Computational analysis of flow-driven string dynamics in a pump and residence time calculation

K Komiya¹, T Kanai¹, Y Otoguro¹, M Kaneko¹, K Hirota¹, Y Zhang², K Takizawa¹, T E Tezduyar¹,², M Nohmi³, T Tsuneda³, M Kawai⁴ and M Isono⁴

¹ Department of Modern Mechanical Engineering, Waseda University
  3-4-1 Ookubo, Shinjuku-ku, Tokyo 169-8555, Japan
² Mechanical Engineering, Rice University 6100 Main Street, TX 77005, USA
³ Research Department, Technologies, R&D Division, Ebara corporation
⁴ Development and Engineering Department, Standard Pump Business Division, Ebara corporation

E-mail: Kenji.Takizawa@tafsm.org

Abstract. We present computational analysis of flow-driven string dynamics in a pump and the related residence time calculation. The objective in the study is to understand how the strings carried by a fluid interact with the pump surfaces, including the blades, and get stuck on or around those surfaces. The residence time calculations help us to have a simplified but quick understanding of the string behavior. The core computational method is the Space–Time Variational Multiscale (ST-VMS) method, and the other key methods are the ST Isogeometric Analysis (ST-IGA), ST Slip Interface (ST-SI) method, ST/NURBS Mesh Update Method (STNMUM), a general-purpose NURBS mesh generation method for complex geometries, and a one-way-dependence model for the string dynamics. The ST-IGA with NURBS basis functions in space is used in both fluid mechanics and string structural dynamics. The ST framework provides higher-order accuracy. The VMS feature of the ST-VMS addresses the computational challenges associated with the turbulent nature of the unsteady flow, and the moving-mesh feature of the ST framework enables high-resolution computation near the rotor surface. The ST-SI enables moving-mesh computation of the spinning rotor. The mesh covering the rotor spins with it, and the SI between the spinning mesh and the rest of the mesh accurately connects the two sides of the solution. The ST-IGA enables more accurate representation of the pump geometry and increased accuracy in the flow solution. The IGA discretization also enables increased accuracy in the structural dynamics solution, as well as smoothness in the string shape and fluid dynamics forces computed on the string. The STNMUM enables exact representation of the mesh rotation. The general-purpose NURBS mesh generation method makes it easier to deal with the complex geometry. With the one-way-dependence model, we compute the influence of the flow on the string dynamics, while avoiding the formidable task of computing the influence of the string on the flow, which we expect to be small.

1. Introduction
In pumps, objects carried by the fluid, such as a piece of string, can get stuck on or around the blades, possibly hindering the rotor motion. Computational analysis of flow-driven string dynamics pumps and the related residence time calculation can enable a better understanding of how the strings carried by the fluid interact with the pump surfaces, including the blades, and
get stuck on or around those surfaces. In the computational analysis we present, the core method is the Space–Time Variational Multiscale (ST-VMS) method [1]. The other key methods are the ST Isogeometric Analysis (ST-IGA) [2], ST Slip Interface (ST-SI) method [3], ST/NURBS Mesh Update Method (STNMUM) [4], a general-purpose NURBS mesh generation method for complex geometries [5], and a one-way-dependence model for the string dynamics [6].

1.1. **ST-VMS and ST-SUPS**

The ST-VMS is the VMS version of the Deforming-Spatial-Domain/Stabilized ST (DSD/SST) method [7]. The DSD/SST was introduced for computation of flows with moving boundaries and interfaces (MBI), including fluid–structure interaction (FSI). In MBI computations the DSD/SST functions as a moving-mesh method. Moving the fluid mechanics mesh to track a fluid–solid interface enables high-resolution representation of the boundary layer. Because the stabilization components of the DSD/SST are the Streamline-Upwind/Petrov-Galerkin (SUPG) [8] and Pressure-Stabilizing/Petrov-Galerkin (PSPG) [7] stabilizations, the method is also called “ST-SUPS.” The VMS components of the ST-VMS are from the residual-based VMS (RBVMS) method [9]. The Arbitrary Lagrangian–Eulerian (ALE) method is an earlier and more commonly used moving-mesh method. The ALE-VMS method [10] is the VMS version of the ALE. The ALE-VMS and RBVMS have been successfully applied to many classes of FSI, MBI and fluid mechanics problems (see the classes of problems listed and references cited in [11]). The ST-SUPS and ST-VMS have also been successfully applied to many classes of FSI, MBI and fluid mechanics problems (see the classes of problems listed and references cited in [11]), including the flow-driven string dynamics in turbomachinery [12]. In the computational analysis here, the ST framework provides higher-order accuracy. The VMS feature of the ST-VMS addresses the computational challenges associated with the multiscale and turbulent nature of the unsteady flow in the pump. The moving-mesh feature of the ST framework enables high-resolution computation near the rotor surface. The advection equation involved in calculation of the residence time is solved with the ST-SUPG method.

1.2. **Discontinuity-capturing term**

When the flow field has a shock or some other discontinuity, stabilized methods are often supplemented with a discontinuity-capturing (DC) term. A number of DC parameters were introduced to be used in the SUPG, ST-SUPG, compressible-flow SUPG and compressible-flow ST-SUPG computations (see the overview and references cited in [13]), including the YZ $\beta$ DC parameter [14]. In the computational analysis presented here, the ST-SUPG method used in solving the residence time advection equation is supplemented with the YZ $\beta$ DC.

1.3. **ST-SI**

The ST-SI was introduced in [3], in the context of incompressible-flow equations, to retain the desirable moving-mesh features of the ST-VMS when we have spinning solid surfaces, such as a turbine rotor. The mesh covering the spinning surface spins with it, retaining the high-resolution representation of the boundary layers. The SI between the spinning mesh and the rest of the mesh accurately connects the two sides of the flow field. The starting point in the development of the ST-SI was the ALE-VMS version for “sliding interfaces” [15]. In the ST-SI, 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. An ST-SI version where the SI is between fluid and solid domains with weakly-enforced Dirichlet boundary conditions for the fluid was also presented in [3]. The SI in this case is a “fluid–solid SI” rather than a standard “fluid–fluid SI.” The ST-SI method introduced in [16] for the coupled incompressible-flow and thermal-transport equations retains the high-resolution representation of the thermo-fluid boundary layers near spinning solid surfaces. These ST-SI methods have been successfully applied to many classes
of problems (see the classes of problems listed and references cited in [11]), including the flow-driven string dynamics in turbomachinery [12]. In the computational analysis presented here, the ST-SI enables moving-mesh computation of the spinning pump rotor. The mesh covering the rotor spins with it, and the SI between the spinning mesh and the rest of the mesh accurately connects the two sides of the solution.

1.4. ST-IGA and STNMUM
The ST-IGA was introduced in [1]. It is the integration of the ST framework with isogeometric discretization. First computations with the ST-VMS and ST-IGA were reported in [1] in a 2D context, with IGA basis functions in space for flow past an airfoil, and in both space and time for the advection equation. The stability and accuracy analysis given [1] for the advection equation showed that using higher-order basis functions in time would be essential in getting full benefit out of using higher-order basis functions in space. In the early stages of the ST-IGA, the emphasis was on IGA basis functions in time. As pointed out in [1] and demonstrated in [4], higher-order NURBS basis functions in time provide a more accurate representation of the motion of the solid surfaces and a mesh motion consistent with that. They also provide more efficiency in temporal representation of the motion and deformation of the volume meshes, and better efficiency in remeshing. That is how the ST/NURBS Mesh Update Method (STNMUM) was introduced [4]. The STNMUM has a wide scope that includes spinning solid surfaces. With the spinning motion represented by quadratic NURBS basis functions in time, and with sufficient number of temporal patches for a full rotation, the circular paths are represented exactly, and a “secondary mapping” [1] enables also specifying a constant angular velocity for invariant speeds along the paths. The ST framework and NURBS in time also enable, with the “ST-C” method, extracting a continuous representation from the computed data and, in large-scale computations, efficient data compression [17, 18]. The STNMUM and desirable features of the ST-IGA with IGA basis functions in time have been demonstrated in many 3D computations (see the classes of problems listed and references cited in [11]). The ST-IGA with IGA basis functions in space provides more accurate representation of the geometry and increased accuracy in the flow solution. Because it accomplishes that with less number of control points, and consequently with larger effective element sizes, it enables using larger time-step sizes while keeping the Courant number at a desirable level for good accuracy. It has been utilized in ST computational flow analysis of several classes of problems (see the classes of problems listed and references cited in [11]). In the flow analysis presented here, the ST-IGA enables more accurate representation of the pump geometry and increased accuracy in the flow solution. The IGA discretization also enables increased accuracy in the structural dynamics solution, as well as smoothness in the string shape and fluid dynamics forces computed on the string. The STNMUM enables exact representation of the mesh rotation.

1.5. General-purpose NURBS mesh generation method
To make the ST-IGA use, and in a wider context the IGA use, even more practical in computational flow analysis with complex geometries, NURBS volume mesh generation needs to be easier and more automated. To that end, a general-purpose NURBS mesh generation method was introduced in [5]. The method is based on multi-block-structured mesh generation with existing techniques, projection of that mesh to a NURBS mesh made of patches that correspond to the blocks, and recovery of the original model surfaces. The recovery of the original surfaces is to the extent they are suitable for accurate and robust fluid mechanics computations. The method is expected to retain the refinement distribution and element quality of the multi-block-structured mesh that we start with. Because there are ample good techniques and software for generating multi-block-structured meshes, the method makes general-purpose mesh generation relatively easy. Mesh-quality performance studies for 2D and 3D meshes, including those for
complex models, were presented in [19]. A test computation for a turbocharger turbine and exhaust manifold was also presented in [19]. The performance studies and test computation demonstrated that the general-purpose NURBS mesh generation method makes the IGA use in fluid mechanics computations even more practical. The general-purpose NURBS mesh generation method is used also in the pump flow analysis presented here.

1.6. ST-C
As we compute the flow field, we store the computed time-dependent data with a special data compression method based on the ST-C method [17]. With the ST-C method, we can represent the data with fewer temporal control points, resulting in reduced computer storage cost. In one of the two ST-C versions introduced in [17], the continuous representation is extracted by projection from a solution already computed. Because we use a successive-projection technique (SPT), with a small number of temporal NURBS basis functions at each projection, the extraction can take place as the original solution is being computed, without the need to first complete the computation and store all that data. This version was named “ST-C-SPT” in [17]. In the work reported in this article, the large time-history data from the flow field computation is stored using the ST-C-SPT method. The stored data is used in computation of the flow-driven string dynamics and in residence time calculation.

1.7. One-way dependence model
Because a string is a very thin object, its influence on the flow will be very small. With the one-way-dependence model [6], we compute the influence of the flow on the string dynamics, while avoiding the formidable task of computing the influence of the string on the flow. The one-way-dependence model has been used in other contexts of computational analysis. The examples we are familiar with are calculating the aerodynamic forces acting on the suspension lines of spacecraft parachutes [6] and calculating the forces acting on the particles in particle-laden flows [20]. In the flow-driven string dynamics computation here, we first compute the flow field and store the time-dependent flow data with the ST-C, and then compute the string dynamics.

1.8. Outline of the remaining sections
We describe the string dynamics and residence time computation methods in Section 2. In Section 3, we present the flow-driven string dynamics computation in a pump and the related residence time calculation. The concluding remarks are given in Section 4.

2. Methods
The ST-VMS and ST-SI can be found in [12], and the ST-IGA in [2].

2.1. String dynamics
The string is modeled with bending-stabilized cable elements [21], using the IGA with cubic NURBS basis functions. This gives us a higher-order method and smoothness in the structure shape. It also gives us smoothness in the fluid forces acting on the string. In the one-way-dependence model, the forces acting on the string are calculated with the method described in [6] for computing the aerodynamic forces acting on the suspension lines of spacecraft parachutes. Contact between the string and solid surfaces is handled with the Surface-Edge-Node Contact Tracking (SENCT-FC) method [22], which is a newer version of the SENCT [23].
2.2. Particle residence time
The residence time in domain $\Omega_s \subset \Omega$ can be written as
\[ \frac{dR}{dt} = s(x), \]
where $s(x) = 1$ on $\Omega_s$ and $s(x) = 0$ on $\Omega \setminus \Omega_s$. The Eulerian form of the equation is
\[ \frac{\partial R}{\partial t} + \mathbf{u} \cdot \nabla R = s, \]
and we solve that with the ST-SUPG method supplemented with the YZ$\beta$ DC [14]. Integration of equation (2) over $\Omega_s$ gives
\[ \int_{\Omega_s} \left( \frac{\partial R}{\partial t} + \mathbf{u} \cdot \nabla R \right) d\Omega = \int_{\Omega_s} s d\Omega. \] (3)
We assume $\nabla \cdot \mathbf{u} = 0$ and obtain
\[ \frac{d}{dt} \left( \int_{\Omega_s} R d\Omega \right) + \int_{\Gamma_s} \mathbf{n} \cdot (\mathbf{u} - \mathbf{v}) R d\Gamma = V, \] (4)
where $\Gamma_s$ is the boundary of $\Omega_s$, $\mathbf{v}$ is the boundary velocity, and
\[ V = \int_{\Omega_s} d\Omega. \] (5)
We define the flow-rate-averaged residence time as
\[ \bar{R}_{\text{out}} = \frac{1}{Q} \int_{(\Gamma_s)_{\text{out}}} \mathbf{n} \cdot \mathbf{u} R d\Gamma, \] (6)
\[ Q = \int_{(\Gamma_s)_{\text{out}}} \mathbf{n} \cdot d\Gamma, \] (7)
where subscript “out” indicates the outlet.
In a typical setting, there is no flow coming back to $\Omega_s$, $\mathbf{u} = \mathbf{v}$ on the part of $\Gamma_s$ corresponding to the rotor, and $\mathbf{v} = 0$ at the inlet and outlet. If we assume that first term in equation (4) is zero, $\bar{R}_{\text{out}} = V/Q$. If any part of $\Omega_s$ is enclosed by streamlines, the first term cannot be zero.

3. Computations
3.1. Flow analysis of the pump
3.1.1. Pump geometry
We use a vortex pump with 6 blades, including two higher-height blades. We are unable to provide more details due to the industrial-partner restrictions.
3.1.2. Mesh
The quadratic NURBS mesh for the flow analysis is shown in Figure 1. The number of control points and elements are 838,222 and 544,466.
3.1.3. Conditions
The pump is used for water, the density is 998.2 kg/m$^3$, and the kinematic viscosity $8.7 \times 10^{-7}$ m$^2$/s. The boundary conditions are shown in Figure 2. At the inlet, $Q = 5.46 \times 10^{-3}$ m$^3$/s. The time-step size is $9.8 \times 10^{-5}$ s. The number of nonlinear iterations per time step is 3, and the number of GMRES iterations per nonlinear iteration is 100. Stabilization parameters of the ST-VMS method are those given by equations (2.4)–(2.6), (2.8) and (2.10) in [3].
3.1.4. Results  Figure 3 shows the second invariant of the velocity gradient tensor. The turbulent nature of the flow is well captured. The solution is compared to the experimental data from Professor Miyagawa’s group (Waseda University). The conditions here are close to those corresponding to the best-efficiency operating point, and the relative error in the efficiency compared to the experimental data is less than 1.5 %.

3.2. String dynamics in the pump

3.2.1. Problem setup and mesh  The string has 10 mm length, 1.5 mm diameter, and circular-shape moment of inertia. The Young’s module and density are 5.0 MPa and 960 kg/m$^3$. We use a cubic NURBS mesh, with 19 control points and 16 elements.

3.2.2. Conditions  The flow field is from the fluid mechanics computation. The time period for rotations 17 through 21 is used repeatedly. There are 17 different initial positions, shown in Figure 4. The initial string velocity is 2.0 m/s, in the flow direction. The time-step size is $9.8 \times 10^{-4}$ s, which is 10 times smaller than the time-step size used in the flow computation. The number of nonlinear iterations per time step is 3, with full GMRES.

3.2.3. Results  Figure 5 shows the string with the initial position at A (see Figure 4). The string first hits the top of the blade, and then moves to the edge of the pump casing.

3.3. Time dependent residence time for the pump

3.3.1. Conditions  We set the entire pump domain as $\Omega_s$ for the residence time. The flow velocity is from the fluid mechanics computation. The time period for rotations 17 through 21 is used repeatedly. The computation is carried out with a time-step size of $4.9 \times 10^{-4}$ s, which

---

**Figure 1.** Control mesh. Red circle represents the control points.

**Figure 2.** Boundary conditions. Inlet (red) and outlet (blue). The wall and the blades (green) have no-slip condition. The circular interface (yellow) is the SI.
\( t = 0 \) s
\( t = 0.023 \) s
\( t = 0.047 \) s
\( t = 0.071 \) s
\( t = 0.094 \) s
\( t = 0.119 \) s

**Figure 3.** Isosurfaces of the second invariant value of velocity gradient tensor, colored by the velocity magnitude (m/s).

**Figure 4.** The initial positions of the strings at the inlet plane.

is 5 times larger than the one for the flow computation. The number of nonlinear iterations per time step is 2, and the number of GMRES iterations per nonlinear iteration is 30.

### 3.3.2. Results

The flow-rate-averaged residence time over the outlet is shown in Figure 6. After 1.2 s it reaches the maximum value. Figure 7 shows the spatial distribution of the residence time at the end of the computation. The residence time under the rotor is much higher than the residence time at the outlet, which is around 0.4 s. This means that this region is not connected to the main flow.
3.4. Discussion
We discuss the relationship between the string dynamics and the residence time. Figure 8 shows the time histories of the string centroid positions in radius and height. We see some strings moving in circles along the bottom edges of the casing. These strings tend to stay there and cannot rise up. Therefore they stay in the pump forever. This can be correlated with the high residence time at the bottom of the pump (Figure 7).

4. Concluding remarks
Computational analysis of flow-driven string dynamics in pumps is helping us better understand how the strings carried by a fluid interact with the pump surfaces, including the blades, and get stuck on or around those surfaces. The related residence time calculation is also helping us increase our understanding, with an approach that is rather simplified but quick. In the
computational analysis, the core method is the ST-VMS, and the other key methods are the ST-IGA, ST-SI, STNMUM, a general-purpose NURBS mesh generation method for complex geometries, and a one-way-dependence model for the string dynamics. The ST framework provides higher-order accuracy. The VMS feature of the ST-VMS addresses the computational challenges associated with the turbulent nature of the unsteady flow in the pump, and the moving-mesh feature of the ST framework enables high-resolution computation near the rotor surface. The ST-SI enables moving-mesh computation of the spinning rotor. The mesh covering the rotor spins with it, and the SI between the spinning mesh and the rest of the mesh accurately connects the two sides of the solution. The ST-IGA enables more accurate representation of the pump geometry and increased accuracy in the flow solution. The IGA discretization also enables increased accuracy in the structural dynamics solution, as well as smoothness in the string shape and fluid dynamics forces computed on the string. The STNMUM enables exact representation of the mesh rotation. The general-purpose NURBS mesh generation method makes it somewhat easier to deal with the complex geometry. With the one-way-dependence model, we are able to compute the influence of the flow on the string dynamics, while avoiding the formidable task of computing the influence of the string on the flow, which we expect to be small. The computation presented shows the effectiveness of these methods in the class of problems we are targeting, and the residence time calculated shows good correlation with the string residence time we observe from the flow-driven string-dynamics computation.

Acknowledgement
This work was supported in part by Grant-in-Aid for Challenging Exploratory Research 16K13779 from Japan Society for the Promotion of Science; Grant-in-Aid for Scientific Research (S) 26220002 from the Ministry of Education, Culture, Sports, Science and Technology of Japan (MEXT); and Rice–Waseda research agreement. This work was also supported (second author) in part by Grant-in-Aid for JSPS Research Fellow 16J10373. This work was also supported (sixth and eighth authors) in part by ARO Grant W911NF-17-1-0046. We thank Professor Miyagawa (Waseda University) for kindly providing the experimental data.

[1] Takizawa K and Tezduyar T E 2011 Computational Mechanics 48 247–267
[2] Takizawa K, Tezduyar T E, Ototguro Y, Terahara T, Kuraishi T and Hattori H 2017 Computers & Fluids 142 15–20
[3] Takizawa K, Tezduyar T E, Mochizuki H, Hattori H, Mei S, Pan L and Montel K 2015 Mathematical Models and Methods in Applied Sciences 25 2377–2406
[4] Takizawa K, Henicke B, Puntel A, Spielman T and Tezduyar T E 2012 Journal of Applied Mechanics 79 010903
[5] Ototguro Y, Takizawa K and Tezduyar T E 2017 Computers & Fluids 158 189–200
[6] Tezduyar T E, Takizawa K, Moorman C, Wright S and Christopher J 2010 International Journal for Numerical Methods in Fluids 64 1201–1218
[7] Tezduyar T E 1992 Advances in Applied Mechanics 28 1–44
[8] Brooks A N and Hughes T J R 1982 Computer Methods in Applied Mechanics and Engineering 32 199–259
[9] Hughes T J R 1995 Computer Methods in Applied Mechanics and Engineering 127 387–401
[10] Bazilevs Y, Calo V M, Hughes T J R and Zhang Y 2008 Computational Mechanics 43 3–37
[11] Ototguro Y, Takizawa K, Tezduyar T E, Nagaoka K and Mei S May 2018 Turbocharger turbine and exhaust manifold flow computation with the Space–Time Variational Multiscale Method and Isogeometric Analysis Computers & Fluids, published online, DOI: 10.1016/j.compfluid.2018.05.019
[12] Takizawa K, Tezduyar T E and Hattori H 2017 Computers & Fluids 142 109–117
[13] Takizawa K, Tezduyar T E and Kanai T 2017 Mathematical Models and Methods in Applied Sciences 27 771–806
[14] Tezduyar T E 2007 Computers & Fluids 36 191–206
[15] Bazilevs Y and Hughes T J R 2008 Computational Mechanics 43 143–150
[16] Takizawa K, Tezduyar T E, Kuraishi T, Tabata S and Takagi H 2016 Computational Mechanics 57 965–977
[17] Takizawa K and Tezduyar T E 2014 Computational Mechanics 53 91–99
[18] Takizawa K, Tezduyar T E and Kuraishi T 2015 Mathematical Models and Methods in Applied Sciences 25 2227–2255
[19] Ototguro Y, Takizawa K and Tezduyar T E 2018 Frontiers in Computational Fluid–Structure Interaction and Flow Simulation: Research from Lead Investigators under Forty – 2018 Modeling and Simulation in Science, Engineering and Technology ed Tezduyar T E (Springer) pp 399–434 ISBN 978-3-319-96468-3 URL http://doi.org/10.1007/978-3-319-96469-0_10
[20] Corsini A, Rispoli F, Sheard A G, Takizawa K, Tezduyar T E and Venturini P 2014 Computational Mechanics 54 1191–1202
[21] Raknes S B, Deng X, Bazilevs Y, Benson D J, Mathisen K M and Kvamsdal T 2013 Computer Methods in Applied Mechanics and Engineering 263 127–143
[22] Takizawa K, Spielman T and Tezduyar T E 2011 Computational Mechanics 48 345–364
[23] Tezduyar T E and Sathe S 2007 International Journal for Numerical Methods in Fluids 54 855–900