Aorta zero-stress state modeling with T-spline discretization

Takafumi Sasaki¹ · Kenji Takizawa¹ · Tayfun E. Tezduyar^{2,3}

Received: 4 September 2018 / Accepted: 26 October 2018 / Published online: 2 November 2018
© The Author(s) 2018

Abstract

The image-based arterial geometries used in patient-specific arterial fluid–structure interaction (FSI) computations, such as aorta FSI computations, do not come from the zero-stress state (ZSS) of the artery. We propose a method for estimating the ZSS required in the computations. Our estimate is based on T-spline discretization of the arterial wall and is in the form of integration-point-based ZSS (IPBZSS). The method has two main components. (1) An iterative method, which starts with a calculated initial guess, is used for computing the IPBZSS such that when a given pressure load is applied, the image-based target shape is matched. (2) A method, which is based on the shell model of the artery, is used for calculating the initial guess. The T-spline discretization enables dealing with complex arterial geometries, such as an aorta model with branches, while retaining the desirable features of isogeometric discretization. With higher-order basis functions of the isogeometric discretization, we may be able to achieve a similar level of accuracy as with the linear basis functions, but using larger-size and much fewer elements. In addition, the higher-order basis functions allow representation of more complex shapes within an element. The IPBZSS is a convenient representation of the ZSS because with isogeometric discretization, especially with T-spline discretization, specifying conditions at integration points is more straightforward than imposing conditions on control points. Calculating the initial guess based on the shell model of the artery results in a more realistic initial guess. To show how the new ZSS estimation method performs, we first present 3D test computations with a Y-shaped tube. Then we show a 3D computation where the target geometry is coming from medical image of a human aorta, and we include the branches in our model.

Keywords Patient-specific arterial FSI · Image-based geometry · Aorta · Zero-stress state · Isogeometric wall discretization · T-spline basis functions · Integration-point-based zero-stress state · Shell-model-based initial guess

1 Introduction

The patient-specific arterial fluid–structure interaction (FSI) computations reported in [1–4] were among the earliest of its kind. The core method in these computations was the early version of the Deforming-Spatial-Domain/Stabilized Space–Time (DSD/SST) method [5,6], which is now called

the “ST-SUPS.” The acronym “SUPS” indicates the stabilization components, the Streamline-Upwind/Petrov–Galerkin (SUPG) [7] and Pressure-Stabilizing/Petrov–Galerkin (PSPG) [5] stabilizations.

The ST computations have been only a small part of the large number of cardiovascular fluid mechanics and FSI computations seen in the last 15 years (see, for example, [8–29]), with the Arbitrary Lagrangian–Eulerian (ALE) method [30] having the largest share in the computations reported. Still, a large number of computations with the ST methods were also reported in the last 15 years. In the first 8 years of that period the ST computations were for FSI of abdominal aorta [31], carotid artery [31] and cerebral aneurysms [32–38]. In the last 7 years, the ST computations focused on even more challenging aspects of cardiovascular fluid mechanics and FSI, including comparative studies of cerebral aneurysms [39,40], stent treatment of cerebral aneurysms [41–45], heart valve flow computation [46–51], aorta flow

✉ 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

analysis [51–54], and coronary arterial dynamics [55]. The large number of computational challenges encountered were addressed by the advances in core methods for moving boundaries and interfaces (MBI) and FSI (see, for example, [20,21,40,46,47,49,50,56–64] and references therein) and in special methods targeting cardiovascular MBI and FSI (see, for example, [20,38,44,45,48–51,54,65] and references therein).

A challenge very specific to patient-specific arterial FSI computations, such as patient-specific aorta FSI computations, is how to use the image-based arterial geometry. The image-based geometry does not come from the zero-stress state (ZSS) of the artery. Special methods targeting cardiovascular MBI and FSI include those designed to account for that. The attempt to find a ZSS for the artery in the FSI computation was first made in a 2007 conference paper [66], where the concept of estimated zero-pressure (EZP) arterial geometry was introduced. The method introduced in [66] for calculating an EZP geometry was also included in a 2008 journal paper on ST arterial FSI methods [32], as “a rudimentary technique” for addressing the issue. It was pointed out in [32,66] that quite often the image-based geometries were used as arterial geometries corresponding to zero blood pressure, and that it would be more realistic to use the image-based geometry as the arterial geometry corresponding to the time-averaged value of the blood pressure. Given the arterial geometry at the time-averaged pressure value, an estimated arterial geometry corresponding to zero blood pressure needed to be built. Special methods developed to address the issue include the newer EZP versions [20,34,37,38,65] and the prestress technique introduced in [16], which was refined in [18] and presented also in [20,65].

We introduced in [67] a method for estimation of the element-based ZSS (EBZSS) in the context of finite element discretization of the arterial wall. The method has three main components. (1) An iterative method, which starts with a calculated initial guess, is used for computing the EBZSS such that when a given pressure load is applied, the image-based target shape is matched. (2) A method for straight-tube segments is used for computing the EBZSS so that we match the given diameter and longitudinal stretch in the target configuration and the “opening angle.” (3) An element-based mapping between the artery and straight-tube is extracted from the mapping between the artery and straight-tube segments. This provides the mapping from the arterial configuration to the straight-tube configuration, and from the estimated EBZSS of the straight-tube configuration back to the arterial configuration, to be used as the initial guess for the iterative method that matches the image-based target shape. Test computations with the method were also presented in [67] for straight-tube configurations with single and three layers, and for a curved-tube configuration with single layer. The method was used also in [55] in coronary

arterial dynamics computations with medical-image-based time-dependent anatomical models.

In [68], we introduced the version of the EBZSS estimation method with isogeometric wall discretization, using NURBS basis functions. With isogeometric discretization, we can obtain the element-based mapping directly, instead of extracting it from the mapping between the artery and straight-tube segments. That is because all we need for the element-based mapping, including the curvatures, can be obtained within an element. With NURBS basis functions, we may be able to achieve a similar level of accuracy as with the linear basis functions, but using larger-size and much fewer elements, and the NURBS basis functions allow representation of more complex shapes within an element. The 2D test computations with straight-tube configurations presented in [68] showed how the EBZSS estimation method with NURBS discretization works. In [69], which is an expanded, journal version of [68], we also showed how the method can be used in a 3D computation where the target geometry is coming from medical image of a human aorta.

In this article we are introducing a new method for estimating the ZSS. The estimate is based on T-spline discretization of the arterial wall and is in the form of integration-point-based ZSS (IPBZSS). The method has two main components. (1) An iterative method, which starts with a calculated initial guess, is used for computing the IPBZSS such that when a given pressure load is applied, the image-based target shape is matched. (2) A method, which is based on the shell model of the artery, is used for calculating the initial guess. The T-spline discretization enables dealing with complex arterial geometries, such as an aorta model with branches, while retaining the desirable features of isogeometric discretization. The IPBZSS is a convenient representation of the ZSS because with isogeometric discretization, especially with T-spline discretization, specifying conditions at integration points is more straightforward than imposing conditions on control points. Calculating the initial guess based on the shell model of the artery results in a more realistic initial guess. To show how the new method for estimating the ZSS performs, we first present 3D test computations with a Y-shaped tube. Then we show a 3D computation where the target geometry is coming from medical image of a human aorta, and we include the branches in our model.

In Sect. 2, we describe the Element-Based Total Lagrangian (EBTL) method, including the EBZSS and IPBZSS concepts. How the initial guess is calculated based on the shell model of the artery is described in Sect. 3. The numerical examples are given in Sect. 4, and the concluding remarks in Sect. 5.

2 EBTL method

In this section we provide an overview of the EBTL method [67], including the EBZSS concept, and describe the IPBZSS concept and the conversion between the two ZSS.

Let $\Omega_0 \subset \mathbb{R}^{n_{sd}}$ be the material domain of a structure in the ZSS, where n_{sd} is the number of space dimensions, and let Γ_0 be its boundary. Let $\Omega_t \subset \mathbb{R}^{n_{sd}}$, $t \in (0, T)$, be the material domain of the structure in the deformed state, and let Γ_t be its boundary. The structural mechanics equations based on the total Lagrangian formulation can be written as

$$\int_{\Omega_0} \mathbf{w} \cdot \rho_0 \frac{d^2 \mathbf{y}}{dt^2} d\Omega + \int_{\Omega_0} \delta \mathbf{E} : \mathbf{S} d\Omega - \int_{\Omega_0} \mathbf{w} \cdot \rho_0 \mathbf{f} d\Omega = \int_{(\Gamma_t)_h} \mathbf{w} \cdot \mathbf{h} d\Gamma. \quad (1)$$

Here, $\mathbf{y}$ is the displacement, $\mathbf{w}$ is the virtual displacement, $\delta \mathbf{E}$ is the variation of the Green–Lagrange strain tensor, $\mathbf{S}$ is the second Piola–Kirchhoff stress tensor, ρ_0 is the mass density in the ZSS, $\mathbf{f}$ is the body force per unit mass, and $\mathbf{h}$ is the external stress vector applied on the subset $(\Gamma_t)_h$ of the boundary Γ_t .

2.1 EBZSS

In the EBTL method the ZSS is defined with a set of positions $\mathbf{X}_0^e$ for each element e . Positions of nodes from different elements mapping to the same node in the mesh do not have to be the same. In the reference state, $\mathbf{X}_{REF}$, all elements are connected by nodes, and we measure the displacement $\mathbf{y}$ from that connected state. The implementation of the method is simple. The deformation gradient tensor $\mathbf{F}$ is evaluated for each element:

$$\mathbf{F}^e \equiv \frac{\partial \mathbf{x}}{\partial \mathbf{X}_0^e}, \quad (2)$$

$$= \frac{\partial (\mathbf{X}_{REF} + \mathbf{y})}{\partial \mathbf{X}_0^e}, \quad (3)$$

where $\mathbf{x}$ is the position in the deformed configuration. The deformation gradient tensors for different elements are on different states, but the terms in Eq. (1), including the second term, do not depend on the orientation. Therefore the rest of the process is the same as it is in the total Lagrangian formulation.

2.2 IPBZSS

The key idea behind the EBZSS method was that, due to the objectivity, all the quantities seen in Eq. (1) can be computed with any orientation of the ZSS. We can extend the way we see $\mathbf{X}_0^e$ to integration-point counterpart of $\mathbf{X}_0^e$. As we did

with the EBZSS, we work with the reference domain. With the reference Jacobian

$$J_{REF} = \det \left(\frac{\partial \mathbf{X}_{REF}}{\partial \mathbf{X}_0} \right), \quad (4)$$

Eq. (1) can be rearranged as

$$\int_{\Omega_{REF}} \mathbf{w} \cdot \rho_0 \frac{d^2 \mathbf{y}}{dt^2} J_{REF}^{-1} d\Omega + \int_{\Omega_{REF}} \delta \mathbf{E} : \mathbf{S} J_{REF}^{-1} d\Omega - \int_{\Omega_{REF}} \mathbf{w} \cdot \rho_0 \mathbf{f} J_{REF}^{-1} d\Omega = \int_{(\Gamma_t)_h} \mathbf{w} \cdot \mathbf{h} d\Gamma. \quad (5)$$

In our implementation, we use the natural coordinates, with covariant basis vectors

$$\mathbf{g}_I = \frac{\partial \mathbf{x}}{\partial \xi^I}, \quad (6)$$

$$\mathbf{G}_I = \frac{\partial \mathbf{X}}{\partial \xi^I}, \quad (7)$$

where ξ^I is the parametric coordinate, and $I = 1, \dots, n_{pd}$, with n_{pd} being the number of parametric dimensions. The contravariant basis vectors can be calculated with the metric tensor components as

$$\mathbf{g}^I = g^{IJ} \mathbf{g}_J, \quad (8)$$

$$\mathbf{G}^I = G^{IJ} \mathbf{G}_J, \quad (9)$$

where

$$g_{IJ} = \mathbf{g}_I \cdot \mathbf{g}_J, \quad (10)$$

$$G_{IJ} = \mathbf{G}_I \cdot \mathbf{G}_J, \quad (11)$$

$$[g^{IJ}] = [g_{IJ}]^{-1}, \quad (12)$$

$$[G^{IJ}] = [G_{IJ}]^{-1}. \quad (13)$$

With those vectors, we can express the deformation gradient tensor:

$$\mathbf{F} = \mathbf{g}_I \mathbf{G}^I, \quad (14)$$

and the Cauchy–Green deformation tensor:

$$\mathbf{C} = \mathbf{F}^T \cdot \mathbf{F} \quad (15)$$

$$= \mathbf{G}^I \mathbf{g}_I \cdot \mathbf{g}_J \mathbf{G}^J \quad (16)$$

$$= g_{IJ} \mathbf{G}^I \mathbf{G}^J. \quad (17)$$

The Jacobian, $J = \det \mathbf{F}$, can be expressed as

$$J^2 = \det \mathbf{C}. \quad (18)$$

We can write $\det \mathbf{C}$ as

$$\det \mathbf{C} = \frac{\det [g_{IJ}]}{\det [G_{IJ}]}, \quad (19)$$

and from that we obtain

$$J = \left(\frac{\det [g_{IJ}]}{\det [G_{IJ}]} \right)^{\frac{1}{2}}. \quad (20)$$

We define the covariant basis vectors corresponding to $\mathbf{X}_{\text{REF}}$:

$$(\mathbf{G}_{\text{REF}})_I = \frac{\partial \mathbf{X}_{\text{REF}}}{\partial \xi^I}, \quad (21)$$

and the components of the metric tensor are

$$(G_{\text{REF}})_{IJ} = (\mathbf{G}_{\text{REF}})_I \cdot (\mathbf{G}_{\text{REF}})_J. \quad (22)$$

In Eq. (20), we replace $\mathbf{g}_I$ and $\mathbf{g}_J$ with $(\mathbf{G}_{\text{REF}})_I$ and $(\mathbf{G}_{\text{REF}})_J$ and obtain an alternative to the expression given by Eq. (4):

$$J_{\text{REF}} = \left(\frac{\det [(G_{\text{REF}})_{IJ}]}{\det [G_{IJ}]} \right)^{\frac{1}{2}}. \quad (23)$$

The Green–Lagrange strain tensor,

$$\mathbf{E} = \frac{1}{2} (\mathbf{C} - \mathbf{I}), \quad (24)$$

where $\mathbf{I}$ is the identity tensor, can be expressed with the contravariant basis vectors as

$$\mathbf{E} = \frac{1}{2} (g_{IJ} - G_{IJ}) \mathbf{G}^I \mathbf{G}^J. \quad (25)$$

The second Piola–Kirchhoff tensor can be expressed with the covariant basis vectors as

$$\mathbf{S} = S^{IJ} \mathbf{G}_I \mathbf{G}_J, \quad (26)$$

where S^{IJ} can be expressed with the components of the metric tensors. Thus, the inner product $\delta \mathbf{E} : \mathbf{S}$, and all the other quantities, in the weak form given by Eq. (5) can be evaluated without actually using the basis vectors $\mathbf{G}_I$. This justifies using $(G_{IJ})_k$ as the integration-point counterpart of $\mathbf{X}_0^e$, with $k = 1, \dots, n_{\text{int}}$, where n_{int} is the number of integration points. Note that G_{IJ} is symmetric, and therefore the IPBZSS representation will in 3D have $6 \times n_{\text{int}}$ parameters for each element.

2.3 EBZSS to IPBZSS

Converting the EBZSS representation to IPBZSS representation is straightforward. From given $\mathbf{X}_0^e$ we can calculate the covariant basis vectors at each integration point ξ_k :

$$(\mathbf{G}_I)_k = \left. \frac{\partial \mathbf{X}_0^e}{\partial \xi^I} \right|_{\xi=\xi_k}, \quad (27)$$

and obtain the components of the metric tensor from Eq. (11).

2.4 IPBZSS to EBZSS

Converting the IPBZSS representation to EBZSS representation, which we might need for visualization purposes, will, in general, not be exact because the IPBZSS has more parameters than the EBZSS. Given $(G_{IJ})_k$, we solve a steady-state element-based problem (with $\mathbf{f} = \mathbf{0}$ and $\mathbf{h} = \mathbf{0}$):

$$\int_{\Omega_{\text{REF}}^e} \delta \mathbf{E} : \mathbf{S} J_{\text{REF}}^{-1} d\Omega = 0, \quad (28)$$

and the solution to that, in the form $\mathbf{X}_{\text{REF}} + \mathbf{y}$, will be the EBZSS representation. If the stress calculated from the solution is zero, then the conversion will be exact. We note that to obtain a steady-state solution, we need to preclude translation and rigid-body rotation by imposing 6 appropriate constraints. To do that we first select three control points: A , B and C . We set all three components of $\mathbf{y}_A$ to be zero and constrain $\mathbf{y}_B$ to be in the direction $(\mathbf{X}_{\text{REF}})_B - (\mathbf{X}_{\text{REF}})_A$. The last constraint is $\mathbf{y}_C$ to be on the plane defined by the vector $((\mathbf{X}_{\text{REF}})_B - (\mathbf{X}_{\text{REF}})_A) \times ((\mathbf{X}_{\text{REF}})_C - (\mathbf{X}_{\text{REF}})_A)$.

2.5 An iterative method

Here we assume that we have a reasonably good initial guess for the IPBZSS (see Sect. 3). The iterative method below is used in calculating the IPBZSS that results in the target state associated with the given load. In the iterative method, we estimate $\mathbf{F}$ from the i th solution. We simply assume

$$\mathbf{F}^{i+1} = \mathbf{F}^i, \quad (29)$$

which means

$$(\mathbf{F}^{i+1})^{-1} = (\mathbf{F}^i)^{-1}. \quad (30)$$

The inverse of the deformation gradient tensor from the i th solution can be written as

$$(\mathbf{F}^i)^{-1} = (\mathbf{G}_I)^i (\mathbf{g}^I)^i. \quad (31)$$

Similarly, the inverse of the target deformation gradient tensor is

$$\left(\mathbf{F}^{i+1}\right)^{-1} = \left(\mathbf{G}_I\right)^{i+1} \left(\mathbf{G}_{\text{REF}}\right)^I. \quad (32)$$

Thus, we obtain the following equation:

$$\left(\mathbf{G}_K\right)^{i+1} \left(\mathbf{G}_{\text{REF}}\right)^K = \left(\mathbf{G}_K\right)^i \left(\mathbf{g}^K\right)^i. \quad (33)$$

Inner-producting both sides of this equation from the right with the covariant basis vectors corresponding to $\mathbf{X}_{\text{REF}}$, we obtain

$$\left(\mathbf{G}_K\right)^{i+1} \left(\mathbf{G}_{\text{REF}}\right)^K \cdot \left(\mathbf{G}_{\text{REF}}\right)_I = \left(\mathbf{G}_K\right)^i \left(\mathbf{g}^K\right)^i \cdot \left(\mathbf{G}_{\text{REF}}\right)_I, \quad (34)$$

which results in

$$\left(\mathbf{G}_I\right)^{i+1} = \left(\mathbf{G}_K\right)^i \left(\mathbf{g}^K\right)^i \cdot \left(\mathbf{G}_{\text{REF}}\right)_I. \quad (35)$$

The components of the metric tensor are

$$\begin{aligned} \left(G_{IJ}\right)^{i+1} &= \left(\left(\mathbf{G}_K\right)^i \left(\mathbf{g}^K\right)^i \cdot \left(\mathbf{G}_{\text{REF}}\right)_I\right) \\ &\quad \cdot \left(\left(\mathbf{G}_L\right)^i \left(\mathbf{g}^L\right)^i \cdot \left(\mathbf{G}_{\text{REF}}\right)_J\right). \end{aligned} \quad (36)$$

Rearranging the terms, we obtain

$$\begin{aligned} \left(G_{IJ}\right)^{i+1} &= \left(G_{KL}\right)^i \left(\left(\mathbf{g}^K\right)^i \cdot \left(\mathbf{G}_{\text{REF}}\right)_I\right) \\ &\quad \times \left(\left(\mathbf{g}^L\right)^i \cdot \left(\mathbf{G}_{\text{REF}}\right)_J\right). \end{aligned} \quad (37)$$

Thus, we update the components of the metric tensor without actually knowing the basis vectors $\mathbf{G}_I$.

Once we obtain the IPBZSS at the end of the iterations, we will have the option to convert it to EBZSS and use that in the subsequent $\mathbf{y}$ computations.

3 Initial guess based on the shell model of the artery

An analytical relationship between the ZS and reference states of straight-tube segments was given in [67]. The relationship was called “straight-tube ZSS template” in [69] and was extended to curved tubes. These were for the EBZSS. Here we directly build the IPBZSS instead of building the EBZSS first. This is simpler because with isogeometric discretization, especially with T-spline discretization, specifying conditions at integration points is far

more straightforward than imposing conditions on control points.

We start with the artery inner surface, which is what the medical images show. Typically, we cannot discern the wall thickness from the medical image. Therefore we first build the inner-surface mesh with T-splines. Then we build a T-spline volume mesh by extruding the surface elements by an estimated thickness.

In our notation here, $\mathbf{x}$ will now imply $\mathbf{X}_{\text{REF}}$, which is our “target” shape, and $\mathbf{X}$ will imply $\mathbf{X}_0$. We explain the method in the context of one element in the thickness direction. Extending the method to multiple elements is straightforward.

3.1 Inner-surface coordinates in the target state

The coordinate system we have here is similar to the one used for the shell modeling in [70]. We note that the “midsurface” of the shell formulation has been shifted to the inner surface here, and $\bullet$ indicates the inner surface. The basis vectors are

$$\bar{\mathbf{g}}_\alpha = \frac{\partial \bar{\mathbf{x}}}{\partial \xi^\alpha}, \quad (38)$$

where $\alpha = 1, \dots, n_{\text{sd}} - 1$, and the third direction is

$$\mathbf{n} = \frac{\bar{\mathbf{g}}_1 \times \bar{\mathbf{g}}_2}{\|\bar{\mathbf{g}}_1 \times \bar{\mathbf{g}}_2\|}. \quad (39)$$

The second fundamental form is defined as

$$\bar{b}_{\alpha\beta} = \frac{\partial \bar{\mathbf{g}}_\alpha}{\partial \xi^\beta} \cdot \mathbf{n}, \quad (40)$$

and the curvature tensor is

$$\hat{\kappa} = \underbrace{-\bar{b}_{\alpha\beta}}_{\hat{\kappa}_{\alpha\beta}} \bar{\mathbf{g}}^\alpha \bar{\mathbf{g}}^\beta. \quad (41)$$

Clearly, $\hat{\kappa}$ is symmetric.

For a given unit vector $\mathbf{t}$ on the surface, we obtain the curvature $\hat{\kappa}$ as

$$\hat{\kappa} = \mathbf{t} \cdot \hat{\kappa} \cdot \mathbf{t}. \quad (42)$$

If $\mathbf{t}$ is a principal direction,

$$\hat{\kappa} \cdot \mathbf{t} = \hat{\kappa} \mathbf{t}, \quad (43)$$

and $\hat{\kappa}$ is the corresponding principal curvature. The eigenvector can be expressed as

$$\mathbf{t} = t^\beta \bar{\mathbf{g}}_\beta. \quad (44)$$

Substituting this into Eq. (43) and inner-producting with $\bar{\mathbf{g}}^\alpha$, we obtain

$$\bar{g}_{\gamma\alpha} \left(\hat{\kappa}_{\bullet\beta}^\alpha - \hat{\kappa} \delta_{\beta}^\alpha \right) t^\beta = 0, \quad (45)$$

where the mixed components indicate

$$\hat{\kappa} = \hat{\kappa}_{\bullet\beta}^\alpha \bar{g}_\alpha \bar{g}^\beta. \quad (46)$$

Since the inverse of $[\bar{g}_{\gamma\alpha}]$ exists,

$$\det [\hat{\kappa}_{\bullet\beta}^\alpha - \hat{\kappa} \delta_{\beta}^\alpha] = 0, \quad (47)$$

$\hat{\kappa}$ is an eigenvalue of the matrix defined by the mixed components $\hat{\kappa}_{\bullet\beta}^\alpha$, and we will call the two eigenvalues $\hat{\kappa}_1$ and $\hat{\kappa}_2$. For $\hat{\kappa}_1 > \hat{\kappa}_2$ we obtain the corresponding eigenvectors:

$$\mathbf{t}_1 = (t_1)^\beta \bar{\mathbf{g}}_\beta, \quad (48)$$

$$\mathbf{t}_2 = (t_2)^\beta \bar{\mathbf{g}}_\beta, \quad (49)$$

where $\mathbf{t}_1$ and $\mathbf{t}_2$ are unit vectors. From Eq. (43), we write

$$\hat{\kappa} \cdot \mathbf{t}_1 = \hat{\kappa}_1 \mathbf{t}_1, \quad (50)$$

$$\hat{\kappa} \cdot \mathbf{t}_2 = \hat{\kappa}_2 \mathbf{t}_2. \quad (51)$$

Since $\hat{\kappa}$ is symmetric,

$$\mathbf{t}_2 \cdot \hat{\kappa} \cdot \mathbf{t}_1 = \mathbf{t}_1 \cdot \hat{\kappa} \cdot \mathbf{t}_2. \quad (52)$$

Substituting Eqs. (50) and (51) into this, we obtain

$$\hat{\kappa}_1 \mathbf{t}_1 \cdot \mathbf{t}_2 = \hat{\kappa}_2 \mathbf{t}_1 \cdot \mathbf{t}_2. \quad (53)$$

Thus, the two vectors are orthonormal, and we can express the curvature tensor as

$$\hat{\kappa} = \hat{\kappa}_1 \mathbf{t}_1 \mathbf{t}_1 + \hat{\kappa}_2 \mathbf{t}_2 \mathbf{t}_2. \quad (54)$$

When $\hat{\kappa}_1 = \hat{\kappa}_2$, an arbitrary orthonormal set of $\mathbf{t}_1$ and $\mathbf{t}_2$ can be used in the above equation.

For more details on calculating the eigenvalues and eigenvectors, see Appendix A.

3.2 Inner-surface coordinates in the ZSS

Since the principal curvature directions $\mathbf{t}_1$ and $\mathbf{t}_2$ of the target shape are orthogonal to each other, we can build the ZSS shape using those directions. The basis vectors on the inner surface in the ZSS are

$$\bar{\mathbf{G}}_\alpha = \frac{\partial \bar{\mathbf{X}}}{\partial \xi^\alpha}, \quad (55)$$

and the third direction is

$$\mathbf{N} = \frac{\bar{\mathbf{G}}_1 \times \bar{\mathbf{G}}_2}{\|\bar{\mathbf{G}}_1 \times \bar{\mathbf{G}}_2\|}. \quad (56)$$

The stretches corresponding to those directions will be $\hat{\lambda}_1$ and $\hat{\lambda}_2$. Then the ZSS basis vectors are calculated from

$$\hat{\lambda}_1 \mathbf{t}_1 = \mathbf{F} \cdot \mathbf{t}_1, \quad (57)$$

$$\hat{\lambda}_2 \mathbf{t}_2 = \mathbf{F} \cdot \mathbf{t}_2. \quad (58)$$

That is

$$\hat{\lambda}_1 \mathbf{F}^{-1} \cdot \mathbf{t}_1 = \mathbf{t}_1, \quad (59)$$

$$\hat{\lambda}_2 \mathbf{F}^{-1} \cdot \mathbf{t}_2 = \mathbf{t}_2. \quad (60)$$

Because the third direction is orthogonal to $\mathbf{t}_1$ and $\mathbf{t}_2$, we can reduce these equations to

$$\hat{\lambda}_1 \bar{\mathbf{G}}_\beta \bar{\mathbf{g}}^\beta \cdot \mathbf{t}_1 = \mathbf{t}_1, \quad (61)$$

$$\hat{\lambda}_2 \bar{\mathbf{G}}_\beta \bar{\mathbf{g}}^\beta \cdot \mathbf{t}_2 = \mathbf{t}_2. \quad (62)$$

Substituting Eqs. (48) and (49) into these, we get

$$\hat{\lambda}_1 (t_1)^\alpha \bar{\mathbf{G}}_\alpha = (t_1)^\alpha \bar{\mathbf{g}}_\alpha, \quad (63)$$

$$\hat{\lambda}_2 (t_2)^\alpha \bar{\mathbf{G}}_\alpha = (t_2)^\alpha \bar{\mathbf{g}}_\alpha. \quad (64)$$

This can also be written as

$$\begin{bmatrix} \hat{\lambda}_1 & 0 \\ 0 & \hat{\lambda}_2 \end{bmatrix} \begin{bmatrix} (t_1)^1 & (t_1)^2 \\ (t_2)^1 & (t_2)^2 \end{bmatrix} \begin{bmatrix} \bar{\mathbf{G}}_1 \\ \bar{\mathbf{G}}_2 \end{bmatrix} = \begin{bmatrix} (t_1)^1 & (t_1)^2 \\ (t_2)^1 & (t_2)^2 \end{bmatrix} \begin{bmatrix} \bar{\mathbf{g}}_1 \\ \bar{\mathbf{g}}_2 \end{bmatrix}, \quad (65)$$

and from that we calculate the basis vectors as

$$\begin{bmatrix} \bar{\mathbf{G}}_1 \\ \bar{\mathbf{G}}_2 \end{bmatrix} = \begin{bmatrix} (t_1)^1 & (t_1)^2 \\ (t_2)^1 & (t_2)^2 \end{bmatrix}^{-1} \begin{bmatrix} \frac{1}{\hat{\lambda}_1} & 0 \\ 0 & \frac{1}{\hat{\lambda}_2} \end{bmatrix} \begin{bmatrix} (t_1)^1 & (t_1)^2 \\ (t_2)^1 & (t_2)^2 \end{bmatrix} \begin{bmatrix} \bar{\mathbf{g}}_1 \\ \bar{\mathbf{g}}_2 \end{bmatrix}. \quad (66)$$

3.3 Wall coordinates in the target state

The position in the target configuration is

$$\mathbf{x} = \bar{\mathbf{x}} + \mathbf{n}\vartheta, \quad (67)$$

where $0 \leq \vartheta \leq h_{\text{th}}$, and h_{th} is the wall thickness in the target configuration. The basis vectors will vary along the thickness direction as

$$\mathbf{g}_\alpha = \frac{\partial \mathbf{x}}{\partial \xi^\alpha} \quad (68)$$

$$= \bar{\mathbf{g}}_\alpha + \frac{\partial \mathbf{n}}{\partial \xi^\alpha} \vartheta \quad (69)$$

$$= \bar{\mathbf{g}}_\alpha - \bar{b}_{\alpha\gamma} \bar{\mathbf{g}}^\gamma \vartheta. \quad (70)$$

The second and third lines are explained in Appendix 1. The third coordinate is mapped as

$$\vartheta = \frac{1 + \xi^3}{2} h_{\text{th}}, \quad (71)$$

where $-1 \leq \xi^3 \leq 1$. The basis vector in the third direction is constant as

$$\mathbf{g}_3 = \frac{h_{\text{th}}}{2} \mathbf{n}, \quad (72)$$

and

$$\mathbf{g}^3 = \frac{2}{h_{\text{th}}} \mathbf{n}. \quad (73)$$

With that, the components of the metric tensor are

$$g_{\alpha\beta} = \bar{g}_{\alpha\beta} - 2\bar{b}_{\alpha\beta} \vartheta + \bar{b}_{\alpha\gamma} \bar{g}^{\gamma\delta} \bar{b}_{\beta\delta} \vartheta^2, \quad (74)$$

$$g_{3\alpha} = 0, \quad (75)$$

$$g_{\alpha 3} = 0, \quad (76)$$

$$g_{33} = \frac{h_{\text{th}}^2}{4}. \quad (77)$$

3.4 Wall coordinates in the ZSS

The position in the ZSS configuration is

$$\mathbf{X} = \bar{\mathbf{X}} + \mathbf{N} \vartheta_0, \quad (78)$$

where $0 \leq \vartheta_0 \leq (h_{\text{th}})_0$, and $(h_{\text{th}})_0$ is the wall thickness in the ZSS configuration. The basis vectors will vary along the thickness direction as

$$\mathbf{G}_\alpha = \frac{\partial \mathbf{X}}{\partial \xi^\alpha} \quad (79)$$

$$= \bar{\mathbf{G}}_\alpha + \frac{\partial \mathbf{N}}{\partial \xi^\alpha} \vartheta_0 \quad (80)$$

$$= \bar{\mathbf{G}}_\alpha - \bar{B}_{\alpha\gamma} \bar{\mathbf{G}}^\gamma \vartheta_0. \quad (81)$$

The curvature tensor in the ZSS configuration is

$$\hat{\kappa}_0 = (\hat{\kappa}_0)_1 \mathbf{t}_1 \mathbf{t}_1 + (\hat{\kappa}_0)_2 \mathbf{t}_2 \mathbf{t}_2. \quad (82)$$

From that,

$$\begin{aligned} \bar{B}_{\alpha\beta} &= -\hat{\kappa}_0 : \bar{\mathbf{G}}_\alpha \bar{\mathbf{G}}_\beta \\ &= -(\hat{\kappa}_0)_1 (\mathbf{t}_1 \cdot \bar{\mathbf{G}}_\alpha) (\mathbf{t}_1 \cdot \bar{\mathbf{G}}_\beta) \end{aligned} \quad (83)$$

$$- (\hat{\kappa}_0)_2 (\mathbf{t}_2 \cdot \bar{\mathbf{G}}_\alpha) (\mathbf{t}_2 \cdot \bar{\mathbf{G}}_\beta). \quad (84)$$

Similar to what we had for $\mathbf{t}_1$ and $\mathbf{t}_2$,

$$\lambda_3 \mathbf{F}^{-1} \cdot \mathbf{n} = \mathbf{n}, \quad (85)$$

which becomes

$$\lambda_3 \mathbf{G}_3 \mathbf{g}^3 \cdot \mathbf{n} = \mathbf{n}. \quad (86)$$

We substitute Eq. (73) into this and obtain

$$\mathbf{G}_3 = \frac{h_{\text{th}}}{2\lambda_3} \mathbf{n}, \quad (87)$$

and

$$\mathbf{G}^3 = \frac{2\lambda_3}{h_{\text{th}}} \mathbf{n}. \quad (88)$$

3.5 Calculating the components of the ZSS metric tensor at each integration point

For an integration point ξ , we can obtain the components of the metric tensor as

$$G_{\alpha\beta} = \bar{G}_{\alpha\beta} - 2\bar{B}_{\alpha\beta} \vartheta_0 + \bar{B}_{\alpha\gamma} \bar{G}^{\gamma\delta} \bar{B}_{\beta\delta} \vartheta_0^2, \quad (89)$$

$$G_{3\alpha} = 0, \quad (90)$$

$$G_{\alpha 3} = 0, \quad (91)$$

$$G_{33} = \frac{h_{\text{th}}^2}{4\lambda_3^2}. \quad (92)$$

The third coordinate can be obtained from

$$\vartheta_0 = \int_{-1}^{\xi^3} \frac{h_{\text{th}}}{2\lambda_3} d\xi^3. \quad (93)$$

Assuming incompressible material, $J = 1$,

$$\lambda_3 = \frac{A_0}{A}, \quad (94)$$

where

$$A^2 = \det [g_{\alpha\beta}], \quad (95)$$

$$A_0^2 = \det [G_{\alpha\beta}]. \quad (96)$$

The components of the matrix tensors are given by Eqs. (74) and (89).

Fig. 1 Y-shaped tube. Target state. The end diameters are 20, 14 and 10 mm

Fig. 2 Y-shaped tube. Mesh made of cubic and quartic T-splines. Red circles represent the control points. The parts with the quartic T-splines, obtained by order elevation [72], are around the two extraordinary points, each connected to six edges. (Color figure online)

3.6 Design of the ZSS

The design parameters are the principal curvatures $(\hat{\kappa}_0)_1$ and $(\hat{\kappa}_0)_2$, and the stretches $\hat{\lambda}_1$ and $\hat{\lambda}_2$ for each principal curvature direction. Those parameters can be determined from $\hat{\kappa}_1$ and $\hat{\kappa}_2$ of the target configuration.

As proposed in [69], the two principal directions are seen as circumferential and longitudinal directions, and $\hat{\kappa}_1$ is in the circumferential direction, giving us

$$(\hat{\kappa}_0)_1 = \frac{2\pi - \phi}{2\pi} \hat{\kappa}_1. \quad (97)$$

Here ϕ is the opening angle, which is seen after a longitudinal cut, based on artery experimental data [71]. The stretch in that direction, $\hat{\lambda}_1$, corresponds to λ_θ in [69] and that is deter-

Fig. 3 Y-shaped tube. Wall thickness distribution

Fig. 4 Y-shaped tube. The IPBZSS initial guess, shown using the EBZSS representation

mined from the 2D computations in [69]. We assume that in the longitudinal direction the ZSS configuration has zero curvature, $(\hat{\kappa}_0)_2 = 0$. The stretch in that direction, $\hat{\lambda}_2$, corresponds to λ_z in [69]. If at an integration point $\hat{\kappa}_2 \approx \hat{\kappa}_1$, we set $(\hat{\kappa}_0)_2 = (\hat{\kappa}_0)_1$ and $\hat{\lambda}_2 = \hat{\lambda}_1$. That makes the assignment of the principal directions less consequential.

4 Numerical examples

In the numerical examples we use the Fung's model [20, 38, 70] with $D_1 = 2.6447 \times 10^3$ Pa, $D_2 = 8.365$, and the Poisson's ratio $\nu = 0.45$. The energy-density function we use is in the form

$$\varphi_{\text{FR}}(\mathbf{C}) = D_1 \left(e^{D_2 \left(\text{tr} J^{-\frac{2}{3}} \mathbf{C} - 3 \right)} - 1 \right)$$

Fig. 5 Y-shaped tube. An element in the target state (top) and the corresponding IPBZSS initial guess, shown using the EBZSS representation (bottom)

Fig. 6 Y-shaped tube. Colored by $\|\mathbf{y}\|$ computed from the converged IPBZSS. (Color figure online)

$$+ \frac{1}{2} \kappa \left(\frac{1}{2} (J^2 - 1) - \ln J \right), \quad (98)$$

where

$$\kappa = \frac{4D_1 D_2 (1 + \nu)}{3(1 - 2\nu)}. \quad (99)$$

Fig. 7 Y-shaped tube. The von Mises strain, from the IPBZSS initial guess (top) and from the converged IPBZSS (bottom)

Fig. 8 Y-shaped tube. The converged IPBZSS element corresponding to the element in Fig. 5, shown using the EBZSS representation

We assume the pressure associated with the target shape is 92 mm Hg.

The initial guess for the iterations is determined as described in Sect. 3, with $\phi = \frac{5}{2}\pi$ and $\hat{\lambda}_2 = 1.05$. After the iterations, explained in Sect. 2.5, for comparison purposes, we convert the IPBZSS representation to EBZSS representation, with the method described in Sect. 2.4. With the EBZSS,

Fig. 9 Y-shaped tube. Colored by $\|\mathbf{y}\|$ computed after converting the converged IPBZSS to EBZSS. (Color figure online)

Fig. 10 Y-shaped tube. The von Mises strain, obtained after converting the converged IPBZSS to EBZSS

we compute $\mathbf{y}$ again, and compare that to what we obtained directly from the IPBZSS.

4.1 Y-shaped tube

The target state of the Y-shaped tube is shown in Fig. 1. The end diameters of the tube are 20, 14 and 10 mm. Figure 2 shows the T-spline mesh. The mesh is based on a mixture of cubic and quartic T-splines. The wall thickness distribution is smooth, outcome of solving the Laplace's equation over the inner surface, with Dirichlet boundary conditions at the tube ends, where the value specified is 0.1 times the end diameter. Figure 3 shows the thickness distribution. The volume mesh is built with one element (cubic Bézier element) in the thickness direction. The number of control points and elements are 5,180 and 2,592.

Fig. 11 Patient-specific aorta geometry. Target state, extracted from medical images

Fig. 12 Patient-specific aorta geometry. Mesh made of cubic and quartic T-splines. Red circles represent the control points. The parts with the quartic T-splines, obtained by order elevation [72], are around the eight extraordinary points. (Color figure online)

Since the IPBZSS cannot be visualized, we show it using the EBZSS representation. Figure 4 shows the initial guess for the IPBZSS. Figure 5 shows an element in the target state and the corresponding IPBZSS initial guess.

We iterate and obtain the converged IPBZSS. Figure 6 shows $\|\mathbf{y}\|$ computed from that. The maximum value of $\|\mathbf{y}\|$ is 1.279×10^{-14} mm. Figure 7 shows the von Mises strain, computed from the IPBZSS initial guess and from the converged IPBZSS. The converged IPBZSS element is shown in Fig. 8.

Remark 1 Figure 8 shows that the opening angle and the longitudinal stretch do not change much between the IPBZSS initial guess and the converted IPBZSS. However, the circumferential stretches are somewhat different. The initial

Fig. 13 Patient-specific aorta geometry. Wall thickness distribution

guess for the circumferential stretch was based on the 2D computations reported in [69]. Alternatively, we can estimate the stretch by an analytical solution, similar to the one described in [70].

We also compute $\mathbf{y}$ after converting the converged IPBZSS to EBZSS. Figure 9 shows $\|\mathbf{y}\|$ computed that way. The maximum value of $\|\mathbf{y}\|$ is 1.626×10^{-2} mm. Figure 10 shows the von Mises strain. There is no visible difference between the strains obtained from the IPBZSS directly and after conversion to EBZSS.

4.2 Patient-specific aorta geometry

The target state of the patient-specific geometry is shown in Fig. 11. Figure 12 shows the T-spline mesh.

The wall thickness distribution is smooth, outcome of solving the Laplace's equation over the inner surface, with Dirichlet boundary conditions at the tube ends, where the value specified is 0.08 times the end diameter. In some parts of the branched area the thickness exceeds the radius of curvature, and there we reduce the thickness to 0.8 times the radius of curvature. Figure 13 shows the thickness distribution.

The volume mesh is built again with one element (cubic Bézier element) in the thickness direction. The number of control points and elements are 9,244 and 4,360.

Figure 14 shows the initial guess for the IPBZSS. We again iterate and obtain the converged IPBZSS. Figure 15

Fig. 14 Patient-specific aorta geometry. The IPBZSS initial guess, shown using the EBZSS representation

Fig. 15 Patient-specific aorta geometry. Colored by $\|\mathbf{y}\|$ computed from the converged IPBZSS. (Color figure online)

shows $\|\mathbf{y}\|$ computed from that. The maximum value of $\|\mathbf{y}\|$ is 1.163×10^{-13} mm. Figure 16 shows the von Mises strain, computed from the IPBZSS initial guess and from the converged IPBZSS.

We again compute $\mathbf{y}$ after converting the converged IPBZSS to EBZSS. Figure 17 shows $\|\mathbf{y}\|$ computed that way. The maximum value of $\|\mathbf{y}\|$ is 1.057×10^{-1} mm. Figure 18 shows the von Mises strain. These results indicate that the EBZSS obtained by conversion from the IPBZSS is also a reasonable representation of the aorta ZSS.

Fig. 16 Patient-specific aorta geometry. The von Mises strain, from the IPBZSS initial guess (top) and from the converged IPBZSS (bottom)

5 Concluding remarks

We have introduced a new method for estimating the ZSS required in patient-specific arterial FSI computations, where the image-based arterial geometries do not come from the ZSS of the artery. The estimate is based on T-spline discretization of the arterial wall and is in the form of IPBZSS. The method has two main components. (1) An iterative method, which starts with a calculated initial guess, is used for computing the IPBZSS such that when a given pressure

Fig. 17 Patient-specific aorta geometry. Colored by $\|y\|$ computed after converting the converged IPBZSS to EBZSS. (Color figure online)

Fig. 18 Patient-specific aorta geometry. The von Mises strain, obtained after converting the converged IPBZSS to EBZSS

load is applied, the image-based target shape is matched. (2) A method, which is based on the shell model of the artery, is used for calculating the initial guess. The T-spline discretization enables dealing with complex arterial geometries, such as an aorta model with branches, while retaining

the desirable features of isogeometric discretization. The desirable features of higher-order basis functions of isogeometric discretization include being able to achieve a similar level of accuracy as with the linear basis functions, but using larger-size and much fewer elements, and being able to represent more complex shapes within an element. The IPBZSS is a convenient representation of the ZSS because with isogeometric discretization, especially with T-spline discretization, specifying conditions at integration points is more straightforward than imposing conditions on control points. Calculating the initial guess based on the shell model of the artery results in a more realistic initial guess. To show how the new ZSS estimation method performs, we first presented 3D test computations with a Y-shaped tube. Then we presented a 3D computation where the target geometry was coming from medical image of a human aorta, and the model included the aorta branches.

Acknowledgements This work was supported in part by JST-CREST; Grant-in-Aid for Scientific Research (S) 26220002 from the Ministry of Education, Culture, Sports, Science and Technology of Japan (MEXT); 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 18J14680. 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 distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

A Eigenvalues and eigenvectors of the curvature tensor

In calculating the eigenvalues from the quadratic equation

$$\hat{\kappa}^2 - \hat{\kappa} \operatorname{tr} [\hat{\kappa}_{\bullet\beta}^{\alpha}] + \det [\hat{\kappa}_{\bullet\beta}^{\alpha}] = 0, \quad (100)$$

we obtain

$$\hat{\kappa} = \frac{\operatorname{tr} [\hat{\kappa}_{\bullet\beta}^{\alpha}] \pm \sqrt{\operatorname{tr}^2 [\hat{\kappa}_{\bullet\beta}^{\alpha}] - 4 \det [\hat{\kappa}_{\bullet\beta}^{\alpha}]}}{2}. \quad (101)$$

This can also be written as

$$\hat{\kappa}_l = \frac{\operatorname{tr} [\hat{\kappa}_{\bullet\beta}^{\alpha}] - (-1)^l \sqrt{\operatorname{tr}^2 [\hat{\kappa}_{\bullet\beta}^{\alpha}] - 4 \det [\hat{\kappa}_{\bullet\beta}^{\alpha}]}}{2}, \quad (102)$$

with $l = 1, 2$. In calculating the eigenvectors, we work with $(v_1)^{\alpha}$ and $(v_2)^{\alpha}$, the contravariant components of the non-normalized eigenvectors. The vectors are normalized with the scaling factors τ_1 and τ_2 :

$$(t_l)^{\alpha} = \tau_l (v_l)^{\alpha}. \quad (103)$$

With that, we obtain the relationship

$$(\tau_l)^2 (v_l)^{\alpha} g_{\alpha\beta} (v_l)^{\beta} = 1, \quad (104)$$

which yields

$$\tau_l = ((v_l)^{\alpha} g_{\alpha\beta} (v_l)^{\beta})^{-\frac{1}{2}}. \quad (105)$$

Given $(v_1)^{\alpha}$, which we will explain later how to calculate, from Eqs. (103) and (105) we obtain

$$(t_1)^{\alpha} = ((v_1)^{\gamma} g_{\gamma\delta} (v_1)^{\delta})^{-\frac{1}{2}} (v_1)^{\alpha}. \quad (106)$$

We calculate $\mathbf{t}_2$ from the orthogonality between $\mathbf{t}_1$ and $\mathbf{t}_2$:

$$\mathbf{t}_1 \cdot \mathbf{t}_2 = 0. \quad (107)$$

With the substitutions $\mathbf{t}_1 = (t_1)^{\alpha} \bar{\mathbf{g}}_{\alpha}$ and $\mathbf{t}_2 = (t_2)_{\beta} \bar{\mathbf{g}}^{\beta}$, we obtain

$$(t_1)^{\alpha} (t_2)_{\alpha} = 0. \quad (108)$$

The expression

$$(v_2)_{\alpha} = R_{\alpha\beta} (t_1)^{\beta}, \quad (109)$$

where

$$[R_{\alpha\beta}] = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \quad (110)$$

satisfies Eq. (108). To calculate τ_2 , we start from the covariant version of (105):

$$\tau_2 = ((v_2)_{\alpha} g^{\alpha\beta} (v_2)_{\beta})^{-\frac{1}{2}}. \quad (111)$$

Substituting Eq. (109) into Eq. (111), we obtain

$$(\tau_2)^{-2} = R_{\alpha\gamma} (t_1)^{\gamma} g^{\alpha\beta} R_{\beta\delta} (t_1)^{\delta}. \quad (112)$$

By expanding the matrices and rearranging, we obtain the relationship

$$R_{\alpha\gamma} g^{\alpha\beta} R_{\beta\delta} = \det[g^{\gamma\delta}] [g^{\gamma\delta}]^{-1} \quad (113)$$

$$= \det^{-1}[g_{\gamma\delta}] g_{\gamma\delta}. \quad (114)$$

Combining Eqs. (112) and (114), we obtain

$$(\tau_2)^{-2} = \det^{-1}[g_{\gamma\delta}] \underbrace{g_{\gamma\delta} (t_1)^\gamma (t_1)^\delta}_{=1}, \quad (115)$$

and then

$$\tau_2 = \det^{\frac{1}{2}}[g_{\gamma\delta}]. \quad (116)$$

Using Eqs. (109) and (116) in the expression

$$(t_2)_\alpha = \tau_2 (v_2)_\alpha, \quad (117)$$

we obtain

$$(t_2)_\alpha = \det^{\frac{1}{2}}[g_{\gamma\delta}] R_{\alpha\beta} (t_1)^\beta. \quad (118)$$

Now we explain how to calculate $(v_1)^\alpha$. From Eqs. (43) and (100), we write the equations that $(v_1)^\alpha$ and $\hat{k}_1$ satisfy:

$$(\hat{k}_{\bullet 1}^1 - \hat{k}_1)(v_1)^1 + \hat{k}_{\bullet 2}^1 (v_1)^2 = 0, \quad (119)$$

$$\hat{k}_{\bullet 1}^2 (v_1)^1 + (\hat{k}_{\bullet 2}^2 - \hat{k}_1)(v_1)^2 = 0, \quad (120)$$

$$(\hat{k}_{\bullet 1}^1 - \hat{k}_1)(\hat{k}_{\bullet 2}^2 - \hat{k}_1) - \hat{k}_{\bullet 1}^2 \hat{k}_{\bullet 2}^1 = 0. \quad (121)$$

If $\hat{k}_{\bullet 2}^1$ or $\hat{k}_1 - \hat{k}_{\bullet 1}^1$ is nonzero, we can write, from Eq. (119),

$$[(v_1)^\alpha] = \begin{bmatrix} \hat{k}_{\bullet 2}^1 \\ \hat{k}_1 - \hat{k}_{\bullet 1}^1 \end{bmatrix}. \quad (122)$$

If $\hat{k}_1 - \hat{k}_{\bullet 2}^2$ or $\hat{k}_{\bullet 1}^2$ is nonzero, we can write, from Eq. (120),

$$[(v_1)^\alpha] = \begin{bmatrix} \hat{k}_1 - \hat{k}_{\bullet 2}^2 \\ \hat{k}_{\bullet 1}^2 \end{bmatrix}. \quad (123)$$

If both $\hat{k}_1 - \hat{k}_{\bullet 1}^1$ and $\hat{k}_{\bullet 1}^2$ are zero, we can write,

$$[(v_1)^\alpha] = \begin{bmatrix} 1 \\ 0 \end{bmatrix}. \quad (124)$$

Taking into account the numerical stability issues related to dealing with very small numbers, we select one of these three equations in the order given below.

1. If $|\hat{k}_1 - \hat{k}_{\bullet 1}^1| > \varepsilon |\hat{k}_{\bullet 2}^1|$, where $\varepsilon = 10^{-6}$, then Eq. (122) is selected.
2. If $|\hat{k}_1 - \hat{k}_{\bullet 1}^1| \leq \varepsilon |\hat{k}_{\bullet 2}^1|$ and if $|\hat{k}_{\bullet 1}^2| > \varepsilon |\hat{k}_1 - \hat{k}_{\bullet 2}^2|$, then Eq. (123) is selected.
3. Otherwise Eq. (124) is selected.

B Derivative of the normal vector

Derivative of the normal vector with respect to ξ^α can be obtained as follows:

$$\bar{\mathbf{n}}_{,\alpha} = \frac{\partial}{\partial \xi^\alpha} \left(\frac{\bar{\mathbf{g}}_1 \times \bar{\mathbf{g}}_2}{(\bar{\mathbf{g}}_1 \times \bar{\mathbf{g}}_2) \cdot \mathbf{n}} \right) \quad (125)$$

$$= (\mathbf{I} - \mathbf{nn}) \cdot \frac{\bar{\mathbf{g}}_{1,\alpha} \times \bar{\mathbf{g}}_2 + \bar{\mathbf{g}}_1 \times \bar{\mathbf{g}}_{2,\alpha}}{(\bar{\mathbf{g}}_1 \times \bar{\mathbf{g}}_2) \cdot \mathbf{n}} \quad (126)$$

$$= (\mathbf{I} - \mathbf{nn}) \cdot \frac{\bar{\mathbf{g}}_{1,\alpha} \times (\mathbf{n} \times \bar{\mathbf{g}}^1) + \bar{\mathbf{g}}_{2,\alpha} \times (\mathbf{n} \times \bar{\mathbf{g}}^2)}{(\bar{\mathbf{g}}^1 \times \bar{\mathbf{g}}^2) \cdot \mathbf{n} (\bar{\mathbf{g}}_1 \times \bar{\mathbf{g}}_2) \cdot \mathbf{n}} \quad (127)$$

$$= (\mathbf{I} - \mathbf{nn}) \cdot (\bar{\mathbf{g}}_{\beta,\alpha} \times (\mathbf{n} \times \bar{\mathbf{g}}^\beta)) \quad (128)$$

$$= (\mathbf{I} - \mathbf{nn}) \cdot ((\bar{\mathbf{g}}_{\beta,\alpha} \cdot \bar{\mathbf{g}}^\beta) \mathbf{n} - (\bar{\mathbf{g}}_{\beta,\alpha} \cdot \mathbf{n}) \bar{\mathbf{g}}^\beta) \quad (129)$$

$$= -(\bar{\mathbf{g}}_{\beta,\alpha} \cdot \mathbf{n}) \bar{\mathbf{g}}^\beta + \underbrace{\mathbf{n} (\mathbf{n} \cdot \bar{\mathbf{g}}^\beta)}_{=0} (\bar{\mathbf{g}}_{\beta,\alpha} \cdot \mathbf{n}) \quad (130)$$

$$= -\bar{\mathbf{g}}^\beta \bar{\mathbf{g}}_{\beta,\alpha} \cdot \mathbf{n} \quad (131)$$

$$= -\bar{\mathbf{g}}^\beta \bar{b}_{\alpha\beta}. \quad (132)$$

In the derivation, we used the following relationships, which generally hold:

$$\mathbf{g}_1 = \frac{\mathbf{g}^2 \times \mathbf{n}}{(\mathbf{g}^1 \times \mathbf{g}^2) \cdot \mathbf{n}}, \quad (133)$$

$$\mathbf{g}_2 = \frac{\mathbf{n} \times \mathbf{g}^1}{(\mathbf{g}^1 \times \mathbf{g}^2) \cdot \mathbf{n}}, \quad (134)$$

$$(\mathbf{g}^1 \times \mathbf{g}^2) \cdot \mathbf{n} = ((\mathbf{g}_1 \times \mathbf{g}_2) \cdot \mathbf{n})^{-1}. \quad (135)$$

References

1. Torii R, Oshima M, Kobayashi T, Takagi K, Tezduyar TE (2004) Computation of cardiovascular fluid–structure interactions with the DSD/SST method. In: Proceedings of the 6th world congress on computational mechanics (CD-ROM), Beijing, China
2. Torii R, Oshima M, Kobayashi T, Takagi K, Tezduyar TE (2004) Influence of wall elasticity on image-based blood flow simulations. Trans Jpn Soc Mech Eng Ser A 70:1224–1231. <https://doi.org/10.1299/kikaia.70.1224> in Japanese
3. Torii R, Oshima M, Kobayashi T, Takagi K, Tezduyar TE (2006) Computer modeling of cardiovascular fluid–structure interactions with the deforming-spatial-domain/stabilized space–time formulation. Comput Methods Appl Mech Eng 195:1885–1895. <https://doi.org/10.1016/j.cma.2005.05.050>
4. Torii R, Oshima M, Kobayashi T, Takagi K, Tezduyar TE (2006) Fluid–structure interaction modeling of aneurysmal conditions with high and normal blood pressures. Comput Mech 38:482–490. <https://doi.org/10.1007/s00466-006-0065-6>
5. 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)
6. 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>

7. 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
8. 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
9. Bazilevs Y, Calo VM, Tezduyar TE, Hughes TJR (2007) $YZ\beta$ discontinuity-capturing for advection-dominated processes with application to arterial drug delivery. *Int J Numer Methods Fluids* 54:593–608. <https://doi.org/10.1002/fld.1484>
10. Bazilevs Y, Calo VM, Hughes TJR, Zhang Y (2008) Isogeometric fluid–structure interaction: theory, algorithms, and computations. *Comput Mech* 43:3–37
11. Isaksen JG, Bazilevs Y, Kvamsdal T, Zhang Y, Kaspersen JH, Waterloo K, Romner B, Ingebrigtsen T (2008) Determination of wall tension in cerebral artery aneurysms by numerical simulation. *Stroke* 39:3172–3178
12. 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
13. 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
14. 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
15. Sugiyama K, Ii S, Takeuchi S, Takagi S, Matsumoto Y (2010) Full Eulerian simulations of biconcave neo-Hookean particles in a Poiseuille flow. *Comput Mech* 46:147–157
16. 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
17. Bazilevs Y, del Alamo JC, Humphrey JD (2010) From imaging to prediction: emerging non-invasive methods in pediatric cardiology. *Progr Pediatr Cardiol* 30:81–89
18. Hsu M-C, Bazilevs Y (2011) Blood vessel tissue prestress modeling for vascular fluid–structure interaction simulations. *Finite Elem Anal Des* 47:593–599
19. Yao JY, Liu GR, Narmoneva DA, Hinton RB, Zhang Z-Q (2012) Immersed smoothed finite element method for fluid–structure interaction simulation of aortic valves. *Comput Mech* 50:789–804
20. Bazilevs Y, Takizawa K, Tezduyar TE (2013) Computational fluid–structure interaction: methods and applications. Wiley, New York ISBN 978-0470978771
21. 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>
22. 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>
23. Esmaily-Moghadam M, Bazilevs Y, Marsden AL (2013) A new preconditioning technique for implicitly coupled multidomain simulations with applications to hemodynamics. *Comput Mech* 52:1141–1152. <https://doi.org/10.1007/s00466-013-0868-1>
24. 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>
25. Yao J, Liu GR (2014) A matrix-form GSM-CFD solver for incompressible fluids and its application to hemodynamics. *Comput Mech* 54:999–1012. <https://doi.org/10.1007/s00466-014-0990-8>
26. 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>
27. 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>
28. 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>
29. 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
30. Hughes TJR, Liu WK, Zimmermann TK (1981) Lagrangian–Eulerian finite element formulation for incompressible viscous flows. *Comput Methods Appl Mech Eng* 29:329–349
31. Tezduyar TE, Sathe S, Cragin T, Nanna B, Conklin BS, Pausewang J, Schwaab M (2007) Modeling of fluid–structure interactions with the space–time finite elements: arterial fluid mechanics. *Int J Numer Methods Fluids* 54:901–922. <https://doi.org/10.1002/fld.1443>
32. Tezduyar TE, Sathe S, Schwaab M, Conklin BS (2008) Arterial fluid mechanics modeling with the stabilized space–time fluid–structure interaction technique. *Int J Numer Methods Fluids* 57:601–629. <https://doi.org/10.1002/fld.1633>
33. Tezduyar TE, Schwaab M, Sathe S (2009) Sequentially-coupled arterial fluid–structure interaction (SCAFSI) technique. *Comput Methods Appl Mech Eng* 198:3524–3533. <https://doi.org/10.1016/j.cma.2008.05.024>
34. Takizawa K, Christopher J, Tezduyar TE, Sathe S (2010) Space–time finite element computation of arterial fluid–structure interactions with patient-specific data. *Int J Numer Methods Biomed Eng* 26:101–116. <https://doi.org/10.1002/cnm.1241>
35. Tezduyar TE, Takizawa K, Moorman C, Wright S, Christopher J (2010) Multiscale sequentially-coupled arterial FSI technique. *Comput Mech* 46:17–29. <https://doi.org/10.1007/s00466-009-0423-2>
36. Takizawa K, Moorman C, Wright S, Christopher J, Tezduyar TE (2010) Wall shear stress calculations in space–time finite element computation of arterial fluid–structure interactions. *Comput Mech* 46:31–41. <https://doi.org/10.1007/s00466-009-0425-0>
37. Takizawa K, Moorman C, Wright S, Purdue J, McPhail T, Chen PR, Warren J, Tezduyar TE (2011) Patient-specific arterial fluid–structure interaction modeling of cerebral aneurysms. *Int J Numer Methods Fluids* 65:308–323. <https://doi.org/10.1002/fld.2360>
38. Tezduyar TE, Takizawa K, Brummer T, Chen PR (2011) Space–time fluid–structure interaction modeling of patient-specific cerebral aneurysms. *Int J Numer Methods Biomed Eng* 27:1665–1710. <https://doi.org/10.1002/cnm.1433>
39. Takizawa K, Brummer T, Tezduyar TE, Chen PR (2012) A comparative study based on patient-specific fluid–structure interaction modeling of cerebral aneurysms. *J Appl Mech* 79:010908. <https://doi.org/10.1115/1.4005071>
40. 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>
41. Takizawa K, Schjodt K, Puntel A, Kostov N, Tezduyar TE (2012) Patient-specific computer modeling of blood flow in cerebral arter-

- ies with aneurysm and stent. *Comput Mech* 50:675–686. <https://doi.org/10.1007/s00466-012-0760-4>
42. Takizawa K, Schjodt K, Puntel A, Kostov N, Tezduyar TE (2013) Patient-specific computational fluid mechanics of cerebral arteries with aneurysm and stent. In: Li S, Qian D (eds) *Multiscale simulations and mechanics of biological materials*, Chapter 7. Wiley, New York, pp 119–147 ISBN 978-1-118-35079-9
 43. 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>
 44. 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>
 45. Takizawa K, Bazilevs Y, Tezduyar TE, Long CC, Marsden AL, Schjodt K (2014) Patient-specific cardiovascular fluid mechanics analysis with the ST and ALE-VMS methods. In: Idelsohn SR (ed) *Numerical simulations of coupled problems in engineering*, volume 33 of computational methods in applied sciences, chapter 4. Springer, Berlin, pp 71–102. https://doi.org/10.1007/978-3-319-06136-8_4 ISBN 978-3-319-06135-1
 46. 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>
 47. 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>
 48. 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>
 49. 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 computational mechanics. Springer, Berlin, pp 77–99. https://doi.org/10.1007/978-3-319-59548-1_6 ISBN 978-3-319-59547-4
 50. 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>
 51. 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
 52. 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>
 53. 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
 54. Takizawa K, Tezduyar TE, Uchikawa H, Terahara T, Sasaki T, Yoshida A (2018) Mesh refinement influence and cardiac-cycle flow periodicity in aorta flow analysis with isogeometric discretization. *Comput Fluids*. <https://doi.org/10.1016/j.compfluid.2018.05.025>
 55. 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>
 56. 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>
 57. Takizawa K, Bazilevs Y, Tezduyar TE, Hsu M-C, Øiseth O, Mathisen KM, Kostov N, McIntyre S (2014) Computational engineering analysis and design with ALE-VMS and ST methods. In: Idelsohn SR (ed) *Numerical simulations of coupled problems in engineering*, volume 33 of computational methods in applied sciences, chapter 13. Springer, Berlin, pp 321–353. https://doi.org/10.1007/978-3-319-06136-8_13 ISBN 978-3-319-06135-1
 58. 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>
 59. 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>
 60. 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
 61. Takizawa K, Tezduyar TE, Otaguro 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>
 62. 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>
 63. 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>
 64. 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
 65. 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>
 66. Tezduyar TE, Cragin T, Sathe S, Nanna B (2007) FSI computations in arterial fluid mechanics with estimated zero-pressure arterial geometry. In: Onate E, Garcia J, Bergan P, Kvamsdal T (eds) *Marine 2007*. CIMNE, Barcelona, Spain
 67. 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>

68. 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
69. 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>
70. Takizawa K, Tezduyar TE, Sasaki T (2018) Isogeometric hyperelastic shell analysis with out-of-plane deformation mapping. *Comput Mech*. <https://doi.org/10.1007/s00466-018-1616-3>
71. Holzapfel GA, Sommer G, Auer M, Regitnig P, Ogden RW (2007) Layer-specific 3D residual deformations of human aortas with non-atherosclerotic intimal thickening. *Ann Biomed Eng* 35:530–545
72. Scott MA, Simpson RN, Evans JA, Lipton S, Bordas SPA, Hughes TJR, Sederberg TW (2013) Isogeometric boundary element analysis using unstructured T-splines. *Comput Methods Appl Mech Eng* 254:197–221. <https://doi.org/10.1016/j.cma.2012.11.001>

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