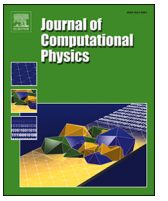
Journal of Computational Physics 540 (2025) 114267

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp



A fully implicit method for numerical simulation of blood flows with incompressible Navier-Stokes equations with slip and penetration boundary conditions

Xiangdong Zhang [a,b], Li Luo [a,*], Ye Li [b,*], Xiao-Chuan Cai [c]

[a] Department of Mathematics, University of Macau, Macao SAR, 999078, China
[b] Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen, 518055, China
[c] Department of Computer Science, University of Colorado Boulder, Boulder, CO 80309, USA

ARTICLE INFO

Keywords:
Blood flows
Incompressible Navier-Stokes equations
Slip and penetration boundary conditions
Finite element
Unstructured mesh
Parallel computing

ABSTRACT

We consider the numerical simulation of blood flows in the human artery using the incompressible Navier-Stokes equations with slip and penetration boundary conditions on part of the arterial wall. Most existing approaches assume the no-slip and no-penetration boundary conditions for the entire wall, but such assumptions do not hold under certain pathological conditions. For these problems, we develop a fully implicit domain decomposition method for the incompressible Navier-Stokes equations with the proposed more realistic boundary conditions. A stabilized finite element method, with a high order boundary discretization, is used for the spatial discretization and a backward Euler method is used for the temporal discretization. The resulting large, sparse, and highly ill-conditioned nonlinear algebraic equations are scaled and solved by a Newton-Krylov method that is preconditioned by a parallel one-level overlapping Schwarz method. Benchmark simulations for flows in 2D and 3D channels demonstrate good agreement with previous studies. An in-depth study is provided for the influence of the slip and penetration boundary conditions on the flow behavior and the parallel solver. The proposed method performs well for blood flows in a realistic patient-specific cerebral artery with some level of stenosis, and the parallel scalability is studied using up to 4096 processor cores.

1. Introduction

Hemodynamic simulations have found valuable applications in various clinical scenarios, including risk assessment of cardiovascular diseases [1], disease diagnosis [2,3], and noninvasive measurement of hemodynamic indexes [4,5]. Typically, these simulations involve modeling the blood flow by the incompressible Navier-Stokes equations with a zero-velocity Dirichlet boundary condition. This boundary condition assumes that the blood flow neither permeates nor slips on the vessel walls. However, realistic arterial walls consist of endothelial cells with permeable membranes and endothelial barriers between them [6]. In certain pathological conditions like inflammation or cancer, the barrier function may be disrupted, resulting in significantly increased permeability [7]. Numerous studies have also demonstrated the existence of slip velocities between blood flows and arterial walls, both in micro arteries [8–12] and large arteries [13,14]. Slip velocities are commonly observed on rough, structured, and soft biological surfaces at fluid-solid interfaces [15,16]. Additionally, blood is a multiphase flow comprising blood cells, platelets, and plasma. Experimental studies have observed significant slip velocities of red blood cells near arterial walls [8–12], and the plasma near the cells moves due to blood

* Corresponding authors.
E-mail addresses: xiangdong_zhang@foxmail.com (X. Zhang), liluo@um.edu.mo (L. Luo), ye.li@siat.ac.cn (Y. Li), cai@cs.colorado.edu (X. Cai).

https://doi.org/10.1016/j.jcp.2025.114267
Received 6 December 2024; Received in revised form 14 July 2025; Accepted 31 July 2025
Available online 5 August 2025
0021-9991/© 2025 Elsevier Inc. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

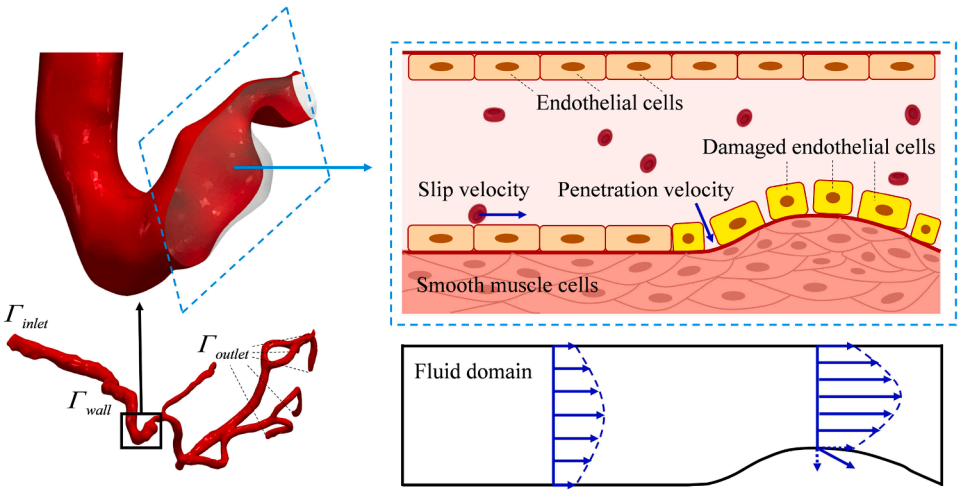


Fig. 1. A schematic representation of the slip and penetration velocities on arterial walls.

cell-induced velocity [17,18]. Therefore, incorporating slip and penetration boundary conditions (SPBC) in hemodynamic simulation is practically necessary, which potentially offers crucial insights into the intricate dynamics of blood flow on the arterial walls. This deeper understanding can shed light on the development and progression of various pathologies, such as aneurysm and dissection, and aid in personalized surgical planning. Fig. 1 illustrates a schematic representation of the SPBC.

Existing studies of nonzero velocities on arterial walls focus on treating the arterial wall as a porous medium with certain thickness. For example, Koshiba et al. [19] developed a multiphysics model to examine the accumulation and distribution of low-density lipoprotein (LDL) on arterial walls. They observed that the properties of the porous medium significantly influence the transportation and accumulation of LDL. Thon et al. [20] built upon this concept and developed a similar model for non-Newtonian fluids to investigate the progression of atherosclerosis. Cheema et al. [21] constructed an ideal channel model for the aorta and conducted simulations comparing the results obtained by assuming porous medium on the walls to those obtained using classical no-slip boundary conditions. Their findings indicated that the average wall shear stress was higher when using the porous medium boundary approach. In a three-dimensional computational model of arteries with an aneurysm, Jiang et al. [22] performed similar comparisons and observed lower wall shear stress in the aneurysm region when treating the walls as porous medium. These studies suggest that boundary velocity on arterial walls has a significant impact on the simulation results.

However, treating the arterial wall as a porous medium has several limitations, which make it hard to develop personalized or patient-specific models. These limitations arise due to the challenges associated with measuring the arterial wall thickness using conventional medical imaging techniques and determining the boundary locations for the porous medium model, given the intricate components of artery walls, such as the endothelial layer and smooth muscle cells. In [23], Lucotte et al. directly visualized the location of water permeability barriers within the arterial wall by using anti-Stokes Raman scattering microscopy combined with fluorescence and deuteroxide perfusion. They found that the water barrier of the artery wall is located at the endothelial basolateral membrane, which is merely a single layer of cells away from the fluid domain. This experimental discovery suggests that the thickness of the porous media model may be considerably thinner than previously assumed. Modeling and conducting numerical methods for the porous medium domain become rather difficult due to the extremely small thickness of the arterial wall.

To overcome the limitations associated with the porous medium assumption, a viable solution is to directly implement SPBC for the fluids and solve the incompressible Navier-Stokes equations solely in the fluid domain. John [24] adopted this approach to study incompressible flows with SPBC in simple fluid domains using structured meshes. Thereafter, several numerical studies have investigated the effects of Navier slip boundary conditions in both two-dimensional (2D) [25,26] and three-dimensional (3D) [27,28] domains. The existence of weak solutions has also been discussed in theoretical studies [29,30].

To our knowledge, there is a lack of research that specifically addresses the utilization of SPBC in the area of hemodynamic simulation. Implementing these boundary conditions brings considerable complexities, particularly when dealing with the intricate geometry of realistic arteries. The complex nature of the arterial geometry introduces difficulties in accurately determining the normal direction required for these boundary conditions, especially in regions with large curvature. Insufficient resolution of discretized mesh faces for approximating the curved surface can lead to substantial computational errors. High-order numerical methods have the potential to enhance computational accuracy. For instance, Costa et al. [28] proposed an eighth-order finite volume scheme capable of accurately capturing slip velocities on complex curved surfaces. Some other high-order approaches, such as the stabilized Fourier spectral methods [31,32], have demonstrated improved accuracy in solving nonlinear equations. Despite these advantages, the application of high-order methods to realistic arterial geometries remains computationally demanding due to the large number of degrees of freedom required. Furthermore, when considering realistic values for slip and penetration coefficients in a diseased artery, the condition number of the system matrix becomes notably high, posing great challenges to the robustness and efficiency of numerical solution algorithms. Besides, the combined effect using the slip and penetration boundary conditions with the resistance

boundary conditions commonly applied at the outlets remains unclear. This is because the impedance offered by the downstream vasculature has a direct impact on the amount of penetrating blood through the artery wall. Further investigation is necessary to fully understand the interplay between these boundary conditions and their impact on hemodynamic simulations.

In this paper, we study the application of SPBC to blood flows in patient-specific arteries modeled by incompressible Navier-Stokes equations. The implementation is based on a fully-implicit stabilized finite element method on 3D unstructured meshes. We reformulate the SPBC so that they can be naturally incorporated into the weak form of the governing equations. To mitigate errors introduced during boundary integration, finite element basis functions are utilized to reconstruct the normal directions on the element faces. We provide implementation details of the SPBC and the stabilization scheme. Achieving accurate and efficient simulation of 3D blood flow becomes a formidable task that exceeds the computational capacity of a sequential computer. To tackle this demand, we develop a robust and scalable algorithm based on the Newton-Krylov-Schwarz framework [33,34] for solving the discretized system on a cluster of parallel computers. Notably, the proposed one-level additive Schwarz preconditioner with a point-block structure proves to be highly effective in reducing the condition number resulting from the integration of SPBC. We conduct extensive numerical experiments to demonstrate the effectiveness of the proposed boundary conditions and algorithms in capturing diverse slip and penetration behaviors. Specifically, we provide a practical approach to estimate the slip coefficient for the case with a patient-specific artery, and study the combined effect of penetration coefficient and outlet resistance on the blood leakage rate. The parallel efficiency of the proposed algorithm is also reported.

The structure of this paper is as follows. Section 2 provides an overview of the modeling of blood flows. Sections 3.1 and 3.2 present the fully implicit finite element discretization and the implementation details. In Section 3.3, we introduce the parallel domain decomposition algorithm for solving the nonlinear system. Section 4 presents the results of numerical experiments conducted on several benchmark tests and a real artery. This section also examines the robustness and scalability of the proposed algorithm. Finally, we provide concluding remarks in Section 5.

2. Modeling of the blood flow in human artery with penetrable wall

Let Ω be the artery domain, as shown in Fig. 1. The blood flow is modeled by the unsteady incompressible Navier-Stokes equations

$$\begin{cases} \rho\dfrac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u}\cdot\nabla)\mathbf{u} - \mu\nabla^2\mathbf{u} + \nabla p = \mathbf{0} \quad \text{in } \Omega, & (1a) \\ \nabla\cdot\mathbf{u} = 0 \quad \text{in } \Omega. & (1b) \end{cases}$$

Here, $\mathbf{u}$ is the velocity, p is the blood pressure, ρ is the density, and μ is the dynamic viscosity. The boundary of the artery $\partial\Omega$ is partitioned into three parts: Γ_{inlet} is the inlet, Γ_{wall} is the artery wall, Γ^j_{outlet} is the j^{th} outlet, and N_{outlet} is the number of outlets. The boundary conditions applied to these three parts have a significant influence on the simulation results. At the inlet Γ_{inlet}, we impose a parabolic velocity profile [35]. We apply a resistant boundary condition on each outlet Γ^j_{outlet} to model the truncated downstream vasculature,

$$p = R_j Q_j = R_j \int_{\Gamma^j_{outlet}} \mathbf{u}\cdot\mathbf{n} d\Gamma \quad \text{on } \Gamma^j_{outlet}, \quad j = 1, \ldots, N_{outlet} \tag{2}$$

where R_j denotes the resistance, Q_j denotes the volumetric flow rate through the outlet Γ^j_{outlet}, and $\mathbf{n}$ is the unit outward normal vector of the boundary.

On the artery wall, we assume the following Robin-type slip and penetration boundary conditions [24], including the penetration boundary condition:

$$C_n\mathbf{u}\cdot\mathbf{n} + \left(\mu\frac{\partial\mathbf{u}}{\partial\mathbf{n}} - p\mathbf{n}\right)\cdot\mathbf{n} = 0, \quad \text{on } \Gamma_{wall}, \tag{3a}$$

and the slip boundary condition:

$$C_\tau\mathbf{u}\cdot\boldsymbol{\tau}_k + \left(\mu\frac{\partial\mathbf{u}}{\partial\mathbf{n}} - p\mathbf{n}\right)\cdot\boldsymbol{\tau}_k = 0, \quad \text{on } \Gamma_{wall}, \ k \in (1,2), \tag{3b}$$

where $\boldsymbol{\tau}_1$ and $\boldsymbol{\tau}_2$ are unit tangential vectors of the boundary. Eq. (3a) assumes the penetration velocity $\mathbf{u}\cdot\mathbf{n}$ is proportional to the fluid stress $\mu\frac{\partial\mathbf{u}}{\partial\mathbf{n}} - p\mathbf{n}$ along $\mathbf{n}$. When C_n tends towards infinity, it enforces an impermeable boundary condition, whereas as C_n approaches zero, it establishes a free outflow boundary condition. On the other hand, Eq. (3b) elucidates the proportional relationship between the tangential fluid stress and the slip velocity $\mathbf{u}\cdot\boldsymbol{\tau}_k$. When C_τ goes to infinity, this becomes a no-slip boundary condition, while when C_τ approaches zero, it results in a free slip boundary condition.

In this work, we reformulate SPBC to eliminate the need for $\boldsymbol{\tau}_1$, $\boldsymbol{\tau}_2$ and address the fluid stress $\mu\frac{\partial\mathbf{u}}{\partial\mathbf{n}} - p\mathbf{n}$ explicitly. Specifically, we change Eq. (3b) to:

$$C_\tau\mathbf{u} - C_\tau(\mathbf{u}\cdot\mathbf{n})\mathbf{n} + \left(\mu\frac{\partial\mathbf{u}}{\partial\mathbf{n}} - p\mathbf{n} - \left(\left(\mu\frac{\partial\mathbf{u}}{\partial\mathbf{n}} - p\mathbf{n}\right)\cdot\mathbf{n}\right)\mathbf{n}\right) = \mathbf{0}, \quad \text{on } \Gamma_{wall}. \tag{4}$$

By substituting Eq. (3a) into Eq. (4), we obtain

$$\mu\frac{\partial\mathbf{u}}{\partial\mathbf{n}} - p\mathbf{n} = (C_\tau - C_n)(\mathbf{u}\cdot\mathbf{n})\mathbf{n} - C_\tau\mathbf{u}, \quad \text{on } \Gamma_{wall}. \tag{5}$$

This reformulation allows for easy incorporation of the SPBC into the weak form.

Let $\mathbf{U} = \{\mathbf{u} \in \left[H^1(\Omega)\right]^3; \mathbf{u}|_{\Gamma_{inlet}} = \mathbf{u}_{in}(t)\}$, $\mathbf{V} = \{\mathbf{v} \in \left[H^1(\Omega)\right]^3; \mathbf{v}|_{\Gamma_{inlet}} = \mathbf{0}\}$, $Q = \left\{q \in L^2(\Omega)\right\}$, the weak form of (1) takes the form: Find $(\mathbf{u}, p) \in (\mathbf{U}, Q)$ such that $\forall\ (\mathbf{v}, q) \in (\mathbf{V}, Q)$,

$$\begin{aligned}
&\int_\Omega \rho\Big(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u}\Big)\cdot\mathbf{v}d\Omega + \int_\Omega \mu\nabla\mathbf{u}:\nabla\mathbf{v}d\Omega - \int_\Omega p\nabla\cdot\mathbf{v}d\Omega \\
&- \left(C_\tau - C_n\right)\int_{\Gamma_{wall}} (\mathbf{u}\cdot\mathbf{n})\mathbf{n}\cdot\mathbf{v}d\Gamma + C_\tau\int_{\Gamma_{wall}} \mathbf{u}\cdot\mathbf{v}d\Gamma \qquad (6a)\\
&- \sum_{j=1}^{N_{outlet}}\int_{\Gamma^j_{outlet}}\left(\mu\frac{\partial\mathbf{u}}{\partial\mathbf{n}} + R_j\int_{\Gamma^j_{outlet}}\mathbf{u}\cdot\mathbf{n}d\Gamma\mathbf{n}\right)\cdot\mathbf{v}d\Gamma = 0,\\
&\int_\Omega q\nabla\cdot\mathbf{u}d\Omega = 0. \qquad (6b)
\end{aligned}$$

3. Numerical methods

3.1. Spatial and temporal discretization

To discretize Eqs. (6a) and (6b), we consider a fully implicit backward Euler scheme in time and a P1-P1 pressure stabilized finite element method in space. Let E be a tetrahedron element and $\Omega_h = \{E\}$ be a conforming tetrahedral mesh of Ω with elements E. We employ a pressure projection method to stabilize the coupled scheme arising from the fully implicit discretization [36]. To describe the method, we first introduce some notations. Consider a piecewise constant space

$$R_0 = \{r \in L^2(\Omega) \mid r|_E \in P_0(E),\ \forall\ E \in \Omega_h\},$$

where $P_0(E)$ is a constant space on E. Given a function $q \in L^2(\Omega)$, we define a projection operator $\Pi : L^2(\Omega) \to R_0$ such that

$$\int_E (q - \Pi q) r d E = 0, \quad \forall\ r \in R_0, \quad \forall\ E \in \Omega_h. \qquad (7)$$

From Eq. (7), it can be derived that $\forall\ q_h \in Q_h$,

$$\Pi q_h|_E = \frac{1}{V(E)}\int_E q_h d E, \qquad (8)$$

where $V(E)$ is the volume of a tetrahedron element E. The projection $\Pi q_h|_E$ gives an element average of q_h on E.

We next introduce three piecewise linear continuous finite element spaces $\mathbf{U}_h \subset \mathbf{U}(\Omega_h)$, $\mathbf{V}_h \subset \mathbf{V}(\Omega_h)$, $Q_h \subset Q(\Omega_h)$. Let $\Gamma^{h,j}_{outlet}$ and Γ^h_{wall} be the boundary of Ω_h with respect to the j^{th} outlet and the wall, respectively. We denote by $(\cdot,\cdot)$ the volume $L^2(\Omega_h)$-inner product, $\langle\cdot,\cdot\rangle_{\Gamma^{h,j}_{outlet}}$ the surface $L^2(\Gamma^{h,j}_{outlet})$ -inner product, and by $\langle\cdot,\cdot\rangle_{\Gamma^h_{wall}}$ the surface $L^2(\Gamma^h_{wall})$ -inner product. Using the element-wise projector Π and a time step length Δt, we denote the finite element solution at the n^{th} time step as $\mathbf{u}^n_h$ and p^n_h for $n = 0, 1, \ldots$. We address the discretized scheme as follows: Find $(\mathbf{u}^{n+1}_h, p^{n+1}_h) \in (\mathbf{U}_h, Q_h)$ such that $\forall\ (\mathbf{v}_h, q_h) \in (\mathbf{V}_h, Q_h)$,

$$\begin{aligned}
&\frac{\rho}{\Delta t}\left(\mathbf{u}^{n+1}_h, \mathbf{v}_h\right) - \frac{\rho}{\Delta t}\left(\mathbf{u}^n_h, \mathbf{v}_h\right) + \rho\left(\left(\mathbf{u}^{n+1}_h\cdot\nabla\right)\mathbf{u}^{n+1}_h, \mathbf{v}_h\right) + \mu\left(\nabla\mathbf{u}^{n+1}_h, \nabla\mathbf{v}_h\right) - \left(p^{n+1}_h, \nabla\cdot\mathbf{v}_h\right)\\
&- \left(C_\tau - C_n\right)\left\langle\left(\mathbf{u}^{n+1}_h\cdot\mathbf{n}_h\right)\mathbf{n}_h, \mathbf{v}_h\right\rangle_{\Gamma^h_{wall}} + C_\tau\left\langle\mathbf{u}^{n+1}_h, \mathbf{v}_h\right\rangle_{\Gamma^h_{wall}} \qquad (9a)\\
&- \sum_{j=1}^{N_{outlet}}\left\langle\mu\left(\mathbf{n}_h\cdot\nabla\right)\mathbf{u}^{n+1}_h + R_j\left\langle\mathbf{u}^{n+1}_h, \mathbf{n}_h\right\rangle_{\Gamma^{h,j}_{outlet}}\mathbf{n}_h, \mathbf{v}_h\right\rangle_{\Gamma^{h,j}_{outlet}} = 0,\\
&\left(\nabla\cdot\mathbf{u}^{n+1}_h, q_h\right) - C_\Pi\left(p^{n+1}_h - \Pi p^{n+1}_h, q_h - \Pi q_h\right) = 0. \qquad (9b)
\end{aligned}$$

The second term on the left-hand side of Eq. (9b) is referred to as the pressure stabilization term, where C_Π is a user-defined coefficient. It was shown in [36–39] that using this term is able to stabilize the lowest equal order pair $(\mathbf{U}_h, Q_h)$.

We denote the vector of unknowns in Eqs. (9a) and (9b) as $\mathbf{x}^{n+1}$. The fully implicit discretization Eqs. (9a) and (9b) results in a nonlinear algebraic system

$$F(\mathbf{x}^{n+1}) = 0 \qquad (10)$$

to be solved at each time step, where F is the nonlinear residual vector.

3.2. Implementation details of the discretized scheme

3.2.1. The pressure stabilization term

In the weak form Eq. (9b), the contribution of the pressure stabilization term to F is computed element by element. Since $p^{n+1}_h \in Q_h$ and $\Pi q_h \in R_0$, according to Eq. (7), we have

$$\int_E \left(p^{n+1}_h - \Pi p^{n+1}_h\right)\Pi q_h d E = 0. \qquad (11)$$

Therefore, the pressure stabilization term becomes

$$-C_\Pi \int_E q_h\left(p_h^{n+1} - \Pi p_h^{n+1}\right) dE. \tag{12}$$

Let $\{\phi_j\}_{j=1}^4$ be basis functions on E, and $\{p_j^{n+1}\}_{j=1}^4$ the nodal values of the pressure, then $p_h^{n+1}|_E = \sum_{j=1}^4 p_j^{n+1}\phi_j$ and $\Pi p_h^{n+1}|_E = \sum_{j=1}^4 p_j^{n+1}/4$. The contribution of the stabilization term to F corresponding to the i^{th} vertex on E is

$$-C_\Pi \int_E q_h\left(p_h^{n+1} - \Pi p_h^{n+1}\right) dE = -C_\Pi \int_E \phi_i \sum_{j=1}^4 \left(\phi_j - \frac{1}{4}\right) p_j^{n+1} dE. \tag{13}$$

3.2.2. The slip and penetration boundary terms

To introduce the slip and penetration boundary terms, we denote $\Gamma_{wall}^E = E \cap \Gamma_{wall}^h$ as the face of a tetrahedron element E on the arterial wall, the boundary integral terms from SPBC takes the form

$$-\left(C_\tau - C_n\right) \int_{\Gamma_{wall}^E} \left(\mathbf{u}_h^{n+1} \cdot \mathbf{n}_h\right)\left(\mathbf{v}_h \cdot \mathbf{n}_h\right) d\Gamma + C_\tau \int_{\Gamma_{wall}^E} \mathbf{u}_h^{n+1} \cdot \mathbf{v}_h d\Gamma. \tag{14}$$

We define a set of vector-value shape functions $\{\mathbf{v}_i\}_{i=1}^{12}$ on E with each $\mathbf{v}_i \in [H^1(E)]^3$, and they are ordered such that the first 4 functions do not vanish only in the first component, the second 4 functions in the second component and the last 4 functions in the third component:

$$\mathbf{v}_i = \begin{cases} (v_{x,i}, 0, 0)^T, & 1 \le i \le 4, \\ (0, v_{y,i}, 0)^T, & 5 \le i \le 8, \\ (0, 0, v_{z,i})^T, & 9 \le i \le 12. \end{cases} \tag{15}$$

Let $\{u_j^{n+1}\}_{j=1}^{12}$ be the nodal values of the velocity of E, by replacing $\mathbf{v}_h$ with $\mathbf{v}_i$ and using Eq. (15) and the following notations

$$\mathbf{u}_h^{n+1}|_E = \sum_{j=1}^{12} u_j^{n+1}\mathbf{v}_j, \quad \mathbf{n}_h = \left(n_x, n_y, n_z\right)^T, \tag{16}$$

we obtain

$$\begin{aligned} &-\left(C_\tau - C_n\right) \int_{\Gamma_{wall}^E} \left(\mathbf{u}_h^{n+1} \cdot \mathbf{n}_h\right)\left(\mathbf{v}_i \cdot \mathbf{n}_h\right) d\Gamma \\ &= \begin{cases} -\left(C_\tau - C_n\right) \sum_{j=1}^{12} u_j^{n+1} \int_{\Gamma_{wall}^E} \left(v_{x,j}n_x + v_{y,j}n_y + v_{z,j}n_z\right) v_{x,i}n_x d\Gamma, & 1 \le i \le 4, \\ -\left(C_\tau - C_n\right) \sum_{j=1}^{12} u_j^{n+1} \int_{\Gamma_{wall}^E} \left(v_{x,j}n_x + v_{y,j}n_y + v_{z,j}n_z\right) v_{y,i}n_y d\Gamma, & 5 \le i \le 8, \\ -\left(C_\tau - C_n\right) \sum_{j=1}^{12} u_j^{n+1} \int_{\Gamma_{wall}^E} \left(v_{x,j}n_x + v_{y,j}n_y + v_{z,j}n_z\right) v_{z,i}n_z d\Gamma, & 9 \le i \le 12, \end{cases} \end{aligned} \tag{17}$$

and

$$C_\tau \int_{\Gamma_{wall}^E} \mathbf{u}_h^{n+1} \cdot \mathbf{v}_i d\Gamma = \begin{cases} C_\tau \sum_{j=1}^{4} u_j^{n+1} \int_{\Gamma_{wall}^E} v_{x,j}v_{x,i} d\Gamma, & 1 \le i \le 4, \\ C_\tau \sum_{j=5}^{8} u_j^{n+1} \int_{\Gamma_{wall}^E} v_{y,j}v_{y,i} d\Gamma, & 5 \le i \le 8, \\ C_\tau \sum_{j=9}^{12} u_j^{n+1} \int_{\Gamma_{wall}^E} v_{z,j}v_{z,i} d\Gamma, & 9 \le i \le 12. \end{cases} \tag{18}$$

We remark that the normal vector $\mathbf{n}_h$ in the boundary integral term Eq. (17) is a constant for each face element, and the value jumps from element to element on a curved surface. This may introduce substantial numerical errors in the boundary integral terms, thereby undermining the desired slip and penetration effects. The situation is even worse when C_τ or C_n is large, as they are coefficients of the boundary integration terms that amplify the errors. To improve the accuracy of the boundary integral terms, we apply finite element interpolation of vertex normal vectors to calculate the normal vectors at various quadrature points, as illustrated in Fig. 2. Consider a face element Γ_{wall}^E that is a triangle consisting of three vertices v_j, $j = 1, 2, 3$, we denote by s_j the number of face elements sharing the vertex v_j. The normal vector $\mathbf{n}_{v_j}$ at the vertex v_j is calculated by averaging the normal vectors $\mathbf{n}_{f_i}$ of all face elements connected to the vertex v_j:

$$\mathbf{n}_{v_j} = \sum_{i=1}^{s_j} \mathbf{n}_{f_i} \Big/ \left\| \sum_{i=1}^{s_j} \mathbf{n}_{f_i} \right\|, \tag{19}$$

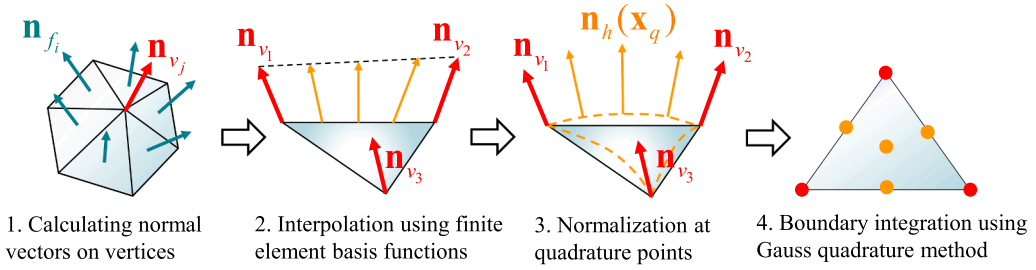


Fig. 2. Construction of the normal vectors at quadrature points on curved surfaces.

where $\|\cdot\|$ is the l^2 norm of a vector. The normal vector at a quadrature point $\mathbf{x}_q$ is then computed by

$$\mathbf{n}_h(\mathbf{x}_q) := \sum_{j=1}^{3} \varphi_j(\mathbf{x}_q)\mathbf{n}_{v_j} \Big/ \left\| \sum_{j=1}^{3} \varphi_j(\mathbf{x}_q)\mathbf{n}_{v_j} \right\|, \tag{20}$$

where $\varphi_j(\mathbf{x}_q)$ is the piecewise linear shape function associated with the j^{th} vertex on Γ^E_{wall}. We refer to $\mathbf{n}_h$ the revised normal vector, distinct from the element face normal vector $\mathbf{n}_f$. Once the interpolated normal vectors at the quadrature points are determined, the boundary integral can be computed using Gaussian quadrature. Our numerical results demonstrate that this treatment plays a crucial role in mitigating integration errors.

3.3. Parallel Newton-Krylov-Schwarz method

In this study, we use the inexact Newton-Krylov-Schwarz method [33,34] to solve the system of nonlinear Eq. (10). The inexact Newton method for Eq. (10) can be described as follows: Given the initial guess $\mathrm{x}_0^{n+1} = \mathrm{x}^n$ and $k = 0$, update the solutions iteratively by

$$\mathrm{x}_{k+1}^{n+1} = \mathrm{x}_k^{n+1} + \lambda_k^{n+1}\Delta\mathrm{x}_k^{n+1}, \tag{21}$$

where λ_k^{n+1} is the step length determined by the cubic backtracking line search strategy, and $\Delta\mathrm{x}_k^{n+1}$ is the Newton direction that can be obtained by solving the preconditioned Jacobian system inexactly as follows,

$$\left\| M^{-1}J\Delta\mathrm{x}_k^{n+1} + M^{-1}F\left(\mathrm{x}_k^{n+1}\right) \right\| \le \gamma_L \left\| M^{-1}F\left(\mathrm{x}_k^{n+1}\right) \right\|, \tag{22}$$

where $J = F'\left(\mathrm{x}_k^{n+1}\right)$ is the Jacobian matrix, M^{-1} is the preconditioner to be introduced later, and γ_L is a relative tolerance. The nonlinear iteration is stopped if

$$\|F\left(\mathrm{x}_{k+1}^{n+1}\right)\| \le \gamma_N, \tag{23}$$

where γ_N is an absolute tolerance.

A popular choice for ordering the unknowns in x_k^{n+1} and $F\left(\mathrm{x}_k^{n+1}\right)$ is the field-by-field ordering, in which the components are ordered as

$$\mathrm{x}_k^{n+1} = (\underbrace{u_{x,1}, \ldots, u_{x,N_m}, u_{y,1}, \ldots, u_{y,N_m}, u_{z,1}, \ldots, u_{z,N_m}}_{\mathrm{u}_k^{n+1}}, \underbrace{p_1, \ldots, p_{N_m}}_{\mathrm{p}_k^{n+1}})^T,$$

where N_m is the number of mesh nodes, $u_{x,i}$, $u_{y,i}$, $u_{z,i}$ are velocity components and p_i is the pressure component of x^{n+1} defined on the i^{th} mesh vertex ($i = 1, \ldots, N_m$). Using this ordering, the unpreconditioned Jacobian system has the form

$$J\Delta\mathrm{x}_k^{n+1} = \begin{pmatrix} A & B \\ B^T & C \end{pmatrix} \begin{pmatrix} \Delta\mathrm{u}_k^{n+1} \\ \Delta\mathrm{p}_k^{n+1} \end{pmatrix} = -F\left(\mathrm{u}_k^{n+1}, \mathrm{p}_k^{n+1}\right). \tag{24}$$

The entries of A are contributed by the convection term, the viscosity term, and the slip and penetration boundary terms which are treated in a fully implicit manner. The entries of C are contributed by the pressure stabilization term. On the element level, the unknowns of x_k^{n+1} associated with the vertices of an element E are ordered as

$$\mathrm{x}_k^{n+1}|_E = (u_{x,1}^E, \ldots, u_{x,4}^E, u_{y,1}^E, \ldots, u_{y,4}^E, u_{z,1}^E, \ldots, u_{z,4}^E, p_1^E, \ldots, p_4^E)^T.$$

The corresponding element Jacobian matrix J^E has a similar 2×2 block form as J in Eq. (24). For $1 \le i \le 4$, the contribution of the slip and penetration boundary terms to $J^E(i,j)$ has the form

$$\begin{aligned}
&\text{for } 1 \le j \le 4: \ -\left(C_\tau - C_n\right)\textstyle\int_{\Gamma^E_{wall}} v_{x,j}n_x v_{x,i}n_x d\Gamma + C_\tau \int_{\Gamma^E_{wall}} v_{x,j}v_{x,i}d\Gamma,\\
&\text{for } 5 \le j \le 8: \ -\left(C_\tau - C_n\right)\textstyle\int_{\Gamma^E_{wall}} v_{y,j}n_y v_{x,i}n_x d\Gamma,\\
&\text{for } 9 \le j \le 12: \ -\left(C_\tau - C_n\right)\textstyle\int_{\Gamma^E_{wall}} v_{z,j}n_z v_{x,i}n_x d\Gamma.
\end{aligned}$$

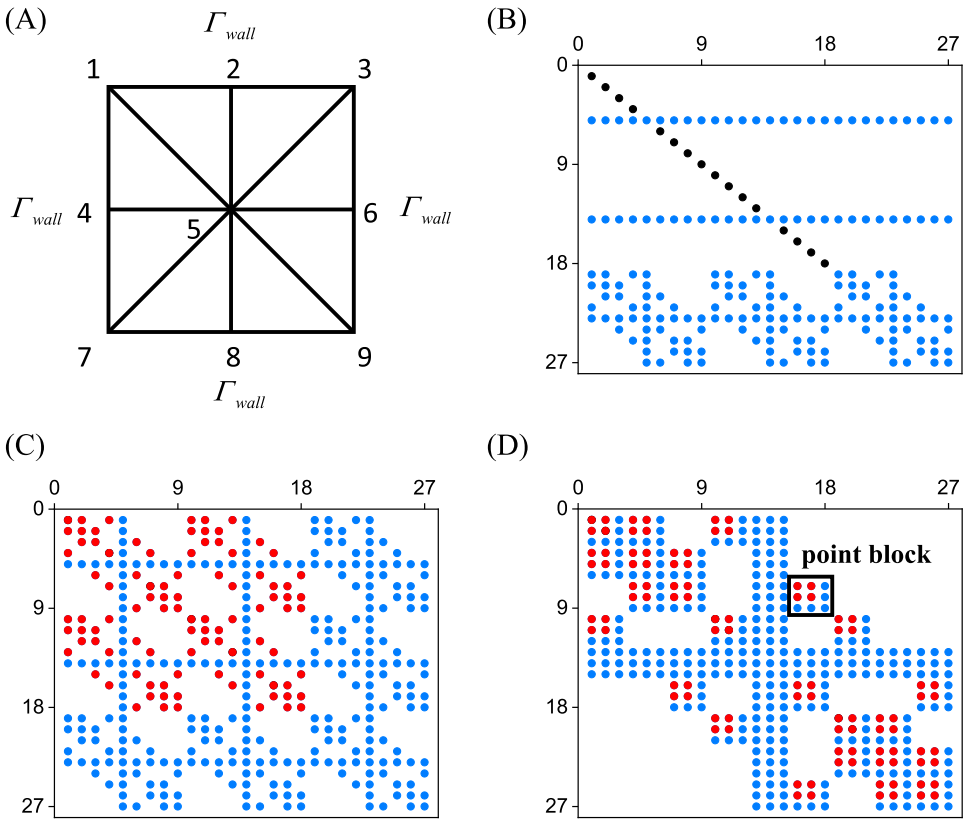


Fig. 3. Sparsity pattern of the Jacobian matrix. (A) a sample mesh on a square domain; (B) the matrix constructed with the classical Dirichlet boundary condition ($\mathbf{u} = \mathbf{0}$) using a field-by-field ordering, and the black entries represent the values of one. (C) the matrix constructed with the SPBC using a field-by-field ordering; (D) the matrix constructed with the SPBC using a node-by-node ordering. The boundary integral terms of SPBC contribute to the red entries. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Similar formulas are obtained for $5 \le i \le 12$. The entries $J^E(i,j)$ with $1 \le i,j \le 12$ assemble to the block A. When the coefficients C_n and C_τ are assumed sufficiently large values, the boundary integral terms dominate over other contributions to the entries of A. On the other hand, the contribution of the pressure stabilization term gives

$$J^E(i,j) = -C_\Pi \int_E \phi_i \left(\phi_j - \frac{1}{4}\right) dE, \quad 13 \le i,j \le 16,$$

which assembles to the block C.

The sparsity pattern of the Jacobian matrix J generated with the Robin-type SPBC significantly differs from the matrix formed with the conventional Dirichlet-type boundary condition $\mathbf{u} = \mathbf{0}$, as illustrated in Fig. 3. Typically, in a standard implementation, rows associated with the Dirichlet-type boundary condition are substituted with zeros except for the diagonal entries. In contrast, utilizing the SPBC maintains the original nonzero pattern of the matrix. Furthermore, organizing the matrix field by field results in a broad bandwidth, this often affects the memory caching performance and poses challenges for devising effective parallel algorithms for solving the linear Jacobian system. In this work, we consider a node-by-node ordering approach, in which the unknowns associated with each mesh point are always placed together, i.e.,

$$\mathbf{x}_k^{n+1} = (u_{x,1}, u_{y,1}, u_{z,1}, p_1, \ldots, u_{x,N_m}, u_{y,N_m}, u_{z,N_m}, p_{N_m})^T,$$

The corresponding Jacobian matrix has a point-block structure that groups the entries into many 4×4 small blocks (3×3 in 2D) with each block representing the coupled correlation among the physical variables on the same mesh point. The node-by-node ordering leads to a more compact nonzero structure compared to the field-by-field ordering, as shown in Fig. 3(D). It can be seen that the slip and penetration boundary integral terms contribute to the small block associated with per mesh point on Γ_{wall}^h.

Choosing practical values for the slip and penetration coefficients often results in an ill-conditioned Jacobian J, and the size of J can be considerably large when applied to 3D blood flow problems. As a consequence, solving the Jacobian system becomes highly challenging. In this study, we propose to solve the Jacobian system in parallel by using a scalable domain decomposition method. Specifically, the Generalized Minimal RESidual (GMRES) method is used as the linear solver, and the preconditioner M^{-1} is designed based on the one-level Restricted Additive Schwarz (RAS) algorithm [40] with the awareness of the features of SPBC, which

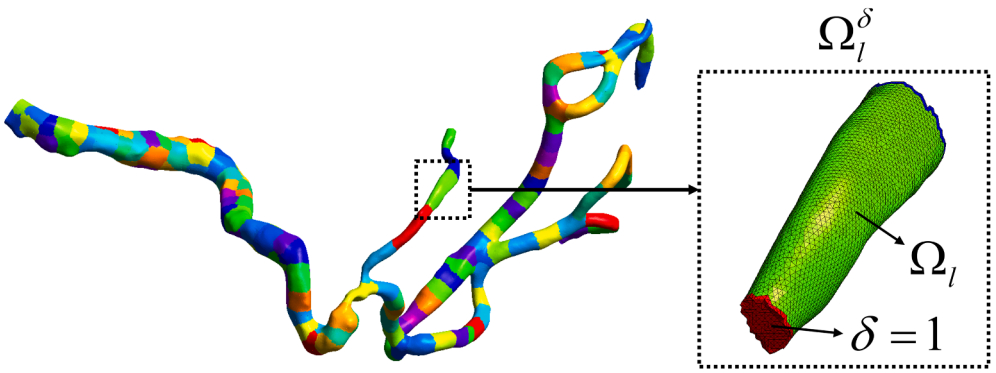


Fig. 4. A sample partition of an artery.

is constructed as

$$M^{-1} = M_{RAS}^{-1} D^{-1}. \tag{25}$$

In Eq. (25), D is a diagonal matrix for scaling the rows of J and F contributed by the slip and penetration terms. Let $\mathcal{N} = \{1, \ldots, 4N_m\}$ be an index set of all components of F and $S \subset \mathcal{N}$ be an index set of the components associated with the SPBC contribution, we define

$$D(i,i) = \begin{cases} C_n, & i \in S, \\ 1, & i \in \mathcal{N} \setminus S. \end{cases} \tag{26}$$

We remark that this scaling procedure helps reduce the condition number of J when a large value is used for C_n. The one-level RAS preconditioner M_{RAS}^{-1} is constructed based on the point-block structure of the Jacobian, without the prior knowledge of physical fields. Denote by n_p the number of processor cores of the parallel computer, we partition the computational mesh Ω_h into n_p nonoverlapping subdomains Ω_l (i.e., $\Omega_i \cap \Omega_j = \emptyset$, $\forall\, i \neq j$) for $l = 1, \ldots, n_p$, such that $\Omega_h = \Omega_1 \cup \cdots \cup \Omega_{n_p}$. The partition is obtained using METIS [41]. A sample partition of the artery is presented in Fig. 4. We then extend each Ω_l by δ layers of mesh elements to overlap with its neighboring subdomains. The resulting overlapping subdomain is denoted as Ω_l^δ. Within each overlapping subdomain, we introduce the corresponding subvector z_l^δ and define the restriction operator R_l^δ, which maps the global vector of unknowns in Ω_h to z_l^δ, i.e.,

$$z_l^\delta = R_l^\delta z = (I \quad 0) \begin{pmatrix} z_l^\delta \\ z \setminus z_l^\delta \end{pmatrix}.$$

We denote R_l^0 as the restriction operator that returns z_l defined on the nonoverlapping subdomain. Then, the RAS preconditioner is defined as

$$M_{\mathrm{RAS}}^{-1} = \sum_{l=1}^{n_p} \left(R_l^0\right)^T \left(J_l\right)^{-1} R_l^\delta, \tag{27}$$

$$J_l = R_l^\delta J \left(R_l^\delta\right)^T. \tag{28}$$

In Eq. (27), $\left(J_l\right)^{-1}$ is the inverse of the subdomain Jacobian matrix, whose product with a vector is computed by using a point-block incomplete LU (ILU) factorization of J_l. In this algorithm, the inverse of each small block is computed exactly and all unknowns associated with the same mesh point are updated simultaneously, giving rise to good convergence and numerical stability of the preconditioner.

Remark 1. In this work, we employ a one-level overlapping Schwarz preconditioner without a coarse space to accelerate the convergence of the linear solver within the fully implicit framework. This choice is motivated by the complexity of the unstructured meshes and realistic geometries inherent to blood flow simulations, which pose significant challenges for constructing efficient and scalable coarse problems. The absence of a natural multilevel hierarchy and the irregularity of domain partitions make the generation of a robust coarse space algorithmically intricate.

4. Numerical experiments

In Sections 4.1–4.3, we first examine the effectiveness of the SPBC and the efficiency of the solution algorithm. Next, we apply the fully-implicit algorithm to blood flow simulation in a patient-specific artery and investigate its robustness and parallel performance in Section 4.4. In all, four numerical test cases are examined: 4.1) the Womersley flow; 4.2) a 2D step flow; 4.3) flow in a curved channel; and 4.4) blood flow in a patient-specific artery.

The initial velocity and pressure fields are typically unknown beforehand. Consequently, we initialize these fields with zero values across the entire domain at the start of the simulation ($t = 0$). The algorithms are implemented using PETSc [42]. Unless explicitly

Table 1
A mesh convergence test for the Womersley flow problem.

Mesh	Number of elements	Number of nodes	Element size h (cm)	$\|u_z - V\|_{L^2}$	order	$\|p - P\|_{L^2}$	order
Mesh1	8425	2027	0.169	4.10×10^{-2}	-	1.65×10^{-1}	-
Mesh2	61,047	12,166	0.085	9.82×10^{-3}	2.06	6.91×10^{-2}	1.26
Mesh3	520,462	92,781	0.042	2.09×10^{-3}	2.24	2.89×10^{-2}	1.26

specified, the following parameters are used in our solvers. The absolute tolerance for the nonlinear iteration is set to 10^{-4} and the relative tolerance for the linear iteration is 10^{-3}. The restart value of GMRES is 400. The size of overlap in the linear RAS preconditioner is $\delta = 1$ for all cases, and we use the point-block ILU with 3 fill-in levels as the subdomain solver. We denote by 'Newton' the averaged number of Newton iterations per time step, and by 'GMRES' the averaged number of GMRES iterations per Newton iteration.

4.1. The Womersley flow

We first consider the Womersley flow as a benchmark to verify the discretization scheme with the pressure stabilization method. The computational domain represents a simplified artery modeled as a straight tube with length $L = 5\ cm$ and radius $R = 0.5\ cm$, centered at $(0, 0, 0)$. The analytical solution to this problem is [43]:

$$V(r,t) = -R^2(\sin(t)J_1(r) + \cos(t)J_2(r))/\mu\omega_0^2, \tag{29a}$$

$$P(z,t) = \cos(t)(z - L/2), \tag{29b}$$

where V is the z-directional velocity, P is the pressure, r is the radius coordinate, $\omega_0 = R\sqrt{\rho/\mu}$ is the Womersley number, J_1 and J_2 are the real and imaginary part of the function $1 - J_0(\omega_0 i^{3/2} r/R)/J_0(\omega_0 i^{3/2})$, and $J_0(x)$ is the zeroth order Bessel function. The viscosity is chosen to 0.035 $dyne \cdot s/cm^2$ and the density is 1.06 g/cm^3 [43]. An analytical velocity profile derived from Eq. (29) is used for the inlet and a traction-free boundary condition is used for the outlet:

$$\mu \frac{\partial \mathbf{u}}{\partial \mathbf{n}} - p\mathbf{n} = \mathbf{0} \quad on\ \Gamma_{outlet}. \tag{30}$$

On the wall, we apply the SPBC with $C_n = 10^8\ g/(cm^2 \cdot s)$ and $C_\tau = 10^5\ g/(cm^2 \cdot s)$ to approximate the no-slip and no-penetration boundary conditions. The time step size is $\Delta t = 0.01\ s$. A mesh convergence test is shown in Table 1, in which the L^2 norms of velocity and pressure errors are computed at $t = 2\pi$ s. According to the table, the axial velocity exhibits a second-order convergence, while the pressure shows an order of about 1.26. Numerical results obtained using different meshes are presented in Fig. 5, including the z-directional velocity at the center $u_z(0, 0, 0, t)$ and the average pressure difference between the inlet and the outlet $grad(p) = (\bar{p}|_{\Gamma_{inlet}} - \bar{p}|_{\Gamma_{outlet}})/L$. To validate the SPBC, we provide the results obtained using a Dirichlet-type implementation of no-slip and no-penetration boundary condition $\mathbf{u} = \mathbf{0}$ on Γ_{wall} for comparison. As shown in Fig. 5(C) and (D), a finer mesh results in reduced error, and the SPBC with large coefficients leads to almost identical results with the Dirichlet-type no-slip and no-penetration boundary condition. To study the impact of the pressure stabilization coefficient C_Π on the numerical accuracy, we compare the error of $grad(p)$ using Mesh3 and summarize the results in Fig. 6. When the stabilization coefficient C_Π is set to zero, significant errors arise in the pressure field. These errors can be reduced by increasing C_Π; however, an excessively large value of C_Π may cause the solution to deviate from the accurate result. Notably, we observe that $C_\Pi = 1$ yields the least error, a finding also supported by references [38,39]. Consequently, for the problems under consideration in this study, we opt to keep $C_\Pi = 1$.

4.2. A 2D step flow

To validate the effectiveness of SPBC and its implementation, we conduct a simulation of a 2D step flow and compare the results with the reference [24]. The setting of the problem is illustrated in Fig. 7(A). A uniform velocity profile $\mathbf{u}_{in} = (1, 0, 0)$ is used for the inlet and the traction-free boundary condition Eq. (30) is used for the outlet. The density is $\rho = 1$ and the kinematic viscosity is $\mu = 0.02$. The SPBC is applied for the upper and lower boundaries in which the coefficient C_n is fixed to 10^{12}. The resulting flow field is characterized by a vortex formation behind the step, as depicted in Fig. 7(B), and a decrease in C_τ leads to an expansion in the vortex size. A reattachment point with zero-slip velocity consistently emerges behind the step due to recirculation of the vortex near the wall. Therefore, the vortex size can be quantitatively assessed by the location of the reattachment point. Fig. 7(C) shows the location of the reattachment point with respect to C_τ for various mesh sizes. The mesh configurations are provided in Table 2. Our results align quite well with the reference [24].

4.3. Flow in a curved channel

As discussed in Section 3.2.2, the revised normal vector is critical to the efficacy of SPBC. We conduct an error analysis for this treatment by using a simplified artery modeled by a curved channel, as shown in Fig. 8. The diameter of the channel is 1. We consider the steady-state Navier-Stokes equations, subject to a parabolic inlet velocity profile given by $\mathbf{u}_{\text{in}} = (0, 0, 1 - 4(x^2 + y^2))$, and a traction-free boundary condition Eq. (30) applied at the outlet. The SPBC is applied on the channel wall for which we set C_τ to three groups

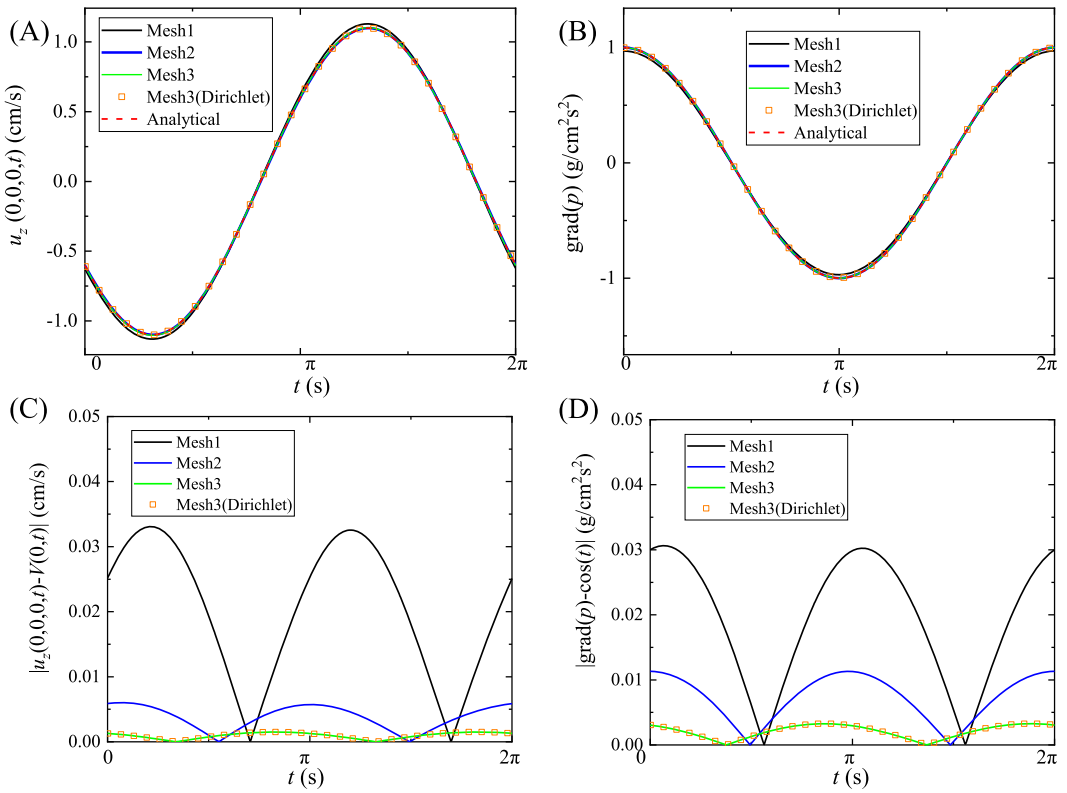


Fig. 5. Results of the Womersley flow: (A) z-directional velocity $u_z(0,0,0,t)$, (B) average pressure difference between the inlet and the outlet $grad(p)$, (C) error of $u_z(0,0,0,t)$; (D) error of $grad(p)$.

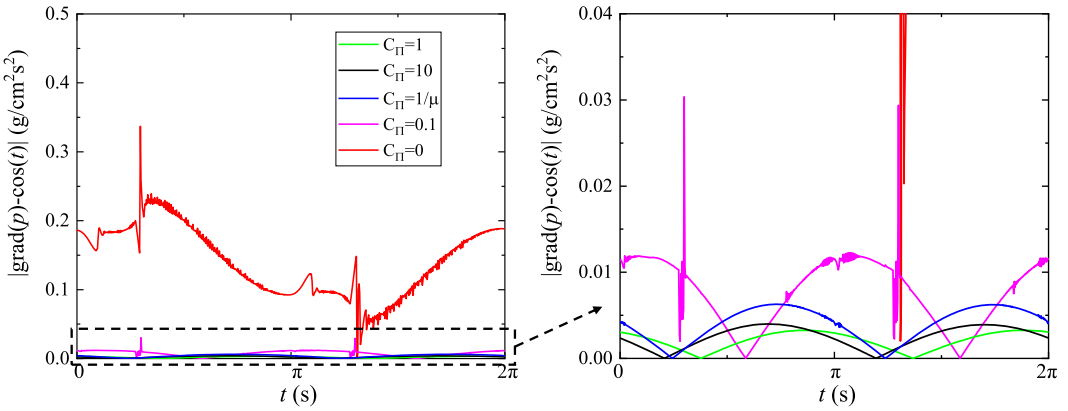


Fig. 6. Error of $grad(p)$ obtained using different stabilization coefficients C_Π on Mesh3.

Table 2
Mesh information of the step flow problem.

Mesh	Number of elements	Number of nodes
Mesh 1	2796	1494
Mesh 2	92,622	46,822
Mesh 3	268,851	135,376
Mesh 4	1,075,408	539,603

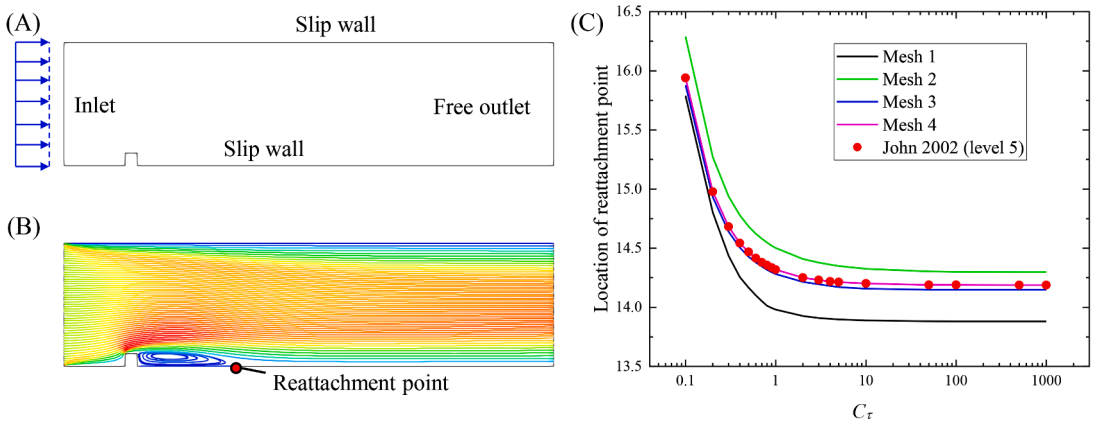


Fig. 7. Results of the step flow: (A) configuration of the problem, (B) streamlines colored with velocity magnitude, illustrating the vortex and the location of reattachment point, (C) location of the reattachment point with respect to C_τ for various mesh sizes, in comparison with the reference results [24].

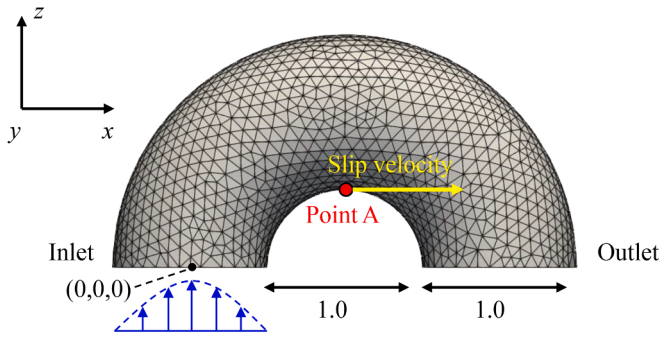


Fig. 8. A simplified artery modeled by a curved channel with a sample mesh (Mesh 1).

of values 0.1, 1, and 10, and within each group we vary C_n from 10 to 10^4. The density is $\rho = 1$ and the dynamic viscosity is $\mu = 0.05$ for this case. To quantitatively assess the implementation of SPBC, we choose an observation point A in the middle of the channel surface (shown in Fig. 8) and compute the following effective slip coefficient at A that can be derived directly from the boundary condition Eq. (3b),

$$C_\tau{}' = -WSS_x/u_x, \tag{31}$$

where u_x is the x-directional component of the velocity $\mathbf{u}$ and WSS_x is the x-directional component of the wall shear stress (WSS), formulated as follows,

$$WSS = \mu\frac{\partial \mathbf{u}}{\partial \mathbf{n}} - \mu\left(\frac{\partial \mathbf{u}}{\partial \mathbf{n}}\cdot\mathbf{n}\right)\mathbf{n}, \quad on\ \Gamma_{wall}. \tag{32}$$

Three meshes are used for the assessment, as listed in Table 3. A comparison of results obtained using the element face normal vector and the revised normal vector in SPBC is provided in Fig. 9. We observe that using element face normal vectors leads to vast discrepancies between the preset coefficient C_τ and the effective coefficient $C_\tau{}'$ when C_n is large. In contrast, using the revised normal vectors results in good consistency between C_τ and $C_\tau{}'$, ensuring the efficacy of SPBC.

Table 3
Mesh information of the curved channel flow problem.

Mesh	Number of elements	Number of nodes
Mesh 1	34,406	6676
Mesh 2	273,965	49,236
Mesh 3	2,208,073	382,152

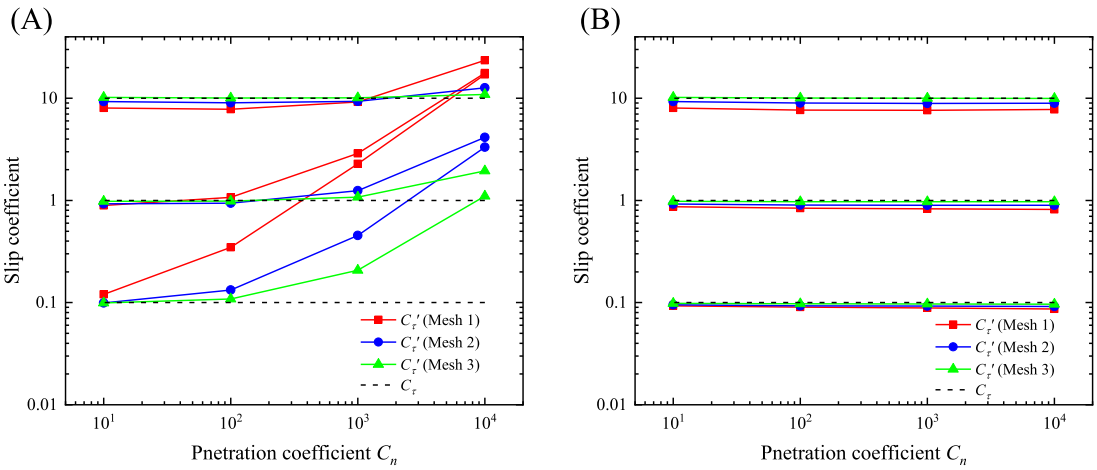


Fig. 9. The effective slip coefficient C_τ' computed at an observation point with different preset coefficient C_τ obtained using element face normal vectors (A) and revised normal vectors (B) for the implementation of SPBC. Each connected line corresponds to the same $C_\tau \in \{0.1, 1, 10\}$.

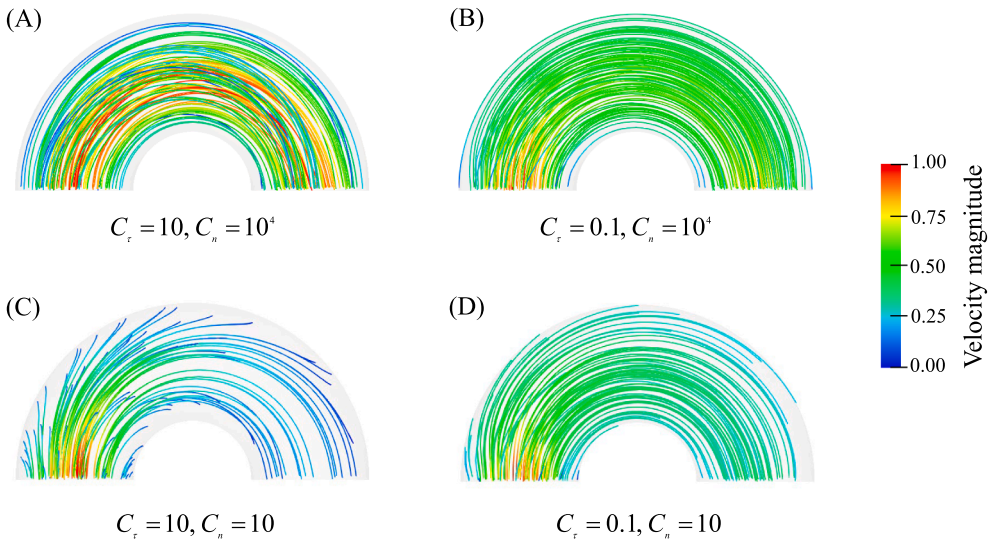


Fig. 10. Streamlines generated with different slip and penetration coefficients for the case with inflow velocity $(0, 0, 1 - 4(x^2 + y^2))$.

In Fig. 10, the streamlines generated with different slip and penetration coefficients on Mesh3 are provided. As expected, with a smaller C_n, certain streamlines are found to penetrate through the permeable boundary. On the other hand, decreasing C_τ leads to greater slip velocities along the boundary.

To investigate the impact of SPBC on unsteady flow patterns, we consider the time-dependent Navier-Stokes equations with the inflow velocity $\mathbf{u}_{in}(t) = (0, 0, (1 - 4(x^2 + y^2)))(5\cos(4\pi t - \pi) + 15))$ on Γ_{inlet}. The streamlines obtained using different slip and penetration coefficients are shown in Fig. 11. These results indicate that decreasing either C_τ or C_n typically generates a vortex in the middle of the channel. Moreover, the intensity of the vortex tends to amplify during the deceleration phase ($t \in [0.25, 0.5]$). It was shown in [44] that the effective slip behavior is influenced by the curvature of the boundary. The difference of curvature between the upper and lower boundary parts in this case leads to variations in slip behavior along the channel that creates the potential for generating a vortex. This curvature-induced variation of slip is more profound when C_τ is decreased. The vortex induced by penetration is also attributed to the channel geometry. As the mainstream within the channel ascends and penetrates the wall, a sharp change in flow direction occurs at the center of the channel. In this scenario, increased penetration effects facilitate the development of a larger vortex.

One important task in this study is to investigate how the slip and penetration coefficients affect the condition numbers and the performance of the linear solver. In Table 4, we list the condition numbers ($cond(M^{-1}A)$) of the preconditioned Jacobian matrix, the number of GMRES iterations, and the compute time for the second Newton iteration. The cases in Fig. 10 are used for the tests, and the case with a larger coefficient $C_n = 10^8$ is also included for comparison. The number of processor cores used is 16. Several

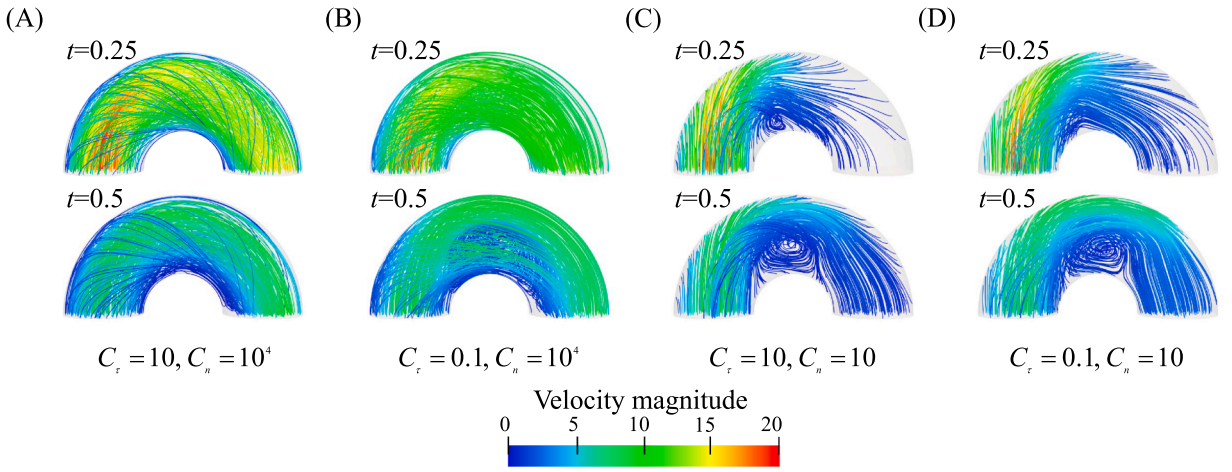


Fig. 11. Streamlines generated with different slip and penetration coefficients for the case with inflow velocity $(0, 0, (1 - 4(x^2 + y^2)))(5\cos\times(4\pi t - \pi) + 15))$.

Table 4
Performance of various preconditioners when using different slip and penetration coefficients for the flow in a curved channel. '$*$ (†, ‡)' in the column of GMRES for the field-split method means that the number of outer GMRES iterations is $*$, and the total numbers of GMRES iterations for the velocity and pressure sub-solvers are † and ‡, respectively.

Preconditioner M^{-1}	C_n	C_τ	$cond(M^{-1}A)$	Newton	GMRES	Compute time (s)
I (No preconditioning)	10	0.1	6.24×10^4	3	1139	614
	10	10	1.44×10^4	3	623	90.7
	10^4	0.1	$> 10^5$	-	diverges	-
	10^4	10	$> 10^5$	-	diverge	-
	10^8	0.1	$> 10^6$	-	diverges	-
	10^8	10	$> 10^6$	-	diverges	-
M_{RAS}^{-1} (Point-block RAS)	10	0.1	25.8	3	27	0.283
	10	10	18.1	3	20	0.236
	10^4	0.1	97.1	3	46	0.412
	10^4	10	16.0	3	31	0.308
	10^8	0.1	1.12×10^5	4	105	0.863
	10^8	10	1.94×10^4	4	88	0.725
$M_{RAS}^{-1}D^{-1}$ (Point-block RAS)	10	0.1	25.7	3	27	0.279
	10	10	18.0	3	20	0.236
	10^4	0.1	31.1	3	42	0.384
	10^4	10	15.6	3	29	0.296
	10^8	0.1	612	3	44	0.400
	10^8	10	59.0	3	35	0.334
$M_{FS}^{-1}D^{-1}$ (Field-split)	10	0.1	1.07	3	3 (364, 42)	0.959
	10	10	1.05	3	3 (244, 39)	0.846
	10^4	0.1	1.12	3	3 (484, 54)	1.081
	10^4	10	1.08	3	3 (339, 45)	0.937
	10^8	0.1	2.53	3	4 (575, 63)	1.186
	10^8	10	1.08	3	3 (418, 50)	1.020

preconditioning strategies are compared, including the proposed RAS preconditioner, and the field-split preconditioner that is popular when the Jacobian is ordered field-by-field. We refer to [45] for more details of the field-split preconditioner. It is seen from the table that when without preconditioning, a large coefficient C_n results in considerably large condition numbers, causing divergence of the GMRES method. The point-block RAS preconditioner effectively reduces the condition number, ensuring the convergence of preconditioned GMRES for all cases. However, for cases with extremely large C_n, the condition number and the number of GMRES iterations remain notably high. Conversely, employing point-block RAS with scaling markedly enhances the convergence, leading to reduced computation time. Although the field-split preconditioner results in a relatively low number of outer GMRES iterations, each outer iteration entails multiple GMRES solves within the velocity and pressure sub-solvers. This nested structure leads to increased computational overhead and reduced overall efficiency. In contrast, the proposed node-by-node preconditioner demonstrates superior performance, in terms of both iteration count and compute time, across a range of slip and penetration scenarios.

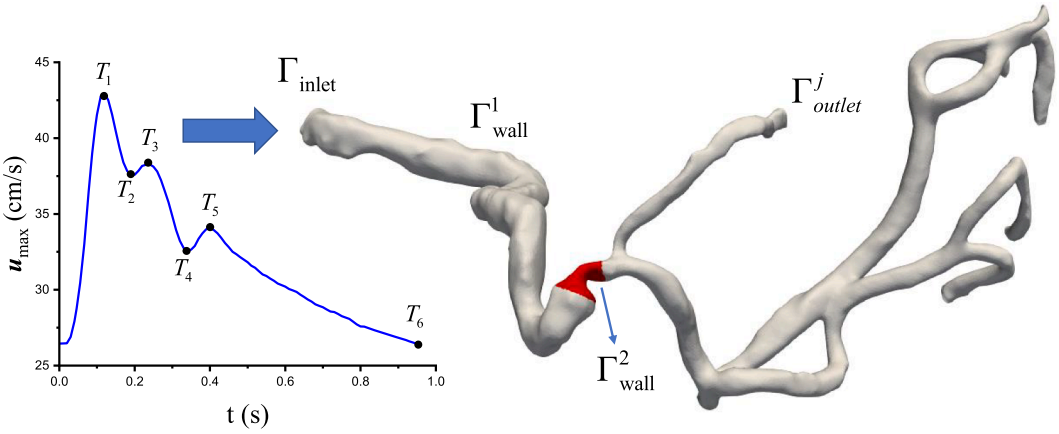


Fig. 12. An intracranial artery with a stenosis. The inlet velocity is derived from a clinically measured flow rate.

4.4. Blood flow in a patient-specific artery

In this section, we apply the SPBC and solution algorithms to the blood flow in an intracranial artery with a stenosis, as shown in Fig. 12. The density of blood is 1.06 g/cm^3, the viscosity is 0.05 $dyne \cdot s/cm^2$. An unstructured mesh with 9,196,574 tetrahedra is used for the tests. The number of mesh nodes is 1,641,136, resulting in 6,564,544 unknowns. We apply the following velocity profile on the inlet [35],

$$\mathbf{u}_{in}(t) = \mathbf{u}_{\max}(t) - \mathbf{u}_{\max}(t)\left(1 - \frac{d}{d_{\max}}\right)^2, \quad \text{on } \Gamma_{inlet}, \tag{33}$$

where d denotes the minimum distance between the mesh point $P(x, y, z) \in \Gamma_{inlet}$ and the boundary of Γ_{inlet}, and $d_{\max} = \max_{P \in \Gamma_{inlet}} d$. $\mathbf{u}_{\max}$ is the velocity at the point that yields $d = d_{\max}$, which is derived from a clinically measured flow rate at the inlet, as illustrated in Fig. 12. A cardiac cycle is considered for the time period. On the outlets, we use the resistance boundary condition Eq. (2) with $R_j = 322,884\ dyne \cdot s/cm^5$.

Experimental studies have been made to understand the slip effects of blood flow in microvessels [8–12] and main arteries [13,14]. Assuming the artery to be ideally circular with radius R, the work in [9] measured the distribution of velocity V on the cross-section and fitted it by the following parabolic profile

$$V(r) = V_{\max}\left(1 - (1 - \alpha)\left(\frac{r}{R}\right)^2\right), \tag{34}$$

where r is the distance from the center of the cross-section, and $\alpha = V_{\text{slip}}/V_{\max}$ is the ratio between the slip velocity V_{slip} and the maximum speed of the cross-section $V_{\max}$. Based on this experimental finding, we estimate the coefficient C_τ by connecting the fitted profile Eq. (34) with the SPBC. Let $\boldsymbol{\tau}$ be a unit vector normal to the cross-section, we have $V = \mathbf{u} \cdot \boldsymbol{\tau}$. Note that the radial direction gives the normal direction to the artery wall, using Eqs. (3b) and (34), we obtain

$$V_{\text{slip}} = (\mathbf{u} \cdot \boldsymbol{\tau})|_{r=R} = -\frac{1}{C_\tau}\mu \frac{\partial V}{\partial r}|_{r=R} = \frac{2(1-\alpha)\mu}{RC_\tau} V_{\max}. \tag{35}$$

Then, based on the definition of the slip ratio

$$\alpha = \frac{V_{\text{slip}}}{V_{\max}} = \frac{2(1-\alpha)\mu}{RC_\tau}, \tag{36}$$

we get

$$C_\tau = \frac{1-\alpha}{\alpha} \cdot \frac{2\mu}{R}. \tag{37}$$

According to [9], α is fitted within the range of 0.1 to 0.4 for major arteries. We choose $\alpha = 0.1$ for the concerned intracranial artery with radius $R \approx 0.1\ cm$, resulting in an estimated slip coefficient of $C_\tau \approx 10\ g/(cm^2 \cdot s)$. To our knowledge, few studies have identified the appropriate range of the penetration coefficient C_n for arteries. In general, healthy arteries are nearly impermeable, but pathological conditions can compromise the endothelial barrier, resulting in blood leakage. We assume that the artery wall consists of two parts $\Gamma_{wall} = \Gamma^1_{wall} \cup \Gamma^2_{wall}$, where $\Gamma^2_{wall} = \Gamma_{wall} \setminus \Gamma^1_{wall}$ is in the vicinity of the stenosis. We consider the following three cases:

(i) Nearly no slip & no penetration boundary condition with $C_n = 10^8\ g/(cm^2 \cdot s)$ and $C_\tau = 10^5\ g/(cm^2 \cdot s)$ for both Γ^1_{wall} and Γ^2_{wall};
(ii) Slip only boundary condition with $C_n = 10^8\ g/(cm^2 \cdot s)$ and $C_\tau = 10\ g/(cm^2 \cdot s)$ for both Γ^1_{wall} and Γ^2_{wall};

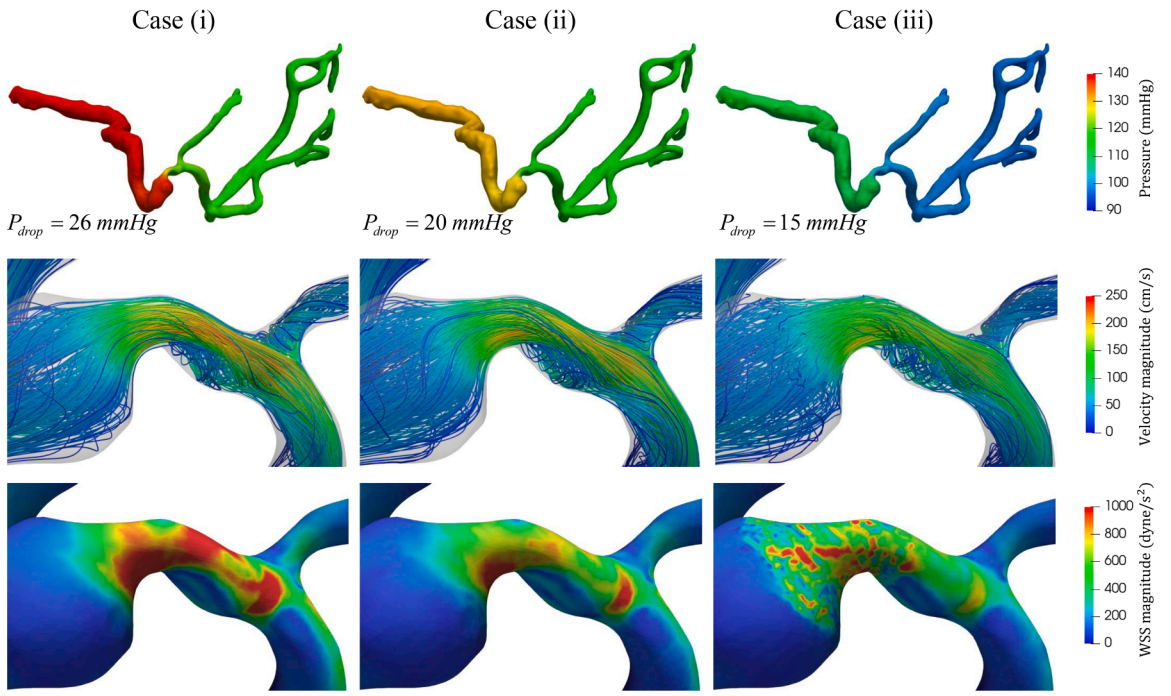


Fig. 13. Results of pressure (top), streamlines colored by velocity magnitude (middle), and wall shear stress (WSS) (bottom) for Case (i)-(iii), $t = T_1$.

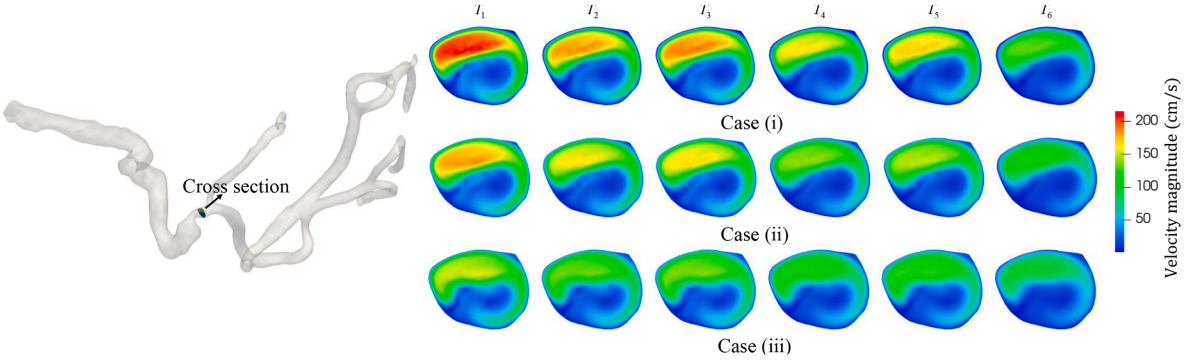


Fig. 14. Velocity profiles at the cross-section at different times for Case (i)-(iii).

(iii) Slip only boundary condition with $C_n = 10^8\ g/(cm^2 \cdot s)$ and $C_\tau = 10\ g/(cm^2 \cdot s)$ for Γ^1_{wall}, and slip & penetration boundary condition with $C_n = 10^5\ g/(cm^2 \cdot s)$ and $C_\tau = 10\ g/(cm^2 \cdot s)$ for Γ^2_{wall}. In this case, Γ^2_{wall} is assumed to be unhealthy and permeable due to conditions like plaque ruptures or erosion.

Results of the pressures, velocity, and wall shear stress (WSS) at $t = T_1$ for these cases are shown in Fig. 13. The velocity profiles at the cross-section are presented in Fig. 14. The time step size is $\Delta t = 0.01\ s$. The number of processor cores used is 1024. The pressure drop P_{drop} represents the pressure difference across the stenosis, which is a critical parameter in assessing the severity of the narrowing in the artery. It is seen from the figure that the slip-only case results in a lower pressure drop and a more moderate WSS magnitude due to reduced friction compared to the nearly no-slip and no-penetration case. Additionally, the maximum velocity is decreased, as the velocity profiles become more flattened at the same volumetric flow rate, as illustrated in Fig. 14. For example, in Case (i), the maximum velocity at the cross-section is 215 cm/s, while the maximum velocity for Case (ii) is 193 cm/s. In Case (iii), when C_n is reduced to 10^5 in the stenosis region, the pressure drop and the velocity are further reduced, and the WSS exhibits a scattering pattern. This result may be caused by the change of velocity directions near the artery wall, as indicated by the chaotic near-wall streamlines in Fig. 13. This result indicates that the blood flow is sensitive to artery wall permeability, and the fluid leakage has a significant impact on existing WSS-based risk assessment methods for pathological conditions.

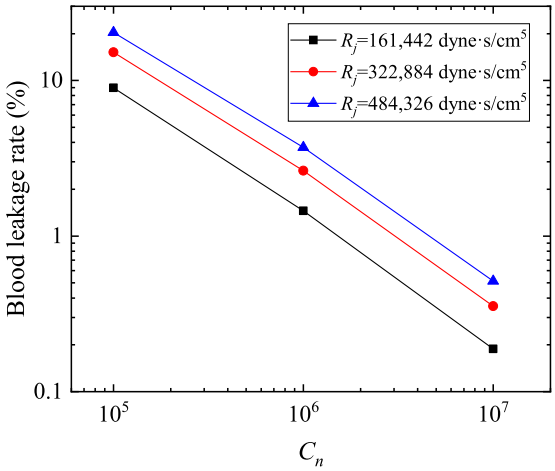


Fig. 15. Effects of C_n and R_j on the blood leakage rate at $t = T_1$.

Blood leakage can result in hemorrhage, which may be life-threatening depending on the volume of blood lost. We define the blood leakage rate as

$$\text{Blood leakage rate} = \frac{Q_{inlet} - \sum_{j=1}^{N_{outlet}} Q_j}{Q_{inlet}}, \tag{38}$$

where $Q_j = \int_{\Gamma^j_{outlet}} \mathbf{u} \cdot \mathbf{n} d\Gamma$ and $Q_{inlet} = \int_{\Gamma_{inlet}} \mathbf{u} \cdot \mathbf{n} d\Gamma$. For Case (iii), we examine how the blood leakage rate is affected by C_n and the resistance parameter R_j. Results for $t = T_1$ are plotted in Fig. 15. As expected, the leakage rate rises with a decrease of C_n and an increase of R_j. This suggests that increasing penetration effect facilitates greater fluid leakage through the arterial wall, while increased resistance may hinder the fluid movement towards downstream, further enhancing leakage. In practice, the blood leakage rate may vary in a broad range with different pathological conditions, and a volumetric leakage rate exceeding 150 ml/min is usually considered a critical hemorrhage [46]. In this study, a leakage rate of approximately 10 % corresponds to a time-averaged volumetric loss of about 0.4 mL/s, or 24 mL/min, which is below the clinical threshold for critical hemorrhage. These findings demonstrate the capability of the newly proposed SPBC in capturing certain physiological and pathological scenarios.

We next compare the performance of the proposed fully-implicit scheme with a semi-implicit scheme in terms of stability and computational efficiency for Case (iii). In the semi-implicit scheme, the nonlinear convection term is discretized in time using $(\mathbf{u}_h^n \cdot \nabla)\mathbf{u}_h^{n+1}$ while other terms are discretized identically with the fully-implicit scheme, resulting in a linear system to be solved at each time step. Both schemes are based on the first-order Euler schemes. Fig. 16 (A) and (B) show the results of pressure and velocity magnitude at an observation point $P(13.67, 10.12, 1.89)$ on Γ^2_{wall} using different time step sizes. We use 1024 processor cores for the tests. When the time step size is sufficiently small, almost identical results are obtained for both schemes. The semi-implicit scheme imposes an upper bound on the time step size, when this stability condition is not satisfied, particularly at large time step sizes, numerical instabilities such as oscillations can occur, as evidenced by the results in Fig. 16(B). To enable quantitative comparisons, we use the results from the fully-implicit scheme with $\Delta t = 0.001\ s$ as the reference and calculate the standard deviations (SD) of the results on Γ^2_{wall} for different time step sizes. As shown in Fig. 16(C) and (D), significant standard deviations are observed for the semi-implicit scheme with larger time step sizes. In contrast, the fully implicit method remains stable, allowing for accurate resolution of the flow dynamics. In Table 5, we show the average number of Newton and GMRES iterations per time step, the average standard deviations over the cardiac cycle, and the compute time for the two schemes with different time step sizes. When comparable standard deviations are required, the fully-implicit scheme allows using a larger time step size and requires less compute time. Alternative semi-implicit strategies, such as Implicit-Explicit (IMEX) methods and operator splitting techniques, offer a framework in which velocity and pressure fields are decoupled and computed alternatively. Typically, the velocity components are advanced using explicit schemes, which impose strict CFL conditions on the time step size. While these approaches can simplify implementation and reduce computational cost, the decoupling of physical variables often introduces error accumulation, potentially compromising the stability and accuracy of the overall algorithm. In addition to the first-order Euler methods considered in this study, the proposed algorithm is adaptable to higher-order time integration schemes, such as the second-order backward differentiation formula (BDF2), while preserving the same sparsity pattern of the Jacobian matrix.

To test the performance of the proposed point-block RAS preconditioner, we show in Table 6 the impact of the overlapping size of meshes and ILU fill-in levels on the compute time and numbers of iterations using 1024 processor cores. The time step size is $\Delta t = 0.01\ s$. It is observed from the table that the number of Newton iterations remains unchanged. An optimal value for overlapping size is 1, and the optimal value for ILU fill-in levels is 3 in terms of the compute time. In Table 7, we compare the performance of the point-block RAS with and without scaling for case (iii), using the optimal setting of parameters. The results indicate that the RAS preconditioner with scaling is approximately 40 % faster than the version without scaling. Compared to the results in Table 4, the

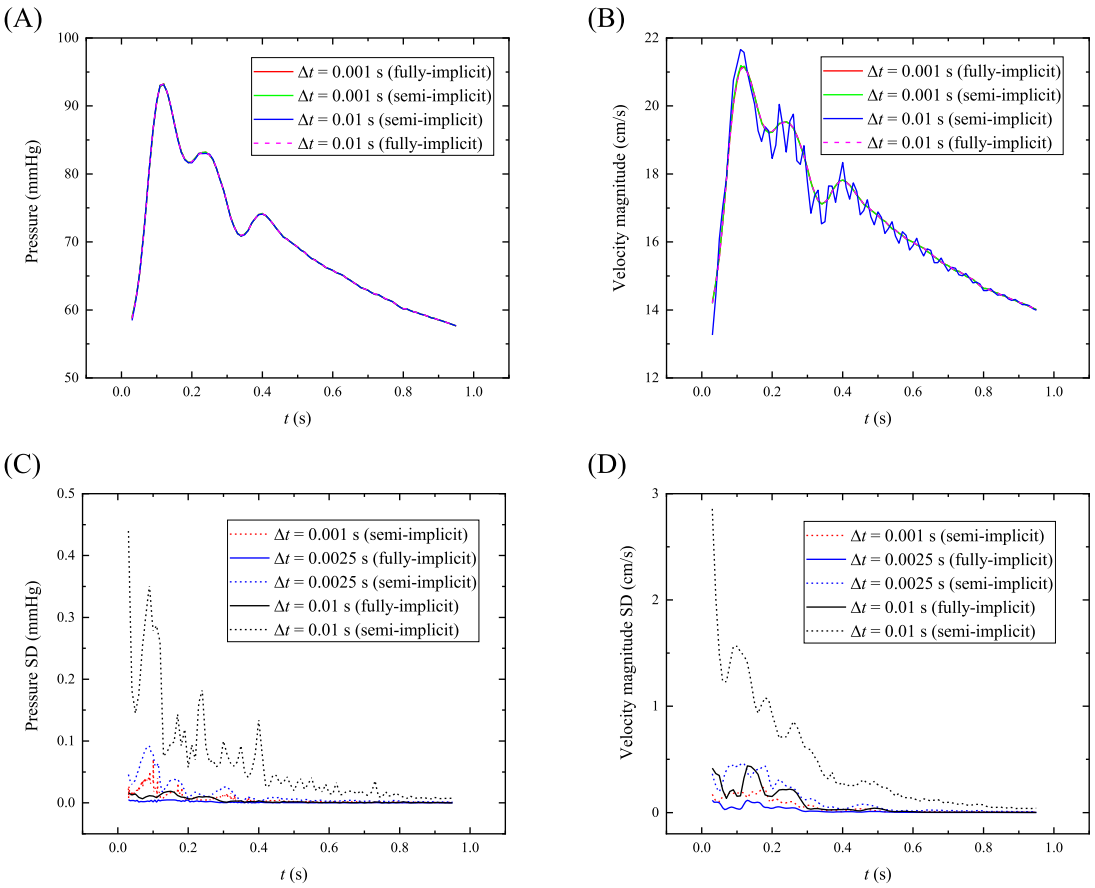


Fig. 16. (A)-(B) Results of pressure and velocity magnitude at an observation point $P(13.67, 10.12, 1.89)$ on Γ^2_{wall} using the fully-implicit and semi-implicit schemes with different time step sizes. (C)-(D) Standard deviations (SD) of the results calculated on Γ^2_{wall} using the fully-implicit scheme with $\Delta t = 0.001\ s$ as the reference.

Table 5
A comparison of performance between the fully-implicit scheme and the semi-implicit scheme for Case (iii). The number of processor cores used is 1024. 'Average SD_{vel}' and 'Average SD_{pres}' refer to the average standard deviation over the cardiac cycle for the velocity and pressure, respectively.

Δt	Scheme	Newton	GMRES	Average SD_{vel}	Average SD_{pres}	Total time (s)
0.001	fully-implicit	1.41	469	–	–	1.92×10^4
0.001	semi-implicit	–	371	0.0501	0.0059	1.26×10^4
0.0025	fully-implicit	1.65	487	0.0193	0.0011	1.01×10^4
0.0025	semi-implicit	–	381	0.1163	0.0136	3.50×10^3
0.01	fully-implicit	2.09	525	0.0794	0.0039	3.48×10^3
0.01	semi-implicit	–	402	0.4654	0.0668	1.36×10^3

numbers of GMRES iteration are significantly higher for the realistic artery model, highlighting the substantial impact of geometric and model complexity on solver performance. To address this, more advanced preconditioning strategies, such as projection methods [47] and two-level Schwarz-type preconditioners [43], may help reduce iteration counts and improve computational efficiency.

Lastly, we study the parallel performance of the overall algorithm for Case (iii). The overlapping size of the RAS preconditioner is fixed to 1 and the ILU fill-in level is fixed to 3. Results of the strong scalability test are summarized in Table 8, where the global problem size is kept fixed. As the number of processor cores n_p increases, good scalability is observed for the number of Newton iterations, while the number of GMRES iterations grows mildly, giving rise to the reduction of the compute time. The condition number of the one-level overlapping Schwarz preconditioner is inversely proportional to the subdomain size. As the number of processor cores increases, each subdomain becomes smaller, leading to reduced effectiveness of the preconditioner. This can lead to slower convergence of the iterative solver. Additionally, the increased frequency of inter-processor communication can contribute to suboptimal parallel efficiency. This behavior is commonly observed in the application of one-level overlapping Schwarz preconditioners [33,34]. Overall,

Table 6
Impact of different overlapping sizes and ILU fill-in levels for Case (iii). The number of processor cores used is 1,024. 'Time (s)' refers to the compute time per time step in seconds.

Overlap	ILU fill-in	Newton	GMRES	Time (s)
1	2	4.7	609	54
1	3	4.7	436	44
1	4	4.7	412	57
2	2	4.7	580	58
2	3	4.7	392	48
2	4	4.7	350	57
3	2	4.7	555	60
3	3	4.7	374	53
3	4	4.7	323	62

Table 7
Impact of different preconditioners for Case (iii). The number of processor cores used is 1,024. The overlapping size is 1. The ILU fill-in level is 3. 'Time (s)' refers to the compute time per time step in seconds.

Preconditioner M^{-1}	Newton	GMRES	Time (s)
M_{RAS}^{-1}	6.3	532	77
$M_{RAS}^{-1} \times D^{-1}$	4.7	436	44

Table 8
Strong scalability test for Case (iii). The overlapping size is 1. The ILU fill-in level is 3. 'Time (s)' refers to the compute time per time step in seconds.

n_p	Newton	GMRES	Time (s)	Efficiency
512	4.7	424	81	-
1024	4.7	436	44	92 %
2048	4.7	504	34	60 %
4096	4.7	563	27	38 %

Table 9
Weak scalability test for Case (iii). The overlapping size is 1. The ILU fill-in level is 3. 'Time (s)' refers to the compute time per time step in seconds.

Element size h (cm)	n_p	Number of elements/n_p	Newton	GMRES	Time (s)	Efficiency
0.030	128	8978	6.3	291	24	-
0.024	256	8979	5.3	312	23	104 %
0.019	512	8980	4.7	320	27	89 %
0.015	1024	8981	4.7	436	44	55 %

a parallel efficiency of 38 % is achieved as n_p increases from 512 to 4,096. Table 9 presents the results of the weak scalability test, in which the size of the local problem remains nearly constant. As n_p increases, the number of GMRES iterations gradually rises. When n_p is increased eightfold, the compute time nearly doubles, resulting in a parallel efficiency of 55 %.

5. Concluding remarks

This study presents an efficient algorithm for solving 3D incompressible Navier-Stokes equations with slip and penetration boundary conditions (SPBC), with application to realistic hemodynamic simulations. We propose a fully-implicit domain decomposition algorithm based on the Newton-Krylov-Schwarz method that is shown to be robust and efficient for the blood flow model. Several benchmark examples validate the efficacy of the SPBC and its implementation. Numerical analysis is conducted to examine the influence of slip and penetration coefficients on the flow behavior and the proposed algorithm. The performance tests show that the fully-implicit scheme results in better efficiency compared with the semi-implicit scheme when using a larger time step size. The overall algorithm is scalable in terms of the number of iterations and the compute time, and the parallel efficiency for the strong scalability test reaches 38 % with up to 4096 processor cores and 55 % for the weak scalability test with up to 1024 processor cores.

CRediT authorship contribution statement

Xiangdong Zhang: Writing – original draft, Software, Methodology, Conceptualization; **Li Luo:** Writing – review & editing, Methodology, Conceptualization; **Ye Li:** Resources, Funding acquisition, Conceptualization; **Xiao-Chuan Cai:** Writing – review & editing, Supervision, Methodology, Conceptualization.

Data availability

Data will be made available on request.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Li Luo reports financial support was provided by The Science and Technology Development Fund (FDCT) of Macao SAR, National Natural Science Foundation of China. Ye Li reports financial support was provided by The Strategic Priority CAS Project, National Natural Science Foundation of China. Xiangdong Zhang reports financial support was provided by Guangdong Basic and Applied Basic Research Foundation, Shenzhen Basic Research Foundation. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is supported in part by National Natural Science Foundation of China 12371442 and U2241210, Macau FDCT 0090/2022/A2, 0035/2025/RIA1, 0146/2024/RIA2, and MYRG 2022-00051-FST, the Strategic Priority CAS Project XDB38040200, the Guangdong Basic and Applied Basic Research Foundation (2022A1515011217), and Shenzhen Basic Research Foundation (JCYJ20220818101216034).

References

[1] M. Kadem, L. Garber, M. Abdelkhalek, B.K. Al-Khazraji, Z. Keshavarz-Motamed, Hemodynamic modeling, medical imaging, and machine learning and their applications to cardiovascular interventions, IEEE Rev. Biomed. Eng. 16 (2022) 403–423.
[2] X. Zhang, D. Wu, H. Li, Y. Fang, H. Xiong, Y. Li, Early diagnosis of intracranial internal carotid artery stenosis using extracranial hemodynamic indices from carotid doppler ultrasound, Bioengineering 9 (9) (2022) 422.
[3] J.F. Gerónimo, A. Keramat, J. Alastruey, H.-F. Duan, Computational modelling and application of mechanical waves to detect arterial network anomalies: diagnosis of common carotid stenosis, Comput. Meth. Programs Biomed. 227 (2022) 107213.
[4] J. Liu, L. Hao, F. van de Vosse, L. Xu, A noninvasive method of estimating patient-specific left ventricular pressure waveform, Comput. Meth. Programs Biomed. 227 (2022) 107192.
[5] A. Swillens, L. Taelman, J. Degroote, J. Vierendeels, P. Segers, Comparison of non-invasive methods for measurement of local pulse wave velocity using FSI-simulations and in vivo data, Ann. Biomed. Eng. 41 (2013) 1567–1578.
[6] L. Fu, H.N. Kim, J.D. Sterling, S.M. Baker, M.S. Lord, The role of the cell surface glycocalyx in drug delivery to and through the endothelium, Adv. Drug Deliv. Rev. 184 (2022) 114195.
[7] L. Claesson-Welsh, E. Dejana, D.M. McDonald, Permeability of the endothelial barrier: identifying and reconciling controversies, Trends. Mol. Med. 27 (4) (2021) 314–331.
[8] G.J. Tangelder, D.W. Slaaf, A.M. Muijtjens, T. Arts, M.G. Oude Egbrink, R.S. Reneman, Velocity profiles of blood platelets and red blood cells flowing in arterioles of the rabbit mesentery, Circ. Res. 59 (5) (1986) 505–514.
[9] Z. Zhong, H. Song, T.Y.P. Chui, B.L. Petrig, S.A. Burns, Noninvasive measurements and analysis of blood velocity profiles in human retinal vessels, IOVS 52 (7) (2011) 4151–4157.
[10] S. Roman, S. Lorthois, P. Duru, F. Risso, Velocimetry of red blood cells in microvessels by the dual-slit method: effect of velocity gradients, Microvasc. Res. 84 (3) (2012) 249–261.
[11] J. Tang, S.E. Erdener, B. Li, B. Fu, S. Sakadzic, S.A. Carp, J. Lee, D.A. Boas, Shear-induced diffusion of red blood cells measured with dynamic light scattering-optical coherence tomography, J. Biophoton. 11 (2) (2018) e201700070.
[12] G. Meng, J. Zhong, Q. Zhang, J.S.J. Wong, J. Wu, K.K. Tsia, N. Ji, Ultrafast two-photon fluorescence imaging of cerebral blood circulation in the mouse brain in vivo, PNAS 119 (23) (2022) e2117346119.
[13] A.V. Kamenskiy, Y.A. Dzenis, J.N. MacTaggart, A.S. Desyatova, I.I. Pipinos, In vivo three-dimensional blood velocity profile shapes in the human common, internal, and external carotid arteries, J. Vasc. Surg. 54 (4) (2011) 1011–1020.
[14] M. Correia, J. Provost, M. Tanter, M. Pernot, 4D Ultrafast ultrasound flow imaging: in vivo quantification of arterial volumetric flow rate in a single heartbeat, Phys. Med. Biol. 61 (23) (2016) L48.
[15] T. Lee, E. Charrault, C. Neto, Interfacial slip on rough, patterned and soft surfaces: a review of experiments and simulations, Adv. Colloid Interface Sci. 210 (2014) 21–38.
[16] L. Luo, Q. Zhang, X.-P. Wang, X.-C. Cai, A parallel finite element method for 3D two-phase moving contact line problems in complex domains, J. Sci. Comput. 72 (2017) 1119–1145.
[17] Y. Nubar, Blood flow, slip, and viscometry, Biophys. J. 11 (3) (1971) 252–264.
[18] F. Yilmaz, M.Y. Gundogdu, et al., A critical review on blood flow in large arteries; relevance to blood rheology, viscosity models, and physiologic conditions, Korea Aust. Rheol. J. 20 (4) (2008) 197–211.
[19] N. Koshiba, J. Ando, X. Chen, T. Hisada, Multiphysics simulation of blood flow and LDL transport in a porohyperelastic arterial wall model, J. Biomech. Eng. 129 (3) (2007) 374–385.
[20] M.P. Thon, A. Hemmler, A. Glinzer, M. Mayr, M. Wildgruber, A. Zernecke-Madsen, M.W. Gee, A multiphysics approach for modeling early atherosclerosis, Biomech. Model. Mechanobiol. 17 (2018) 617–644.
[21] T.A. Cheema, G.M. Kim, C.Y. Lee, J.G. Hong, M.K. Kwak, C.W. Park, Characteristics of blood vessel wall deformation with porous wall conditions in an aortic arch, Appl. Rheol. 24 (2) (2014) 17–24.
[22] Y. Jiang, L. Ge, R. Di, G. Lu, L. Huang, G. Li, X. Leng, S. Zhang, H. Wan, D. Geng, et al., Differences in hemodynamic characteristics under high packing density between the porous media model and finite element analysis in computational fluid dynamics of intracranial aneurysm virtual treatment, J. NeuroInterventional Surg. 11 (8) (2019) 853–858.

[23] B.M. Lucotte, C. Powell, J.R. Knutson, C.A. Combs, D. Malide, Z.-X. Yu, M. Knepper, K.D. Patel, A. Pielach, E. Johnson, et al., Direct visualization of the arterial wall water permeability barrier using CARS microscopy, PNAS 114 (18) (2017) 4805–4810.
[24] V. John, Slip with friction and penetration with resistance boundary conditions for the Navier–Stokes equations–numerical tests and aspects of the implementation, J. Comput. Appl. Math. 147 (2) (2002) 287–300.
[25] J. Pavlova, A. Fasano, J. Janela, A. Sequeira, Numerical validation of a synthetic cell-based model of blood coagulation, J. Theor. Biol. 380 (2015) 367–379.
[26] I.G. Gjerde, L.R. Scott, Kinetic-energy instability of flows with slip boundary conditions, J. Math. Fluid Mech. 24 (4) (2022) 97.
[27] D. Nolte, C. Bertoglio, Reducing the impact of geometric errors in flow computations using velocity measurements, Int. J. Numer. Meth. Biomed. 35 (6) (2019) e3203.
[28] R. Costa, S. Clain, G.J. Machado, J.M. Nóbrega, H.B. da Veiga, F. Crispo, Imposing slip conditions on curved boundaries for 3D incompressible flows with a very high-order accurate finite volume scheme on polygonal meshes, Comput. Meth. Appl. Mech. Eng. 415 (2023) 116274.
[29] P. Acevedo, C. Amrouche, C. Conca, A. Ghosh, Stokes and Navier–Stokes equations with Navier boundary condition, C. R. Math. 357 (2) (2019) 115–119.
[30] P.A. Tapia, C. Amrouche, C. Conca, A. Ghosh, Stokes and Navier-Stokes equations with Navier boundary conditions, J. Differ. Equ. 285 (2021) 258–320.
[31] H. Zhang, X. Jiang, F. Zeng, G.E. Karniadakis, A stabilized semi-implicit Fourier spectral method for nonlinear space-fractional reaction-diffusion equations, J. Comput. Phys. 405 (2020) 109141.
[32] J.J. X. Jiang, Improved uniform error bounds of exponential wave integrator method for long-time dynamics of the space fractional Klein-Gordon equation with weak nonlinearity, J. Sci. Comput. 97 (3) (2023) 58.
[33] L. Luo, W.-S. Shiu, R. Chen, X.-C. Cai, A nonlinear elimination preconditioned inexact newton method for blood flow problems in human artery with stenosis, J. Comput. Phys. 399 (2019) 108926.
[34] L. Luo, X.-C. Cai, Z. Yan, L. Xu, D.E. Keyes, A multilayer nonlinear elimination preconditioned inexact Newton method for steady-state incompressible flow problems in three dimensions, SIAM J. Sci. Comput. 42 (6) (2020) B1404–B1428.
[35] B. Feiger, M. Vardhan, J. Gounley, M. Mortensen, P. Nair, R. Chaudhury, D. Frakes, A. Randles, Suitability of lattice Boltzmann inlet and outlet boundary conditions for simulating flow in image-derived vasculature, Int. J. Numer. Meth. Biomed. 35 (6) (2019) e3198.
[36] E.C. Cyr, J.N. Shadid, R.S. Tuminaro, Stabilization and scalable block preconditioning for the Navier–Stokes equations, J. Comput. Phys. 231 (2) (2012) 345–363.
[37] C.R. Dohrmann, P.B. Bochev, A stabilized finite element method for the stokes problem based on polynomial pressure projections, Int. J. Numer. Methods Fluids 46 (2) (2004) 183–201.
[38] P.B. Bochev, C.R. Dohrmann, M.D. Gunzburger, Stabilization of low-order mixed finite elements for the stokes equations, SIAM J. Numer. Anal. 44 (1) (2006) 82–101.
[39] J. Li, Y. He, Z. Chen, A new stabilized finite element method for the transient Navier–Stokes equations, Comput. Methods Appl. Mech. Eng. 197 (1–4) (2007) 22–35.
[40] X.-C. Cai, M. Sarkis, A restricted additive Schwarz preconditioner for general sparse linear systems, SIAM J. Sci. Comput. 21 (2) (1999) 792–797.
[41] G. Karypis, V. Kumar, METIS: A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill-Reducing Orderings of Sparse Matrices, 1998.
[42] S. Balay, S. Abhyankar, M.F. Adams, S. Benson, J. Brown, P. Brune, K. Buschelman, E.M. Constantinescu, L. Dalcin, A. Dener, V. Eijkhout, J. Faibussowitsch, W.D. Gropp, V. Hapla, T. Isaac, P. Jolivet, D. Karpeev, D. Kaushik, M.G. Knepley, F. Kong, S. Kruger, D.A. May, L.C. McInnes, R.T. Mills, L. Mitchell, T. Munson, J.E. Roman, K. Rupp, P. Sanan, J. Sarich, B.F. Smith, S. Zampini, H. Zhang, H. Zhang, J. Zhang, PETSc Web page, 2025, (https://petsc.org/).
[43] Y. Liu, F. Qi, X.-C. Cai, A one-dimensional coarse preconditioner for three-Dimensional unsteady incompressible Navier–stokes flows in patient-Specific arteries, SIAM J. Sci. Comput. 46 (2) (2024) S1–S23.
[44] D. Einzel, P. Panzer, M. Liu, Boundary condition for fluid flow: curved or rough surfaces, Phys. Rev. Lett. 64 (1990) 2269–2272.
[45] N. Yang, H. Yang, C. Yang, Multilevel field-split preconditioners with domain decomposition for steady and unsteady flow problems, Comput. Phys. Commun. 280 (2022) 108496.
[46] K. Irita, Risk and crisis management in intraoperative hemorrhage: human factors in hemorrhagic critical events, Korean J. Anesthesiol. 60 (3) (2011) 151–160.
[47] S.Y.K. D. A. Knoll, A Jacobian-free Newton Krylov implicit-explicit time integration method for incompressible flow problems, Commun. Comput. Phys. 13 (5) (2013) 1408–1431.