronary arterial dynamics computation with medical-image-based time-dependent anatomical models and element-based zero-stress state estimates. *Comput Mech* 54:1047–1053. <https://doi.org/10.1007/s00466-014-1049-6>
147. Takizawa K, Tezduyar TE, Sasaki T (2018) Estimation of element-based zero-stress state in arterial FSI computations with isogeometric wall discretization. In: Wriggers P, Lenarz T (eds) *Biomedical technology: modeling, experiments and simulation, lecture notes in applied and computational mechanics*. Springer, Berlin, pp 101–122. https://doi.org/10.1007/978-3-319-59548-1_7 (ISBN 978-3-319-59547-4)
148. Takizawa K, Tezduyar TE, Sasaki T (2017) Aorta modeling with the element-based zero-stress state and isogeometric discretization. *Comput Mech* 59:265–280. <https://doi.org/10.1007/s00466-016-1344-5>
149. Sasaki T, Takizawa K, Tezduyar TE (2019) Aorta zero-stress state modeling with T-spline discretization. *Comput Mech* 63:1315–1331. <https://doi.org/10.1007/s00466-018-1651-0>
150. Sasaki T, Takizawa K, Tezduyar TE (2019) Medical-image-based aorta modeling with zero-stress-state estimation. *Comput Mech* 64:249–271. <https://doi.org/10.1007/s00466-019-01669-4>
151. Takizawa K, Tezduyar TE, Sasaki T (2019) Isogeometric hyperelastic shell analysis with out-of-plane deformation mapping. *Comput Mech* 63:681–700. <https://doi.org/10.1007/s00466-018-1616-3>
152. Konings MK, Jansen R, Bosman LP, Rienks R, Chamuleau SAJ, Rademakers FE, Cramer MJ (2017) Non-invasive measurement of volume-time curves in patients with mitral regurgitation and in healthy volunteers, using a new operator-independent screening tool. *Physiol Meas* 38:241–258
153. Vieira ML, Nomura CH, Tranchesi JB, Oliveira WA, Serpa BS, Passos RB, Funari MB, Fischer CH, Morhy SS (2009) Left ventricular ejection fraction and volumes as measured by 3d echocardiography. *ABC Cardiol J* 92:294–301
154. Murray CD (1926) The physiological principle of minimum work: I. The vascular system and the cost of blood volume. *Proc Natl Acad Sci USA* 12:207–214
155. Tezduyar TE, Behr M, Mittal S, Johnson AA (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*, ASME, New York, pp 7–24
156. Tezduyar T, Aliabadi S, Behr M, Johnson A, Mittal S (1993) Parallel finite-element computation of 3D flows. *Computer* 26(10):27–36. <https://doi.org/10.1109/2.237441>
157. Johnson AA, Tezduyar TE (1994) Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces. *Comput Methods Appl Mech Eng* 119:73–94. [https://doi.org/10.1016/0045-7825\(94\)00077-8](https://doi.org/10.1016/0045-7825(94)00077-8)
158. Stein K, Tezduyar T, Benney R (2003) Mesh moving techniques for fluid–structure interactions with large displacements. *J Appl Mech* 70:58–63. <https://doi.org/10.1115/1.1530635>
159. Masud A, Hughes TJR (1997) A space–time Galerkin/least-squares finite element formulation of the Navier–Stokes equations for moving domain problems. *Comput Methods Appl Mech Eng* 146:91–126

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