A stable method for 4D CT-based CFD simulation in the right ventricle of a TGA patient

Yuri V. Vassilevski[*]   Alexander Danilov [†]   Alexander Lozovskiy[‡]

Maxim A. Olshanskii[§]   Victoria Yu. Salamatova[¶]   Su Min Chang [‖]

Yushui Han [**]   Huie Lin [††]

Abstract

The paper discusses a stabilization of a finite element method for the equations of fluid motion in a time-dependent domain. After experimental convergence analysis, the method is applied to simulate a blood flow in the right ventricle of a post-surgery patient with the transposition of the great arteries disorder. The flow domain is reconstructed from a sequence of 4D CT images. The corresponding segmentation and triangulation algorithms are also addressed in brief.

1 Introduction

Last decades evidenced a remarkable progress in the development of mathematical and computational models for physiological flows and their interaction with surrounding tissues and organs. In particular, computer simulations of the human cardiovascular system and its parts have been a focus of intense research. We refer to the recent monographs [1–3] for the overview of the field. Based on these advances, the CFD visualization and quantification of blood flow in the heart and large vesicles has a potential to become a clinical standard and to complement Doppler sonography as a decision supporting tool for practicing cardiologists [4]. Nevertheless, reliable and predictive patient-specific simulations of flow in the heart chambers remains a challenge, especially under pathological or post-surgery conditions. For example, for the CFD reconstruction of the flow in a heart ventricle, the challenge consists of quality image acquisition, image segmentation, recovery of tissue motions, a suitable volume tessellation (meshing), a discretization of the system of governing partial differential equations, fast solution methods and postprocessing of the computed solution. While the present paper briefly addresses several of these stages of personalized CFD simulations, the focus here is on building a stable discrete approximation of the Navier–Stokes equations for a transitional flow in an evolving domain. A concrete practical problem we are interested in here is the simulation of blood flow in the

---

[*]Marchuk Institute of Numerical Mathematics RAS, Sechenov University, and Moscow Institute of Physics and Technology; `yuri.vassilevski@gmail.com`

[†]Marchuk Institute of Numerical Mathematics RAS and Moscow Institute of Physics and Technology;

[‡]Marchuk Institute of Numerical Mathematics RAS;

[§]Department of Mathematics, University of Houston;

[¶]Sechenov University and Moscow Institute of Physics and Technology;

[‖]DeBakey Heart&Vascular Center, Houston Methodist Hospital;

[**]DeBakey Heart&Vascular Center, Houston Methodist Hospital;

[††]DeBakey Heart&Vascular Center, Houston Methodist Hospital;

right ventricle reconstructed from 4D CT images of the heart of a patient with transposition of the great arteries (TGA).

The rest of the paper is organized in five sections. In section 2 we introduce a system of the Navier–Stokes equations governing the motion of incompressible viscous fluid (blood) in a time-dependent domain (the heart chamber) with appropriate boundary conditions. Section 3 discusses a discretization method with particular attention to spatial stabilization. Section 4 adds details of the medical image processing and meshing techniques. Section 5 splits into two parts. In the first part, we show the results of convergence tests for the numerical method, which confirm the expected accuracy rates. The second part presents the CFD visualization and analysis of flow in the right ventricle.

2 Mathematical model

We represent the heart chamber as a 3D time-dependent domain $\Omega(t) \subset \mathbb{R}^3$ occupied by a viscous fluid (blood) for the simulation time $t \in [0, T]$. We assume that the deformation of $\Omega(t)$ is smooth enough in the sense that there exists a two times continuously differentiable one-to-one mapping $\boldsymbol{\xi}$ from the reference domain $\Omega_0 = \Omega(0)$ to the physical domain, i.e., $\boldsymbol{\xi} : \Omega_0 \to \Omega(t)$ for $t \in [0, T]$ and $\boldsymbol{\xi} \in C^2([0, T] \times \overline{\Omega}_0)$.

For the blood flow in the heart, it is reasonable to assume that the fluid is Newtonian and incompressible [3]. The dynamics of incompressible Newtonian fluid is governed by the system of Navier–Stokes equations,

$$\begin{cases} \dfrac{\mathrm{d}}{\mathrm{dt}}\mathbf{u} - 2\nu\mathrm{div}\left(\mathbf{D}(\mathbf{u})\right) + \nabla p = \mathbf{f} \\ \mathrm{div}\,\mathbf{u} = 0 \end{cases} \quad \text{in}\;\; \Omega(t), \;\; \text{for}\; t \in (0, T), \tag{1}$$

written for the unknown fluid velocity vector field $\mathbf{u}(\mathbf{x}, t)$ and the unknown pressure function $p(\mathbf{x}, t)$. In equation (1), ν is the kinematic viscosity coefficient, $\mathbf{D}(\mathbf{u})$ stands for the rate of deformation tensor, and $\frac{\mathrm{d}}{\mathrm{dt}}$ denotes the material (Lagrangian) derivative, i.e. the derivative along material trajectories of particles. For numerical purposes, it is convenient to expand the material derivative in Eulerian terms and to re-write the fluid system in the so-called arbitrary Lagrangian–Eulerian (ALE) form,

$$\begin{cases} \left.\dfrac{\partial \mathbf{u}}{\partial t}\right|_{\Omega_0} + ((\mathbf{u} - \mathbf{w}) \cdot \nabla)\mathbf{u} - 2\nu\mathrm{div}\left(\mathbf{D}(\mathbf{u})\right) + \nabla p = \mathbf{f} \\ \mathrm{div}\,\mathbf{u} = 0 \end{cases} \quad \text{in}\;\; \Omega(t), \;\; \text{for}\; t \in (0, T). \tag{2}$$

Here $\mathbf{w} := \dfrac{\partial \boldsymbol{\xi}(t, \boldsymbol{\xi}^{-1}(\mathbf{x}))}{\partial t}$ is the ALE velocity at $\mathbf{x} \in \Omega(t)$ and $\left.\dfrac{\partial \mathbf{u}}{\partial t}\right|_{\Omega_0} := \dfrac{\partial \hat{\mathbf{u}}(t, \boldsymbol{\xi}^{-1}(\mathbf{x}))}{\partial t}$, with $\hat{\mathbf{u}} := \mathbf{u} \circ \boldsymbol{\xi}$, is the time derivative of velocity in the reference frame. Further we shall write $\mathbf{u}_t$ for the sake of notation simplicity.

The system (2) should be supplemented with boundary conditions for velocity or stress. These boundary conditions drive the flow and so they are important part of the model. On the walls of the chamber, denoted by $\partial\Omega^{\mathrm{ns}}(t)$, we set no-penetration and no-slip boundary condition for fluid, which for the moving domain take the following form:

$$\mathbf{u} = \boldsymbol{\xi}_t \circ \boldsymbol{\xi}^{-1} \quad \text{on}\; \partial\Omega^{\mathrm{ns}}(t). \tag{3}$$

The part of the boundary corresponding to the tricuspid valve is designated as the inflow part, $\partial\Omega^{\mathrm{in}}(t)$, with the following boundary conditions

$$\mathbf{u} = \boldsymbol{\xi}_t \circ \boldsymbol{\xi}^{-1} \quad \text{on}\; \partial\Omega^{\mathrm{in}}(t), \text{ for systolic phase and } \boldsymbol{\sigma}\mathbf{n} = 0 \quad \text{on}\; \partial\Omega^{\mathrm{in}}(t), \text{ for diastolic phase,} \tag{4}$$

where $\boldsymbol{\sigma} = -2\nu\mathbf{D}(\mathbf{u}) + p\mathbf{I}$ is the Cauchy stress tensor, $\mathbf{n}$ is the unit normal vector on the boundary of $\Omega(t)$. These conditions imply that the blood freely flows in the ventricle through the tricuspid valve driven by the compartment expansion. The part of the boundary with the pulmonary valve represents the outflow part, $\partial\Omega^{\text{out}}(t)$, with the boundary condition

$$\boldsymbol{\sigma}\mathbf{n} = 0 \quad \text{on } \partial\Omega^{\text{out}}(t), \text{ for systolic phase} \quad \text{and} \quad \mathbf{u} = \boldsymbol{\xi}_t \circ \boldsymbol{\xi}^{-1} \quad \text{on } \partial\Omega^{\text{out}}(t), \text{ for diastolic phase.} \tag{5}$$

Finally, we need to define an initial state of the system, which in the absence of other data we assume to be the fluid at rest, $\mathbf{u} = 0$ in $\Omega(0)$. One may need to simulate several cardiac cycles to obtain $\mathbf{u}$ and p non-sensitive to the error induced by such non-physiological initial state.

To build a discrete model, we first need an integral formulation of the system. We multiply the first equation in (2) by a smooth vector function $\boldsymbol{\psi} : \Omega(t) \to \mathbb{R}^3$ such that $\boldsymbol{\psi} = \mathbf{0}$ on $\partial\Omega^{\text{ns}}(t) \cup \partial\Omega^{\text{in}}(t)$ during systolic phase and $\boldsymbol{\psi} = \mathbf{0}$ on $\partial\Omega^{\text{ns}}(t) \cup \partial\Omega^{\text{out}}(t)$ during diastolic phase. Integrating the resulting identity over $\Omega(t)$ and by parts we get

$$\int_{\Omega(t)} \{\mathbf{u}_t + ((\mathbf{u} - \mathbf{w}) \cdot \nabla)\mathbf{u}\} \cdot \boldsymbol{\psi}\,dx + 2\nu \int_{\Omega(t)} \mathbf{D}(\mathbf{u}) : \mathbf{D}(\boldsymbol{\psi})\,dx$$
$$- \int_{\Omega(t)} p\,\text{div}\,\boldsymbol{\psi}\,dx + \int_{\Omega(t)} q\,\text{div}\,\mathbf{u}\,dx = \int_{\Omega(t)} \mathbf{f} \cdot \boldsymbol{\psi}\,dx, \tag{6}$$

where we used boundary conditions and added the second equation in (2) tested by a smooth function $q : \Omega(t) \to \mathbb{R}$. For a regular solution to (2)–(5), i.e. for smooth velocity field $\mathbf{u}$ and pressure p, the identity (6) holds for any $t \in (0, T)$ and any smooth test functions q and $\boldsymbol{\psi}$ as specified above.

3 Discrete models

We start with building a triangulation of the reference domain Ω_0: Let $\mathcal{T}_h$ be a collection of tetrahedra such that $\overline{\Omega}_0 = \bigcup_{T \in \mathcal{T}_h} \overline{T}$ and any two tetrahedra from $\mathcal{T}_h$ intersect by either an entire face, an entire edge, a vertex, or the empty set. We also assume that the mesh is regular in the sense that the minimal angle condition holds for all tetrahedra from $\mathcal{T}_h$, cf., e.g., [5]. A triangulation satisfying all above conditions is called admissible. In turn, triangulations $\mathcal{T}_h(t)$ of $\Omega(t)$ are built using $\boldsymbol{\xi}(t)$ to map tetrahedra vertices from Ω_0 to $\Omega(t)$. The mapping $\boldsymbol{\xi}$ should be such that the resulting sequence of meshes has the same connectivity and delivers admissible triangulations of $\Omega(t)$. Consider conforming FE spaces $\mathbb{V}_h(t) \subset H^1(\Omega(t))^d$ and $\mathbb{Q}_h(t) \subset L^2(\Omega(t))$, spaces of continuous piecewise polynomial functions on $\mathcal{T}_h(t)$; $\mathbb{V}_h^0$ is a subspace of $\mathbb{V}_h$ of functions vanishing on the same part of the boundary as the test function from the integral formulation (6). In this paper, we choose the Taylor–Hood (P2/P1) pair of finite element spaces for the velocity–pressure pair [6]:

$$\begin{aligned} \mathbb{V}_h(t) &= \{\mathbf{u}_h \in C(\Omega(t))^3 \,:\, \mathbf{u}_h|_T \in [P_2(T)]^3, \forall\, T \in \mathcal{T}_h(t)\}, \\ \mathbb{Q}_h(t) &= \{q_h \in C(\Omega(t)) \,:\, q_h|_T \in P_1(T), \forall\, T \in \mathcal{T}_h(t)\}, \end{aligned} \tag{7}$$

which are known to satisfy the necessary inf-sup stability condition [7]. The mapping $\boldsymbol{\xi}$ is also approximated with a piecewise polynomial mapping $\boldsymbol{\xi}_h \in \mathbb{V}_h(0)$, which is constructed by the interpolation of $\boldsymbol{\xi}$ using its nodal values.

We now consider discretization in time. Assume a constant time step $\Delta t = \frac{T}{N}$, where N is the total number of steps. We use the notations $t_k = k\Delta t$, $\mathbf{u}^k := \mathbf{u}(t_k, \mathbf{x})$, and similar for p, $\boldsymbol{\xi}$, and $\Omega_k = \Omega(t_k)$. For a sequence of functions f^i, $i = 0, \ldots, k$, all defined in the

reference domain, $[f]_t^k := \frac{f^k - f^{k-1}}{\Delta t}$ denotes the backward finite difference at t_k. Let $\mathbf{u}_h^0$ be the Lagrange interpolant of the initial velocity field. The fully discrete problem builds on the integral formulation (6) and reads: For $k = 1, 2, \ldots$, find $\mathbf{u}_h^k \in \mathbb{V}_h(t_k)$, $p_h^k \in \mathbb{Q}_h(t_k)$ satisfying Dirichlet boundary conditions in (3),(4),(5), and the integral equality

$$\int_{\Omega_k} \{[\widehat{\mathbf{u}}_h]_t^k \circ (\boldsymbol{\xi}_h^k)^{-1} + ((\widetilde{\mathbf{u}}_h^{k-1} - \mathbf{w}_h^k) \cdot \nabla)\mathbf{u}_h^k\} \cdot \boldsymbol{\psi}_h \, dx + 2\nu \int_{\Omega_k} \mathbf{D}(\mathbf{u}_h^k) : \mathbf{D}(\boldsymbol{\psi}_h) \, dx$$

$$- \int_{\Omega_k} p_h^k \operatorname{div} \boldsymbol{\psi}_h \, dx + \int_{\Omega_k} q_h \operatorname{div} \mathbf{u}_h^k \, dx = \int_{\Omega_k} \mathbf{f} \cdot \boldsymbol{\psi}_h \, dx, \quad (8)$$

for all $\boldsymbol{\psi}_h \in \mathbb{V}_h^0(t_k)$, $q_h \in \mathbb{Q}_h(t_k)$. Here the advection velocity is computed with the help of the mapping to the reference domain, $\mathbf{w}_h^k(\mathbf{x}) = [\boldsymbol{\xi}_h]_t^k((\boldsymbol{\xi}_h^k)^{-1}(\mathbf{x}))$, as well as $\widetilde{\mathbf{u}}_h^{k-1}(\mathbf{x}) := \mathbf{u}_h^{k-1}(\boldsymbol{\xi}_h^{k-1}(\boldsymbol{\xi}_h^k)^{-1}(\mathbf{x}))$, for $\mathbf{x} \in \Omega^k$, and $\widehat{\mathbf{u}}_h^k := \mathbf{u}_h^k \circ \boldsymbol{\xi}_h^k$.

Note that the inertia terms are linearized so that a *linear* algebraic system should be solved on each time step.

Blood flow of the heart ventricles is characterized by transitional or even turbulent regimes, see, e.g., [8–10]. Therefore, sufficiently fine resolution of spatial and time scales in $\mathbf{u}$ is required for the direct numerical simulation of blood flows in the ventricles. Adopting such resolution would lead to computations prohibitively expensive for patient specific modelling. An alternative to employing very fine triangulation and time stepping is the use of a Large Eddy Simulation (LES) model of turbulence [10] or another subgrid method designed to model the effect of unresolved scales and to dissipate excessive energy. Such models aim to deliver stable numerical approximations for the mean flow statistics. This is the approach taken in the present paper. More specifically, we consider a combination of the classical Smagorinski LES model [11] and the streamline upwinded Petrov–Galerkin (SUPG) method [12].

SUPG method introduces subgrid modelling on the level of discretization [13]. The variant we use consists in adding to (8) the element-wise residual term of the form

$$\sum_{T \in \mathcal{T}_h(t_k)} \tau_T \int_T (\mathbf{a}_h^k \cdot \nabla)\boldsymbol{\psi}_h \left\{ [\widehat{\mathbf{u}}_h]_t^k + (\mathbf{a}_h^k \cdot \nabla)\mathbf{u}_h^k - 2\nu \operatorname{div}(\mathbf{D}(\mathbf{u}_h^k)) + \nabla p_h^k - \mathbf{f} \right\} dx, \quad (9)$$

with $\mathbf{a}_h^k := \widetilde{\mathbf{u}}_h^{k-1} - \mathbf{w}_h^k$,

The parameters τ_T are defined elementwise as follows

$$\tau_T = \frac{1}{\sqrt{d_1 + d_2 + d_3}},$$

with

$$d_1 = 60\nu^2 \operatorname{tr}(\mathbf{G}\mathbf{G}^T), \quad d_2 = \frac{4}{|\Delta t|^2}, \quad d_3 = (\mathbf{a}_h^k)^T \mathbf{G} \mathbf{a}_h^k.$$

Matrix $\mathbf{G} \in \mathbb{R}^{3\times3}$, is the element metric tensor defined as $\mathbf{G} = \mathbf{B}^T\mathbf{B}$, where $\mathbf{B} = \frac{\partial \boldsymbol{\alpha}}{\partial \mathbf{x}} \in \mathbb{R}^{4\times3}$, and $\boldsymbol{\alpha}$ is the mapping from the bulk to the barycentric coordinates of T.

The SUPG method is sufficiently accurate, since it is a residual type method. However, we found SUPG formulation alone not always sufficiently stable for flow regimes typical to the ventricles on acceptable computational grids. Therefore, we suggest to increase the robustness of the SUPG method by combining it with the Smagorinski model. In the Smagorinski approach the effect of unresolved scales is modeled through introducing additional (turbulent) viscosity in the equation. The turbulent viscosity depends on the local strain rate and adds to the physical viscosity. The Smagorinski model is relatively simple, but known to be excessively

dissipative, especially near walls [14]. We incorporate it in SUPG formulation by replacing ν in (8) with

$$\nu_T = \nu + M \cdot (0.2 h_T)^2 \sqrt{2\mathbf{D}(\widetilde{\mathbf{u}}_h^{k-1}) : \mathbf{D}(\widetilde{\mathbf{u}}_h^{k-1})}, \tag{10}$$

where $h_T = \operatorname{diam}(T)$, for any tetrahedral cell $T \in \mathcal{T}_h$. Factor M regulates how much turbulent viscosity we add to the method. Our goal is to set M to a minimal possible value, which makes SUPG stabilization robust for flow regimes typical for the ventricles. Modification of viscosity increases the finite element error, yet the method is consistent in the sense that the added viscosity vanishes as $h_T \to 0$.

4   Image segmentation and mesh generation

In this work, we apply the finite element CFD simulation for personalized modelling of the flow in the right ventricle of a patient with transposition of the great arteries (TGA), a rare congenital defect. The motion of the heart is reconstructed from 4D CT images. The incoming data set is a series of 10 contrast enhanced CT images with $512 \times 512 \times 304$ voxels and $0.355 \times 0.355 \times 0.5$ mm resolution. We crop and resample each CT image to $162 \times 112 \times 136$ voxels with 1 mm isotropic resolution. The right ventricle is manually segmented for each of the ten CT images using a semi-automatic level-set method from ITK-SNAP package [15].

Since the temporal resolution of ten images per cardiac cycle is not enough for smooth mesh transition, we also perform temporal resampling. For each frame we convert the binary mask of the segmented right ventricle to a signed distance function, which is negative inside the ventricle, positive outside, and zero at the ventricle boundary. These ten scalar frames are resampled to 90 frames per cycle using a cubic interpolation in time with the periodic conditions at the end points of the time interval. For each new frame, the boundary of the right ventricle is recovered as a zero isosurface of this interpolant. Thus, the set of 90 frames represent a periodic motion. We next apply a cyclic shift of the frame indices to set the starting frame to be the one with the maximum volume of the ventricle. Then the frame 44 shows the minimum volume of the ventricle. We assume that frames from 0 to 44 represent systole of the cardiac cycle, and frames from 45 to 89 represent diastole.

We manually select two static cutoff planes to represent the position of valves in the right ventricle. We note that actual valves are moving during the cardiac cycle and do not stay in the same planes. Therefore, we estimate the average position of the valves based on the available sparse temporal resolution.

The mesh generation process is similar to the one proposed in our previous work [16] and we refer to that paper for any omitted details. The algorithm requires a reference domain, which is defined implicitly as an enclosed volume of the averaged distance function over all 90 frames. This volume is also bounded by static valve planes. A reference quasi-uniform unstructured tetrahedral mesh is first constructed with the help of Delaunay triangulation algorithm from CGAL Mesh library [17]. We next improve the reference mesh quality using aniMBA library from the Ani3D package [18] and also enforce each tetrahedron to have at least one internal node.

At the next stage, we deform the reference mesh to sequentially adapt to all frames from 0 to 89, which is followed by the second cycle of the mesh adaptation to ensure a smooth periodic transition of the meshes from one cardiac cycle to the next one. Each step of the mesh deformation is split into two substeps. First, we move only boundary nodes while simultaneously propagating and smoothing the surface mesh. Each boundary node is shifted in the direction of the weighted sum of two vectors: the surface normal vector (weight 0.5) and the
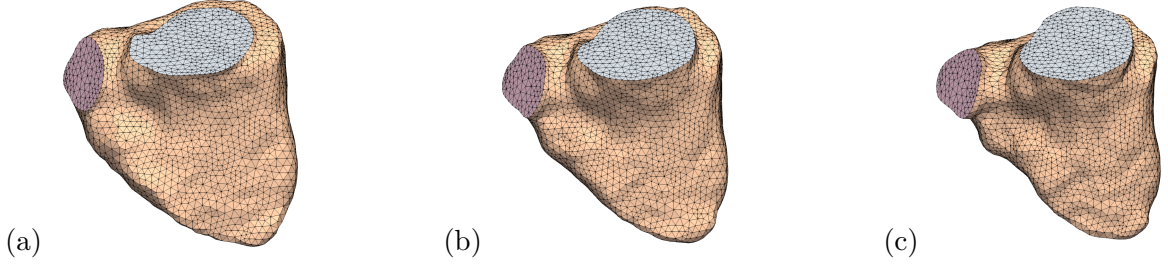
Figure 1: The right ventricle surface mesh: (a) beginning of systole (frame 0), (b) middle of systole (frame 23), (c) end of systole (frame 44).
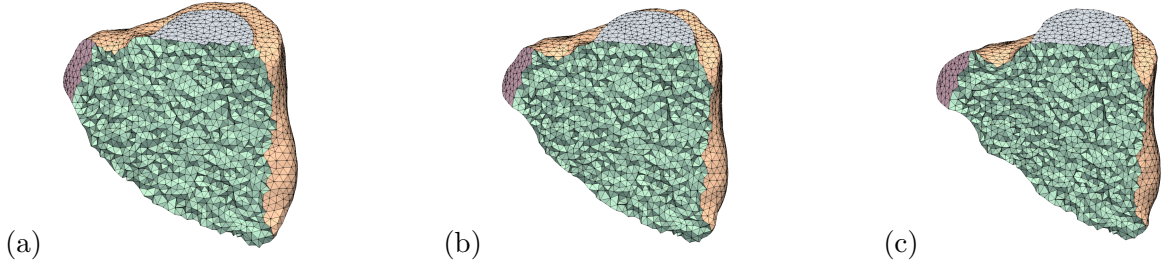

Figure 2: The right ventricle volume mesh, the cutplane passes through ventricle apex and the centers of the valves: (a) beginning of systole (frame 0), (b) middle of systole (frame 23), (c) end of systole (frame 44).

vector pointing at the center of surrounding nodes (weight 0.04). This procedure is repeated until the maximum displacement drops below $\varepsilon = 0.0001$ mm, or until the maximum number of 2000 iterations is exceeded. For vertices lying on the valve planes, the displacement vectors are projected to these planes, thus ensuring the vertices stay on valve planes (Fig. 1). At the second step, we apply a simultaneous untangling and smoothing algorithm [19]; the boundary nodes are then fixed, and only the internal nodes are shifted. The untangling stage is robust due to the presence of internal nodes in all tetrahedra.

The final result is a 'periodic' series of 90 topologically invariant meshes with 13222 nodes, 86920 edges and 70533 tetrahedra for the right ventricle; see Figures 1 and 2.

5 Numerical examples

Numerical examples include convergence and stability tests for a problem with known analytical solution and personalized simulation of the flow in the right ventricle of a post-surgery patient with TGA. The finite element method (8),(9),(10) is implemented within the open source Ani3D software (www.sf.net/p/ani3d).

5.1 Convergence and stability tests

To set up convergence test with a synthetic solution, we let the reference domain Ω_0 to be the axisymmetric tube defined in cylindrical coordinates (r, y, ϕ) as:

$$\Omega_0 = \{(r, y, \phi) : -4 \leq y \leq 4, r^2 \leq e^{y/4+1}\}.$$

Neumann boundary condition is set at outflow boundary $\partial\Omega_0^{\text{out}} = \{(r, y, \phi) : r \leq e, y = 4\}$, and Dirichlet boundary condition is set on the remaining part of the boundary $\partial\Omega_0^{\text{ns}} = \partial\Omega_0 \setminus \partial\Omega_0^{\text{out}}$.

The time-dependent domain and the analytical solution $\{\mathbf{u}, p\}$ to (2),(3) are defined as follows:

$$\Omega(t) = \left\{ (r, y, \phi) : -4 \leq y \leq 4, r^2 \leq e^{y/4+1} \left(1 - \frac{1}{4}t \right) \right\}, \quad t \in [0, 3],$$

$$u_r = -\frac{2e^{-\frac{1}{4}(y+4)}r^3}{(4-t)^2}, \quad u_y = \frac{8}{4-t} - \frac{32e^{-\frac{1}{4}(y+4)}r^2}{(4-t)^2}, \quad u_\phi = 0,$$

$$p = 512\nu \frac{e^{-\frac{1}{4}(y+4)}}{(4-t)^2} - 8\frac{y}{(t-4)^2} + \tilde{p}(t),$$

where $\tilde{p}(t)$ depends only on t. Requirement $p = 0$ on $\partial\Omega^{\text{out}}(t)$ guarantees the unique pressure solution. To simplify calculation of the corresponding body force, we replace the term $2\nu \int_{\Omega(t)} \mathbf{D}(\mathbf{u}) : \mathbf{D}(\psi) \, dx$ in the Galerkin formulation (6) to $\nu \int_{\Omega(t)} \nabla\mathbf{u} : \nabla\psi \, dx$, and in the SUPG residual of (9) we use $-\nu\Delta\mathbf{u}_h^k$ for the viscous term.

The right-hand side $\mathbf{f} = (f_r, f_y, f_\phi)^T$ is obtained by substituting $\{\mathbf{u}, p\}$ to (2) for the given viscosity ν:

$$f_r = \nu \frac{e^{-\frac{1}{4}(y+4)}}{(4-t)^2} \left(16r + \frac{1}{8}r^3 \right) - 4\frac{e^{-\frac{1}{2}(y+4)}}{(t-4)^4}r^5, \quad f_y = 2\nu \frac{e^{-\frac{1}{4}(y+4)}}{(4-t)^2}r^2 - 128\frac{e^{-\frac{1}{2}(y+4)}}{(t-4)^4}r^4, \quad f_\phi = 0.$$

To check the convergence of the discrete velocity and pressures, we run a series of five simulations using a sequence of unstructured quasi-uniform tetrahedral meshes with mesh sizes $h_i = 2^{(1-i)/2}$, $i = 1, \ldots, 5$. The time integration interval is $[0, 0.2]$. According to numerical analysis [20] and numerical evidence, the scheme (8) without convective stabilizations delivers a two-times reduction of the energy error norm if the time step Δt decreases by two for each next mesh in this sequence. For $\mathbf{u}^k := \mathbf{u}(k\Delta t), p^k := p(k\Delta t)$, we define the error of the finite element solution $\{\mathbf{u}_h^k, p_h^k\}$ as $\{\mathbf{e}^k, e^k\} := \{\mathbf{u}^k - \mathbf{u}_h^k, p^k - p_h^k\}$. To check the effect of the SUPG stabilization on the convergence rate, we perform the experiment with and without the stabilization and report the computed error norms in Table 1. We see that the second order asymptotic convergence in the energy norm is observed both for the SUPG-stabilized and non-stabilized finite element schemes. The L^2-norm of the error demonstrates the third order convergence. In the numerical test we set viscosity to $\nu = \frac{\mu}{\rho} = \frac{4.2 \cdot 10^{-5}}{1.05 \cdot 10^{-3}} = 0.04$ (all units are SI except length which is in cm) which corresponds to that of blood. Note that the mesh size and the analytical solution are such that the scheme (8) can be used without stabilization.

Now we fix the third mesh withe $h = 0.5$ and perform $N = 300$ time steps with $\Delta t = 0.01$. This setting allows the numerical instability to develop for smaller viscosities. In Table 2 we present the velocity norm $\sqrt{\max_{0 \leqslant k \leqslant N} \frac{1}{2} \|\mathbf{u}_h^k\|^2 + \nu \sum_{k=1}^N \Delta t \|\nabla\mathbf{u}_h^k\|^2}$ for different values of viscosity. Here and further $\|\cdot\|$ denotes the L^2 norm. SUPG stabilization as well as the combined SUPG/Smagorinski stabilization with $M = 0.01$ produce stable solutions while the scheme (8) without convective stabilization blows up.

In order to study the impact of the scaling given by M in the definition (10) of the turbulent viscosity, we compute the finite element errors in the setup of the previous experiment. In Table 3 we show the L^2- and energy norm errors of the finite element solutions stabilized by SUPG or the combined SUPG/Smagorinski method. For relatively large viscosity, the addition of the turbulent viscosity increases the L^2- and the energy error norms by 100% and 50%, respectively. To the contrary, for small viscosity the partial addition of the turbulent viscosity reduces the error norms by 30% and 40%, respectively suggesting slightly oscillatory behavior of the pure SUPG-stabilized solution and warranting the inclusion of the weighted Smagorinski term.

Mesh size	1.0	$1.0/\sqrt{2}$	0.5	$0.5/\sqrt{2}$	0.25
Time step	0.04	0.02	0.01	0.005	0.0025
			no stabilization		
$\displaystyle\max_{0\leqslant k\leqslant N}\|\mathbf{e}^k\|$	0.067	0.038	0.0179	0.00761	0.00240
Error ratio		1.76	2.1	2.35	3.17
$\sqrt{\sum_{k=1}^{N}\Delta t\|\nabla\mathbf{e}^k\|^2}$	0.203	0.138	0.0820	0.0466	0.0212
Error ratio		1.47	1.68	1.76	2.20
			SUPG stabilization		
$\displaystyle\max_{0\leqslant k\leqslant N}\|\mathbf{e}^k\|$	0.066	0.037	0.0178	0.00760	0.00241
Error ratio		1.76	2.1	2.34	3.15
$\sqrt{\sum_{k=1}^{N}\Delta t\|\nabla\mathbf{e}^k\|^2}$	0.200	0.137	0.0813	0.0464	0.0212
Error ratio		1.46	1.69	1.75	2.19

Table 1: Finite element errors for the given analytical solution, $\nu = 0.04$.

Viscosity	0.04	0.004	0.0004
no stabilization	33.2	29.8	$3.62 \cdot 10^5$
SUPG stabilization	33.2	29.8	29.4
Combined SUPG/Smagorinski stabilization	33.2	29.7	29.3

Table 2: Numeric velocity norm $\sqrt{\max_{0\leq k\leq N}\frac{1}{2}\|\mathbf{u}_h^k\|^2 + \nu\sum_{k=1}^{N}\Delta t\|\nabla\mathbf{u}_h^k\|^2}$ over the period of $N = 300$ time steps of size $\Delta t = 0.01$ each for the triangulation with mesh size $h = 0.5$.

Viscosity	0.04	0.004	0.0004
		SUPG stabilization	
$\displaystyle\max_{1\leqslant k\leqslant N}\|\mathbf{e}^k\|$	0.106	0.198	0.562
$\sqrt{\sum_{k=1}^{N}\Delta t\|\nabla\mathbf{e}^k\|^2}$	1.00	2.089	7.00
$\sqrt{\max_{0\leqslant k\leqslant N}\frac{1}{2}\|\mathbf{e}^k\|^2 + \nu\sum_{k=1}^{N}\Delta t\|\nabla\mathbf{e}^k\|^2}$	0.214	0.192	0.422
		Combined SUPG/Smagorinski stabilization	
$\displaystyle\max_{0\leqslant k\leqslant N}\|\mathbf{e}^k\|$	0.198	0.328	0.396
$\sqrt{\sum_{k=1}^{N}\Delta t\|\nabla\mathbf{e}^k\|^2}$	1.47	3.08	4.24
$\sqrt{\max_{0\leqslant k\leqslant N}\frac{1}{2}\|\mathbf{e}^k\|^2 + \nu\sum_{k=1}^{N}\Delta t\|\nabla\mathbf{e}^k\|^2}$	0.326	0.303	0.293

Table 3: Finite element errors for the given analytical solution after $N = 300$ time steps with $\Delta t = 0.01$ and $h = 0.5$.

5.2 Blood flow in the right ventricle

We now apply the finite element methods with Smagorinski, SUPG and the combined stabilization to simulate blood flow in the right ventricle. For this purpose we use a sequence of

Smagorinski   0.01*Smagorinski+SUPG   SUPG

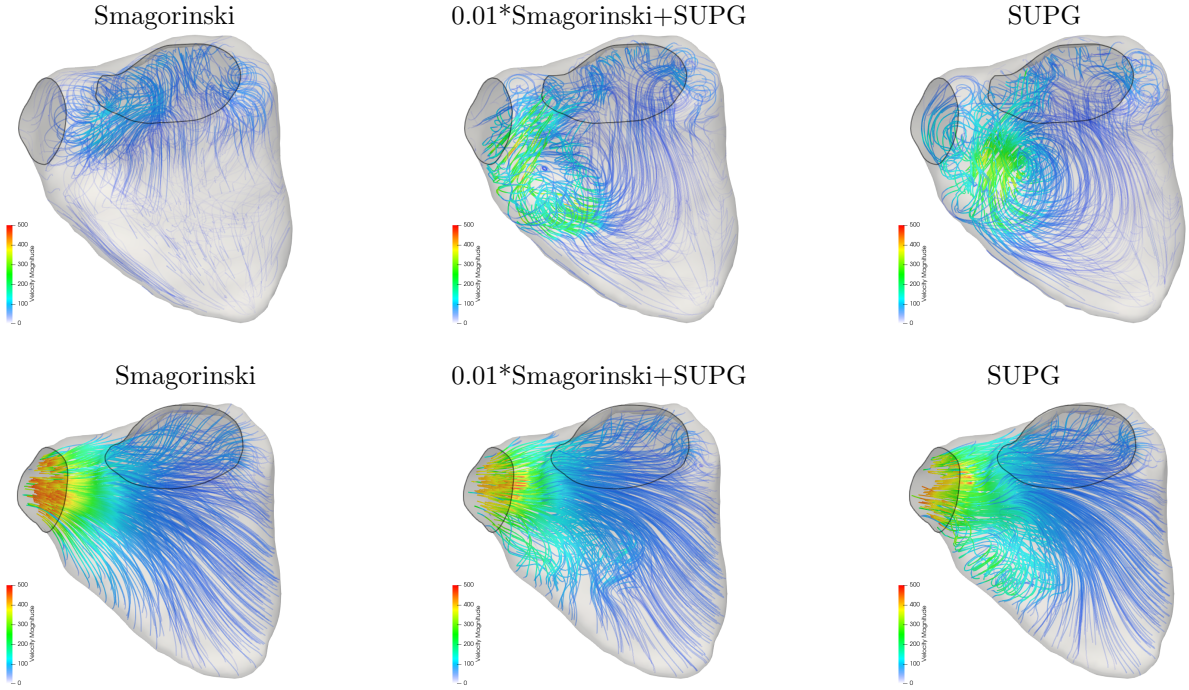Smagorinski   0.01*Smagorinski+SUPG   SUPG

Figure 3: Visualization of streamlines. The upper row is for the beginning of systole. The bottom row is the middle of systole.

topologically equivalent meshes described in section 4. The duration of the cardiac cycle is $0.87s$. Since we have 90 time frames, the time step Δt in (8) was set to be equal to $0.87/90s$. Boundary conditions were set according to (3)–(5), with $\boldsymbol{\xi}_t$ computed on the boundary by the first order finite-difference stencil directly from the displacement at the nodes and edges. Numerical simulations start from the flow at rest as the initial condition and run over two cardiac cycles. All results below are shown for the second cycle.

From the numerical results for the problem with the synthetic solution, we see that both SUPG and the combination of weighted Smagorinski and SUPG stabilizations deliver convergent solutions with the pure SUPG being closer to the borderline of numerically stable simulations. Pure Smagorinski stabilization is a well-known turbulent LES model that, however, may over-dissipate. We are now interested in how these properties affect the patient specific CFD visualization of blood flow in the right ventricle.

In Figure 3 we show streamlines that are computed as particle tracers in blood flow at a given time. The flow is visualized at the beginning and middle of the systole. We see that among three models the pure SUPG and the weighted Smagorinski combined with SUPG stabilizations capture the large recirculation zone in front of the pulmonary valve. This observation is confirmed by the picture of the Q-criterion levels in Figure 4: the zones where $Q = \frac{1}{2}(\text{tr}^2(\nabla \mathbf{u}) - \text{tr}([\nabla \mathbf{u}]^2))$ is positive are identified as vortexes in incompressible flow [21]. Again, pure SUPG model suggests most distinct vortical regions and using standard Smagorinski model suppresses vorticity. The combination of SUPG and weighted Smagorinski model is doing a reasonably good job in identifying the vortical region. All these observations confirm that standard Smagorinski stabilization is over-diffusive for numerical simulations of the flow in the heart.

Finally, we calculate the wall shear stress (WSS) predicted by the three models and show them in Figure 5. The WSS is likely underestimated by the Smagorinski model, which is a consequence of its overall poor accuracy. Less diffusive methods predict larger wall shear stress.

Smagorinski   0.01*Smagorinski+SUPG   SUPG

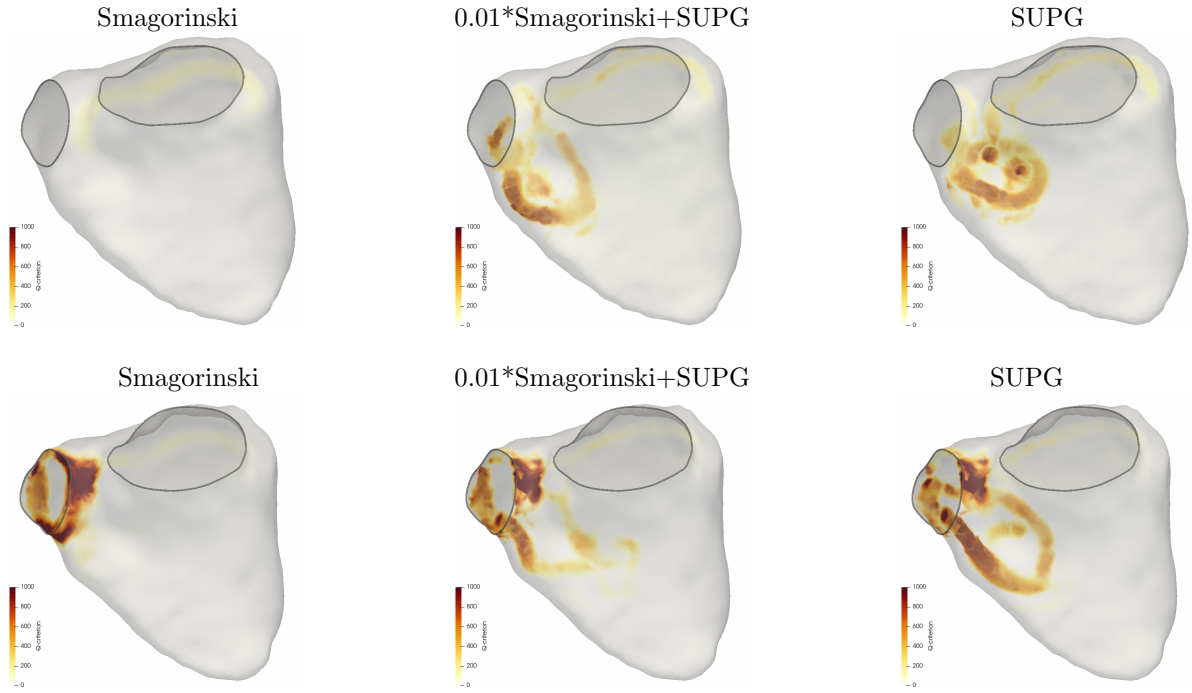Smagorinski   0.01*Smagorinski+SUPG   SUPG

Figure 4: Visualization of vortex regions according to the Q-criterion. The upper row is for the beginning of systole. The bottom row is the middle of systole.
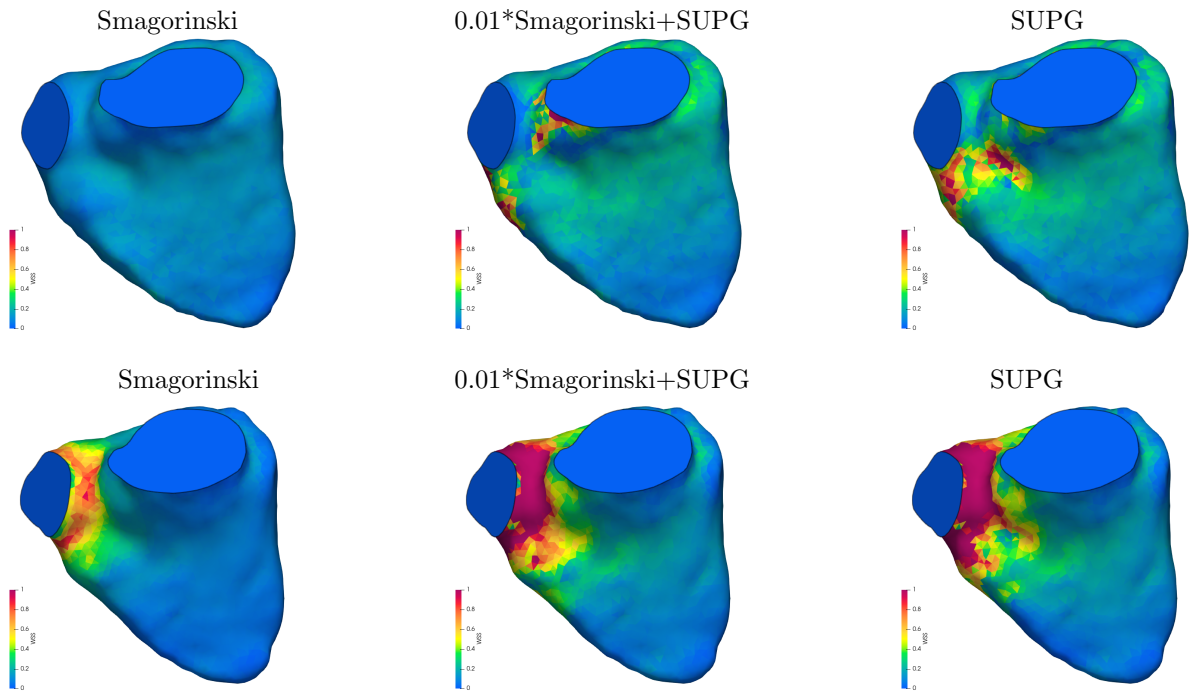
Smagorinski   0.01*Smagorinski+SUPG   SUPG

Smagorinski   0.01*Smagorinski+SUPG   SUPG

Figure 5: Visualization of the wall shear stress. The upper row is for the beginning of systole. The bottom row is the middle of systole.

We conclude that both SUPG and the combination of SUPG and weighted Smagorinski stabilizations of the finite element method can be used in a predictive CFD visualization of cardiac flows with SUPG being the least diffusive, while SUPG/weighted Smagorinski adds extra stability.

References

[1] J. Tu, K. Inthavong, K. K. L. Wong, Computational Hemodynamics–Theory, Modelling and Applications, Springer, 2015.

[2] A. Quarteroni, L. Dede, A. Manzoni, C. Vergara, Mathematical modelling of the human cardiovascular system: data, numerical approximation, clinical applications, Vol. 33, Cambridge University Press, 2019.

[3] Y. Vassilevski, M. Olshanskii, S. Simakov, A. Kolobov, A. Danilov, Personalized Computational Hemodynamics: Models, Methods, and Applications for Vascular Surgery and Antitumor Therapy, Academic Press, 2020.

[4] D. Rodriguez Muñoz, M. Markl, J. L. Moya Mur, A. Barker, C. Fernández-Golfín, P. Lancellotti, J. L. Zamorano Gómez, Intracardiac flow visualization: current status and future directions, European Heart Journal–Cardiovascular Imaging 14 (11) (2013) 1029–1038.

[5] P. G. Ciarlet, The finite element method for elliptic problems, Vol. 40, Siam, 2002.

[6] P. Hood, C. Taylor, Navier-Stokes equations using mixed interpolation, University of Alabama in Huntsville Press, Huntsville.

[7] M. Bercovier, O. Pironneau, Error estimates for finite element method solution of the Stokes problem in the primitive variables, Numerische Mathematik 33 (2) (1979) 211–224.

[8] G. Querzoli, S. Fortini, A. Cenedese, Effect of the prosthetic mitral valve on vortex dynamics and turbulence of the left ventricular flow, Physics of Fluids 22 (4) (2010) 041901.

[9] A. Falahatpisheh, A. Kheradvar, High-speed particle image velocimetry to assess cardiac fluid dynamics in vitro: from performance to validation, European Journal of Mechanics-B/Fluids 35 (2012) 2–8.

[10] C. Chnafa, S. Mendez, F. Nicoud, Image-based large-eddy simulation in a realistic left heart, Computers & Fluids 94 (2014) 173–187.

[11] J. Smagorinsky, General circulation experiments with the primitive equations: I. The basic experiment, Monthly weather review 91 (3) (1963) 99–164.

[12] A. N. Brooks, T. J. 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 (1-3) (1982) 199–259.

[13] T. J. Hughes, G. Scovazzi, L. P. Franca, Multiscale and stabilized methods, Encyclopedia of Computational Mechanics, Second Edition (2018) 1–64.

[14] Y. Zang, R. L. Street, J. R. Koseff, A dynamic mixed subgrid-scale model and its application to turbulent recirculating flows, Physics of Fluids A: Fluid Dynamics 5 (12) (1993) 3186–3196.

[15] P. A. Yushkevich, J. Piven, H. C. Hazlett, R. G. Smith, S. Ho, J. C. Gee, G. Gerig, User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability, NeuroImage 31 (3) (2006) 1116–1128.

[16] A. Danilov, A. Lozovskiy, M. Olshanskii, Y. Vassilevski, A finite element method for the Navier-Stokes equations in moving domain with application to hemodynamics of the left ventricle, Russian Journal of Numerical Analysis and Mathematical Modelling 32 (4) (2017) 225–236.

[17] L. Rineau, M. Yvinec, A generic software design for Delaunay refinement meshing, Computational Geometry 38 (1-2) (2007) 100–110.

[18] K. Lipnikov, Y. Vassilevski, A. Danilov, et al., Advanced Numerical Instruments 3D, https://sourceforge.net/projects/ani3d.

[19] J. Escobar, E. Rodríguez, R. Montenegro, G. Montero, J. González-Yuste, Simultaneous untangling and smoothing of tetrahedral meshes, Computer Methods in Applied Mechanics and Engineering 192 (25) (2003) 2775–2787.

[20] A. Lozovskiy, M. A. Olshanskii, Y. V. Vassilevski, A quasi-lagrangian finite element method for the Navier–Stokes equations in a time-dependent domain, Computer Methods in Applied Mechanics and Engineering 333 (2018) 55–73.

[21] Y. Dubief, F. Delcayre, On coherent-vortex identification in turbulence, Journal of turbulence 1 (1) (2000) 011–011.