Journal of Computational Physics 508 (2024) 113027

Contents lists available at ScienceDirect

Journal of Computational Physics

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

A highly parallel algorithm for simulating the elastodynamics of a patient-specific human heart with four chambers using a heterogeneous hyperelastic model

Yi Jiang [a,b], Zhengzheng Yan [a,b], Xinhong Wang [c], Rongliang Chen [a,b,*], Xiao-Chuan Cai [d,*]

[a] Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China
[b] Shenzhen Key Laboratory for Exascale Engineering and Scientific Computing, Shenzhen, China
[c] Second Affiliated Hospital, School of Medicine, Zhejiang University, Hangzhou, China
[d] University of Macau, Macau, China

ARTICLE INFO

Keywords:
Human heart
Hyperelasticity model
Finite element
Fully implicit method
Domain decomposition
Parallel computing

ABSTRACT

In this paper a highly parallel method is developed for simulating the elastodynamics of a four-chamber human heart with patient-specific geometry. The heterogeneous hyperelastic model is discretized by a finite element method in space and a fully implicit adaptive method in time, and the resulting nonlinear algebraic systems are solved by a scalable domain decomposition algorithm. The deformations of the cardiac muscles are quite complex due to the realistic geometry, the heterogeneous hyperelasticity of the cardiac tissue, and the myocardial fibers with active stresses. Moreover, the deformations in different chambers and at different phases of the cardiac cycle are very different. To simulate all the muscle movements including the atrial diastole, the atrial systole, the isovolumic contraction, the ventricular ejection, the isovolumic relaxation, and the ventricular filling, the temporal-spatial mesh needs to be sufficiently fine, but not too fine so that the overall computing time is manageable, we introduce a baseline mesh in space and a two-level time stepping strategy including a uniform baseline time step size to obtain the desired time accuracy and an adaptive time stepping method within a baseline time step to guarantee the convergence of the nonlinear solver. Through numerical experiments, we investigate the performance of the proposed method with respect to the material coefficients, the fiber orientations, as well as the mesh sizes and the time step sizes. For an unstructured tetrahedral mesh with more than 200 million degree of freedoms, the method scales well for up to 16,384 processor cores for all steps of an entire cardiac cycle.

1. Introduction

Computational simulation is a powerful tool for investigating the biomechanical properties of the human heart, and has been increasingly used for certain clinical applications [1–6]. However, simulating the heart is a very challenging problem due to its complex, patient-specific geometry, hyperelastic tissue properties, anisotropic myocardial fiber structure, and other parameters of

* Corresponding authors.
E-mail addresses: rl.chen@siat.ac.cn (R. Chen), xccai@um.edu.mo (X.-C. Cai).

https://doi.org/10.1016/j.jcp.2024.113027
Received 9 October 2023; Received in revised form 6 April 2024; Accepted 15 April 2024

Available online 21 April 2024

the constitutive model. To capture the details of deformation in all four chambers of the heart throughout the cardiac cycle, a fine, unstructured finite element mesh is required, necessitating a highly scalable and robust numerical method. In this paper, we present a highly parallel implicit algorithm with adaptive time stepping for simulating the whole heart for an entire cardiac cycle. Our algorithm addresses the challenges by the complex geometry of the heart and the nonlinear properties of its tissue, while also accommodating the need for a fine mesh to accurately capture the deformation of the heart.

In recent years, significant progress has been made in developing computational techniques for understanding the functionalities of the heart, but most studies focus on the modeling of one or two chambers, and the geometry is sometimes simplified into a regular shape in order to increase the chance of convergence of the numerical methods and to reduce the computational cost. Noteworthy advancements in this field can be found in [7] and [8]. In these investigations, the authors applied Newton and quasi-Newton type methods, combined with Krylov subspace Jacobian solvers, along with algebraic multigrid and/or BDDC preconditioners, to solve the nonlinear elasticity model for understanding the cardiac mechanics. Further contributions are highlighted in [9–13], where authors showcase the scalability of their algorithms, with up to $16,000$ processor cores and millions of degrees of freedom.

While simplified models provide valuable insights into cardiac function and the performance of the numerical methods, but the models lack the ability to capture the functionality of the heart as a result of the complex geometry of a realistic heart. They ignore the significant effects of the deformation of one chamber on the rest of the heart, which often leads to inaccuracies in simulations [5]. For these reasons, there is a growing interest in developing numerical algorithms that can simulate the whole human heart with patient-specific geometry to more accurately model the cardiac functions and pathology. One representative study on this topic is [14], in which the authors developed an elasticity model of the whole human heart and investigated the pericardial effects from the boundary conditions. In [15], a database with a cohort of 24 virtual four-chamber hearts was used to simulate free contractions of the myocardial muscles to understand ventricular ejection fractions, while [16] developed many powerful tools for modeling the human heart. Another important work can be found in [17], in which the authors proposed an efficient parallel numerical algorithm and verified that it can scale up to 8,192 processor cores in solving the cardiac hyperelastic system. A series recent advances in this area can be found in [18–20] which proposed and studied several comprehensive biophysically detailed electromechanical models of the human heart that incorporates atrial and ventricular contraction, including anatomically accurate geometry, myocardial fiber architecture, and a closed-loop circulatory system model. These methods successfully reproduce healthy cardiac function and highlight the importance of considering atrial contraction and fibers-stretch-rate feedback in regulating blood flows. The immersed boundary method (IBM) [21] offers another promising approach for simulating the whole heart [22]. While earlier attempts, such as [23], utilized simplified geometry rather than realistic representations, recent advancements have focused on incorporating more accurate anatomical details. In [24,25], a multiphysics model was formulated utilizing the interaction potential method [26], with a focus on augmenting the realism and predictive accuracy of cardiac dynamics across both healthy and pathophysiological scenarios. Moreover, [27] presented a fluid-structure interaction model with detailed representations of all four cardiac valves, incorporating biomechanically calibrated descriptions of cardiac structures from human tissue specimens, resulting in realistic physiological dynamics and fine-scale flow features. Moreover, in a recent advancement, [28] presented a comprehensive fluid-structure interaction model for the entire heart, with a focus on valve dynamics and interactions between the pericardium and the heart, marking a significant progress in cardiac modeling.

Nevertheless, simulating the whole heart remains a highly challenging task due to various factors, including the complex geometry, nonlinear and anisotropic properties of the myocardial tissue, and the significant deformation experienced by the four chambers at different stages of a cardiac cycle, all of which necessitate the solution of some sparse, large-scale, highly nonlinear algebraic systems. Addressing these challenges requires sophisticated numericals algorithms with high parallel scalability and robustness.

In this paper, we present a novel numerical method that extends our previous work on simulating the left ventricle to the whole heart [29]. Our method tackles the challenges posed by the increased physical size and geometric complexity of the heart, as well as the higher deformation and imbalanced nonlinearity during the cardiac cycle. Specifically, we use an elastodynamic equation with a heterogeneous hyperelastic constitutive law to describe the motion of the heart, where the myocardial tissues and the attached vessel roots are modeled using a Guccione transverse material model reinforced by cardiac fibers and an isotropic Mooney-Rivlin material, respectively. The partial differential equations in the model are spatially discretized using a finite element method based on a fully unstructured tetrahedral mesh to accurately capture the deformation of the entire heart. For the temporal discretization, we propose a two-level time stepping strategy: Initially, the entire cardiac cycle is divided into a sequence of baseline time steps, all with a uniform step size; Subsequently, an adaptive time-stepping technique is applied to each baseline time step to guarantee the convergence of the nonlinear solver. This approach not only achieves the desired temporal accuracy but also allows large time steps when possible. In addition, we develop an efficient and robust rule-based method for the fiber and active stress distribution over the whole heart for the entire cardiac cycle. Finally, to solve the resulting nonlinear algebraic systems at each time step, we adopt a Newton-Krylov-Schwarz method with an analytically computed Jacobian matrix and a restricted additive Schwarz preconditioner [30–33].

The rest of the paper is organized as follows. Section 2 describes a heterogeneous hyperelasticity model of a patient-specific human heart with four chambers. In Section 3, we describe the details of the methodology to solve the cardiac elastodynamic system, which includes a fully implicit finite element scheme, and an efficient Newton-Krylov-Schwarz method. Section 4 devotes to the generation of the cardiac fibers of the myocardial tissues, and the settings of the active stresses in an asynchronous form for the atria and ventricles. A series of numerical experiments are carried out in Section 5 to validate the proposed algorithm and analyze the convergence behavior of the numerical method with respect to temporal and spatial mesh sizes, the model and solver parameters. Some final remarks are given in Section 6.

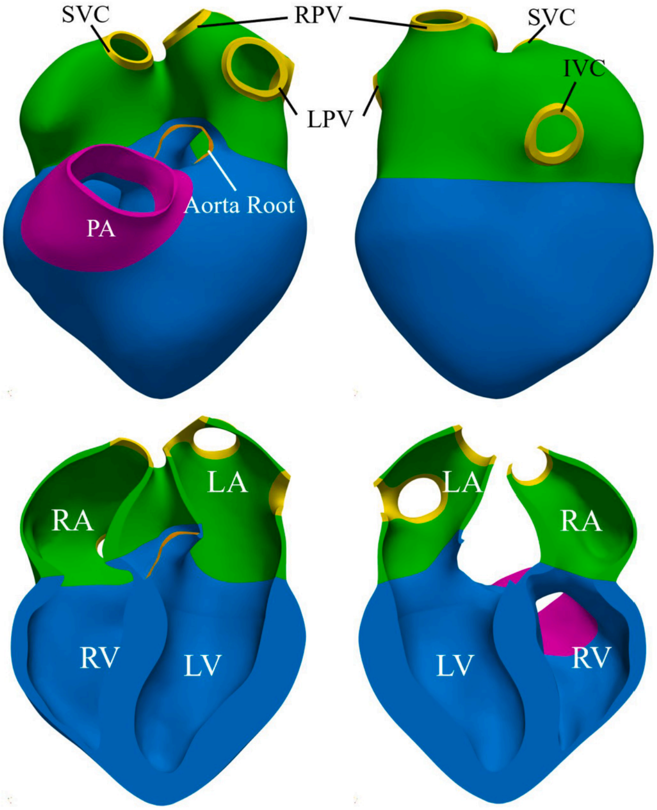

Fig. 1. Geometry of a patient-specific heart with four chambers. The myocardium is composed of the atria (green) and the ventricles (blue), while the roots of the vessels (SVC, IVC, RPV, LPV, and PA) are colored in yellow. The aorta root (orange) is an annular-like surface portion that connects the LV endocardium and the epicardium, which is used to apply a mandatory boundary condition. Through the cross-section views on the bottom row, one can see the thickness of the muscular wall and the shape of the chamber cavity. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

2. A heterogeneous hyperelastic model of a human heart

A normal human heart is composed of four chambers – the left atrium (LA), the right atrium (RA), the left ventricle (LV), and the right ventricle (RV) (Fig. 1). The elasticity of the heart is mainly determined by the myocardium, which is a thick muscular wall made up of self-contracting long, thin cells that are arranged in bundles known as cardiac fibers. These fibers spiral in a specific helical pattern that is crucial to the heart's ability to effectively and efficiently pump blood into the arteries. In addition, the heart chambers are connected to large vessels, including the superior vena cava (SVC), inferior vena cava (IVC), left pulmonary veins (LPV), right pulmonary veins (RPV), pulmonary artery (PA), and aorta (AO). The proximal root segments of these vessels are also considered as part of the heart wall.

Mathematically, the heart can be described as a continuum elastomer $\Omega \subset \mathbb{R}^3$ made of two types of materials: $\Omega = \Omega_{\text{myo}} \cup \Omega_{\text{ves}}$, the myocardium and the vessel roots, respectively. Inside the myocardium, the embedded fibers are densely distributed, each of which can be seen as a three-dimensional curve due to the large length-to-diameter ratio. The myocardial elasticity is anisotropic with respect to the fibers, as its local response along the longitudinal direction is different from a transverse direction, upon internal and/or external stresses. On the other hand, the vessel roots contain no fiber and are isotropic. Based on the model, to specify the transient kinetic status of the heart, one can refer to the finite elasticity theory. That is, at a given time t, for a material point $\mathbf{X} \in \Omega$, its displacement $\mathbf{d} = \mathbf{d}(\mathbf{X}, t)$ and velocity $\mathbf{v} = \mathbf{v}(\mathbf{X}, t)$ shall satisfy the following partial differential equations:

$$\begin{cases} \rho \dfrac{\partial \mathbf{v}}{\partial t} - \nabla \cdot \mathbf{P} = 0 \\ \qquad\qquad \dfrac{\partial \mathbf{d}}{\partial t} = \mathbf{v} \end{cases} \quad \text{in} \quad \Omega \times (0, T), \tag{2.1}$$

where $(0, T)$ is a time interval of interest, $\nabla\cdot$ is the divergence operator, $\mathbf{P}$ is the first Piola-Kirchhoff stress, ρ is the material density, and Ω can be regarded as the reference domain occupied by the heart in the stress-free state, without loss of generality.

Usually, it is more convenient to measure $\mathbf{P}$ according to the relation

$$\mathbf{P} = \mathbf{F}\mathbf{S}, \tag{2.2}$$

where $\mathbf{F} = \mathbf{I} + \nabla \mathbf{d}$ is the deformation gradient tensor, $\mathbf{I}$ is the identity tensor of the second rank, $\nabla \mathbf{d}$ is the gradient of the displacement, and $\mathbf{S}$ is the second Piola-Kirchhoff stress. In the context of cardiac elastodynamics, $\mathbf{S}$ can be defined as a superposition of two parts:

$$\mathbf{S} = \mathbf{S}_{\text{act}} + \mathbf{S}_{\text{pas}}, \tag{2.3}$$

where the active part $\mathbf{S}_{\text{act}}$ is raised by the autonomic shortenings of the cardiac fibers in Ω_{myo} along the longitudinal directions [34]:

$$\mathbf{S}_{\text{act}} = \begin{cases} T_a \dfrac{\mathbf{f} \otimes \mathbf{f}}{\sqrt{\mathbf{f} \cdot \mathbf{C}\mathbf{f}}} & \text{in} \quad \Omega_{\text{myo}}, \\ 0 & \text{in} \quad \Omega_{\text{ves}}. \end{cases} \tag{2.4}$$

Here, T_a represents the scalar magnitude of the active stress, $\mathbf{f}$ is the unit vector along the fiber direction in the reference configuration, and $\otimes$ denotes the tensor product operator. Besides, the passive part $\mathbf{S}_{\text{pas}}$ accounts for the elastic responses of all tissues in Ω upon deformations. Given the heart's ability to bear large deformations and return to the original position when the loading is removed, a hyperelastic material assumption is commonly used to define the specific form of $\mathbf{S}_{\text{pas}}$. That it, there exists a strain energy density $\mathcal{W}$, such that:

$$\mathbf{S}_{\text{pas}} = \frac{\partial \mathcal{W}}{\partial \mathbf{E}}, \tag{2.5}$$

where $\mathbf{E}$ is the Green-Lagrange strain tensor:

$$\mathbf{E} = \frac{1}{2}(\mathbf{F}^T \mathbf{F} - \mathbf{I}). \tag{2.6}$$

In this paper, because of the material heterogeneity, $\mathcal{W}$ is defined domain-wisely on Ω_{myo} and Ω_{ves}, and thereby as $\mathcal{W}_{\text{myo}}$ and $\mathcal{W}_{\text{ves}}$ in the following discussion.

In particular, we assume that the myocardial material satisfies the orthotropic Guccione constitutive law [35], by which the strain energy reads

$$\mathcal{W}_{\text{myo}} = \frac{C}{2}(e^Q - 1) + \frac{\kappa}{2}[\ln(J)]^2, \tag{2.7}$$

where $J = \det \mathbf{F}$, and

$$\begin{aligned} Q = b_f E_{11}^2 &+ b_t(E_{22}^2 + E_{33}^2 + E_{23}^2 + E_{32}^2) \\ &+ b_{fs}(E_{12}^2 + E_{21}^2 + E_{13}^2 + E_{31}^2), \end{aligned} \tag{2.8}$$

each E_{ij} denotes a tensorial component of $\mathbf{E}$ in a local orthonormal coordinate system, in which the first coordinate axis aligns with the fiber direction, and C, b_f, b_t, and b_{fs} are model parameters that specify the bulk and axial stiffness of the material, respectively. The affiliated vessel roots are assumed to satisfy the two-term Mooney-Rivlin (MR) model:

$$\mathcal{W}_{\text{ves}} = C_1(\bar{I}_1 - 3) + C_2(\bar{I}_2 - 3) + \frac{\kappa}{2}[\ln(J)]^2, \tag{2.9}$$

where $\bar{I}_1$ and $\bar{I}_2$ are the first and second principal invariants of the unimodular component of the right Cauchy-Green strain tensor $\mathbf{C} = \mathbf{F}^T\mathbf{F}$:

$$\bar{I}_1 = J^{-2/3}\text{trace}(\mathbf{C}), \quad \bar{I}_2 = \frac{J^{-4/3}}{2}\Big((\text{trace}(\mathbf{C}))^2 - \text{trace}(\mathbf{C}^2)\Big), \tag{2.10}$$

C_1 and C_2 are the MR-model parameters that characterize the material shear modulus. Note that, a penalty term is introduced for both material models to enforce a nearly-incompressible constraint, in which κ denotes the corresponding penalty parameter.

Based on the above assumptions, and via straightforward calculations, it can be shown that $\mathbf{S}_{\text{pas}}$ has the following analytical form:

$$\mathbf{S}_{\text{pas}} = \begin{cases} Ce^Q \mathbf{A} * \mathbf{E} + \kappa \ln J \mathbf{C}^{-1}, & \text{in} \quad \Omega_{\text{myo}}, \\ c_1 \mathbf{I} + c_2 \mathbf{C} + c_3 \mathbf{C}^{-1} + \kappa \ln J \mathbf{C}^{-1}, & \text{in} \quad \Omega_{\text{ves}}, \end{cases} \tag{2.11}$$

where $*$ denotes a component-wise multiplication operator, $\mathbf{A}$ is a coefficient matrix

$$\mathbf{A} = \begin{pmatrix} b_f & b_{fs} & b_{fs} \\ b_{fs} & b_t & b_t \\ b_{fs} & b_t & b_t \end{pmatrix},$$

and

$$c_1 = 2J^{-2/3}(C_1 + C_2\bar{I}_1), \quad c_2 = -2C_2J^{-4/3}, \quad c_3 = -\frac{2}{3}(C_1\bar{I}_1 + 2C_2\bar{I}_2).$$

In addition, to ensure a unique solution in the problem domain $\Omega \times (0, T)$, the system of partial differential equations (2.1) is complemented by a set of initial and boundary conditions in the following form:

$$\mathbf{d} = 0 \quad \text{on} \quad \Gamma_d \qquad \text{and} \qquad \mathbf{PN} = \mathbf{g} \quad \text{on} \quad \Gamma_n, \tag{2.12}$$

and

$$\mathbf{d}(0, \mathbf{X}) = 0 \qquad \text{and} \qquad \mathbf{v}(0, \mathbf{X}) = 0, \tag{2.13}$$

where Γ_d and Γ_n denote the Dirichlet and Neumann boundaries and $\Gamma_d \cup \Gamma_n = \partial\Omega$, $\mathbf{N}$ is the outward unit normal vector of Γ_n, and $\mathbf{g}$ represents an external traction force applied on Γ_n. In this paper, the aorta orifice is fixed as a homogeneous Dirichlet boundary, and the inner surfaces of all chambers are regarded as free walls. Additionally, inspired by the approach outlined in [14], we consider two types of Neumann conditions on the remaining parts of the heart surfaces, which mimic the motion resistances exerted by the major vessels and the surrounding organs on the heart. First, the epicardium is subject to a loading that is normal to its surface, taking into account the projection of the displacement and velocity along its outward normal direction. Interested readers may refer to the diagrams in [14] for illustration. On the other hand, the surface loading on the cross-sections of the vessel roots is simplified as a superposition of the opposite of the scaled displacement and velocity. More specifically, a surface force being applied on the epicardial surface according to:

$$\mathbf{g} = -(k_d \mathbf{d} \cdot \mathbf{N} + k_v \mathbf{v} \cdot \mathbf{N})\mathbf{N}, \tag{2.14}$$

where k_d and k_v are constants, $\mathbf{N}$ is the unit outward normal on the epicardium, measured in the reference configuration. Besides, on the cross-section of each vessel root, the heart is subject to a loading given by:

$$\mathbf{g} = -(k_d \mathbf{d} + k_v \mathbf{v}). \tag{2.15}$$

These conditions can be regarded as a series of springs and dashpots that are attached to the surface of the heart, such that its movements are impeded. Depending on the type, either (2.14) or (2.15) is substituted into (2.12) on an exterior surface of the heart.

3. Methodology

To solve (2.1) numerically, we first propose a fully implicit discretization in time with varying time step sizes and a finite element discretization on a fully unstructured tetrahedral mesh in space. Then, at each time step, a highly scalable domain decomposition algorithm is invoked to solve the large sparse nonlinear algebraic system of equations.

3.1. A fully implicit finite element discretization

Let the temporal interval $(0, T)$ be subdivided into N_{T} steps $\{t_n | t_0 = 0,\ t_n = t_{n-1} + \Delta t,\ n = 1, 2, \dots, N_{\mathrm{T}}\}$ where Δt denotes a uniform time step size such that $\Delta t = T / N_{\mathrm{T}}$. Then, corresponding to (2.1), we form the following variational problem [36]: Find $(\mathbf{d}, \mathbf{v}) \in \mathcal{D} \times \mathcal{V}$, such that

$$\mathcal{F}((\mathbf{d}, \mathbf{v}), (\mathbf{p}, \mathbf{q})) = 0 \qquad \forall (\mathbf{p}, \mathbf{q}) \in \mathcal{D} \times \mathcal{V}, \tag{3.1}$$

where $\mathcal{D} = \left\{ \mathbf{w} \in [H^1(\Omega)]^3 \mid \mathbf{w} = 0 \text{ on } \Gamma_d \right\}$ and $\mathcal{V} = [H^1(\Omega)]^3$ are the vectorial Sobolev spaces, $\mathbf{p}$ and $\mathbf{q}$ represent the corresponding test functions, and $\mathcal{F}$ is defined by

$$\begin{aligned} \mathcal{F}((\mathbf{d}, \mathbf{v}), (\mathbf{p}, \mathbf{q})) = & \rho \int_\Omega \frac{\partial \mathbf{v}}{\partial t} \cdot \mathbf{p} \mathrm{d}\mathbf{X} + \int_\Omega \mathbf{P} : \nabla \mathbf{p} \mathrm{d}\mathbf{X} - \int_{\Gamma_n} \mathbf{g} \cdot \mathbf{p} \mathrm{d}\Gamma \\ & + \int_\Omega \frac{\partial \mathbf{d}}{\partial t} \cdot \mathbf{q} \mathrm{d}\mathbf{X} - \int_\Omega \mathbf{v} \cdot \mathbf{q} \mathrm{d}\mathbf{X}. \end{aligned} \tag{3.2}$$

Let $\mathcal{T}_h$ be a conformal tetrahedral mesh that covers Ω, based on which we introduce $\mathcal{D}_h \subset \mathcal{D}$ and $\mathcal{V}_h \subset \mathcal{V}$ two piecewise linear continuous Galerkin finite element spaces with basis functions $\{\boldsymbol{\psi}_h\}$ and $\{\boldsymbol{\varphi}_h\}$, respectively. Then, by denoting $(\mathbf{d}_h^n, \mathbf{v}_h^n) = (\mathbf{d}_h(t_n, \cdot), \mathbf{v}_h(t_n, \cdot)) \in \mathcal{D}_h \times \mathcal{V}_h$ as the finite element solutions at the n^{th} time step, their time derivatives can be approximated by:

$$\frac{\partial \mathbf{d}_h^n}{\partial t} \approx \frac{\mathbf{d}_h^n - \mathbf{d}_h^{n-1}}{\Delta t} \quad \text{and} \quad \frac{\partial \mathbf{v}_h^n}{\partial t} \approx \frac{\mathbf{v}_h^n - \mathbf{v}_h^{n-1}}{\Delta t} \quad \text{for} \quad n = 1, 2, \dots, N_{\mathrm{T}}, \tag{3.3}$$

where $\mathbf{d}_h^0$ and $\mathbf{v}_h^0$ denote the given initial values. Consequently, substituting (3.3) into (3.2), the sequence of $\{(\mathbf{d}_h^n, \mathbf{v}_h^n)\}$ can be found by solving the following finite dimensional problems for $n = 1, 2, \dots, N_{\mathrm{T}}$, in a consecutive order:

$$\mathcal{F}_h((\mathbf{d}_h^n, \mathbf{v}_h^n), (\boldsymbol{\psi}_h, \boldsymbol{\varphi}_h); (\mathbf{d}_h^{n-1}, \mathbf{v}_h^{n-1}), \Delta t) = 0, \text{ for all } (\boldsymbol{\psi}_h, \boldsymbol{\varphi}_h) \in \mathcal{D}_h \times \mathcal{V}_h, \tag{3.4}$$

where

$$\begin{aligned}
&\mathcal{F}_h((\mathbf{d}_h^n,\mathbf{v}_h^n),(\boldsymbol{\psi}_h,\boldsymbol{\varphi}_h);(\mathbf{d}_h^{n-1},\mathbf{v}_h^{n-1}),\Delta t)\\
=&\rho\int_\Omega\left(\mathbf{v}_h^n-\mathbf{v}_h^{n-1}\right)\cdot\boldsymbol{\psi}_h\mathrm{d}\mathbf{X}+\Delta t\int_\Omega\mathbf{P}(\mathbf{d}_h^n):\nabla\boldsymbol{\psi}_h\mathrm{d}\mathbf{X}-\Delta t\int_{\Gamma_n}\mathbf{g}\cdot\boldsymbol{\psi}_h\mathrm{d}\Gamma\\
&+\int_\Omega\left(\mathbf{d}_h^n-\mathbf{d}_h^{n-1}\right)\cdot\boldsymbol{\varphi}_h\mathrm{d}\mathbf{X}-\Delta t\int_\Omega\mathbf{v}_h^n\cdot\boldsymbol{\varphi}_h\mathrm{d}\mathbf{X}.
\end{aligned}\tag{3.5}$$

Note that we include $(\mathbf{d}_h^{n-1},\mathbf{v}_h^{n-1})$ and Δt in (3.4) to emphasize that the definition of the finite element functional $\mathcal{F}_h$ relies on the solution of the previous time step and the time step size.

(3.4) is a large, sparse, and highly nonlinear system of algebraic equations, whose degree of freedoms (dofs) are nodal values $(\mathbf{d}_h^n,\mathbf{v}_h^n)$ that are collectively arranged in the following order

$$\{\cdots,d_{i,X}^n,d_{i,Y}^n,d_{i,Z}^n,v_{i,X}^n,v_{i,Y}^n,v_{i,Z}^n,\cdots\}\qquad i=1,2,\ldots,N_{\mathrm{V}},\tag{3.6}$$

where N_{V} is the total number of mesh points, and $d_{i,\{X,Y,Z\}}^n$ and $v_{i,\{X,Y,Z\}}^n$ are the $\{X,Y,Z\}$-component of $\mathbf{d}_h^n$ and $\mathbf{v}_h^n$ at the i-th mesh point, respectively. In other words, the mesh points are traversed one-by-one, and all corresponding nodal values of a point are placed together to form a 6×1 block-vector. For easing the notations, (3.4) is re-written as

$$\mathcal{F}(\mathbf{x}^n;\mathbf{x}^{n-1},\Delta t)=0,\tag{3.7}$$

where $\mathbf{x}^n$ denotes the solution vector (i.e., nodal values) to be solved at the n^{th} time step, and $\mathbf{x}^{n-1}$ is the solution of the previous step which is referred to as the initial data of (3.7) in the rest of the paper.

3.2. An inexact Newton-Krylov method with Schwarz preconditioned analytic Jacobian

The system of nonlinear algebraic equations (3.7) is solved using an inexact Newton method, in each of the inexact Newton iteration, (3.7) is linearized to form a Jacobian system, and an approximate solution is obtained using a Schwarz preconditioned GMRES method. The algorithm is usually referred to as the Newton-Krylov-Schwarz (NKS) method, it has been successfully applied to elastodynamic simulations of a left ventricle in our previous work [29], but unfortunately, the algorithm doesn't always work for the case of a whole heart. The algorithm is summarized in **Algorithm 1** below, and a modification of (3.7) is introduced later to ensure the convergence of the NKS method.

Algorithm 1 A Newton-Krylov-Schwarz for solving (3.7) with the initial data $\mathbf{x}^{n-1}$ and the time step size Δt as input. Output $\mathbf{x}^n$.

```
Let $m=0$ and set the initial guess $\mathbf{x}_0=\mathbf{x}^{n-1}$.
while $\|\mathcal{F}(\mathbf{x}_m)\|\geq\max\{\mathrm{a_N},\mathrm{r_N}\|\mathcal{F}(\mathbf{x}_0)\|\}$ do
    Calculate the Jacobian matrix analytically
        $\mathcal{J}(\mathbf{x}_m)=\frac{\partial\mathcal{F}}{\partial\mathbf{x}}(\mathbf{x}_m).$
    Find an inexact Newton direction $\delta\mathbf{x}$ by approximately solving the right-preconditioned system:
        $\mathcal{J}(\mathbf{x}_m)\mathcal{M}_m^{-1}\mathcal{M}_m\delta\mathbf{x}=-\mathcal{F}(\mathbf{x}_m),$   (3.8)
    such that
        $\|\mathcal{J}(\mathbf{x}_m)\delta\mathbf{x}+\mathcal{F}(\mathbf{x}_m)\|\leq\max\{\mathrm{a_L},\mathrm{r_L}\|\mathcal{F}(\mathbf{x}_m)\|\},$
    where $\mathcal{M}_m^{-1}$ is a restricted additive Schwarz (RAS) preconditioner corresponding to $\mathcal{J}(\mathbf{x}_m)$.
    Find $\alpha$ via a cubic line search method.
    Let $m=m+1$, update $\mathbf{x}_m=\mathbf{x}_{m-1}+\alpha\delta\mathbf{x}$.
    if $m>m_{\max}$ then
        Terminate the algorithm and report it is not convergent.
    end if
end while
Output $\mathbf{x}^n=\mathbf{x}_m$.
```

Here, $\mathrm{a_L}$, $\mathrm{r_L}$, $\mathrm{a_N}$ and $\mathrm{r_N}$ are pre-chosen small positive values used to determine whether the GMRES and Newton iterations have converged, respectively. The specific values of these parameters will be provided in the numerical experiments section. In addition, $m_{\max}$ specifies the maximal allowed number of Newton iterations, and the solution process is considered non-converged for $m>m_{\max}$. Later, we will introduce an adaptive temporal refinement scheme to handle the non-converged cases. **Algorithm 1** is reliant on two essential ingredients to ensure its effective and efficient implementation: the construction of the Jacobian matrix $\mathcal{J}$ and the preconditioner $\mathcal{M}^{-1}$ that are crucial for the convergence, and the robustness of the NKS algorithm. In the following, we present the structure of $\mathcal{J}$, while construction of $\mathcal{M}^{-1}$ will be discussed in the next subsection.

In many applications that use the Newton-Krylov family of methods (e.g. [37–39]), $\mathcal{J}$ is computed through a finite difference approximation that involves a large number of evaluations of the nonlinear function $\mathcal{F}$ with small perturbations on the input vector

x. Although it is easy to program, this approach is slow and lacks the level of robustness that we need. In this paper, the Jacobian matrix is constructed analytically, and arranged in a point-block fashion to match the ordering of **x** in (3.6):

$$\mathcal{J} = [\mathcal{J}_{ij}] \qquad \text{for} \quad i, j = 1, 2, \dots, N_{\mathrm{V}}, \tag{3.9}$$

in which each $\mathcal{J}_{ij}$ is a 6×6 block

$$\mathcal{J}_{ij} = \begin{bmatrix} \frac{\partial \mathcal{F}_{d_X,i}}{\partial d_{j,X}} & \frac{\partial \mathcal{F}_{d_X,i}}{\partial d_{j,Y}} & \frac{\partial \mathcal{F}_{d_X,i}}{\partial d_{j,Z}} & \rho M_{ij} & 0 & 0 \\ \frac{\partial \mathcal{F}_{d_Y,i}}{\partial d_{j,X}} & \frac{\partial \mathcal{F}_{d_Y,i}}{\partial d_{j,Y}} & \frac{\partial \mathcal{F}_{d_Y,i}}{\partial d_{j,Z}} & 0 & \rho M_{ij} & 0 \\ \frac{\partial \mathcal{F}_{d_Z,i}}{\partial d_{j,X}} & \frac{\partial \mathcal{F}_{d_Z,i}}{\partial d_{j,Y}} & \frac{\partial \mathcal{F}_{d_Z,i}}{\partial d_{j,Z}} & 0 & 0 & \rho M_{ij} \\ M_{ij} & 0 & 0 & -\Delta t M_{ij} & 0 & 0 \\ 0 & M_{ij} & 0 & 0 & -\Delta t M_{ij} & 0 \\ 0 & 0 & M_{ij} & 0 & 0 & -\Delta t M_{ij} \end{bmatrix}. \tag{3.10}$$

In the matrix, the component functions $\mathcal{F}_{d_{\{X,Y,Z\}},i}$ are defined by

$$\begin{aligned} \mathcal{F}_{d_X,i} &= \Delta t \int\limits_{\Omega} \left(P_{11} \frac{\partial \psi_i}{\partial X} + P_{12} \frac{\partial \psi_i}{\partial Y} + P_{13} \frac{\partial \psi_i}{\partial Z} \right) \mathrm{d}\mathbf{X} - \Delta t \int\limits_{\Gamma_n} g_X \psi_i \mathrm{d}\Gamma, \\ \mathcal{F}_{d_Y,i} &= \Delta t \int\limits_{\Omega} \left(P_{21} \frac{\partial \psi_i}{\partial X} + P_{22} \frac{\partial \psi_i}{\partial Y} + P_{23} \frac{\partial \psi_i}{\partial Z} \right) \mathrm{d}\mathbf{X} - \Delta t \int\limits_{\Gamma_n} g_Y \psi_i \mathrm{d}\Gamma, \\ \mathcal{F}_{d_Z,i} &= \Delta t \int\limits_{\Omega} \left(P_{31} \frac{\partial \psi_i}{\partial X} + P_{32} \frac{\partial \psi_i}{\partial Y} + P_{33} \frac{\partial \psi_i}{\partial Z} \right) \mathrm{d}\mathbf{X} - \Delta t \int\limits_{\Gamma_n} g_Z \psi_i \mathrm{d}\Gamma, \end{aligned} \tag{3.11}$$

and M_{ij} is defined by

$$M_{ij} = \int\limits_{\Omega} \psi_i \psi_j \mathrm{d}\mathbf{X}, \tag{3.12}$$

where $P_{ab}(a, b = 1, 2, 3)$ and $g_{\{X,Y,Z\}}$ are the components of **P** and **g**, and ψ_i is the (scalar) nodal basis corresponding to the i-th mesh point. The detailed analytic expressions of (3.10) and (3.11) can be found in our previous paper [29] for the Guccione material, where the superiority of using the analytical Jacobian over the finite difference approximation is also demonstrated.

Because of the four-chamber geometry and the highly nonlinear stress tensor **P**, these large-scale, non-symmetric, sparse linear systems (3.8) are quite difficult to solve. Consequently, finding the inexact Newton direction $\delta\mathbf{x}$ in each Newton step becomes a challenging task. A well-designed preconditioner $\mathcal{M}^{-1}$ is crucial to ensure the convergence of GMRES. Below, we propose a restricted additive Schwarz (RAS) preconditioner specifically tailored for the Jacobian system.

Let the finite element mesh $\mathcal{T}_h$ be partitioned into n_p non-overlapping subdomains: $\mathcal{T}_h = \cup_{i=1}^{n_p} \mathcal{T}_i$, as in Fig. 2 for an illustration with 120 subdomains. In our numerical experiments, n_p is chosen to be the same as the number of processor cores that we use to run the tests. For each subdomain $\mathcal{T}_i$, an extended subdomain $\mathcal{T}_i^\delta$ is created by including δ neighboring layers of elements. Solution vectors are defined for both the whole mesh and subdomains, with dimensions D, D_i, and D_i^δ, respectively. Let $\mathcal{J}_i$ be the restriction of $\mathcal{J}$ on the overlapping subdomain $\mathcal{T}_i^\delta$, the RAS preconditioner $\mathcal{M}^{-1} : \mathbb{R}^D \to \mathbb{R}^D$ is defined as

$$\mathcal{M}^{-1} = \sum_{i=1}^{n_p} (\mathcal{R}_i^0)^T \mathcal{J}_i^{-1} \mathcal{R}_i^\delta, \tag{3.13}$$

where $\mathcal{R}_i^0 : \mathbb{R}^D \to \mathbb{R}^{D_i}$ and $\mathcal{R}_i^\delta : \mathbb{R}^D \to \mathbb{R}^{D_i^\delta}$ are restriction operators that map a global vector $\mathbf{u} \in \mathbb{R}^D$ onto the corresponding non-overlapping and overlapping subdomains. The transposes $(\mathcal{R}_i^0)^T$ and $(\mathcal{R}_i^\delta)^T$ are used as the extension operators, respectively. $\mathcal{J}_i^{-1}$ denotes a subdomain inverse and its product with a vector $\mathbf{u}_i^\delta$ is obtained by an approximate solution $\mathbf{w}_i^\delta$ of the subdomain linear system of equations:

$$\mathcal{J}_i \mathbf{w}_i^\delta = \mathbf{u}_i^\delta, \tag{3.14}$$

which is solved by an incomplete-LU (ILU) factorization method with an appropriate level of fill-in.

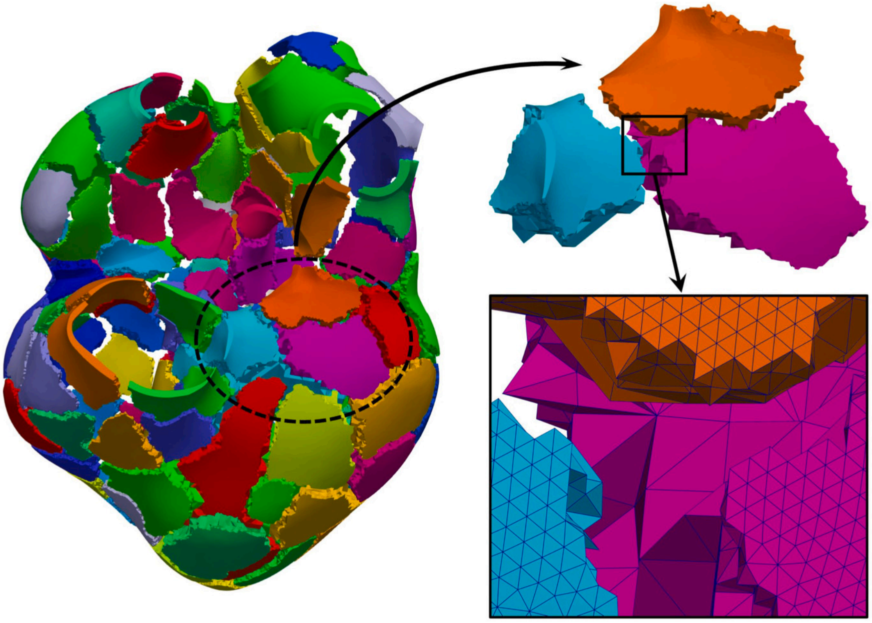

Fig. 2. A partition of a mesh into 120 subdomains.

3.3. A two-level adaptive time stepping scheme

The discussion in the previous sections assumes a uniform time step size is used for the entire cardiac cycle, however, this approach may need to be modified for certain time steps, because some portions of the heart may experience significantly larger deformation at certain moments while other portion remains relatively unchanged, leading to an un-balanced distribution of nonlinearity across the problem domain. This poses considerable challenges for the nonlinear solver to converge in a reasonable number of iterations. In this section, we propose a two-pronged method to handle the time discretization of the elastodynamic system such that both the time accuracy and the solver convergence can be guaranteed. First, based on computational experiences we select a uniform time step size that we refer to as the baseline time step size that offers a reasonable time accuracy throughout the cardiac cycle, but doesn't concern the convergence of the linear and nonlinear solvers. Second, within a baseline time step, we adaptively refine the time step size so that the nonlinear solver converges with a reasonable small number of iterations. To be more specific, we do:

1. The cardiac cycle $(0, T)$ is divided into N_{T} steps, each of which is referred to as a *baseline* time step, and it is selected so that the desired time accuracy is achieved at all the time steps $t_B^n = n \times \Delta t_B (n = 1, \cdots, N_{\mathrm{T}})$ covering the entire cardiac cycle even if no temporal refinement is performed.
2. Within each baseline time interval $[t_B^n, t_B^{n+1}]$, **Algorithm 1** is used to solve the nonlinear system (3.7) with $\mathbf{x}^{n-1}$ as the initial data and Δt_B as the time step size. If the algorithm successfully converges, we proceed to solve the next baseline time interval. However, if **Algorithm 1** fails to converge within the desired number of iterations, the interval is evenly split into two smaller sub-intervals $[t_B^n, t_B^{n+1/2}]$ and $[t_B^{n+1/2}, t_B^{n+1}]$, and the algorithm is applied to solve each consecutively; the solution from the first interval serves as the initial data for the problem on the second interval. This procedure is recursively applied to any reduced time intervals whenever **Algorithm 1** fails. We refer to a time step, whether it is a baseline step or a reduced step, at which convergence is achieved as an *effective* time step, and each a baseline time step may require multiple effective steps along with several unsuccessful trials.

For clarity, the established rules are summarized in **Algorithm 2** for solving the nonlinear system (3.7) in any time interval with provided initial data and time step size.

Algorithm 2 A temporal adaptive refinement algorithm for solving the nonlinear system $\mathcal{F}(\mathbf{x}^{\mathrm{c}}; \mathbf{x}^{\mathrm{p}}, \Delta t) = 0$ for $\mathbf{x}^{\mathrm{c}}$ with the initial data $\mathbf{x}^{\mathrm{p}}$ and the time step size Δt.

```
1: Solve $\mathcal{F}(\mathbf{x}^{\mathrm{c}}; \mathbf{x}^{\mathrm{p}}, \Delta t) = 0$ to obtain $\mathbf{x}^{\mathrm{c}}$ using Algorithm 1.
2: if Algorithm 1 in Step 1 does not converge within the desired number of Newton iterations. then
3:     Set $\Delta t = \Delta t/2$.
4:     Solve $\mathcal{F}(\mathbf{x}^{\mathrm{m}}; \mathbf{x}^{\mathrm{p}}, \Delta t) = 0$ by Algorithm 2 to obtain an intermediate solution $\mathbf{x}^{\mathrm{m}}$.
5:     Solve $\mathcal{F}(\mathbf{x}^{\mathrm{c}}; \mathbf{x}^{\mathrm{m}}, \Delta t) = 0$ by Algorithm 2 to obtain the updated solution $\mathbf{x}^{\mathrm{c}}$.
6: end if
```

Based on **Algorithm 2**, a two-level solution scheme for the entire cardiac cycle is summarized in **Algorithm 3** which includes: (1) the first level time integration over all baseline time steps, and (2) the second level with an adaptive refinement method within each baseline step.

Algorithm 3 A two-level time integration strategy for simulating elastodynamics of the heart during a cardiac cycle $(0, T)$.

```
Set $\mathbf{x}^0 = 0$ according to the initial condition.
for baseline time step $n = 1, 2, \ldots, N_T$ do
    Set $\Delta t_B \equiv T/N_T$ for the temporal discretization.
    Solve $\mathcal{F}(\mathbf{x}^n; \mathbf{x}^{n-1}, \Delta t_B) = 0$ to obtain $\mathbf{x}^n$ using Algorithm 2.
end for
```

Remark 3.1. We remark that using a small, uniform time step size is possible to satisfy the accuracy requirement and also the convergence of the algebraic solver, but the overall computing time may be unacceptably long.

Remark 3.2. The proposed temporal adaptive refinement strategy is different from most adaptive methods that are often based on some error estimate, not based on the failure of the nonlinear solver.

4. Cardiac fibers

Cardiac fibers play an essential role in the heart's functionality as they are bundles of specialized muscle cells that are responsible for the contraction of the heart and supporting its geometric configuration. The cardiac fibers exhibit a significant helix pattern in both the ventricles and atria. In the ventricles, the cardiac fibers take a spiral pattern around the heart's long axis, creating a twisting motion during contraction to pump blood into the aorta efficiently. In the atria, the cardiac fibers are arranged in a double helix formation around the atrial opening toward the ventricles, ensuring the effective transfer of blood from the atria to the ventricles. This section first proposes a methodology for synthesizing fibers in a patient-specific heart that reflect the essential anatomical features of the helix patterns. Then, we introduce a scheme to determine the transient magnitude of active stresses generated by the fibers in a domain-wise form.

4.1. Generating cardiac fibers

The helix pattern of the cardiac fibers is a crucial factor in the elastodynamics of the heart. However, given the complex geometry of a human heart, finding the *in-vivo* atlas of its myocardial fibers is very difficult, if not impossible. Mathematically, the fibers can be viewed as a continuously distributed three-dimensional vector field $\mathbf{f}$ over the heart domain and are used as material parameters to define the elasticity constitutive model. In the numerical method, a unit vector is required at each mesh point that aligns with the local fiber tangent direction. In this paper we employ an efficient rule-based method to accomplish this task. This approach generalizes the interpolation scheme introduced in a previous study [40], which was initially proposed to approximate the fiber distribution in the two ventricles of the lower half of a heart.

We start by defining a set of six potential functions: $\{\phi_{\mathrm{ab}}, \phi_{\mathrm{epi}}, \phi_{\mathrm{ra}}, \phi_{\mathrm{rv}}, \phi_{\mathrm{la}}, \phi_{\mathrm{lv}}\}$ (Fig. 3), each of which solves a Laplacian equation

$$\Delta \phi = 0 \tag{4.1}$$

that satisfies the following boundary condition, respectively:

$$\begin{cases} \phi_{\mathrm{ab}} & : 1 \text{ on the roots of SVC and RPV}, 0 \text{ on the apex.} \\ \phi_{\mathrm{epi}} & : 1 \text{ on epicardium}, 0 \text{ on all chamber endocardia.} \\ \phi_{\mathrm{ra}} & : 1 \text{ on RA endocardium}, 0 \text{ on the epicardium and other endocardia.} \\ \phi_{\mathrm{rv}} & : 1 \text{ on RV endocardium}, 0 \text{ on the epicardium and other endocardia.} \\ \phi_{\mathrm{la}} & : 1 \text{ on LA endocardium}, 0 \text{ on the epicardium and other endocardia.} \\ \phi_{\mathrm{lv}} & : 1 \text{ on LV endocardium}, 0 \text{ on the epicardium and other endocardia.} \end{cases} \tag{4.2}$$

The potentials are utilized to specify the local azimuths at any location on the boundary surfaces through their gradients. Specifically, $\nabla\phi_{\mathrm{ab}}$ is considered as the local longitudinal axis, representing the top-bottom direction, while other gradients are considered as the local transmural directions on the corresponding surfaces. In particular, $\nabla\phi_{\mathrm{epi}}$ specifies the transmural direction on the epicardium, on which the fiber direction $\mathbf{f}(\mathbf{X})$ at any point $\mathbf{X}$ is computed as follows:

1. Let $\mathbf{e}_1 = -\nabla\phi_{\mathrm{ab}}(\mathbf{X})$ and $\mathbf{e}_2 = -\nabla\phi_{\mathrm{epi}}(\mathbf{X})$.
2. Calculate $\mathbf{e}_0 = (\mathbf{e}_1 \times \mathbf{e}_2)/\|\mathbf{e}_1 \times \mathbf{e}_2\|$.
3. The fiber direction is obtained by rotating $\mathbf{e}_0$ toward $\mathbf{e}_1$ by an angle of α: $\mathbf{f}(\mathbf{X}) = \mathbf{e}_0 \cos\alpha + \mathbf{e}_1 \sin\alpha$.

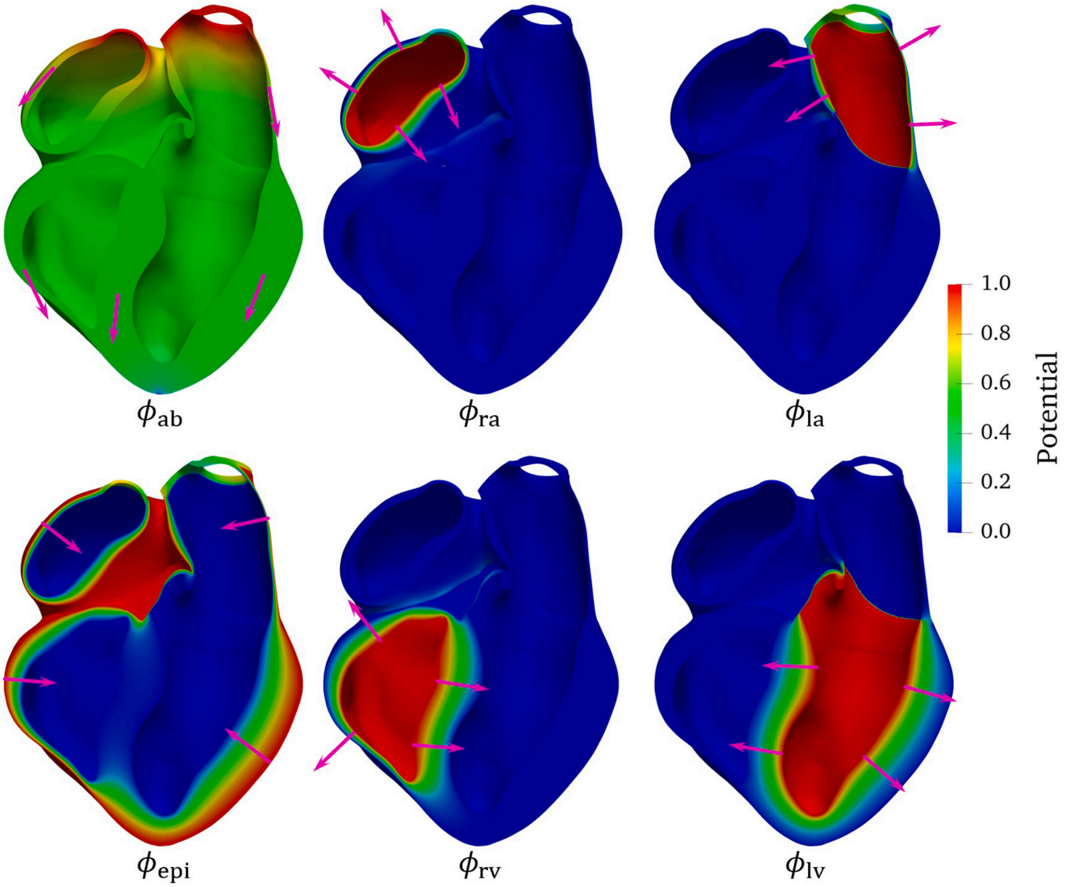

Fig. 3. Cross-section view of the Laplacian potentials that solve the boundary value problems according to (4.1) and (4.2), their gradients at some representative points are indicated by magenta arrows.

Likewise, the process of determining the fiber direction on each chamber's endocardium is done by repeating the above steps, in which ϕ_{epi} is replaced by the corresponding potential $\phi_{\{\text{la, ra, lv, rv}\}}$ to calculate $\mathbf{e}_2$, respectively. On each surface, the rotation angle α can be varied to match the synthetic fibers with the real ones as accurately as possible.

Provided above, the fiber vector field $\mathbf{f}$ over the entire heart domain is determined through interpolations. This is achieved by solving three additional Laplacian boundary value problems, each for a component of $\mathbf{f}$. The corresponding surface value is used as a mandatory condition in each problem. To prepare for this task and ensure consistency, all Laplacian problems are resolved before performing the primary task of cardiac elastodynamics. These problems are discretized using the same finite element space and mesh as the main task, and the conjugate gradient method is used to solve them. An additive Schwarz preconditioner is employed with the incomplete Cholesky factorization as the subdomain solver. In this study, a fiber distribution is generated for numerical experiments using the aforementioned scheme (Fig. 4). The surface rotation angles α are set as follows: 70° for the LV endocardium, 140° for the RV endocardium, 110° for the LA endocardium, 140° for the RA endocardium, 70° for the epicardium of the ventricles, and 0° for the epicardium of the atria.

Remark 4.1. In a normal heart, the atrial walls are significantly thinner than the ventricular walls, and the fiber distribution in the atria is more diverse. Therefore, a mapping from a template is often used to depict atrial fiber layout instead of a rule-based method. Nevertheless, since this paper's primary focus is investigating the numerical algorithm, we consider the proposed rule-based method an acceptable alternative for our purposes.

4.2. Active stresses

The elastodynamics of the heart is driven by active stresses created by the cardiac fibers. To formulate the stresses, in addition to the fiber distribution layout, it is necessary to know the stress magnitude T_a of the fibers. For this purpose, we define T_a for all fibers in the two atria and two ventricles, respectively. Specifically, we use the magnitude-time curve of the active stress produced by a single fiber as a template and shift its phase accordingly to obtain the time-variant curve of the stress magnitude for all fibers in the atria and ventricles.

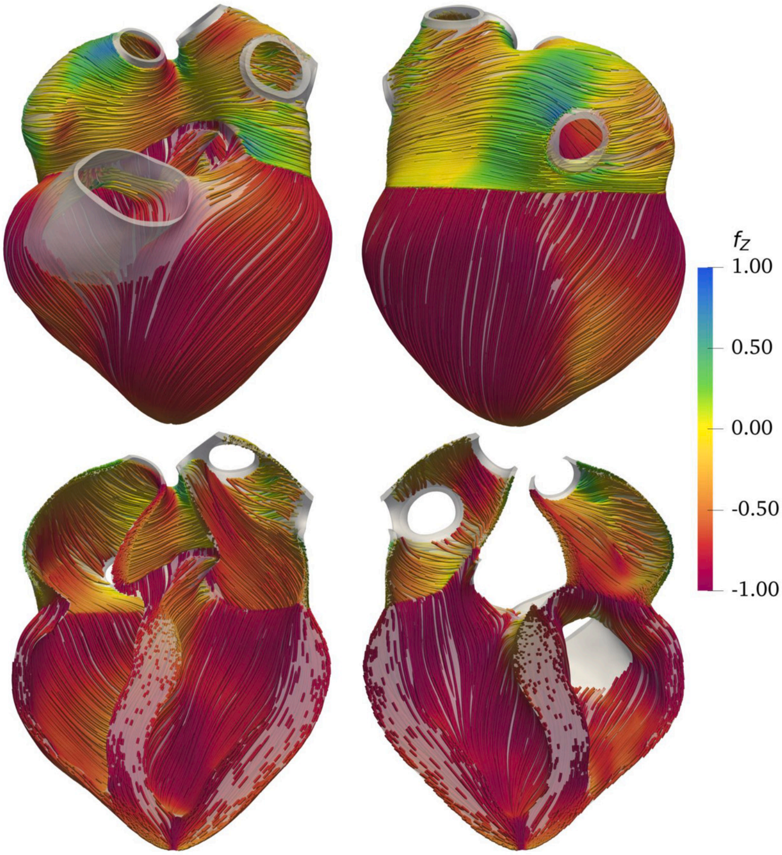

Fig. 4. Distribution of the cardiac fibers over the myocardial domain of the patient-specific heart, which are viewed from front (upper left) and back (upper right), with two corresponding cross-section views (lower left and right), respectively. For better visualization, the fibers are colored according to the magnitude of the Z-(vertical) axis component.

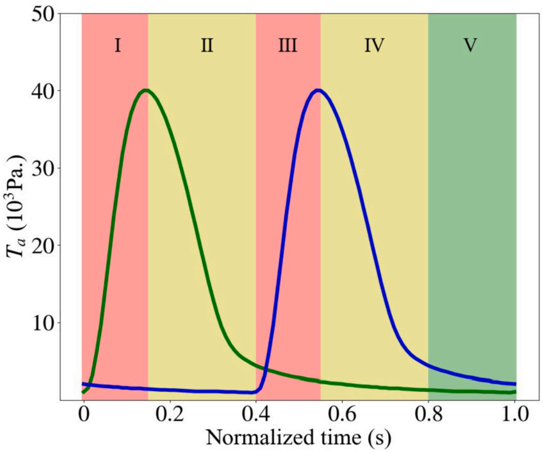

Fig. 5. The magnitude of the active stresses generated by fibers in the atria and ventricles, respectively.

As shown in Fig. 5, we normalize the time for one cardiac cycle to be one second, during which, T_a for the atrial fibers, denoted as the green curve, starts at a low value and then quickly increases to the maximal value $T_{\max}$. Afterward, it decreases to roughly one-eighth of $T_{\max}$ at 0.4 s. After that, T_a decreases gradually at a low rate for the rest of the cycle, and returns to the initial state at the end of the cycle. T_a of the ventricular fibers also varies, represented by the blue curve that is obtained by periodically shifting the green (atrial) curve by 0.4 s. Considering both together, a cardiac cycle is split into five stages (I, II, III, IV, V), and the changes of T_a in each stage are summarized in Table 1.

Table 1
Changing mode of T_a in the atrial and the ventricular domain in each stage of one cardiac cycle, in which "incr." and "decr." represent increasing and decreasing, respectively.

Stage Time (s)	I [0, 0.15]	II [0.15, 0.4]	III [0.4, 0.55]	IV [0.55, 0.8]	V [0.8, 1.0]
Atria	Fast incr.	Fast decr.	Slow decr.	Slow decr.	Slow decr.
Ventricles	Slow decr.	Slow decr.	Fast incr.	Fast decr.	Slow decr.

Table 2
Parameters of the elasticity model and the numerical solver.

Parameter	Value	Unit	Meaning
C	2×10^3	Pa	Guccione bulk stiffness.
κ	5×10^4	Pa	Incompressibility penalty.
$T_{\max}$	4×10^4	Pa	Maximal active stress magnitude.
C_1, C_2	$1 \times 10^5, 2 \times 10^4$	Pa	Mooney-Rivlin model parameters.
k_d	2×10^3	kg·m^{-2}·s^{-2}	Impedance boundary force parameter on displacement.
k_v	2×10^2	kg·m^{-2}·s^{-1}	Impedance boundary force parameter on velocity.
α	70, 140, 110, 140, 70, 0	angle degree	Fiber rotation angle on the endocardium of LV, RV, LA, RA, and the epicardium of the ventricles and the atria, respectively.
b_f, b_{fs}, b_t	8, 4, 2	N/A	Guccione strain energy parameter in fiber, fiber shear, and transverse direction, respectively.
ILU	1	N/A	Fill-in level for the subdomain solver.
Overlap	1	N/A	Overlapping size for Schwarz preconditioner.
$\mathrm{a_N}, \mathrm{r_N}$	10^{-8}, 10^{-6}	N/A	Absolute and relative stopping conditions for Newton.
$\mathrm{a_K}, \mathrm{r_K}$	10^{-8}, 10^{-4}	N/A	Absolute and relative stopping conditions for GMRES.

Remark 4.2. In the heart, the active stress generated by cardiac fibers depends on the cellular ionic current and is triggered by physiological electrical impulses. Numerical modeling of the complex phenomena requires a more sophisticated approach that is beyond the scope of this paper. The proposed active stress function provides a simplified alternative that captures several essential features of the heart including: (1) both atria and ventricles follow the same rhythm of contracting and relaxing in each cardiac cycle, (2) the systolic phases of the atria and ventricles are asynchronous, i.e., when the atria contract the ventricles relax, and vice versa, and (3) the systolic phase is much shorter than the diastolic phase for both the atria and the ventricles.

5. Numerical experiments

In this section, we present some numerical experiments to show the computed deformation of a full size patient-specific heart with four chambers during a cardiac cycle, and the main focus is to understand the performance of the proposed numerical method in terms of its efficiency, robustness, and parallel scalability. The 3D geometry of a patient-specific heart is first extracted from a CT image with a spatial resolution of 0.5 mm, taken at the end of the diastole phase, and the noise and local non-smoothness are removed. Fully unstructured tetrahedral meshes are generated for the reconstructed geometry with smaller cells near the surfaces. This approach ensures that the solution in the irregular endocardia and epicardium is well resolved and that the total number of cells is manageable. Several meshes based on the same geometry are created with different sizes for the study of the mesh dependency of the solution and the parallel scalability, and one of the meshes with 1.01 million vertices and 4.67 million cells is referred to as the *baseline mesh* and used for most of the experiments. Moreover, the time for a cardiac cycle is normalized to be 1 second, and the period $[0, 1]$ is partitioned into 100 time intervals. This initial temporal partition is referred to as the *baseline temporal partition* with the *baseline time step size* $\Delta t_B = 0.01$ s. All experiments are carried out using in-house software that utilizes PETSc [41] and ParMetis [42].

The proposed method features several levels of nested iterations and the convergence is quite difficult to achieve for one entire cardiac cycle since the dynamics of the systems are very stiff at some spatial-temporal locations that are not known in advance. The nonlinearity is very high to the point that **Algorithm 1** fails to converge. In such a situation, the proposed adaptive time stepping scheme presented in **Algorithm 2** becomes important, it reduces the time step size, which effectively reduces the level of deformation for that particular time step, but also increases the total number of time steps. In the experiments, the failure of the solver is determined by two conditions: either the inexact Newton fails to converge in 10 iterations, or GMRES fails to converge in 3000 steps. These conditions are established based on our experience with a large number of experiments. For clarity, Table 2 lists the values of all parameters of the material model and the numerical solvers used for the baseline case.

5.1. Simulation for a cardiac cycle

In this section, we discuss the details of the computed elastodynamics of a patient-specific heart for one cardiac cycle. The computation uses the baseline mesh defined before, with parameters given in Table 2. To reduce the artifacts caused by the impact of the initial condition, we run a preliminary cycle as a "warm-up" step with both the initial displacement and active stress set to

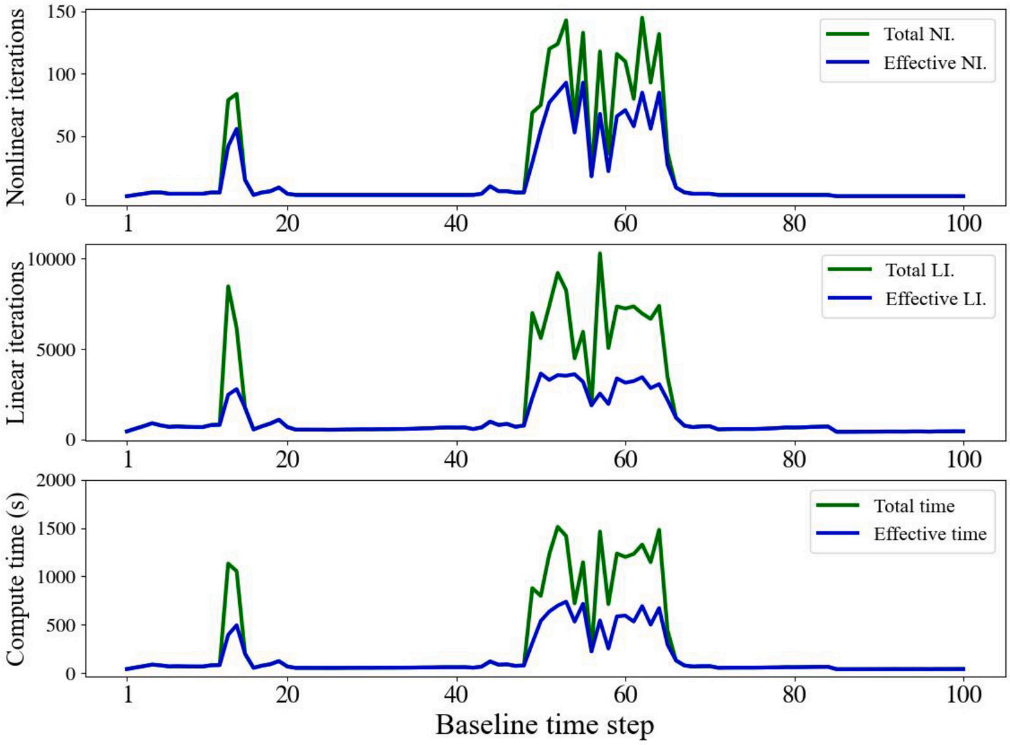

Fig. 6. Numerical performances of the nonlinear and linear solvers in each baseline time step, for one cardiac cycle of a patient-specific four-chamber heart. The top and middle figures present the effective and the total numbers of nonlinear and linear iterations for each baseline time step, respectively, while the bottom figure counts the effective and the total compute time for each baseline time step.

zero. The final solution of this cycle is used as the initial condition for the second cycle, with which we discuss all the performance of the algorithm and its temporal and spatial accuracy.

Selecting the right time step size is not a trivial task since it has to be *small enough* (1) for the solution to be reasonably accuracy, and (2) for the resulting nonlinear algebraic systems to be solvable, meanwhile *large enough* so that the total number of time steps for the cardiac cycle is reasonably small and the total compute time is not too large. Recall from **Algorithm 3** that the cardiac cycle is normalized to 1 second and initially partitioned into 100 the baseline time steps, and the simulation commences with a trial using the baseline time step size, denoted as Δt_B, for each step. Our experiments show that at the beginning part of the cardiac cycle, all components of the methods perform well with the baseline time step size. However, after the beginning phase the system experiences increased stiffness, necessitating the adoption of the adaptive scheme for some of the baseline time steps for ensuring the convergence of the nonlinear and linear solvers within the prescribed iteration limits. Note that the adaptive time stepping **Algorithm 3** involves both successful and unsuccessful trials, which implies the use of varying numbers of Newton and GMRES iterations. Fig. 6 presents the detailed behavior of the linear and nonlinear solvers throughout the whole cardiac cycle, in terms of the effective and the total numbers of nonlinear and linear iterations, as well as the compute time for each baseline time step, respectively. In addition, Table 3 reports the statistics of the numerical behavior collected per each of the five stages of the cardiac cycle (see Table 1). It can be seen that, for most of the 100 baseline time steps, the solver converges successfully using the baseline time step size Δt_B without any adaptive refinement and their statistics per stage are provided in rows 2-5 of Table 3. For the other baseline time steps, the calculating starts with Δt_B, but the proposed adaptive time stepping strategy is used to adjust the time step size to achieve convergence. The statistics of these steps per stage are provided in rows 6-11 of Table 3. Moreover, a summary of all baseline steps is given in rows 12-19 of the table.

In Table 3, for the baseline time steps without requiring adaptive time stepping, we report the average number of nonlinear iterations per time step (Ave. NI/TS), the average number of GMRES iterations per Newton step (Ave. LI/NI), and the average compute time per time step (Ave. Time/TS). On the other hand, for the baseline time steps that require adaptive time steppings, each involves a certain number of successful and unsuccessful trials that correspond to effective and failed steps, respectively. For these intervals, we count the average number of all trials (Ave. Trial/TS) and successful trials (Ave. Effective/TS) per interval, respectively, and further calculate the average number of nonlinear (Ave. ENI/ETS) and linear (Ave. ELI/ENI) iterations for the solver to converge in each successful trial. Moreover, we report the average compute time for each baseline time step (Ave. Time/TS), and record the minimal effective time step size (Min. Size) and the computational time efficiency (Efficiency) per stage, where the latter is defined as the compute time of all successful trials divided by the compute time consumed by the whole baseline time step. Table 3 presents the detailed statistics for each stage, and Fig. 7 illustrates the average number of total trials and effective steps per baseline time step, as well as the numbers of Newton and GMRES iterations per effective step, in each stage of the cardiac cycle.

Table 3
Convergence history of the proposed method in simulating the elastodynamics of a patient-specific heart in a complete cardiac cycle. Rows 2-5 show the average inexact Newton and GMRES iterations as well as the compute time in solving a baseline time step that does not need the adaptive time stepping, rows 6-11 present the solver behaviors in solving the baseline time intervals that require the adaptive time stepping, and the overall performances of solver in solving all baseline time intervals are presented in rows 12-19, per each stage (see Table 1) of the cardiac cycle.

	Stage	I	II	III	IV	V	All stages
No adaptive stepping	# Time Steps	12	25	8	14	20	79
	Ave. NI / TS	4.08	3.48	5.25	3.36	2.20	3.41
	Ave. LI / NI	173.5	177.09	142.57	188.30	219.66	179.97
	Ave. Time / TS	69.56	60.32	78.34	60.65	44.55	59.62
With adaptive stepping	# Time Steps	3	0	7	11	0	21
	Ave. Trial / TS	13.00	N/A	16.14	14.45	N/A	14.81
	Ave. Effective / TS	10.67	N/A	12.29	11.27	N/A	11.52
	Ave. ENI / ETS	3.53	N/A	5.64	4.56	N/A	4.81
	Ave. ELI / ENI	61.93	N/A	47.62	51.02	N/A	50.66
	Ave. Time / TS	795.88	0	1101.20	964.46	0	985.96
All steps	# Time Steps	15	25	15	25	20	100
	Ave. Trial / TS	3.40	1	8.07	6.92	1	3.90
	Ave. Effective / TS	2.93	1	6.27	5.52	1	3.21
	Ave. ENI / ETS	3.68	3.48	5.61	4.43	2.20	4.46
	Ave. ELI / ENI	95.69	177.09	55.18	61.56	219.66	74.95
	Min. Size	3.125E−4	0.01	3.125E−4	3.125E−4	0.01	3.125E−4
	Ave. Time / TS	214.83	60.32	555.68	458.33	44.55	254.15
	Efficiency	59.69%	100%	57.59%	51.25%	100%	59.00%

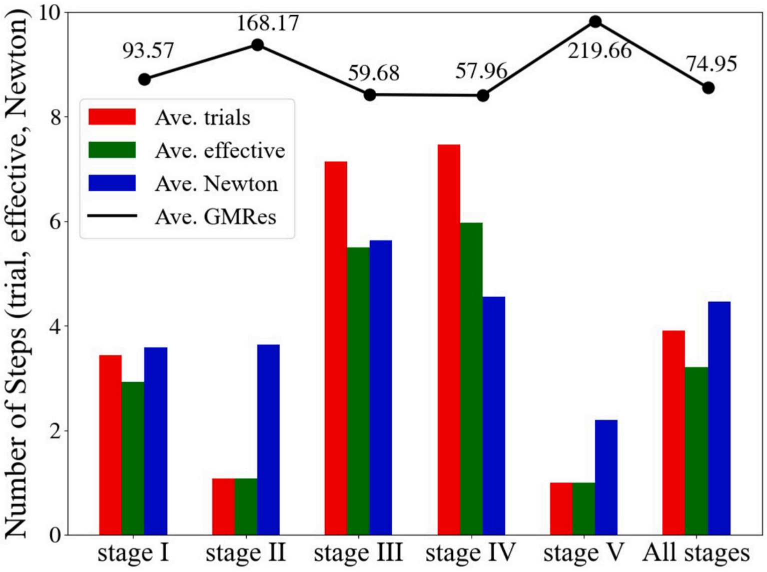

Fig. 7. Collective performances of the numerical nonlinear and linear solvers in each stage, during one cardiac cycle simulation of a patient-specific four-chamber heart.

The numerical algorithm behaves differently from stage to stage. None of the baseline time steps in stages II and V require any refinement, which results in an ideal computational efficiency. In contrast, in stages I, III, and IV, 3 out of 15, 7 out of 15, and 11 out of 25 baseline time steps, respectively, require some level of adaptive refinement. At each of these $21(= 3 + 7 + 11)$ adaptive time steps, the nonlinear solver takes an average of 10 to 13 effective steps to complete, accompanied by 2 to 4 unsuccessful trials. For all the time steps that do not require adaptive time stepping throughout the cardiac cycle, the average number of nonlinear iterations per time step and linear iterations per Newton step are about 3.41 and 179.97, respectively. On the other hand, for successful adaptive trials, these values are around 4.81 and 50.66. These findings suggest that (i) significant nonlinearity is a crucial factor in activating the adaptive time stepping scheme, and (ii) when the inexact Newton converges with an appropriate time step size, it typically requires fewer number of iterations. Notably, as the minimal effective time step decreases to 3.125×10^{-4} s, which is one-thirty-second of 0.01 s, by the use of adaptive reduction rules, the average number of all and successful trials in each baseline time step are 3.90 and 3.21, respectively. This result clearly demonstrates the effectiveness of using adaptive time stepping to reduce the overall computational cost, compared to using a uniform small time step size throughout the entire cardiac cycle.

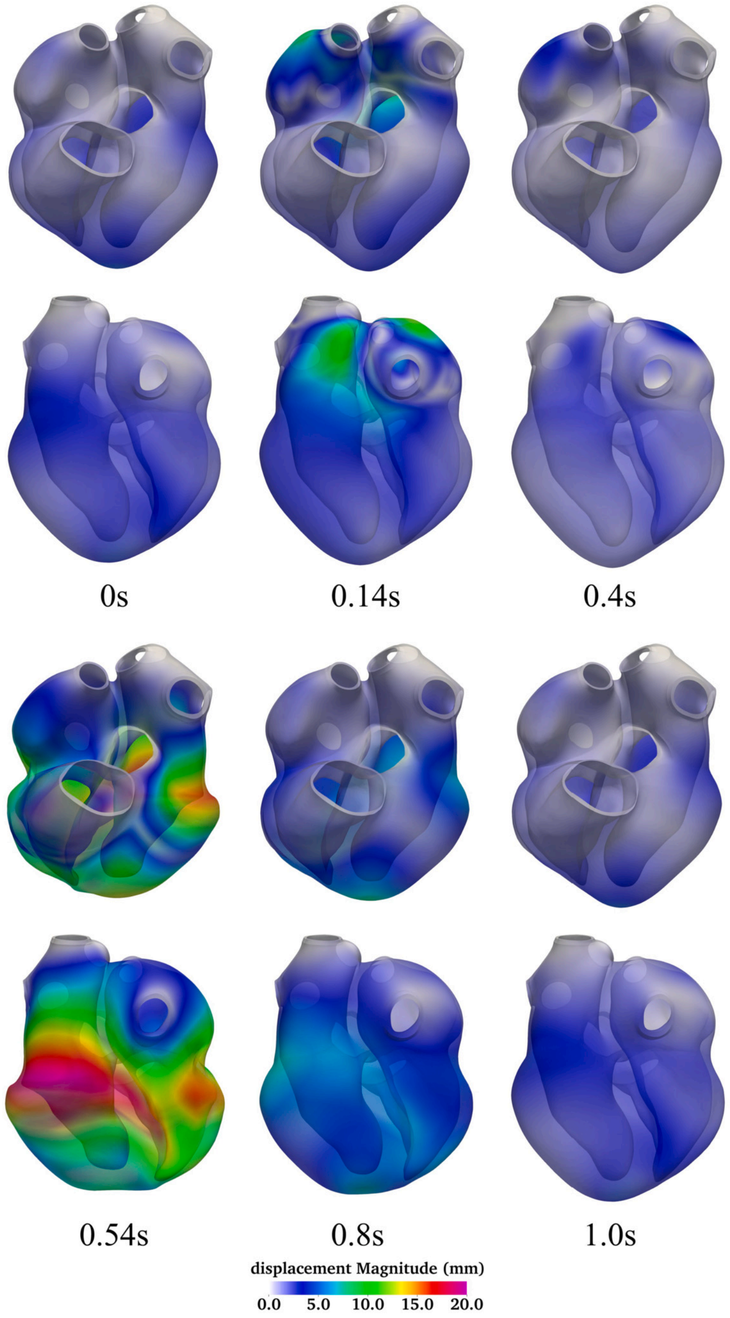

Fig. 8. The magnitude of the displacement over the myocardial domain at six typical time moments, at 0 s, 0.14 s, 0.4 s, 0.54 s, 0.8 s, and 1.0 s (from left to right, top to bottom), respectively. Each panel presents two plots vertically, illustrating the anterior and posterior views of the heart.

The convergence behavior of the proposed method is closely related to several factors, including the extent of the deformation of the heart muscles, the magnitude of the active stress generated by the fibers, and their changing rates. Collectively, these factors dictate the level of difficulty of the nonlinear elastodynamical problem. Table 3 and Fig. 7 confirm that stage I, III, and IV are more challenging to solve than stages II and V. This is because stages I and III represent phases in which the active stress increases rapidly (in the atrial and ventricular domains, respectively), and stage IV is characterized by a high level of deformation in the ventricular domain. Conversely, the changing rate of the active stress and/or deformation extent is much weaker during stages II and V. Noticeably, in both stages II and IV, the process involves restoring the heart to its relaxed state from a deformed shape. The primary distinction between the two lies in the extent of deformation, which directly impacts the complexity of solving the entire system. In stage II, the deformation predominantly affects the atrial region of the heart, with the ventricular part experiencing very

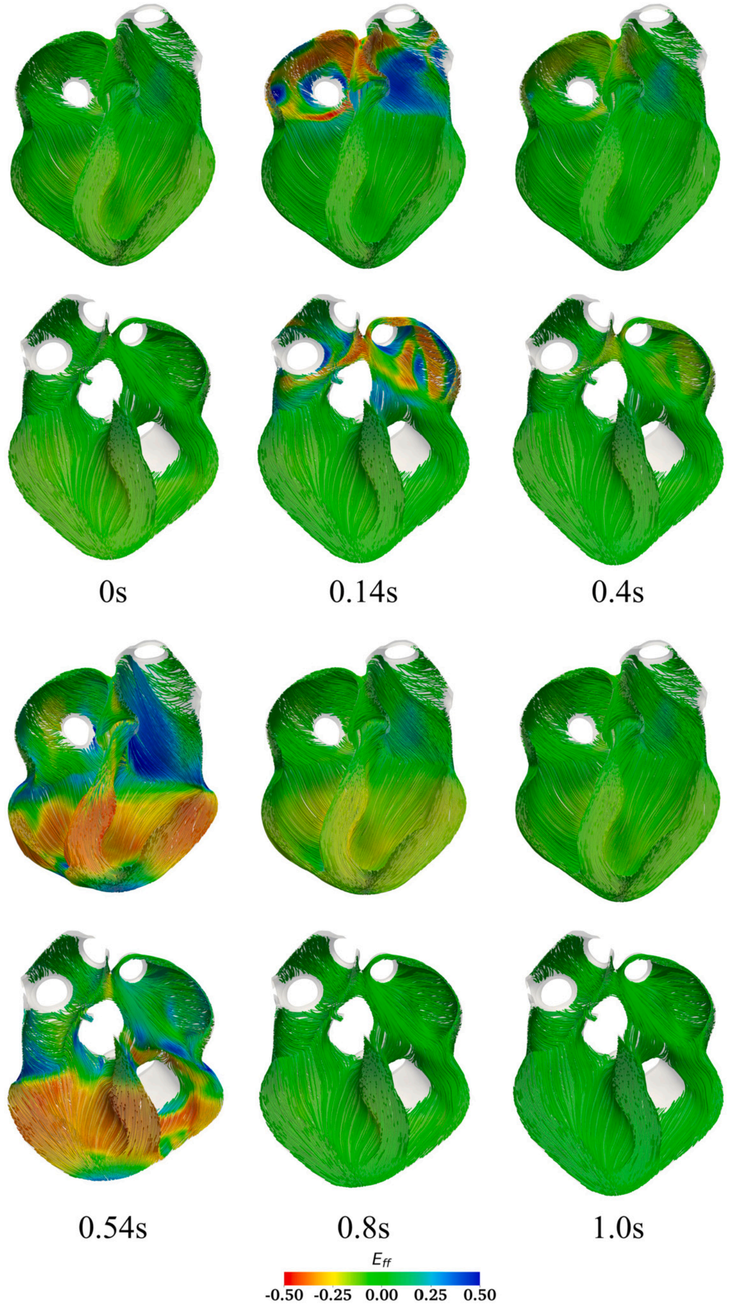

Fig. 9. The fiber strain E_{ff} over the myocardial domain at six typical time moments, at 0 s, 0.14 s, 0.4 s, 0.54 s, 0.8 s, and 1.0 s (from left to right, top to bottom), respectively. Each panel presents two vertically aligned plots, illustrating the left and right halves of the heart over a vertical cross-section through the center, respectively.

small deformation. Conversely, in stage IV, the situation is reversed, with the ventricular portion undergoing significant deformation. Given the much larger size and thickness of the ventricular section compared to the atrial part, the overall deformation and local strains are notably higher in stage IV than in stage II. Consequently, stage IV presents a considerably greater challenge in terms of the solution complexity compared to stage II. These observations are further validated by visualizing the deformed heart at six representative moments throughout the five stages of a cardiac cycle. Specifically, Fig. 8, 9 and 10 illustrate the magnitudes of displacement $|\mathbf{d}|$, as well as the strain E_{ff} and the stress S_{ff} along the fiber direction over the myocardial domain, respectively, where the latter two are defined by

$$E_{ff} = \mathbf{f} \cdot \mathbf{E}\mathbf{f} \qquad \text{and} \qquad S_{ff} = \frac{\mathbf{F}\mathbf{f}}{|\mathbf{F}\mathbf{f}|} \cdot \mathbf{P}\mathbf{f}. \tag{5.1}$$

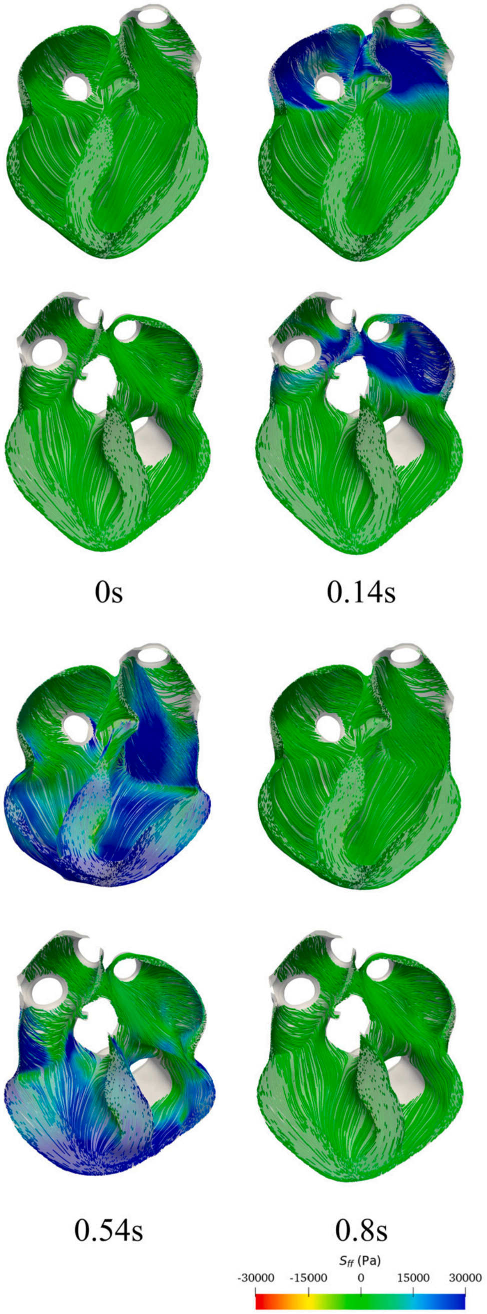

Fig. 10. The fiber stress S_{ff} over the myocardial domain at six typical time moments, at 0 s, 0.14 s, 0.4 s, 0.54 s, 0.8 s, and 1.0 s (from left to right, top to bottom), respectively. Each panel presents two vertically aligned plots, illustrating the left and right halves of the heart over a vertical cross-section through the center, respectively.

Particularly, the definition of S_{ff} is adopted from [18]. In each panel of Fig. 8, 9 and 10, the six pairs of snapshots (from left to right, and top and bottom) are taken at $t = 0$ s, 0.14 s, 0.4 s, 0.54 s, 0.8 s, and 1.0 s, respectively, which show that the deformation starts at a low level before the atria shrink to their maximal contraction at $t = 0.14$ s while the ventricles remain relaxed. Subsequently, the atria undergo diastole, and the ventricles begin to contract, reaching their maximum at $t = 0.54$ s, before returning to diastole. By $t = 0.8$ s, strains in both the atria and ventricles have decreased significantly, and they further reduce to almost the initial status at the end of the cardiac cycle. Significantly, the atrial and ventricular regions exhibit the highest magnitude of deformations, strains, and stresses at $t = 0.14$ s and $t = 0.54$ s, respectively. This corresponds to the peak growth of active stresses in these respective regions.

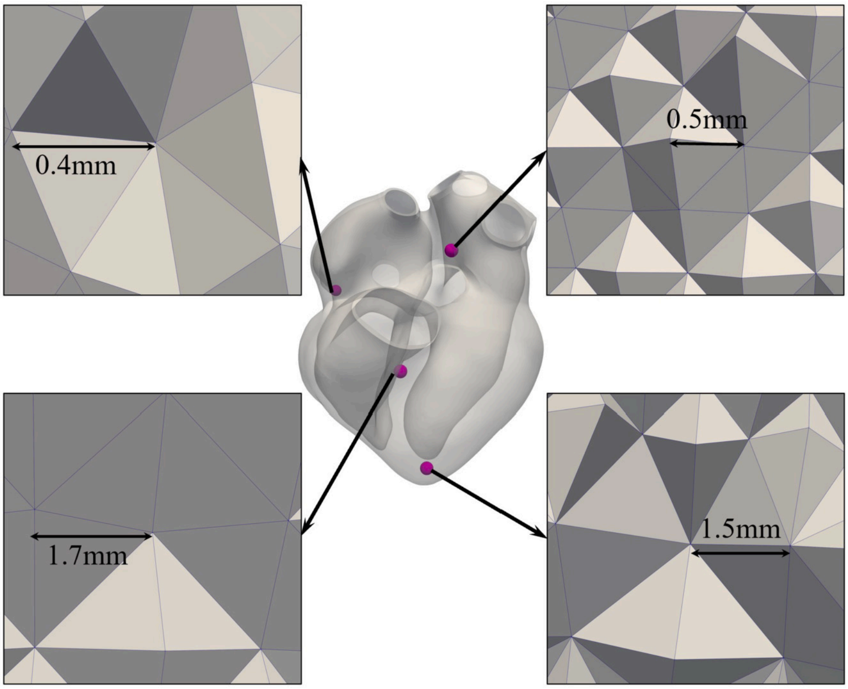

Fig. 11. The locations and baseline mesh structures of the four small regions in the LA, RA, ventricle septum and close to the apex, respectively. Mesh cells inside these regions are locally refined for investigating the convergence of E_{ff} with respect to the mesh size.

5.2. Study on the mesh dependence

Determining the appropriate spatial resolution of the computational mesh is crucial for the numerical simulations to be useful in clinic applications. Simulation of the cardiac elastodynamics requires a sufficiently fine mesh to accurately depict the complicated shape and fiber distributions of the heart muscle, both of which are essential for the desired accuracy of the results. Theoretically speaking, a uniform fine mesh would solve the accuracy issue, but the corresponding compute time would not be acceptable even on the largest supercomputers. To balance the accuracy and the computational time, we must choose appropriate mesh sizes to avoid overburdening the computational resources while simulating the essential geometric deformation of the heart.

In the following, we study the influence of the mesh size on the numerical results, with the intention to provide some guidelines on choosing appropriate mesh sizes for cardiac elastodynamic simulations. In particular, we compare the results by using a series of meshes with different spatial resolutions, generated by locally refining the baseline mesh in four representative regions inside the heart wall of LA, RA, the septum between the ventricles and close to the apex. The diameter of a typical cell in these regions is around 0.5 mm, 0.4 mm, 1.7 mm and 1.5 mm, respectively, and the local structure of the baseline mesh in these regions are shown in Fig. 11. With this type of mesh refinement, the total number of degrees of freedom can be confined within a relatively low range, in contrast to uniformly refining the whole mesh.

Inside each of these regions, the baseline mesh is refined three times independently. Using each series of meshes, the simulation results are compared at $t = 0.15$ s for the LA and RA regions and $t = 0.55$ s for the septum and apex regions. A point is put at the center of each region to monitor the solution. For each case, the fiber strain at the corresponding monitor point is tracked for the preceding 15 time steps (i.e., 0 s–0.15 s for the LA and RA regions and 0.4 s–0.55 s for the septum and apex regions). Details of the study are presented in Figs. 12–16.

For each case, the finest mesh produces the smoothest result, and the results obtained by coarser meshes approach the finest one as the mesh is refined. Notably, E_{ff} at the monitor point in the LA region is almost the same for all meshes. These experiments indicate the solution converges as the mesh is refined in the regions of interests.

5.3. Impact of several material and algorithmic parameters

In this section, we investigate the impact of several Guccione material parameters as well as algorithmic parameters in constructing the proposed numerical method. We run the simulation for one complete cardiac cycle and count, for solving each baseline time step with the baseline time step size, the average number of trials (Trials) and the successful trials (Effective), the average number of inexact Newton iterations per successful trial (NI), the average number of GMRES iterations per inexact Newton step (LI), the average compute time (Time), and the computational efficiency (Efficiency) by using the adaptive time stepping scheme, respectively.

First, we study the impact of ILU fill-in level of the subdomain solver and the overlapping size between neighboring subdomains, both of which are the key parameters for the Schwarz preconditioner. In particular, the level of fill-in determines how accurately each

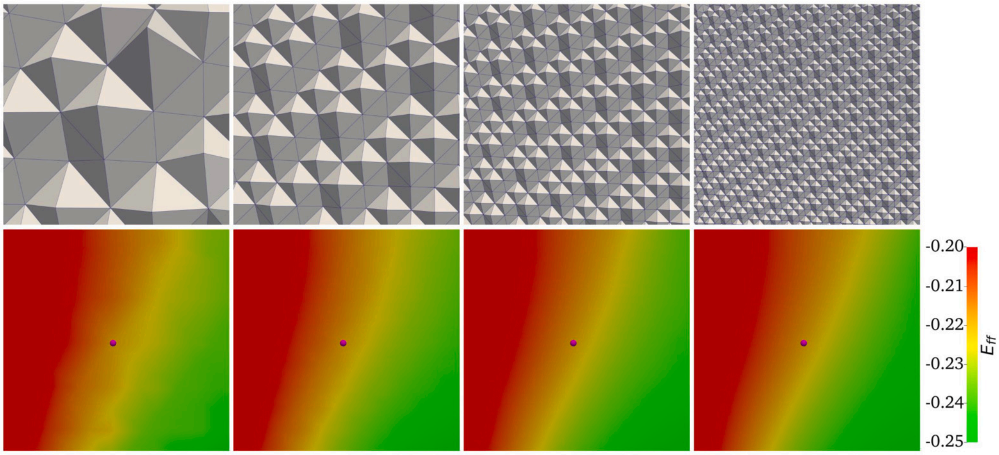

Fig. 12. The fiber strains E_{ff} in a small region inside the LA of the heart, by using a series of meshes that are locally refined for three times, respectively. The purple ball at the center of the subfigures indicates the location of the monitor point.

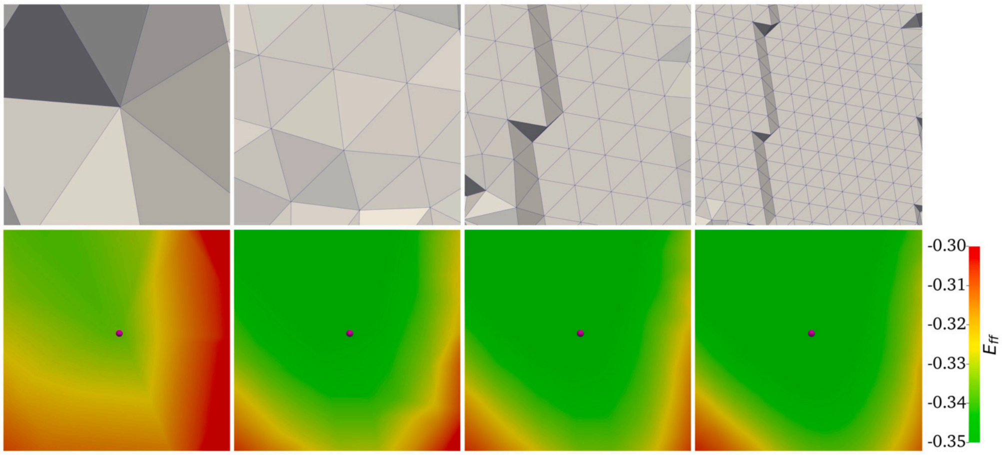

Fig. 13. The fiber strains E_{ff} in a small region inside the RA of the heart, by using a series of meshes that are locally refined for three times, respectively. The purple ball at the center of the subfigures indicates the location of the monitor point.

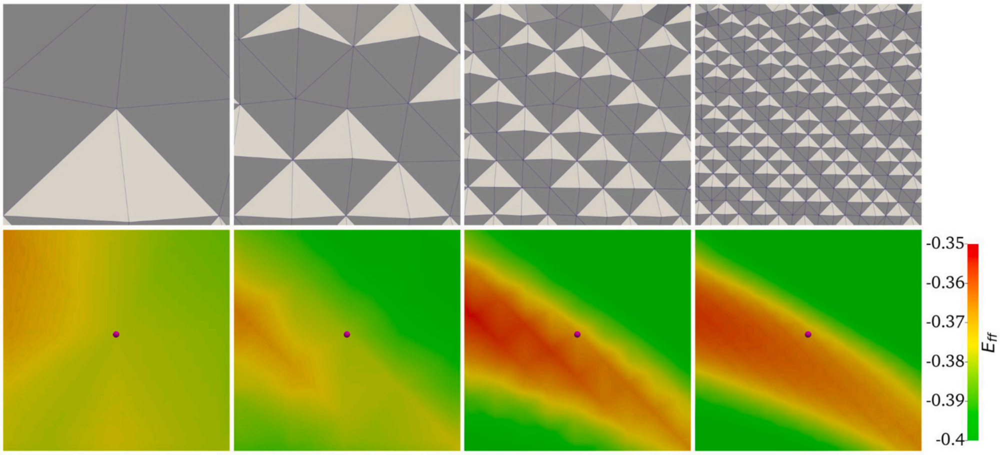

Fig. 14. The fiber strains E_{ff} in a small region inside the ventricle septum of the heart, by using a series of meshes that are locally refined for three times, respectively. The purple ball at the center of the subfigures indicates the location of the monitor point.

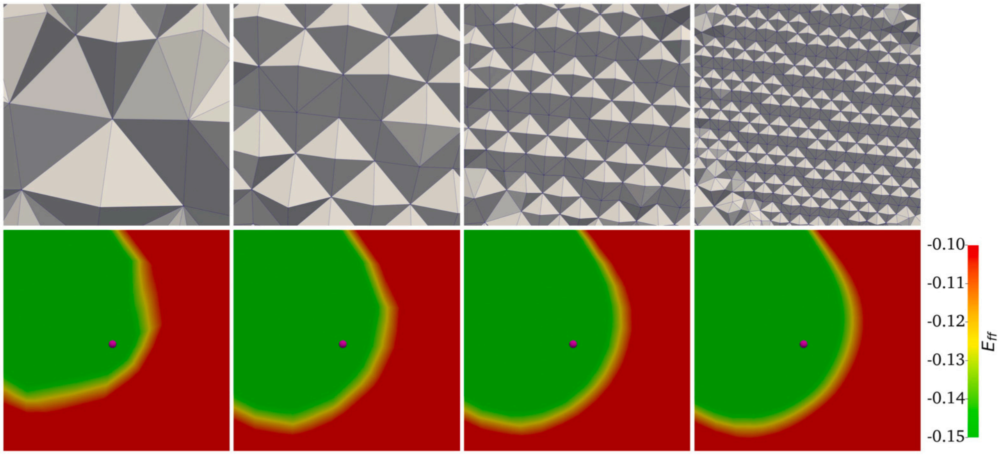

Fig. 15. The fiber strains E_{ff} in a small region close to the apex of the heart, by using a series of meshes that are locally refined for three times, respectively. The purple ball at the center of the subfigures indicates the location of the monitor point.

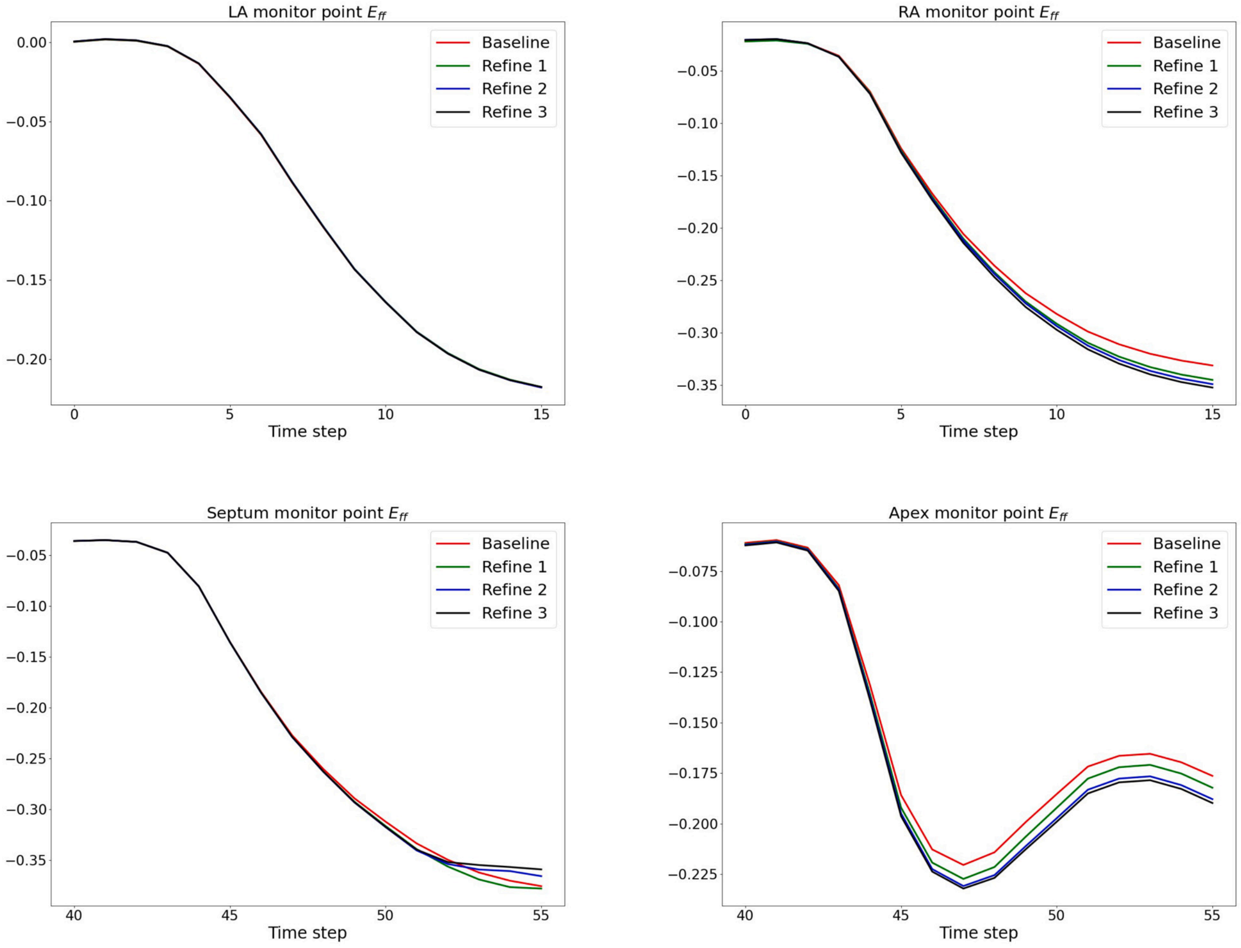

Fig. 16. E_{ff} at the monitor points in the four local refinement regions for 15 time steps: 0 s–0.15 s for LA and RA regions, and 0.4 s–0.55 s for ventricle septum and apex regions, respectively.

sub-domain problem is solved, and the overlapping size determines the amount of the information exchanged between neighboring processor cores. We test various combinations of ILU fill-in levels and overlapping sizes in a range of $0, 1, 2$ to observe the impact on

Table 4
Convergence behavior of **Algorithm 1** as the ILU fill-in level and the overlapping size are adjusted. The reported data are calculated based on all time steps during one complete cardiac cycle.

ILU	Overlap	Trial	Effective	NI	LI	Time	Efficiency
0	0	7.17	5.16	3.55	293.35	531.43	62.01%
	1	7.61	5.55	3.51	271.14	565.16	62.35%
	2	7.72	5.65	3.49	271.61	616.98	62.27%
1	0	4.39	3.61	4.13	91.47	278.31	59.07%
	1	3.90	3.21	4.46	74.95	254.15	59.00%
	2	3.99	3.29	4.34	69.73	265.96	60.04%
2	0	4.12	3.44	4.15	76.55	305.49	59.95%
	1	4.22	3.46	4.26	51.89	312.57	58.06%
	2	4.10	3.38	4.23	47.92	312.39	59.84%

Table 5
Convergence behaviors of the NKS numerical solver and the averaged volumetric changes of the deformed heart region as the Guccione model parameter C and the incompressibility penalty κ are adjusted. The reported data are calculated based on all time steps during one complete cardiac cycle. For the case $C = 1.0 \times 10^3$ and $\kappa = 4 \times 10^4$, the numerical solver failed to converge at $t = 0.51$ s as the adaptive time step size has been reduced to below the minimal allowed value 10^{-5} s.

C	κ	Trial	Effective	NI	LI	Time	Efficiency	Ave. J
1×10^3	4×10^4	–	–	–	–	–	–	–
	5×10^4	15.72	12.10	4.33	42.15	920.78	49.12%	0.9831
	6×10^4	11.24	8.67	4.30	76.58	796.95	51.08%	0.9849
2×10^3	4×10^4	8.52	6.78	4.15	43.62	458.51	52.89%	0.9811
	5×10^4	3.90	3.21	4.46	74.95	254.15	59.00%	0.9834
	6×10^4	3.10	2.60	4.04	95.81	206.06	62.41%	0.9851
3×10^3	4×10^4	2.83	2.39	3.89	80.03	171.15	60.48%	0.9816
	5×10^4	1.40	1.25	3.81	137.16	93.32	76.29%	0.9838
	6×10^4	1.25	1.19	3.39	149.48	72.90	87.27%	0.9855

Table 6
Convergence behaviors of the NKS numerical solver as the Guccione model parameters b_f, b_{fs}, b_t are adjusted. The reported data are calculated based on all time steps during one complete cardiac cycle.

(b_f, b_{fs}, b_t)	Trial	Effective	NI	LI	Time	Efficiency
(16, 4, 1)	2.67	2.30	3.50	107.59	162.57	62.44%
(8, 4, 2)	3.90	3.21	4.46	74.95	254.15	59.00%
(4, 4, 4)	9.15	7.15	4.42	45.33	534.11	51.95%

the convergence behavior of the linear and nonlinear solvers. The detailed statistics are listed in Table 4, from which one can see that the optimal result is achieved when both parameters are equal to 1.

Next we vary several parameters of the Guccione model, including the bulk modulus C and the incompressible penalty parameter κ, as well as the combination of the strain coefficients b_f, b_{fs}, b_t. The detailed results for these cases are summarized in Tables 5 and 6. It can be seen from Table 5, as both the bulk modulus and the incompressible penalty parameter become larger, both the average number of the total and the successful trials to complete one baseline time step become smaller, the compute time reduces and the computational efficiency increases. These observations imply that these two parameters influence the overall difficulty of the problem, in other words, the larger the parameters are, the easier the nonlinear elastodynamic problems become, within the range of the selected values. Additionally, the average volumetric changes of the deformed heart region are reported in the last column of Table 5 (denoted as Ave. J). This metric is defined as the volume of the deformed heart region divided by the volume of the reference region, averaged over all baseline time steps of one cardiac cycle. It can be seen that the deformed volumes range between 98% and 99% for all combinations of C and κ, indicating a good adherence to the incompressibility constraint. On the other hand, Table 6 suggests that the problem becomes more difficult to solve as the values of the strain parameters b_f, b_{fs}, b_t become unified, as the total trials and successful trials both increase, the compute time becomes larger, and the computational efficiency becomes lower.

Finally, we investigate how the fiber rotation angle alters the deformation of the heart. We change the orientation angle α on the heart surface in the fiber generation process by $-10°$, $-5°$, $5°$, and $10°$ relative to the baseline case, and carry out a separate simulation for each variation. To visualize the effects of these modifications, two timepoints are chosen to plot the contours of a common vertical cross-section of the heart in its deformed state, at $t = 0.14$ s (maximal atrial active stress) and $t = 0.54$ s (maximal

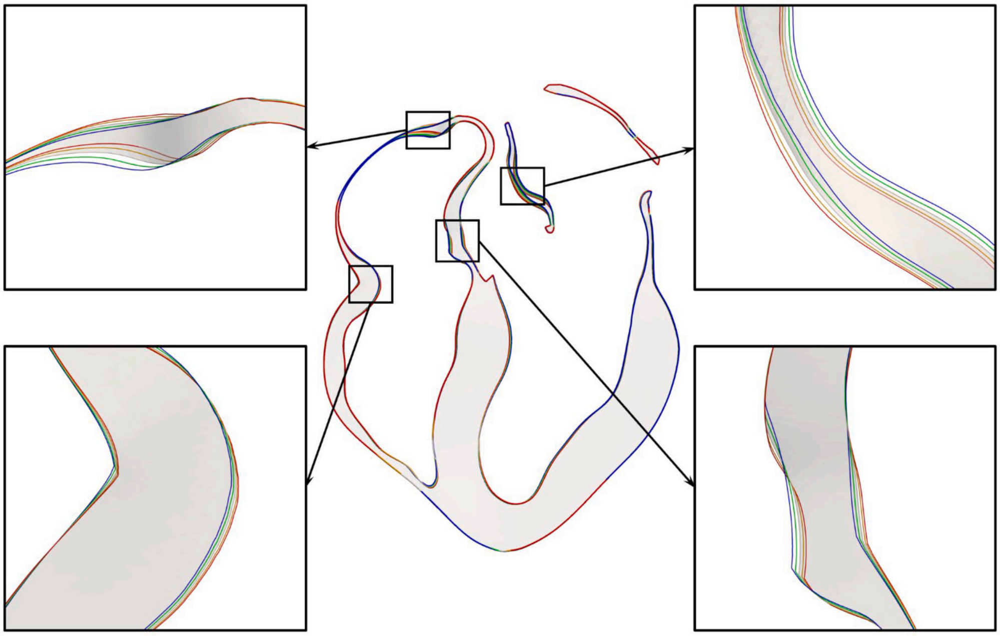

Fig. 17. Contours of a vertical cross-section obtained by setting the fiber orientation angle α to be −10 (blue), −5 (green), +5 (orange), and +10 (red) degrees comparing to the baseline case at $t = 0.14$ s, respectively. The baseline result is colored in gray for reference.

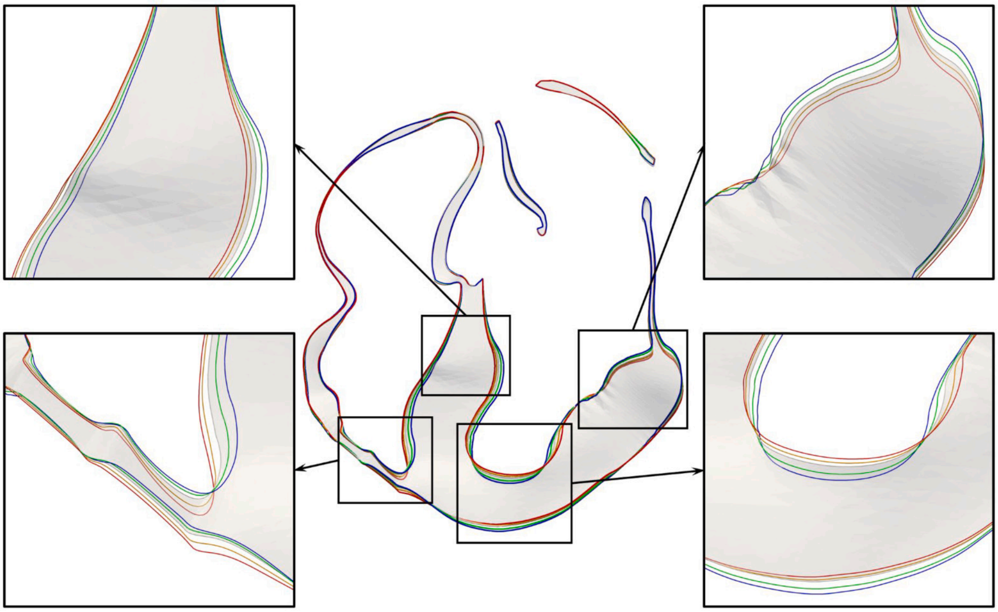

Fig. 18. Contours of a vertical cross-section obtained by setting the fiber orientation angle α to be −10 (blue), −5 (green), +5 (orange), and +10 (red) degrees comparing to the baseline case at $t = 0.54$ s, respectively. The baseline result is colored in gray for reference.

ventricular active stress) in Fig. 17 and Fig. 18, respectively. It can be seen that, comparing to the baseline case, the contour of the cross-section changes in a progressive way as the value of α changes. For example, at $t = 0.14$ s, in a top portion of the right atrium (the top left subplot of Fig. 17), the contours show that the atrial wall moves up further as the change of α increases from $-10°$ to $10°$. More interestingly, at $t = 0.54$ s, in a region closing to the apex of the heart, the right ventricle moves downward (the bottom left subplot of Fig. 18) while the left ventricular moves upward (the bottom right subplot of Fig. 18) as the change of α increases from $-10°$ to $10°$, respectively.

Table 7
Compute time and parallel efficiency using three different meshes calculated based on the first 10 time steps with step size 0.001 s and ILU fill-in level 3.

	Mesh 1 (36M dofs)				Mesh 2 (127M dofs)				Mesh 3 (235M dofs)			
Cores	Time	NI	LI	Efficiency	Time	NI	LI	Efficiency	Time	NI	LI	Efficiency
1024	451.9	2.3	54.1	100%								
2048	241.4	2.3	58.0	93.6%	1016.1	2.3	63.4	100%				
3072	191.4	2.3	60.7	78.7%	696.1	2.3	65.5	97.3%	2118.9	3.0	83.4	100%
4096	152.8	2.3	62.8	74.0%	525.8	2.3	67.2	96.6%	1603.9	3.0	85.1	99.1%
6144	114.0	2.3	65.8	66.1%	370.1	2.3	70.6	91.5%	1112.9	3.0	89.5	95.2%
8192	94.7	2.3	69.2	59.6%	376.4	2.3	73.0	67.5%	859.3	3.0	92.4	92.5%
16384									765.1	3.0	100.2	52.7%

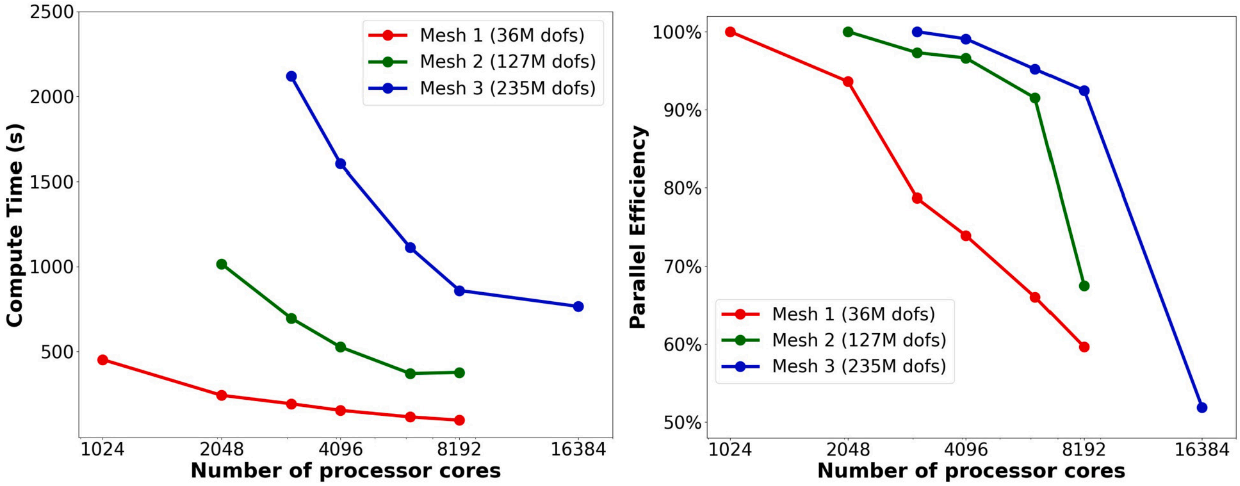

Fig. 19. Compute time and parallel efficiency for three meshes. The reference numbers of processor cores are set to 1024 cores for the 36 millions mesh, 2048 cores for the 127 millions mesh, and 3072 for the 235 millions mesh, respectively.

5.4. Parallel scalability

In this subsection, we first investigate the strong parallel scalability of the proposed algorithm by running simulation on three meshes with around 36 million, 127 million, and 235 million degrees of freedom (DOF). For each mesh, four or five sets of processor cores are used to repeat the simulation, and we report the compute time and the parallel efficiency. Let us note the term parallel efficiency should not be confused with the adaptive time stepping efficiency. We calculate the strong parallel efficiency using the following formula:

$$\text{Efficiency}_s = \frac{\text{compute time with } p_0 \text{ processor cores}}{\text{compute time with } p_i \text{ processor cores}} \times \frac{p_0}{p_i} \times 100\%, \tag{5.2}$$

where p_i is the number of processor cores used for a certain case label by i, and p_0 denotes the reference number of processor cores, which is set to be 1024 cores for the 36 million mesh, 2048 cores for the 127 million mesh, and 3072 cores for the 235 million mesh. Additionally, we run the simulations for the first 10 time steps with an ILU fill-in level set to 3, and the time step size is set to 10^{-3} seconds. All cases converge without using the adaptive time stepping scheme. As can be seen in Table 7 and Fig. 19, except for one case (8192 cores on Mesh 2), the convergence accelerates as more processor cores are utilized to solve the problem, and can maintain a good parallel scalability with up to 16,384 processor cores.

In addition, we evaluate the weak scalability to investigate how the compute time is affected as we increase the number of processor cores while the size of the problem per core remains the same. Due to the complex geometry of a patient specific heart, it is very difficult to generate a set of unstructured tetrahedral meshes which (1) have appropriate sizes and (2) are proportionally larger by integral multiples. In this paper, we evaluate a scaled weak parallel efficiency that is defined by:

$$\text{Efficiency}_{sw} = \frac{\text{compute time with } p_0 \text{ processor cores for mesh}_0}{\text{compute time with } p_i \text{ processor cores for mesh}_i} \times \frac{p_0}{p_i} \times \frac{\text{DOF of mesh}_i}{\text{DOF of mesh}_0} \times 100\%, \tag{5.3}$$

where mesh$_0$ denotes the mesh used for the reference case, and mesh$_i$ for a certain case i. The above definition allows us to approximately evaluate the weak scalability based on the same metrics as calculating the strong scalability, without the strict requirement for uniform problem sizes per processor core. Provided this definition, we further define the ideal scaled compute time for a case i that uses p_i processor cores for mesh$_i$ by:

Table 8
The compute time, ideal time and scaled weak parallel efficiencies of the two groups of mesh-processor core pairs. Rows 1-5 correspond to group 1, while rows 6-10 correspond to group 2.

Group 1	Mesh 1 1024 cores	Mesh 2 3072 cores	Mesh 3 6144 cores
Compute time	451.9	696.1	1112.9
Ideal time	451.9	531.4	491.7
Scaled efficiency	100%	76.35%	44.18%
Group 2	Mesh 1 2048 cores	Mesh 2 6144 cores	Mesh 3 16384 cores
Compute time	241.4	370.1	765.1
Ideal time	241.4	283.9	197.0
Scaled efficiency	100%	76.70%	25.75%

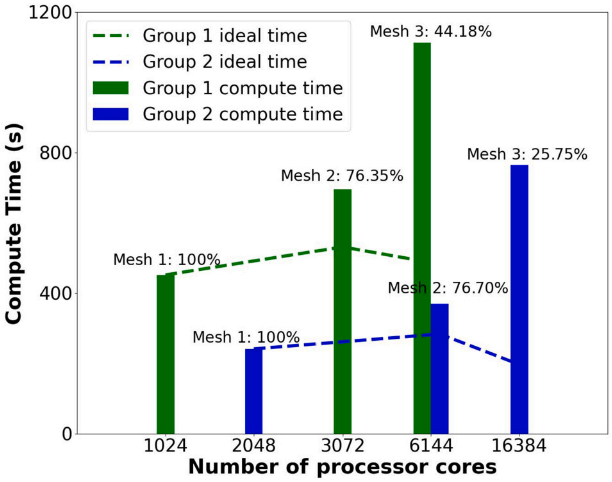

Fig. 20. Compute time and scaled weak parallel efficiency for the two selected groups. The compute time is presented by vertical bars for each case. For each group, the mesh 1 case serves as the reference for calculating the scaled weak parallel efficiencies, marked on the top of vertical bars, for cases utilizing the other two meshes. The ideal compute time is indicated by the dashed lines, respectively.

$$\text{Ideal time} = \text{compute time with } p_0 \text{ processor cores for mesh}_0 \times \frac{p_0}{p_i} \times \frac{\text{DOF of mesh}_i}{\text{DOF of mesh}_0}. \tag{5.4}$$

In this paper, we consider two groups of mesh-and-processor core pairs: Group 1 includes mesh 1 with 1024 cores, mesh 2 with 3072 cores, and mesh 3 with 6144 cores; Group 2 includes mesh 1 with 2048 cores, mesh 2 with 6144 cores, and mesh 3 with 16384 cores. Both two groups use the smallest mesh as the reference case, and they correspond to the cases that each processor core handles around 38,000 DOF and 17000 DOF, respectively. The ideal compute time and the scaled weak scalability efficiency are presented in Table 8, as well as in Fig. 20.

6. Conclusion

In this paper, we developed a highly scalable, fully implicit, two-level adaptive time stepping algorithm for simulating the elastodynamics of a patient-specific human heart with four chambers and fibers. The heart is modeled by a hyperelasticity equation discretized using a finite element method on a fully unstructured mesh with finer mesh cells near the surface and regions with large deformation, and we solve the resulting nonlinear algebraic system using the Newton-Krylov-Schwarz method at each time step. To enhance the robustness of algorithm, we introduce a two-level adaptive time-stepping strategy which offers the desired level of accuracy and accelerates the nonlinear convergence considerably. The numerical experiments demonstrate the proposed algorithm is robust in simulating one complete cardiac cycle and highly scalable for solving the problem with over 200 million degrees of freedom using more than 16,000 processor cores.

CRediT authorship contribution statement

Yi Jiang: Methodology, Writing – original draft, Writing – review & editing, Conceptualization, Software. **Zhengzheng Yan:** Data curation. **Xinhong Wang:** Resources. **Rongliang Chen:** Supervision. **Xiao-Chuan Cai:** Conceptualization, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

This work was partially supported by the NKP of China 2023YFC2507501 and NSFC 12001520 and 62161160312.

References

[1] L. Cai, Y. Hao, P. Ma, G. Zhu, X. Luo, H. Gao, Fluid-structure interaction simulation of calcified aortic valve stenosis, Math. Biosci. Eng. 19 (12) (2022) 13172–13192.

[2] N. Thekkethil, S. Rossi, H. Gao, S.I.H. Richardson, B.E. Griffith, X. Luo, A stabilized linear finite element method for anisotropic poroelastodynamics with application to cardiac perfusion, Comput. Methods Appl. Mech. Eng. 405 (2023) 115877.

[3] E. Kayvanpour, T. Mansi, F. Sedaghat-Hamedani, A. Amr, D. Neumann, B. Georgescu, P. Seegerer, A. Kamen, J. Haas, K.S. Frese, et al., Towards personalized cardiology: multi-scale modeling of the failing heart, PLoS ONE 10 (7) (2015) e0134869.

[4] S.A. Niederer, J. Lumens, N.A. Trayanova, Computational models in cardiology, Nat. Rev. Cardiol. 16 (2) (2019) 100–111.

[5] N.A. Trayanova, Whole-heart modeling: applications to cardiac electrophysiology and electromechanics, Circ. Res. 108 (1) (2011) 113–128.

[6] C. Yamamoto, N.A. Trayanova, Atrial fibrillation: Insights from animal models, computational modeling, and clinical studies, eBioMedicine 85 (2022) 104310.

[7] N.A. Barnafi, L.F. Pavarino, S. Scacchi, Parallel inexact Newton–Krylov and quasi-Newton solvers for nonlinear elasticity, Comput. Methods Appl. Mech. Eng. 400 (2022) 115557.

[8] N.A. Barnafi, L.F. Pavarino, S. Scacchi, A comparative study of scalable multilevel preconditioners for cardiac mechanics, J. Comput. Phys. 492 (2023) 112421.

[9] P.C. Franzone, L.F. Pavarino, S. Scacchi, Parallel multilevel solvers for the cardiac electro-mechanical coupling, Appl. Numer. Math. 95 (2015) 140–153.

[10] P.C. Franzone, L.F. Pavarino, S. Scacchi, A numerical study of scalable cardiac electro-mechanical solvers on HPC architectures, Front. Physiol. 9 (2018) 268.

[11] E. Garcia-Blanco, R. Ortigosa, A.J. Gil, J. Bonet, Towards an efficient computational strategy for electro-activation in cardiac mechanics, Comput. Methods Appl. Mech. Eng. 356 (2019) 220–260.

[12] V. Gurev, P. Pathmanathan, J.-L. Fattebert, H.-F. Wen, J. Magerlein, R.A. Gray, D.F. Richards, J.J. Rice, A high-resolution computational model of the deforming human heart, Biomech. Model. Mechanobiol. 14 (4) (2015) 829–849.

[13] L. Pavarino, S. Scacchi, C. Verdi, E. Zampieri, S. Zampini, Scalable BDDC algorithms for cardiac electromechanical coupling, in: Domain Decomposition Methods in Science and Engineering XXIII, Springer, 2017, pp. 261–268.

[14] M.R. Pfaller, J.M. Hörmann, M. Weigl, A. Nagler, R. Chabiniok, C. Bertoglio, W.A. Wall, The importance of the pericardium for cardiac biomechanics: from physiology to computational modeling, Biomech. Model. Mechanobiol. 18 (2) (2019) 503–529.

[15] M. Strocchi, C.M. Augustin, M.A. Gsell, E. Karabelas, A. Neic, K. Gillette, O. Razeghi, A.J. Prassl, E.J. Vigmond, J.M. Behar, et al., A publicly available virtual cohort of four-chamber heart meshes for cardiac electro-mechanics simulations, PLoS ONE 15 (6) (2020) e0235145.

[16] B. Baillargeon, I. Costa, J.R. Leach, L.C. Lee, M. Genet, A. Toutain, J.F. Wenk, M.K. Rausch, N. Rebelo, G. Acevedo-Bolton, et al., Human cardiac function simulator for the optimal design of a novel annuloplasty ring with a sub-valvular element for correction of ischemic mitral regurgitation, Cardiovasc. Eng. Technol. 6 (2) (2015) 105–116.

[17] C.M. Augustin, A. Neic, M. Liebmann, A.J. Prassl, S.A. Niederer, G. Haase, G. Plank, Anatomically accurate high resolution modeling of human whole heart electromechanics: a strongly scalable algebraic multigrid solver method for nonlinear deformation, J. Comput. Phys. 305 (2016) 622–646.

[18] F. Regazzoni, M. Salvador, P.C. Africa, M. Fedele, L. Dede, A. Quarteroni, A cardiac electromechanical model coupled with a lumped-parameter model for closed-loop blood circulation, J. Comput. Phys. 457 (2022) 111083.

[19] M. Bucelli, A. Zingaro, P.C. Africa, I. Fumagalli, L. Dede', A. Quarteroni, A mathematical model that integrates cardiac electrophysiology, mechanics, and fluid dynamics: Application to the human left heart, Int. J. Numer. Methods Biomed. Eng. 39 (3) (2023) e3678.

[20] M. Fedele, R. Piersanti, F. Regazzoni, M. Salvador, P.C. Africa, M. Bucelli, A. Zingaro, A. Quarteroni, et al., A comprehensive and biophysically detailed computational model of the whole human heart electromechanics, Comput. Methods Appl. Mech. Eng. 410 (2023) 115983.

[21] C.S. Peskin, The immersed boundary method, Acta Numer. 11 (2002) 479–517.

[22] R. Mittal, J.H. Seo, V. Vedula, Y.J. Choi, H. Liu, H.H. Huang, S. Jain, L. Younes, T. Abraham, R.T. George, Computational modeling of cardiac hemodynamics: Current status and future outlook, J. Comput. Phys. 305 (2016) 1065–1082.

[23] D.M. McQueen, C.S. Peskin, A three-dimensional computer model of the human heart for studying cardiac fluid dynamics, ACM SIGGRAPH Comput. Graph. 34 (1) (2000) 56–60.

[24] R. Verzicco, Electro-fluid-mechanics of the heart, J. Fluid Mech. 941 (2022) P1.

[25] F. Viola, G. Del Corso, R. Verzicco, High-fidelity model of the human heart: An immersed boundary implementation, Phys. Rev. Fluids 8 (10) (2023) 100502.

[26] D.A. Fedosov, B. Caswell, G.E. Karniadakis, Systematic coarse-graining of spectrin-level red blood cell models, Comput. Methods Appl. Mech. Eng. 199 (29–32) (2010) 1937–1948.

[27] M. Davey, C. Puelz, S. Rossi, M.A. Smith, D.R. Wells, G. Sturgeon, W.P. Segars, J.P. Vavalle, C.S. Peskin, B.E. Griffith, Simulating cardiac fluid dynamics in the human heart, arXiv preprint, arXiv:2307.02680, 2023.

[28] L. Feng, H. Gao, X. Luo, Whole-heart modelling with valves in a fluid–structure interaction framework, Comput. Methods Appl. Mech. Eng. 420 (2024) 116724.

[29] Y. Jiang, R. Chen, X.-C. Cai, A highly parallel implicit domain decomposition method for the simulation of the left ventricle on unstructured meshes, Comput. Mech. 66 (6) (2020) 1461–1475.

[30] X.-C. Cai, W.D. Gropp, D.E. Keyes, M.D. Tidriri, Newton-Krylov-Schwarz methods in CFD, in: Numerical Methods for the Navier-Stokes Equations, Springer, 1994, pp. 17–30.

[31] X.-C. Cai, M. Sarkis, A restricted additive Schwarz preconditioner for general sparse linear systems, SIAM J. Sci. Comput. 21 (2) (1999) 792–797.

[32] R. Chen, X.-C. Cai, Parallel one-shot Lagrange–Newton–Krylov–Schwarz algorithms for shape optimization of steady incompressible flows, SIAM J. Sci. Comput. 34 (5) (2012) B584–B605.

[33] F. Kong, X.-C. Cai, A highly scalable multilevel Schwarz method with boundary geometry preserving coarse spaces for 3D elasticity problems on domains with complex geometry, SIAM J. Sci. Comput. 38 (2) (2016) C73–C95.

[34] F. Regazzoni, Mathematical modeling and machine learning for the numerical simulation of cardiac electromechanics, 2020.

[35] J.M. Guccione, A.D. McCulloch, L. Waldman, Passive material properties of intact ventricular myocardium determined from a cylindrical model, ASME J. Biomech. Eng. 113 (1991) 42–55.

[36] O.C. Zienkiewicz, R.L. Taylor, The Finite Element Method for Solid and Structural Mechanics, Elsevier, 2005.

[37] F.-N. Hwang, X.-C. Cai, A parallel nonlinear additive Schwarz preconditioned inexact Newton algorithm for incompressible Navier–Stokes equations, J. Comput. Phys. 204 (2) (2005) 666–691.

[38] D.A. Knoll, D.E. Keyes, Jacobian-free Newton–Krylov methods: A survey of approaches and applications, J. Comput. Phys. 193 (2) (2004) 357–397.

[39] E.E. Prudencio, R. Byrd, X.-C. Cai, Parallel full space SQP Lagrange–Newton–Krylov–Schwarz algorithms for PDE-constrained optimization problems, SIAM J. Sci. Comput. 27 (4) (2006) 1305–1328.

[40] J.D. Bayer, R.C. Blake, G. Plank, N.A. Trayanova, A novel rule-based algorithm for assigning myocardial fiber orientation to computational heart models, Ann. Biomed. Eng. 40 (10) (2012) 2243–2254.

[41] S. Balay, S. Abhyankar, M.F. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, A. Dener, V. Eijkhout, W.D. Gropp, D. Karpeyev, D. Kaushik, M.G. Knepley, D.A. May, L.C. McInnes, R.T. Mills, T. Munson, K. Rupp, P. Sanan, B.F. Smith, S. Zampini, H. Zhang, H. Zhang, PETSc users manual, Tech. Rep. ANL-95/11 - Revision 3.13, Argonne National Laboratory, 2020, https://www.mcs.anl.gov/petsc.

[42] G. Karypis, K. Schloegel, V. Kumar, Parmetis: Parallel Graph Partitioning and Sparse Matrix Ordering Library, University of Minnesota, 1997, p. 22.