Space–Time method for flow computations with slip interfaces and topology changes (ST-SI-TC)

Kenji Takizawa^{a,*}, Tayfun E. Tezduyar^b, Shohei Asada^a, Takashi Kuraishi^a

^a*Department of Modern Mechanical Engineering, Waseda University
1-6-1 Nishi-Waseda, Shinjuku-ku, Tokyo 169-8050, Japan*

^b*Mechanical Engineering, Rice University – MS 321
6100 Main Street, Houston, TX 77005, USA*

Abstract

The Space–Time Variational Multiscale (ST-VMS) method was introduced to function as a moving-mesh method. It is the VMS version of the Deforming–Spatial-Domain/Stabilized ST (DSD/SST) method. It has reasonably good turbulence modeling features and serves as a core computational method. The ST Slip Interface (ST-SI) method was introduced to address the challenge involved in high-resolution representation of the boundary layers near spinning solid surfaces. The mesh covering a spinning solid surface spins with it and thus maintains the high-resolution representation near it. The ST-TC method was introduced for moving-mesh computation of flow problems with topology changes, such as contact between solid surfaces. It deals with the TC while maintaining high-resolution boundary layer representation near solid surfaces. The “ST-SI-TC” method we introduce here integrates the ST-SI and ST-TC methods in the ST-VMS framework. It enables accurate flow analysis when we have a spinning solid surface that is in contact with a solid surface. We present two test computations with the ST-SI-TC method, and they are both with models of flow around a rotating tire with road contact and prescribed deformation, one with a 2D model, and one with a 3D model.

Keywords: Space–Time Variational Multiscale method, ST-VMS, ST Slip

*Corresponding author. Tel.: +81-3-5286-2143.

Email addresses: Kenji.Takizawa@tafsm.org (Kenji Takizawa), tezduyar@tafsm.org (Tayfun E. Tezduyar)

1. Introduction

The Space–Time Variational Multiscale (ST-VMS) method <1; 2> was introduced to function as a moving-mesh method, mainly targeting fluid–structure interactions (FSI) and, in a more general context, flows with moving boundaries and interfaces (MBI). It is the VMS version of the Deforming–Spatial–Domain/Stabilized ST (DSD/SST) method <3; 4; 5>. The VMS components are from the residual-based VMS (RBVMS) method <6; 7; 8; 9>. The stabilization components of the original DSD/SST method are the Streamline–Upwind/Petrov–Galerkin (SUPG) <10> and Pressure–Stabilizing/Petrov–Galerkin (PSPG) <3> stabilizations, and for that the method is now also called “ST-SUPS” (the acronym was first used in <11>). The ST-VMS method is an augmented version of the ST-SUPS method, with two additional stabilization terms beyond the three that the ST-SUPS method has. Alternatively, one can view the ST-SUPS method as a “reduced” version of the ST-VMS method. The ALE-VMS method <12; 13>, first presented in <14>, is the VMS version of the Arbitrary Lagrangian–Eulerian (ALE) finite element method <15>, which is a more commonly used moving-mesh method. With the moving-mesh methods, by moving the fluid mechanics mesh to follow a fluid–solid interface, we can control the mesh resolution near the interface, have high-resolution representation of the boundary layers, and obtain accurate solutions in such critical flow regions.

The RBVMS and ALE-VMS methods have been successfully used for different types of FSI, MBI and fluid mechanics problems (see, for example, <16; 17; 14; 18; 19; 20; 21; 22; 13; 23; 24; 11; 25; 26; 27; 28; 29; 30; 31; 32; 33; 34; 35; 36; 37; 38; 39>). The ST-SUPS and ST-VMS methods have also been successfully used for different classes of FSI, MBI and fluid mechanics problems. The classes of problems include spacecraft parachute FSI <11; 40; 41; 42; 43; 44>,

wind-turbine aerodynamics <21; 45; 46; 11; 47; 40; 48; 28; 49>, flapping-wing aerodynamics <50; 51; 52; 11; 53; 54; 55; 40; 48; 56>, cardiovascular fluid mechanics <57; 58; 54; 59; 40; 48; 60; 61>, spacecraft aerodynamics <62; 63>, thermo-fluid analysis of ground vehicles and their tires <64>, thermo-fluid analysis of disk brakes <65>, flow-driven string dynamics in turbomachinery <66>, and flow analysis for the turbine part of a turbocharger <67>.

The ST Slip Interface (ST-SI) method <49> was introduced to retain the desirable moving-mesh features of the ST-VMS method when we have spinning solid surfaces, such as a wind-turbine rotor. With the ST-SI method, the mesh covering the spinning solid surface spins with it and we maintain the high-resolution representation of the boundary layers. The first method to accomplish that objective in the ST framework was the Shear-Slip Mesh Update Method (SSMUM) <68; 69; 70>, which was introduced in <68; 69> and named “SSMUM” in <70>. Later the objective was also accomplished in ST framework with the ST/NURBS Mesh Update Method (STNMUM), which was introduced in <50; 51; 52> and named in <47>. The STNMUM is a general-purpose method with a wide scope that includes spinning solid surfaces, in which case NURBS basis functions are used for the temporal representation of the spinning motion, mesh motion and also in remeshing. With the spinning motion represented by quadratic temporal NURBS basis functions, and with sufficient number of temporal patches for a full rotation, the circular paths associated with the spinning motion are represented exactly. An added “secondary mapping” <1; 50; 2; 11> enables also specifying a constant angular velocity corresponding to the invariant speeds along those paths. The STNMUM was successfully used in <47> in ST-VMS computation of flow past a wind-turbine rotor, with the tower included in the model. The STNMUM is more general than the SSMUM and simpler to use. Even if it meant giving up that generality and limiting the use to spinning structures, we wanted something that is even simpler to use. That motivated the development of the ST-SI method.

The starting point in the development of the ST-SI method was the version of the ALE-VMS method designed for computations with “sliding interfaces”

<24; 71>. This ALE-VMS version has been used successfully in a number of computations with spinning solid surfaces <24; 30; 26; 29; 71>. In the ST-SI method, interface terms similar to those in the ALE-VMS version are added to the ST-VMS formulation to account for the compatibility conditions for the velocity and stress. While having high-resolution representation of the boundary layers near a spinning solid surface, by using NURBS functions in temporal representation of the spinning motion, the ST-SI method has exact representation of the circular paths associated with the spinning. The ST-SI method has been successfully applied to aerodynamic analysis of vertical-axis wind turbines <49>, thermo-fluid analysis of disk brakes <65>, flow-driven string dynamics in turbomachinery <66>, and flow analysis for the turbine part of a turbocharger <67>.

The ST Topology Change (ST-TC) method <54> was introduced for moving-mesh computation of flow problems with TC, such as contact between solid surfaces. As commented in <54>, the ST-SUPS and ST-VMS methods, when supplemented with robust mesh update methods, have been proven effective even 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. For examples of that from earlier computations, see the references mentioned in <54>. In some classes of flow problems, however, the “nearness” that can be modeled with a moving-mesh method without actually bringing the surfaces into contact might not be “near” enough. A good example is flow computation of heart valves, where the flow has to be completely blocked at contact. The Fluid–Solid Interface-Tracking/Interface-Capturing Technique (FSITICT) <72> was motivated by such flow problems. In the FSITICT, we track the interfaces wherever and whenever we can with a moving mesh, and capture over that moving mesh the interfaces we cannot track, specifically the interfaces where and when we need to have an actual contact between the solid surfaces. As commented in <61>, essentially, the FSITICT is based on giving up on the interface-tracking accuracy in the parts of the domain where and when we expect an actual contact. The FSITICT has been successfully extended

in <33> to 3D FSI computation of a bioprosthetic heart valve. In that case the interface-tracking technique was the ALE-VMS method, and the interface-capturing technique was a variational immersed-boundary method, later referred to as the Immersogeometric method in <73>.

The ST-TC method does not give up on interface-tracking accuracy even where there is an actual contact between solid surfaces or other TC. It can handle an actual TC while maintaining high-resolution boundary layer representation near solid surfaces. The ST-TC method is based on special mesh generation and update, and a master-slave system that maintains the connectivity of the “parent” mesh when there is a TC. It has the moving-mesh accuracy, the TC flexibility, and the computational practicality. The ST-TC method has been successfully applied to 3D computational analysis of heart valve models <61> and wing clapping <56>.

The “ST-SI-TC” method we introduce in this paper integrates the ST-SI and ST-TC methods in the ST-VMS framework. It enables accurate flow analysis when we have a spinning solid surface that is in contact with a solid surface. The integration of the ST-SI and ST-TC methods is challenging, partly because the two methods have somewhat conflicting parts. The ST-TC method is based on element “collapse.” In the ST-SI method, on the other hand, the interface terms require non-collapsed elements near the interface. We address that challenge successfully in the integration of the two methods. We present two test computations with the ST-SI-TC method. They are both with models of flow around a rotating tire with road contact and prescribed deformation, one with a 2D model, and one with a 3D model.

In Section 2 we describe the ST-VMS and ST-SI formulations. The ST-SI-TC method is described in Section 3. The test computations are presented in Section 4, and the concluding remarks are given in Section 5.

2. ST-VMS and ST-SI formulations

In this section, we include from <49> the ST-VMS and ST-SI formulations.

2.1. ST-VMS formulation

The ST-VMS formulation is given as

$$\begin{aligned}
& \int_{Q_n} \mathbf{w}^h \cdot \rho \left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{u}^h - \mathbf{f}^h \right) dQ \\
& + \int_{Q_n} \boldsymbol{\varepsilon}(\mathbf{w}^h) : \boldsymbol{\sigma}(\mathbf{u}^h, p^h) dQ - \int_{(P_n)_h} \mathbf{w}^h \cdot \mathbf{h}^h dP \\
& + \int_{Q_n} q^h \nabla \cdot \mathbf{u}^h dQ + \int_{\Omega_n} (\mathbf{w}^h)_n^+ \cdot \rho ((\mathbf{u}^h)_n^+ - (\mathbf{u}^h)_n^-) d\Omega \\
& + \sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} \frac{\tau_{\text{SUPS}}}{\rho} \left[\rho \left(\frac{\partial \mathbf{w}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{w}^h \right) \right. \\
& \quad \left. + \nabla q^h \right] \cdot \mathbf{r}_M(\mathbf{u}^h, p^h) dQ \\
& + \sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} \nu_{\text{LSIC}} \nabla \cdot \mathbf{w}^h \rho r_C(\mathbf{u}^h) dQ \\
& - \sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} \tau_{\text{SUPS}} \mathbf{w}^h \cdot (\mathbf{r}_M(\mathbf{u}^h, p^h) \cdot \nabla \mathbf{u}^h) dQ \\
& - \sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} \frac{\tau_{\text{SUPS}}^2}{\rho} \mathbf{r}_M(\mathbf{u}^h, p^h) \cdot (\nabla \mathbf{w}^h) \cdot \mathbf{r}_M(\mathbf{u}^h, p^h) dQ \\
& = 0,
\end{aligned} \tag{1}$$

where

$$\mathbf{r}_M(\mathbf{u}^h, p^h) = \rho \left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{u}^h - \mathbf{f}^h \right) - \nabla \cdot \boldsymbol{\sigma}(\mathbf{u}^h, p^h), \tag{2}$$

$$r_C(\mathbf{u}^h) = \nabla \cdot \mathbf{u}^h \tag{3}$$

are the residuals of the momentum equation and incompressibility constraint. Here, ρ , $\mathbf{u}$, p , $\mathbf{f}$, $\boldsymbol{\sigma}$, $\boldsymbol{\varepsilon}$, and $\mathbf{h}$ are the density, velocity, pressure, external force, stress tensor, strain rate tensor, and the traction specified at the boundary. The test functions associated with the velocity and pressure are $\mathbf{w}$ and q . A superscript “ h ” indicates that the function is coming from a finite-dimensional space. The symbol Q_n represents the ST slice between time levels n and $n+1$, $(P_n)_h$ is the part of the lateral boundary of that slice associated with the traction boundary condition $\mathbf{h}$, and Ω_n is the spatial domain at time level n . The

superscript “ e ” is the ST element counter, and n_{el} is the number of ST elements. The functions are discontinuous in time at each time level, and the superscripts “ $-$ ” and “ $+$ ” indicate the values of the functions just below and just above the time level. There are various ways of defining the stabilization parameters τ_{SUPS} and ν_{LSIC} . See [4; 5; 47; 64; 49] for the stabilization parameter definitions used here. For more ways of calculating the stabilization parameters in finite element computation of flow problems, see [74; 75; 76; 77; 78; 79; 80; 81; 82; 83; 84; 85; 86; 87; 88; 89; 90; 91; 92; 93; 94; 95].

The expression for $\mathbf{r}_M(\mathbf{u}^h, p^h)$ includes second derivatives of the velocity. For linear basis functions these terms vanish, and for bilinear and trilinear basis functions they are very much underrepresented. This means that $\mathbf{r}_M(\mathbf{u}^h, p^h)$ does not explicitly depend on the Reynolds number. When we use quadratic or higher-order basis functions, on the other hand, the term is nonzero and therefore explicit dependence of the residual on the Reynolds number is taken into account.

2.2. ST-SI formulation

2.2.1. Two-side formulation

We use the labels “Side A” and “Side B” to represent the two sides of the SI. In the ST-SI formulation, we add boundary terms to Eq. (1). The boundary terms for the two sides are first added separately, using test functions $\mathbf{w}_A^h$ and q_A^h and $\mathbf{w}_B^h$ and q_B^h . Then, putting together the terms added to each side, the complete set of terms added becomes

$$\begin{aligned}
& - \int_{(P_n)_{\text{SI}}} (q_B^h \mathbf{n}_B - q_A^h \mathbf{n}_A) \cdot \frac{1}{2} (\mathbf{u}_B^h - \mathbf{u}_A^h) \, dP \\
& - \int_{(P_n)_{\text{SI}}} \rho \mathbf{w}_B^h \cdot \frac{1}{2} ((\mathcal{F}_B^h - |\mathcal{F}_B^h|) \mathbf{u}_B^h \\
& \quad - (\mathcal{F}_B^h - |\mathcal{F}_B^h|) \mathbf{u}_A^h) \, dP \\
& - \int_{(P_n)_{\text{SI}}} \rho \mathbf{w}_A^h \cdot \frac{1}{2} ((\mathcal{F}_A^h - |\mathcal{F}_A^h|) \mathbf{u}_A^h \\
& \quad - (\mathcal{F}_A^h - |\mathcal{F}_A^h|) \mathbf{u}_B^h) \, dP
\end{aligned}$$

$$\begin{aligned}
& + \int_{(P_n)_{\text{SI}}} (\mathbf{n}_B \cdot \mathbf{w}_B^h + \mathbf{n}_A \cdot \mathbf{w}_A^h) \frac{1}{2} (p_B^h + p_A^h) \, dP \\
& - \int_{(P_n)_{\text{SI}}} (\mathbf{w}_B^h - \mathbf{w}_A^h) \cdot (\hat{\mathbf{n}}_B \cdot \mu (\boldsymbol{\varepsilon}(\mathbf{u}_B^h) + \boldsymbol{\varepsilon}(\mathbf{u}_A^h))) \, dP \\
& - \gamma \int_{(P_n)_{\text{SI}}} \hat{\mathbf{n}}_B \cdot \mu (\boldsymbol{\varepsilon}(\mathbf{w}_B^h) + \boldsymbol{\varepsilon}(\mathbf{w}_A^h)) \cdot (\mathbf{u}_B^h - \mathbf{u}_A^h) \, dP \\
& + \int_{(P_n)_{\text{SI}}} \frac{\mu C}{h} (\mathbf{w}_B^h - \mathbf{w}_A^h) \cdot (\mathbf{u}_B^h - \mathbf{u}_A^h) \, dP,
\end{aligned} \tag{4}$$

where

$$\mathcal{F}_B^h = \mathbf{n}_B \cdot (\mathbf{u}_B^h - \mathbf{v}_B^h), \tag{5}$$

$$\mathcal{F}_A^h = \mathbf{n}_A \cdot (\mathbf{u}_A^h - \mathbf{v}_A^h), \tag{6}$$

$$h = \frac{h_B + h_A}{2}, \tag{7}$$

$$h_B = 2 \left(\sum_{\alpha=1}^{n_{\text{ent}}} \sum_{a=1}^{n_{\text{ens}}} |\mathbf{n}_B \cdot \nabla N_a^\alpha| \right)^{-1} \quad (\text{for Side B}), \tag{8}$$

$$h_A = 2 \left(\sum_{\alpha=1}^{n_{\text{ent}}} \sum_{a=1}^{n_{\text{ens}}} |\mathbf{n}_A \cdot \nabla N_a^\alpha| \right)^{-1} \quad (\text{for Side A}), \tag{9}$$

$$\hat{\mathbf{n}}_B = \frac{\mathbf{n}_B - \mathbf{n}_A}{\|\mathbf{n}_B - \mathbf{n}_A\|}. \tag{10}$$

Here, $(P_n)_{\text{SI}}$ is the SI in the ST domain, $\mathbf{n}$ is the unit normal vector, $\mathbf{v}$ is the mesh velocity, n_{ens} and n_{ent} are the number of spatial and temporal element nodes, N_a^α is the basis function associated with spatial and temporal nodes a and α , $\gamma = 1$, and C is a nondimensional constant. From our experience, for our element length definition, usually $C = 1$ is large enough for stability.

A number of remarks were provided in <49> to explain the added terms and to comment on related interpretations. We refer the reader interested in those details to <49>.

Remark 1. In <49>, the expression given by Eq. (7) was in the form

$$h = \min(h_B, h_A). \tag{11}$$

With the new form in Eq. (7), the expression gives zero value only if the element length is zero on both sides.

Remark 2. *In this article we generalize the ST-SI method by adding a coefficient γ to the sixth integration so that we have the option of using $\gamma = -1$. Using $\gamma = 1$ in a discontinuous Galerkin method was introduced in the symmetric interior penalty Galerkin method <96>, and using $\gamma = -1$ was introduced in the nonsymmetric interior penalty Galerkin method <97>. Stabilized methods based on both $\gamma = 1$ and -1 were reported in <98> in the context of the advection–diffusion equation.*

Remark 3. *We propose, in general, to use the ST-SI method also as a way of imposing the periodicity in ST-VMS computations with spatial periodicity, including rotational periodicity (i.e., discrete rotational symmetry). In such cases, we place an SI where we want to impose the periodicity and carry out the computations while including the corresponding ST-SI terms.*

2.2.2. One-side formulation

On solid surfaces where we prefer to have weakly-imposed Dirichlet conditions for the fluid <98; 23>, we use the ST-SI version where the SI is between the fluid and solid domains. This version is obtained (see <49>) by starting with the terms added to Side B and replacing the Side A velocity with the velocity $\mathbf{g}^h$ coming from the solid domain. Then the terms added to Eq. (1) to represent the weakly-imposed Dirichlet conditions become

$$\begin{aligned}
& - \int_{(P_n)_{\text{SI}}} q_{\text{B}}^h \mathbf{n}_{\text{B}} \cdot \mathbf{u}_{\text{B}}^h dP - \int_{(P_n)_{\text{SI}}} \rho \mathbf{w}_{\text{B}}^h \cdot \mathcal{F}_{\text{B}}^h \mathbf{u}_{\text{B}}^h dP \\
& + \int_{(P_n)_{\text{SI}}} q_{\text{B}}^h \mathbf{n}_{\text{B}} \cdot \mathbf{g}^h dP \\
& + \int_{(P_n)_{\text{SI}}} \rho \mathbf{w}_{\text{B}}^h \cdot \frac{1}{2} ((\mathcal{F}_{\text{B}}^h + |\mathcal{F}_{\text{B}}^h|) \mathbf{u}_{\text{B}}^h \\
& \quad + (\mathcal{F}_{\text{B}}^h - |\mathcal{F}_{\text{B}}^h|) \mathbf{g}^h) dP \\
& - \int_{(P_n)_{\text{SI}}} \mathbf{w}_{\text{B}}^h \cdot (\mathbf{n}_{\text{B}} \cdot \boldsymbol{\sigma}_{\text{B}}^h) dP \\
& - \gamma \int_{(P_n)_{\text{SI}}} \mathbf{n}_{\text{B}} \cdot 2\mu \boldsymbol{\epsilon}(\mathbf{w}_{\text{B}}^h) \cdot (\mathbf{u}_{\text{B}}^h - \mathbf{g}^h) dP
\end{aligned}$$

$$+ \int_{(P_n)_{\text{SI}}} \frac{\mu C}{h_B} \mathbf{w}_B^h \cdot (\mathbf{u}_B^h - \mathbf{g}^h) dP. \quad (12)$$

3. ST-SI-TC method

We first overview the aspects of the ST-SI <49> and ST-TC <54> formulations relevant to their integration, and then describe the integration.

3.1. ST-SI method

We note that the ST-SI method allows mesh slipping also in the one-side formulation, that is, when the SI is between the fluid and solid domains where we have weakly-imposed Dirichlet conditions. The boundary terms added to Eq. (1) to connect the two sides and to connect the fluid to the solid in the one-side formulation were given in Sections 2.2.1 and 2.2.2. The added terms (see Eqs. (4) and (12)) include derivatives in the direction normal to the SI. Therefore the elements bordering the SI need to have finite thickness in the normal direction. This places a limitation on the meshes that can be used with the ST-SI method because if an element bordering the interface degenerates it might lead to a zero element thickness in the normal direction.

3.2. ST-TC method

The ST-TC method can deal with TC in ST moving-mesh computations. The discretization is unstructured in time, but based on a “parent mesh” that is structured in time. The ST parent mesh is extruded from a single spatial mesh. The key technology is massive element degeneration by using a special master–slave system. This special system allows changing, within an ST slab, master nodes to slave nodes and slave nodes to master nodes. With that, elements can collapse or be reborn. This way, within an ST slab, we can represent closing and opening motions. Since an ST method naturally allows discretizations that are unstructured in time, the rest of the method needs no modification. The method is very flexible, and computationally as effective as a typical moving-mesh method. However, the master–slave relationship has to be node to node; a point on a solid surface that is not a node cannot be a master or slave node.

3.3. Integration of the ST-SI and ST-TC methods

There are two motivations for the integration of the ST-SI and ST-TC methods. (i) It enables moving-mesh computation when we have a spinning solid surface that is in contact with a solid surface. (ii) It enables dealing with contact location change or contact and sliding on the SI. An example for the first one is flow around a rotating tire with road contact and prescribed deformation. The second one is about overcoming the ST-TC restriction that a point on a solid surface that is not a node cannot be a master or slave node. An example for that is moving-mesh flow analysis of a heart valve, where the valve leaflets may slide on each other before they separate.

Figure 1 is an example of Case (ii), where we have two bars. Figure 2

Figure 1: An example of contact and sliding. The time sequence is from *left to right*, and *top to bottom*. The two bars are deforming, and when they are in contact they are sliding on each other. During the contact, the lower bar extends beyond the upper one in the fourth picture and falls short in the sixth, illustrating that the solid surfaces are sliding on each other.

illustrates, for the example in Figure 1, how the ST-SI-TC method works. An SI is created between the two bars. The SI allows the two bars slide relative to each other. In some cases the ST-TC element collapse occurs between only one side of the SI and the solid surface, as seen in the third picture of Figure 2. If the two bars are in contact, on both sides, the elements between the bar and

Figure 2: Illustration of how the ST-SI-TC method works for the example in Figure 1. The time sequence is from *left to right*, and *top to bottom*. Some of the elements around the bars (blue and red) are shown. The two middle (green) lines are the boundaries creating the SI. The circles on the lines are the integration points.

the SI need to collapse. On the other hand, the SI side with the boundary not connected to a solid surface must have finite element thickness in the normal direction. To adapt to these situations, we switch between the two- and one-side formulations. The decision to switch is made based on the element thickness in the normal direction, as given by Eqs. (8) and (9), evaluated at each integration point. Here we explain the boundary integrations corresponding to the test functions for Side B (i.e., $\mathbf{w}_B^h$ and q_B^h). The integrations are performed with the following rules:

- If $h_B = 0$, we do not integrate on Side B, regardless of the value of h_A .
- If $h_B \neq 0$ and $h_A = 0$, the one-side formulation is used for Side B.
- In cases other than those two above, the two-side formulation is used as usual in the ST-SI method.

Figures 3 and 4 are the close-up views of the third and fifth pictures in Figure 2.

Remark 4. *A node on an SI coinciding with a solid surface must be a slave of*

Figure 3: Close-up view of the third picture in Figure 2. The two integration points B_1 and B_2 belong to an element of Side B bordering the SI. The element of Side A closest to B_1 has finite thickness in the normal direction, and the element closest to B_2 has zero thickness.

Figure 4: Close-up view of the fifth picture in Figure 2. The red circles marked by n_1 and n_2 are the nodes on the lower bar.

the corresponding node on that solid surface.

Remark 5. *The decision to switch between the two- and one-side formulations is made for each integration point individually. Therefore, the integration points within an element might require different formulations. For example, in Figure 3, for B_1 and B_2 we use the two- and one-side formulations, respectively.*

Remark 6. *When for all integration points of an element edge $h_B = h_A = 0$, that edge is a contact surface, and the element is a collapsed element as in the ST-TC method. Therefore an SI node shared only by collapsed elements will, from an implementation point of view, be slave of a node on the solid surface. Node n_2 in Figure 4 is an example of that. Pressure is not treated as an unknown for such nodes, and in that sense those nodes have no role in the equation system. In terms of the fluid mechanics equations, while n_2 cannot be a master node, there will be a master node at n_1 location.*

Remark 7. *If n_2 becomes like n_1 within an ST slab Q_n , we will need predictors*

for pressure values within that slab. The predictors will come from p_{n-1}^- for a nearby node that has pressure values for in Q_{n-1} , such as n_1 .

4. Test computations: flow around a rotating tire with road contact and prescribed deformation

4.1. 2D model

4.1.1. Problem setup

The geometry of the model, which includes 12 treads, and the deformation pattern are shown in Figure 5. The rotation speed corresponds to a linear speed

Figure 5: 2D model. The deformation region is the circular sector with central angle 36° .

of 100 km/h at the tire periphery. The density and kinematic viscosity of the air are 1.205 kg/m^3 and $1.511 \times 10^{-5} \text{ m}^2/\text{s}$.

4.1.2. Computational domain and mesh

Figure 6 shows the computational domain and boundary conditions. The domain size is $6.75 \text{ m} \times 2.52 \text{ m}$. Figure 7 shows the mesh. The number of nodes is 68,491, and the number of elements is 123,037. The mesh between the tire surface and the SI is made of quadrilateral elements and generated manually. There is one more layer of manually generated quadrilateral elements beyond the SI. The rest of the mesh is made of triangular elements. Only the mesh between the tire surface and the SI moves, deforms and has collapsed elements in the

Figure 6: 2D model. Boundary conditions. The left (*red*) and bottom (*yellow*) boundaries represent the inflow and road, where the velocity is set to the linear speed at the tire periphery. The innermost (*blue*) circle is the tire surface, which is rotating. The larger (*pink*) circle is the SI. The bottom of the SI is coinciding with the road and the tire–road interface. The conditions at the right (*green*) and upper (*cyan*) boundaries are traction-free and slip, respectively.

ST-TC method. The master–slave relationship is assigned manually. Figures 8 and 9 show the mesh at 20 equally spaced instants during the rotation.

4.1.3. Computational parameters

The computation is carried out with the ST-SUPS method, which can be obtained from Eq. (1) by dropping the eighth and ninth integrations. The stabilization parameters are those given by Eqs. (2.4)–(2.8) in <49>, without “ τ_{SUGN4} .” There are 2,880 time steps per rotation, which is equivalent to a time-step size of 2.474×10^{-5} s. The number of nonlinear iterations per time step is 5. The number of GMRES <99> iterations per nonlinear iteration is 100 for the first 4 nonlinear iterations, and 300 for the last.

4.1.4. Results

Figures 10 and 11 show, for the second rotation, the velocity magnitude at the same instants as in Figures 8 and 9. Figures 12 and 13 show, for the second rotation, close-up view of the velocity magnitude around the tire at the same instants as in Figures 8 and 9. The overall solution quality is consistent with using a moving-mesh method, including the solution quality near the contact location. Figure 14 shows, for a time period close to the end of the second rotation, how the flow field changes when a tread contacts the road. Being able to accurately compute such flow details illustrates what is special about the ST-SI-TC method.

4.2. 3D model

4.2.1. Problem setup

This is the 3D version of the 2D model. The tire model is obtained by extruding the 2D model 27.5 cm in the axial direction.

4.2.2. Computational domain and mesh

The domain size is $9.93 \text{ m} \times 5.28 \text{ m}$ in the plane perpendicular to the tire axis, and 6.83 m in the tire-axis direction (see Figure 15). Figure 16 shows the mesh. The number of nodes is 2,370,840, and number of elements is 5,415,467. The mesh between the tire surface and the SI are made of hexahedral elements and generated manually. There is one more layer of manually generated hexahedral elements beyond the SI and then one layer of manually generated pyramid elements. The rest of the mesh is made of tetrahedral elements. As in the 2D case, only the mesh between the tire surface and the SI moves, deforms and has collapsed elements.

4.2.3. Computational parameters

The method, the stabilization parameters and the time-step size are the same as in the 2D model. The number of nonlinear iterations per time step is 5. The number of GMRES iterations per nonlinear iteration is 300.

4.2.4. Results

Figure 17 shows the preliminary results. As in the 2D case, the overall solution quality is consistent with using a moving-mesh method.

5. Concluding remarks

We have introduced the ST-SI-TC method, which is an integration of the ST-SI and ST-TC methods in the context of the ST-VMS method. The ST-VMS method is used in computation of FSI and other MBI problems as a moving-mesh method. Therefore it has the desirable features of a method where the mesh moves to follow the fluid–solid interfaces to maintain high-resolution

representation of boundary layers. The ST-SI method retains those desirable features even when we have a spinning solid surface. The mesh covering a spinning solid surface spins with it, and the SI between that inner mesh and the outer mesh is handled with the ST-SI terms added to the formulation. The ST-TC method retains the desirable features in computations with TC, such as contact between moving solid surfaces. The ST-SI-TC method enables accurate flow analysis when we have a spinning solid surface that is in contact with a solid surface. It also overcomes a restriction of the ST-TC method. The integration of the ST-SI and ST-TC methods required reconciling the conflicting parts of the two methods. We have presented two test computations with the ST-SI-TC method, both with models of flow around a rotating tire with road contact and prescribed deformation. The first one is with a 2D model, and the second one with a 3D model. Both computations show that in flow analysis with a spinning solid surface that is in contact with a solid surface, the ST-SI-TC method yields a solution quality consistent with using a moving-mesh method.

Acknowledgement

This work was supported (first, third and fourth authors) in part by Grant-in-Aid for Young Scientists (B) 24760144 from Japan Society for the Promotion of Science (JSPS); Grant-in-Aid for Scientific Research (S) 26220002 from the Ministry of Education, Culture, Sports, Science and Technology of Japan (MEXT); Council for Science, Technology and Innovation (CSTI), Cross-ministerial Strategic Innovation Promotion Program (SIP), “Innovative Combustion Technology” (Funding agency: JST); and Rice–Waseda research agreement. The work was also supported (second author) in part by ARO Grant W911NF-12-1-0162.

- [1] K. Takizawa and T.E. Tezduyar, “Multiscale space–time fluid–structure interaction techniques”, *Computational Mechanics*, **48** (2011) 247–267, doi: 10.1007/s00466-011-0571-z.

- [2] K. Takizawa and T.E. Tezduyar, “Space–time fluid–structure interaction methods”, *Mathematical Models and Methods in Applied Sciences*, **22** (supp02) (2012) 1230001, doi: 10.1142/S0218202512300013.
- [3] T.E. Tezduyar, “Stabilized finite element formulations for incompressible flow computations”, *Advances in Applied Mechanics*, **28** (1992) 1–44, doi: 10.1016/S0065-2156(08)70153-4.
- [4] T.E. Tezduyar, “Computation of moving boundaries and interfaces and stabilization parameters”, *International Journal for Numerical Methods in Fluids*, **43** (2003) 555–575, doi: 10.1002/fld.505.
- [5] T.E. Tezduyar and S. Sathe, “Modeling of fluid–structure interactions with the space–time finite elements: Solution techniques”, *International Journal for Numerical Methods in Fluids*, **54** (2007) 855–900, doi: 10.1002/fld.1430.
- [6] T.J.R. Hughes, “Multiscale phenomena: Green’s functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles, and the origins of stabilized methods”, *Computer Methods in Applied Mechanics and Engineering*, **127** (1995) 387–401.
- [7] T.J.R. Hughes, A.A. Oberai, and L. Mazzei, “Large eddy simulation of turbulent channel flows by the variational multiscale method”, *Physics of Fluids*, **13** (2001) 1784–1799.
- [8] Y. Bazilevs, V.M. Calo, J.A. Cottrell, T.J.R. Hughes, A. Reali, and G. Scovazzi, “Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows”, *Computer Methods in Applied Mechanics and Engineering*, **197** (2007) 173–201.
- [9] Y. Bazilevs and I. Akkerman, “Large eddy simulation of turbulent Taylor–Couette flow using isogeometric analysis and the residual–based variational multiscale method”, *Journal of Computational Physics*, **229** (2010) 3402–3414.

- [10] A.N. Brooks and T.J.R. Hughes, “Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations”, *Computer Methods in Applied Mechanics and Engineering*, **32** (1982) 199–259.
- [11] Y. Bazilevs, K. Takizawa, and T.E. Tezduyar, *Computational Fluid–Structure Interaction: Methods and Applications*. Wiley, February 2013, ISBN 978-0470978771.
- [12] K. Takizawa, Y. Bazilevs, and T.E. Tezduyar, “Space–time and ALE–VMS techniques for patient-specific cardiovascular fluid–structure interaction modeling”, *Archives of Computational Methods in Engineering*, **19** (2012) 171–225, doi: 10.1007/s11831-012-9071-3.
- [13] Y. Bazilevs, M.-C. Hsu, K. Takizawa, and T.E. Tezduyar, “ALE–VMS and ST–VMS methods for computer modeling of wind-turbine rotor aerodynamics and fluid–structure interaction”, *Mathematical Models and Methods in Applied Sciences*, **22** (supp02) (2012) 1230002, doi: 10.1142/S0218202512300025.
- [14] Y. Bazilevs, V.M. Calo, T.J.R. Hughes, and Y. Zhang, “Isogeometric fluid–structure interaction: theory, algorithms, and computations”, *Computational Mechanics*, **43** (2008) 3–37.
- [15] T.J.R. Hughes, W.K. Liu, and T.K. Zimmermann, “Lagrangian–Eulerian finite element formulation for incompressible viscous flows”, *Computer Methods in Applied Mechanics and Engineering*, **29** (1981) 329–349.
- [16] T.J.R. Hughes, J.A. Cottrell, and Y. Bazilevs, “Isogeometric analysis: CAD, finite elements, NURBS, exact geometry, and mesh refinement”, *Computer Methods in Applied Mechanics and Engineering*, **194** (2005) 4135–4195.
- [17] Y. Bazilevs, V.M. Calo, Y. Zhang, and T.J.R. Hughes, “Isogeometric fluid–

- structure interaction analysis with applications to arterial blood flow”, *Computational Mechanics*, **38** (2006) 310–322.
- [18] Y. Bazilevs and T.J.R. Hughes, “NURBS-based isogeometric analysis for the computation of flows about rotating components”, *Computational Mechanics*, **43** (2008) 143–150.
- [19] Y. Bazilevs, J.R. Gohean, T.J.R. Hughes, R.D. Moser, and Y. Zhang, “Patient-specific isogeometric fluid–structure interaction analysis of thoracic aortic blood flow due to implantation of the Jarvik 2000 left ventricular assist device”, *Computer Methods in Applied Mechanics and Engineering*, **198** (2009) 3534–3550.
- [20] Y. Bazilevs, M.-C. Hsu, D. Benson, S. Sankaran, and A. Marsden, “Computational fluid–structure interaction: Methods and application to a total cavopulmonary connection”, *Computational Mechanics*, **45** (2009) 77–89.
- [21] Y. Bazilevs, M.-C. Hsu, I. Akkerman, S. Wright, K. Takizawa, B. Henicke, T. Spielman, and T.E. Tezduyar, “3D simulation of wind turbine rotors at full scale. Part I: Geometry modeling and aerodynamics”, *International Journal for Numerical Methods in Fluids*, **65** (2011) 207–235, doi: 10.1002/fld.2400.
- [22] Y. Bazilevs, M.-C. Hsu, J. Kiendl, R. Wüchner, and K.-U. Bletzinger, “3D simulation of wind turbine rotors at full scale. Part II: Fluid–structure interaction modeling with composite blades”, *International Journal for Numerical Methods in Fluids*, **65** (2011) 236–253.
- [23] M.-C. Hsu, I. Akkerman, and Y. Bazilevs, “Wind turbine aerodynamics using ALE-VMS: Validation and role of weakly enforced boundary conditions”, *Computational Mechanics*, **50** (2012) 499–511.
- [24] M.-C. Hsu and Y. Bazilevs, “Fluid–structure interaction modeling of wind turbines: simulating the full machine”, *Computational Mechanics*, **50** (2012) 821–833.

- [25] Y. Bazilevs, K. Takizawa, and T.E. Tezduyar, “Challenges and directions in computational fluid–structure interaction”, *Mathematical Models and Methods in Applied Sciences*, **23** (2013) 215–221, doi: 10.1142/S0218202513400010.
- [26] A. Korobenko, M.-C. Hsu, I. Akkerman, J. Tippmann, and Y. Bazilevs, “Structural mechanics modeling and FSI simulation of wind turbines”, *Mathematical Models and Methods in Applied Sciences*, **23** (2013) 249–272.
- [27] A. Korobenko, M.-C. Hsu, I. Akkerman, and Y. Bazilevs, “Aerodynamic simulation of vertical-axis wind turbines”, *Journal of Applied Mechanics*, **81** (2013) 021011, doi: 10.1115/1.4024415.
- [28] Y. Bazilevs, K. Takizawa, T.E. Tezduyar, M.-C. Hsu, N. Kostov, and S. McIntyre, “Aerodynamic and FSI analysis of wind turbines with the ALE-VMS and ST-VMS methods”, *Archives of Computational Methods in Engineering*, **21** (2014) 359–398, doi: 10.1007/s11831-014-9119-7.
- [29] Y. Bazilevs, A. Korobenko, X. Deng, J. Yan, M. Kinzel, and J.O. Dabiri, “FSI modeling of vertical-axis wind turbines”, *Journal of Applied Mechanics*, **81** (2014) 081006, doi: 10.1115/1.4027466.
- [30] M.-C. Hsu, I. Akkerman, and Y. Bazilevs, “Finite element simulation of wind turbine aerodynamics: Validation study using NREL Phase VI experiment”, *Wind Energy*, **17** (2014) 461–481.
- [31] C.C. Long, M. Esmaily-Moghadam, A.L. Marsden, and Y. Bazilevs, “Computation of residence time in the simulation of pulsatile ventricular assist devices”, *Computational Mechanics*, **54** (2014) 911–919, doi: 10.1007/s00466-013-0931-y.
- [32] C.C. Long, A.L. Marsden, and Y. Bazilevs, “Shape optimization of pulsatile ventricular assist devices using FSI to minimize thrombotic risk”, *Computational Mechanics*, **54** (2014) 921–932, doi: 10.1007/s00466-013-0967-z.

- [33] M.-C. Hsu, D. Kamensky, Y. Bazilevs, M.S. Sacks, and T.J.R. Hughes, “Fluid–structure interaction analysis of bioprosthetic heart valves: significance of arterial wall deformation”, *Computational Mechanics*, **54** (2014) 1055–1071, doi: 10.1007/s00466-014-1059-4.
- [34] B. Augier, J. Yan, A. Korobenko, J. Czarnowski, G. Ketterman, and Y. Bazilevs, “Experimental and numerical FSI study of compliant hydrofoils”, *Computational Mechanics*, **55** (2015) 1079–1090, doi: 10.1007/s00466-014-1090-5.
- [35] Y. Bazilevs, A. Korobenko, J. Yan, A. Pal, S.M.I. Gohari, and S. Sarkar, “ALE–VMS formulation for stratified turbulent incompressible flows with applications”, *Mathematical Models and Methods in Applied Sciences*, **25** (2015) 2349–2375, doi: 10.1142/S0218202515400114.
- [36] Y. Bazilevs, K. Takizawa, and T.E. Tezduyar, “New directions and challenging computations in fluid dynamics modeling with stabilized and multiscale methods”, *Mathematical Models and Methods in Applied Sciences*, **25** (2015) 2217–2226, doi: 10.1142/S0218202515020029.
- [37] Y. Bazilevs, A. Korobenko, X. Deng, and J. Yan, “FSI modeling for fatigue-damage prediction in full-scale wind-turbine blades”, *Journal of Applied Mechanics*, **83** (6) (2016) 061010.
- [38] J. Yan, B. Augier, A. Korobenko, J. Czarnowski, G. Ketterman, and Y. Bazilevs, “FSI modeling of a propulsion system based on compliant hydrofoils in a tandem configuration”, *Computers and Fluids*, (2015), Published online, doi:10.1016/j.compfluid.2015.07.013.
- [39] J. Yan, A. Korobenko, X. Deng, and Y. Bazilevs, “Computational free-surface fluid–structure interaction with application to floating offshore wind turbines”, *Computers & Fluids*, (2016), Published online, doi:10.1016/j.compfluid.2016.03.008.

- [40] K. Takizawa, Y. Bazilevs, T.E. Tezduyar, M.-C. Hsu, O. Øiseth, K.M. Mathisen, N. Kostov, and S. McIntyre, “Engineering analysis and design with ALE-VMS and space–time methods”, *Archives of Computational Methods in Engineering*, **21** (2014) 481–508, doi: 10.1007/s11831-014-9113-0.
- [41] K. Takizawa, T.E. Tezduyar, R. Kolesar, C. Boswell, T. Kanai, and K. Montel, “Multiscale methods for gore curvature calculations from FSI modeling of spacecraft parachutes”, *Computational Mechanics*, **54** (2014) 1461–1476, doi: 10.1007/s00466-014-1069-2.
- [42] K. Takizawa, T.E. Tezduyar, C. Boswell, R. Kolesar, and K. Montel, “FSI modeling of the reefed stages and disreefing of the Orion spacecraft parachutes”, *Computational Mechanics*, **54** (2014) 1203–1220, doi: 10.1007/s00466-014-1052-y.
- [43] K. Takizawa, T.E. Tezduyar, C. Boswell, Y. Tsutsui, and K. Montel, “Special methods for aerodynamic-moment calculations from parachute FSI modeling”, *Computational Mechanics*, **55** (2015) 1059–1069, doi: 10.1007/s00466-014-1074-5.
- [44] K. Takizawa, T.E. Tezduyar, and R. Kolesar, “FSI modeling of the Orion spacecraft drogue parachutes”, *Computational Mechanics*, **55** (2015) 1167–1179, doi: 10.1007/s00466-014-1108-z.
- [45] K. Takizawa, B. Henicke, T.E. Tezduyar, M.-C. Hsu, and Y. Bazilevs, “Stabilized space–time computation of wind-turbine rotor aerodynamics”, *Computational Mechanics*, **48** (2011) 333–344, doi: 10.1007/s00466-011-0589-2.
- [46] K. Takizawa, B. Henicke, D. Montes, T.E. Tezduyar, M.-C. Hsu, and Y. Bazilevs, “Numerical-performance studies for the stabilized space–time computation of wind-turbine rotor aerodynamics”, *Computational Mechanics*, **48** (2011) 647–657, doi: 10.1007/s00466-011-0614-5.

- [47] K. Takizawa, T.E. Tezduyar, S. McIntyre, N. Kostov, R. Kolesar, and C. Habluetzel, “Space–time VMS computation of wind-turbine rotor and tower aerodynamics”, *Computational Mechanics*, **53** (2014) 1–15, doi: 10.1007/s00466-013-0888-x.
- [48] K. Takizawa, “Computational engineering analysis with the new-generation space–time methods”, *Computational Mechanics*, **54** (2014) 193–211, doi: 10.1007/s00466-014-0999-z.
- [49] K. Takizawa, T.E. Tezduyar, H. Mochizuki, H. Hattori, S. Mei, L. Pan, and K. Montel, “Space–time VMS method for flow computations with slip interfaces (ST-SI)”, *Mathematical Models and Methods in Applied Sciences*, **25** (2015) 2377–2406, doi: 10.1142/S0218202515400126.
- [50] K. Takizawa, B. Henicke, A. Puntel, T. Spielman, and T.E. Tezduyar, “Space–time computational techniques for the aerodynamics of flapping wings”, *Journal of Applied Mechanics*, **79** (2012) 010903, doi: 10.1115/1.4005073.
- [51] K. Takizawa, B. Henicke, A. Puntel, N. Kostov, and T.E. Tezduyar, “Space–time techniques for computational aerodynamics modeling of flapping wings of an actual locust”, *Computational Mechanics*, **50** (2012) 743–760, doi: 10.1007/s00466-012-0759-x.
- [52] K. Takizawa, N. Kostov, A. Puntel, B. Henicke, and T.E. Tezduyar, “Space–time computational analysis of bio-inspired flapping-wing aerodynamics of a micro aerial vehicle”, *Computational Mechanics*, **50** (2012) 761–778, doi: 10.1007/s00466-012-0758-y.
- [53] K. Takizawa, B. Henicke, A. Puntel, N. Kostov, and T.E. Tezduyar, “Computer modeling techniques for flapping-wing aerodynamics of a locust”, *Computers & Fluids*, **85** (2013) 125–134, doi: 10.1016/j.compfluid.2012.11.008.

- [54] K. Takizawa, T.E. Tezduyar, A. Buscher, and S. Asada, “Space–time interface-tracking with topology change (ST-TC)”, *Computational Mechanics*, **54** (2014) 955–971, doi: 10.1007/s00466-013-0935-7.
- [55] K. Takizawa, T.E. Tezduyar, and N. Kostov, “Sequentially-coupled space–time FSI analysis of bio-inspired flapping-wing aerodynamics of an MAV”, *Computational Mechanics*, **54** (2014) 213–233, doi: 10.1007/s00466-014-0980-x.
- [56] K. Takizawa, T.E. Tezduyar, and A. Buscher, “Space–time computational analysis of MAV flapping-wing aerodynamics with wing clapping”, *Computational Mechanics*, **55** (2015) 1131–1141, doi: 10.1007/s00466-014-1095-0.
- [57] K. Takizawa, K. Schjodt, A. Puntel, N. Kostov, and T.E. Tezduyar, “Patient-specific computer modeling of blood flow in cerebral arteries with aneurysm and stent”, *Computational Mechanics*, **50** (2012) 675–686, doi: 10.1007/s00466-012-0760-4.
- [58] K. Takizawa, K. Schjodt, A. Puntel, N. Kostov, and T.E. Tezduyar, “Patient-specific computational analysis of the influence of a stent on the unsteady flow in cerebral aneurysms”, *Computational Mechanics*, **51** (2013) 1061–1073, doi: 10.1007/s00466-012-0790-y.
- [59] K. Takizawa, Y. Bazilevs, T.E. Tezduyar, C.C. Long, A.L. Marsden, and K. Schjodt, “ST and ALE-VMS methods for patient-specific cardiovascular fluid mechanics modeling”, *Mathematical Models and Methods in Applied Sciences*, **24** (2014) 2437–2486, doi: 10.1142/S0218202514500250.
- [60] H. Suito, K. Takizawa, V.Q.H. Huynh, D. Sze, and T. Ueda, “FSI analysis of the blood flow and geometrical characteristics in the thoracic aorta”, *Computational Mechanics*, **54** (2014) 1035–1045, doi: 10.1007/s00466-014-1017-1.
- [61] K. Takizawa, T.E. Tezduyar, A. Buscher, and S. Asada, “Space–time fluid

- mechanics computation of heart valve models”, *Computational Mechanics*, **54** (2014) 973–986, doi: 10.1007/s00466-014-1046-9.
- [62] K. Takizawa, D. Montes, M. Fritze, S. McIntyre, J. Boben, and T.E. Tezduyar, “Methods for FSI modeling of spacecraft parachute dynamics and cover separation”, *Mathematical Models and Methods in Applied Sciences*, **23** (2013) 307–338, doi: 10.1142/S0218202513400058.
- [63] K. Takizawa, D. Montes, S. McIntyre, and T.E. Tezduyar, “Space-time VMS methods for modeling of incompressible flows at high Reynolds numbers”, *Mathematical Models and Methods in Applied Sciences*, **23** (2013) 223–248, doi: 10.1142/s0218202513400022.
- [64] K. Takizawa, T.E. Tezduyar, and T. Kuraishi, “Multiscale ST methods for thermo-fluid analysis of a ground vehicle and its tires”, *Mathematical Models and Methods in Applied Sciences*, **25** (2015) 2227–2255, doi: 10.1142/S0218202515400072.
- [65] K. Takizawa, T.E. Tezduyar, T. Kuraishi, S. Tabata, and H. Takagi, “Computational thermo-fluid analysis of a disk brake”, *Computational Mechanics*, published online, DOI: 10.1007/s00466-016-1272-4, February 2016, doi: 10.1007/s00466-016-1272-4.
- [66] K. Takizawa, T.E. Tezduyar, and H. Hattori, “Computational analysis of flow-driven string dynamics in turbomachinery”, *Computers & Fluids*, published online, DOI: 10.1016/j.compfluid.2016.02.019, March 2016, doi: 10.1016/j.compfluid.2016.02.019.
- [67] K. Takizawa, T.E. Tezduyar, Y. Otoguro, T. Terahara, T. Kuraishi, and H. Hattori, “Turbocharger flow computations with the space-time isogeometric analysis (ST-IGA)”, *Computers & Fluids*, published online, DOI: 10.1016/j.compfluid.2016.02.021, March 2016, doi: 10.1016/j.compfluid.2016.02.021.

- [68] V. Kalro and T.E. Tezduyar, “Parallel finite element computation of 3D incompressible flows on MPPs”, in W.G. Habashi, editor, *Solution Techniques for Large-Scale CFD Problems*, John Wiley & Sons, 1995.
- [69] T. Tezduyar, S. Aliabadi, M. Behr, A. Johnson, V. Kalro, and M. Litke, “Flow simulation and high performance computing”, *Computational Mechanics*, **18** (1996) 397–412, doi: 10.1007/BF00350249.
- [70] M. Behr and T. Tezduyar, “The Shear-Slip Mesh Update Method”, *Computer Methods in Applied Mechanics and Engineering*, **174** (1999) 261–274, doi: 10.1016/S0045-7825(98)00299-0.
- [71] Y. Bazilevs, A. Korobenko, X. Deng, and J. Yan, “Novel structural modeling and mesh moving techniques for advanced FSI simulation of wind turbines”, *International Journal for Numerical Methods in Engineering*, **102** (2015) 766–783, doi: 10.1002/nme.4738.
- [72] T.E. Tezduyar, K. Takizawa, C. Moorman, S. Wright, and J. Christopher, “Space–time finite element computation of complex fluid–structure interactions”, *International Journal for Numerical Methods in Fluids*, **64** (2010) 1201–1218, doi: 10.1002/fld.2221.
- [73] D. Kamensky, M.-C. Hsu, D. Schillinger, J.A. Evans, A. Aggarwal, Y. Bazilevs, M.S. Sacks, and T.J.R. Hughes, “An immersogeometric variational framework for fluid-structure interaction: Application to bioprosthetic heart valves”, *Computer Methods in Applied Mechanics and Engineering*, **284** (2015) 1005–1053.
- [74] T.E. Tezduyar and D.K. Ganjoo, “Petrov-Galerkin formulations with weighting functions dependent upon spatial and temporal discretization: Applications to transient convection-diffusion problems”, *Computer Methods in Applied Mechanics and Engineering*, **59** (1986) 49–71, doi: 10.1016/0045-7825(86)90023-X.

- [75] G.J. Le Beau, S.E. Ray, S.K. Aliabadi, and T.E. Tezduyar, “SUPG finite element computation of compressible flows with the entropy and conservation variables formulations”, *Computer Methods in Applied Mechanics and Engineering*, **104** (1993) 397–422, doi: 10.1016/0045-7825(93)90033-T.
- [76] T.E. Tezduyar, “Finite elements in fluids: Stabilized formulations and moving boundaries and interfaces”, *Computers & Fluids*, **36** (2007) 191–206, doi: 10.1016/j.compfluid.2005.02.011.
- [77] T.E. Tezduyar and M. Senga, “Stabilization and shock-capturing parameters in SUPG formulation of compressible flows”, *Computer Methods in Applied Mechanics and Engineering*, **195** (2006) 1621–1632, doi: 10.1016/j.cma.2005.05.032.
- [78] T.E. Tezduyar and M. Senga, “SUPG finite element computation of inviscid supersonic flows with $YZ\beta$ shock-capturing”, *Computers & Fluids*, **36** (2007) 147–159, doi: 10.1016/j.compfluid.2005.07.009.
- [79] T.E. Tezduyar, M. Senga, and D. Vicker, “Computation of inviscid supersonic flows around cylinders and spheres with the SUPG formulation and $YZ\beta$ shock-capturing”, *Computational Mechanics*, **38** (2006) 469–481, doi: 10.1007/s00466-005-0025-6.
- [80] T.E. Tezduyar and S. Sathe, “Enhanced-discretization selective stabilization procedure (EDSSP)”, *Computational Mechanics*, **38** (2006) 456–468, doi: 10.1007/s00466-006-0056-7.
- [81] A. Corsini, F. Rispoli, A. Santoriello, and T.E. Tezduyar, “Improved discontinuity-capturing finite element techniques for reaction effects in turbulence computation”, *Computational Mechanics*, **38** (2006) 356–364, doi: 10.1007/s00466-006-0045-x.
- [82] F. Rispoli, A. Corsini, and T.E. Tezduyar, “Finite element computation of turbulent flows with the discontinuity-capturing directional

- p>
dissipation (DCDD)”,
- Computers & Fluids*
- ,
- 36**
- (2007) 121–126, doi: 10.1016/j.compfluid.2005.07.004.
- [83] T.E. Tezduyar, S. Ramakrishnan, and S. Sathe, “Stabilized formulations for incompressible flows with thermal coupling”, *International Journal for Numerical Methods in Fluids*, **57** (2008) 1189–1209, doi: 10.1002/fld.1743.
 - [84] F. Rispoli, R. Saavedra, A. Corsini, and T.E. Tezduyar, “Computation of inviscid compressible flows with the V-SGS stabilization and $YZ\beta$ shock-capturing”, *International Journal for Numerical Methods in Fluids*, **54** (2007) 695–706, doi: 10.1002/fld.1447.
 - [85] Y. Bazilevs, V.M. Calo, T.E. Tezduyar, and T.J.R. Hughes, “ $YZ\beta$ discontinuity-capturing for advection-dominated processes with application to arterial drug delivery”, *International Journal for Numerical Methods in Fluids*, **54** (2007) 593–608, doi: 10.1002/fld.1484.
 - [86] A. Corsini, C. Menichini, F. Rispoli, A. Santoriello, and T.E. Tezduyar, “A multiscale finite element formulation with discontinuity capturing for turbulence models with dominant reactionlike terms”, *Journal of Applied Mechanics*, **76** (2009) 021211, doi: 10.1115/1.3062967.
 - [87] F. Rispoli, R. Saavedra, F. Menichini, and T.E. Tezduyar, “Computation of inviscid supersonic flows around cylinders and spheres with the V-SGS stabilization and $YZ\beta$ shock-capturing”, *Journal of Applied Mechanics*, **76** (2009) 021209, doi: 10.1115/1.3057496.
 - [88] A. Corsini, C. Iossa, F. Rispoli, and T.E. Tezduyar, “A DRD finite element formulation for computing turbulent reacting flows in gas turbine combustors”, *Computational Mechanics*, **46** (2010) 159–167, doi: 10.1007/s00466-009-0441-0.
 - [89] M.-C. Hsu, Y. Bazilevs, V.M. Calo, T.E. Tezduyar, and T.J.R. Hughes, “Improving stability of stabilized and multiscale formulations in flow simu-

- lations at small time steps”, *Computer Methods in Applied Mechanics and Engineering*, **199** (2010) 828–840, doi: 10.1016/j.cma.2009.06.019.
- [90] A. Corsini, F. Rispoli, and T.E. Tezduyar, “Stabilized finite element computation of NOx emission in aero-engine combustors”, *International Journal for Numerical Methods in Fluids*, **65** (2011) 254–270, doi: 10.1002/fld.2451.
- [91] A. Corsini, F. Rispoli, and T.E. Tezduyar, “Computer modeling of wave-energy air turbines with the SUPG/PSPG formulation and discontinuity-capturing technique”, *Journal of Applied Mechanics*, **79** (2012) 010910, doi: 10.1115/1.4005060.
- [92] A. Corsini, F. Rispoli, A.G. Sheard, and T.E. Tezduyar, “Computational analysis of noise reduction devices in axial fans with stabilized finite element formulations”, *Computational Mechanics*, **50** (2012) 695–705, doi: 10.1007/s00466-012-0789-4.
- [93] P.A. Kler, L.D. Dalcin, R.R. Paz, and T.E. Tezduyar, “SUPG and discontinuity-capturing methods for coupled fluid mechanics and electrochemical transport problems”, *Computational Mechanics*, **51** (2013) 171–185, doi: 10.1007/s00466-012-0712-z.
- [94] A. Corsini, F. Rispoli, A.G. Sheard, K. Takizawa, T.E. Tezduyar, and P. Venturini, “A variational multiscale method for particle-cloud tracking in turbomachinery flows”, *Computational Mechanics*, **54** (2014) 1191–1202, doi: 10.1007/s00466-014-1050-0.
- [95] F. Rispoli, G. Delibra, P. Venturini, A. Corsini, R. Saavedra, and T.E. Tezduyar, “Particle tracking and particle–shock interaction in compressible-flow computations with the V-SGS stabilization and $YZ\beta$ shock-capturing”, *Computational Mechanics*, **55** (2015) 1201–1209, doi: 10.1007/s00466-015-1160-3.
- [96] M.F. Wheeler, “An elliptic collocation-finite element method with interior penalties”, *SIAM Journal on Numerical Analysis*, **15** (1978) 152–161.

- [97] P. Houston, C. Schwab, and E. Suli, “Discontinuous hp-finite element methods for advection-diffusion reaction problems”, *SIAM Journal on Numerical Analysis*, **39** (2002) 2133–2163.
- [98] Y. Bazilevs and T.J.R. Hughes, “Weak imposition of Dirichlet boundary conditions in fluid mechanics”, *Computers and Fluids*, **36** (2007) 12–26.
- [99] Y. Saad and M. Schultz, “GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems”, *SIAM Journal of Scientific and Statistical Computing*, **7** (1986) 856–869.

Figure 7: 2D model. Global mesh (top) and local mesh around the tire (bottom). The number of nodes is 68,491, and the number of elements is 123,037. Around the tire, there are 360 elements in the circumferential direction, 10 thin layers of elements in the tread grooves, and 10 more thin layers beyond the tread. The mesh between the tire surface and the SI is made of quadrilateral elements and generated manually. There is one more layer of manually generated quadrilateral elements beyond the SI. The rest of the mesh is made of triangular elements.

Figure 8: 2D model. Mesh at the first 10 of 20 equally spaced instants during the rotation. Some of the elements are colored red to show the mesh rotation and deformation and element collapse.

Figure 9: 2D model. Mesh at the second 10 of 20 equally spaced instants during the rotation. Some of the elements are colored red to show the mesh rotation and deformation and element collapse.

Figure 10: 2D model. Velocity magnitude at the same instants as in Figures 8.

Figure 11: 2D model. Velocity magnitude at the same instants as in Figures 9.

Figure 12: 2D model. Close-up view of the velocity magnitude around the tire at the same instants as in Figures 8.

Figure 13: 2D model. Close-up view of the velocity magnitude around the tire at the same instants as in Figures 9.

Figure 14: 2D model. Close-up view of the velocity magnitude around a tread for a time period close to the end of the second rotation.

Figure 15: 3D model. Boundary conditions. The left (*red*) and bottom (*yellow*) boundaries represent the inflow and road, where the velocity is set to the liner speed at the tire periphery. The innermost (*blue*) cylinder-like shape is the tire surface, which is rotating. The larger (*pink*) cylinder-like shape is the SI. The bottom of the SI is coinciding with the road and the tire–road interface. The conditions at the right (*green*) boundary and remaining three lateral boundaries are traction-free and slip, respectively.

Figure 16: 3D model. Mesh at the external boundaries and the inner side of the SI (*top*), inner side of the SI together with two cut surfaces (*middle*), inner side of the SI (*bottom left*), and the tire surface (*bottom right*). The number of nodes is 2,370,840, and number of elements is 5,415,467. The number of elements in the circumferential direction and the layering with thin elements in the tread groves and beyond the tread are the same as in the 2D model. The mesh between the tire surface and the SI are made of hexahedral elements and generated manually. There is one more layer of manually generated hexahedral elements beyond the SI and then one layer of manually generated pyramid elements. The rest of the mesh is made of tetrahedral elements.

Figure 17: 3D model. Velocity magnitude on the central plane and the tire surface after approximately 0.8 rotations. Global view (*top*) and near the contact area (*bottom*).