Computational Mechanics (2020) 66:1461–1475
https://doi.org/10.1007/s00466-020-01912-3

ORIGINAL PAPER

A highly parallel implicit domain decomposition method for the simulation of the left ventricle on unstructured meshes

Yi Jiang[1,2] · Rongliang Chen[1,2] · Xiao-Chuan Cai[3]

Received: 2 May 2020 / Accepted: 17 August 2020 / Published online: 7 September 2020

Abstract
We consider the numerical simulation of the left ventricle of the human heart by a hyperelastic fiber reinforced transversely isotropic model. This is an important model problem for the understanding of the mechanical properties of the human heart but its calculation is very time consuming because the lack of fast, scalable method that is also robust with respect to the model parameters. In this paper, we propose and study a fully implicit overlapping domain decomposition method on unstructured meshes for the discretized system. The algorithm is constructed within the framework of Newton–Krylov methods with an analytically constructed Jacobian. We show numerically that the algorithm is highly parallel and robust with respect to the material parameters, the large deformation, the fiber reinforcement, and the geometry of the patient-specific left ventricle. Numerical experiments show that the algorithm scales well on a supercomputer with more than 8000 processor cores.

Keywords Fiber-reinforced hyperelasticity · Left ventricle · Domain decomposition · Finite element · Parallel computing

1 Introduction

Computational modeling is used increasingly to study the mechanical properties of the human heart [10,14,28,39,41]. To have more realistic simulations, sophisticated material models are being considered and very fine meshes are used to discretize the highly nonlinear partial differential equations. Such simulations are extremely time consuming and require highly parallel algorithms that are also robust with respect the material properties. In this paper, we develop a domain decomposition method for the simulation of the left ventricle described by a hyperelastic model reinforced by transversely isotropic fibers.

Several parallel algorithms for cardiac mechanics have been introduced in the past few years. Among them, one category is developed based on structured or mapped structured grids that are relatively easy to parallelize. For example, in [34], an electro-mechanical model problem was investigated by Reumann et al. on a structured grid, the calculation uses 16,000 processor cores for a mesh with 128 million cells. Pavarino et al. developed a Newton-Krylov type method for a cardiac mechanical model discretized by finite element methods on mapped regular grids, the resulting problem is solved by either an algebraic multigrid [12] or BDDC method [13,31,32]. The method scales well for up to 16000 processor cores for a mesh with 6.5 million degrees of freedom. The other class of methods uses fully unstructured meshes for which the geometrical details of the heart can be preserved, which is essential for patient-specific clinic studies. However, such methods are difficult to parallelize due to the lack of memory alignment, unbalanced computational load and irregular communication requirements. Most of published works in the literature are limited to small number of processors or simulations on coarse meshes [19,29], or using explicit methods [37]. In [21], a very large simulation was carried out for the whole heart using an approach based the Schur complement which is formed explicitly. Generally speaking, forming the Schur complement matrix

✉ Rongliang Chen
rl.chen@siat.ac.cn

Xiao-Chuan Cai
xccai@um.edu.mo

1 Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, Guangdong, China

2 Shenzhen Key Laboratory for Exascale Engineering and Scientific Computing, Shenzhen, Guangdong, China

3 Department of Mathematics, University of Macau, Macau, China

is considered prohibitively expensive, but the authors of [21,30] applied a homogenization technique to the stiffness matrix to reduce considerably the arithmetic complexity of the problem, at the cost of reduced fidelity of the simulation. Another remarkable work is [2], in which Augustin et al. computed an electromechanic problem of the whole heart with a Newton-Krylov method on a machine with 8192 processor cores. However, as pointed out by the authors, the ellipticity required by the method may be lost when the deformation is large [26]. The focus of the present paper is a method without requiring the ellipticity condition.

The aim of this paper is to develop an efficient and robust method for cardiac elastodynamic problems with large deformation. More specifically, a patient-specific human left ventricle is described by a fiber-reinforced hyperelastic model, and the problem is discretized on a fully unstructured tetrahedral mesh by a finite element method in space and a fully implicit method in time. For solving the resulting nonlinear algebraic system, we use an inexact Newton method to deal with the nonlinearity in the governing equations, combined with the GMRES method [36] to solve the Jacobian system. One key ingredient of this approach is that we construct the Jacobian matrix analytically which improves the robustness and computational efficiency considerably. For the scalability, and also for the robust of the method, we adopt a restricted additive Schwarz (RAS) preconditioner [7] which works well without requiring the ellipticity of the Jacobian system [7–9,22]. Numerical experiments show that the proposed algorithm is highly parallel and robust with respect to the material coefficients, the large deformation, and the fiber reinforcement.

The rest of the paper is organized as follows. We first describe a hyperelastic material model with fiber-reinforcement designed for cardiac elastodynamics simulations in Sect. 2. In Sect. 3, we introduce a fully implicit finite element discretization of the model problems on unstructured meshes, and a highly parallel Newton-Krylov-Schwarz method for solving the nonlinear algebraic system at each time step. In Sect. 4, numerical evidence is presented for validating the proposed algorithm and developed numerical solver, some representative simulation results on a patient-specific left ventricle are also demonstrated. At last, we summerize some final remarks in Sect. 5.

2 Description of the material model

Let $\Omega \subset \mathbb{R}^3$ be the undeformed reference domain of an elastomer, and $\mathbf{X} \in \Omega$ a material point. We consider the following elastodynamic system for the displacement $\mathbf{d}(t, \mathbf{X})$ and the velocity $\mathbf{v}(t, \mathbf{X})$ in the Lagrangian coordinate system [11]:

$$\begin{cases} \rho \dfrac{\partial \mathbf{v}}{\partial t} - \nabla \cdot \mathbf{P} = 0 \\ \dfrac{\partial \mathbf{d}}{\partial t} = \mathbf{v} \end{cases} \quad \text{in} \quad \Omega \times (0, T), \tag{1}$$

with the boundary conditions $\mathbf{d} = 0$ on Γ_d and $\mathbf{PN} = \mathbf{f}_T$ on Γ_n, and the initial displacement $\mathbf{d}(0, \mathbf{X}) = \mathbf{d}^0(\mathbf{X})$ and velocity $\mathbf{v}(0, \mathbf{X}) = \mathbf{v}^0(\mathbf{X})$. Here Γ_n and Γ_d are the Neumann and Dirichlet boundaries of Ω, respectively, $\mathbf{N}$ is the outward unit normal vector, and $\mathbf{f}_T$ denotes a traction force on Γ_n. In addition, the first Piola-Kirchhoff stress tensor $\mathbf{P}$ is defined as:

$$\mathbf{P} = \mathbf{FS}, \tag{2}$$

where $\mathbf{S}$ is the second Piola-Kirchhoff stress tensor (to be specified later), $\mathbf{F} = \mathbf{I} + \nabla \mathbf{d}$ is the deformation gradient tensor, $\mathbf{I}$ is the identity tensor of the second rank. Together, (1) and (2) form a general formulation of finite elastodynamics.

The heart is a sophisticated organ that collects and pumps blood via repeatedly diastole and systole of its muscles. Under normal physiological conditions, it has been observed that the human heart can bear large continuum deformations when it is subject to internal and/or external force loadings, and return to its original shape after the loading is removed. The heart muscles are formed by long thin cells known as the cardiac fibers, and they response very differently when the loadings are applied along different directions. These material properties can be conveniently described by a hyperelasticity law through the passive part of the second Piola-Kirchoff tensor. That is, there exists a strain energy functional W, such that

$$\mathbf{S}_{\text{pas}} = \frac{\partial W}{\partial \mathbf{E}}, \tag{3}$$

where $\mathbf{E} = \frac{1}{2}(\mathbf{F}^T\mathbf{F} - \mathbf{I})$ is the Green-Lagrange strain tensor, and $\mathbf{S}_{\text{pas}}$ represents the passive part of $\mathbf{S}$. Further, we assume Ω is occupied by the left ventricle of a heart in the stress-free state, for each point $\mathbf{X} \in \Omega$, there is a fiber described as a three dimensional curve whose tangential direction at $\mathbf{X}$ is denoted by a unit vector $\mathbf{f}(\mathbf{X})$. In this paper, we define

$$W = \frac{C}{2}(e^Q - 1) + \frac{\kappa}{2}[\ln(\det \mathbf{F})]^2. \tag{4}$$

The first term in (4) comes from an transversely isotropic constitutive model (Guccione model [18]), in which

$$\begin{aligned} Q = {} & b_f \widetilde{E}_{11}^2 + b_t(\widetilde{E}_{22}^2 + \widetilde{E}_{33}^2 + \widetilde{E}_{23}^2 + \widetilde{E}_{32}^2) \\ & + b_{fs}(\widetilde{E}_{12}^2 + \widetilde{E}_{21}^2 + \widetilde{E}_{13}^2 + \widetilde{E}_{31}^2), \end{aligned} \tag{5}$$

where $\widetilde{E}_{ab}$ $(a, b = 1, 2, 3)$ denote the components of $\mathbf{E}$ measured in a local Cartesian coordinate system at each $\mathbf{X} \in \Omega$.

More specifically, this local coordinate system is composed of an orthonormal basis $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$ such that the $\mathbf{e}_1$-axis aligns to the fiber direction $\mathbf{f}$ at $\mathbf{X}$. In fact, according to the transformation law of tensors [1],

$$\widetilde{E}_{ab} = (\mathbf{R}[\mathbf{E}]\mathbf{R}^T)_{ab}, \tag{6}$$

where $[\mathbf{E}]$ denotes the matrix representation of tensor $\mathbf{E}$, and $\mathbf{R}$ is a rotation matrix for the coordinate transformation, such that $\mathbf{Rf} = (1, 0, 0)^T$ at each $\mathbf{X} \in \Omega$. In (4) and (5), C is a scaling parameter that measures the bulk stiffness of the material, and the dimensionless coefficients b_f, b_t, b_{fs} specify the exponential weights of the strain energy, for the subscript f representing the fiber normal component, the subscript t the transverse normal component, and the subscript fs the fiber-shear component of the strain tensor $\mathbf{E}$, respectively. Note that $\mathbf{e}_2$ and $\mathbf{e}_3$ can be arbitrarily chosen (as along as they form an orthonormal basis with $\mathbf{e}_1$) and are not distinguished in (5), the transversely isotropy of the material is implied. Besides, the second term of (4) appears due to a volumetric constraint: the human heart is nearly incompressible under deformation. Consequently, the ratio of volume change $\det(\mathbf{F})$ can not deviate from 1 much, and κ stands for a user-defined penalty parameter.

Other than $\mathbf{S}_{\text{pas}}$ that determines the elastic passive property, the elastic dynamics of the heart is prompted by the active contraction of the heart muscles. This mechanism can be described by introducing an internal stress $\mathbf{S}_{\text{act}}$, as the active part of the second Piola-Kirchhoff stress tensor. In this paper, we assume $\mathbf{S}_{\text{act}}$ has the following form [13]:

$$\mathbf{S}_{\text{act}} = T_{\text{act}}\mathbf{f}_0 \otimes \mathbf{f}_0, \tag{7}$$

where T_{act} is the scalar magnitude of the active stress that may vary with time and location, $\mathbf{f}_0$ is a unit vector that points to the fiber direction in the reference configuration, and $\otimes$ denotes the tensor product operator. In the case of (7), it specifies that the contractile force produced by heart muscles is along the direction of the myocardium fibers.

The complete second Piola–Kirchhoff stress $\mathbf{S}$ in (2) is defined by combining the passive and active part:

$$\mathbf{S} = \mathbf{S}_{\text{pas}} + \mathbf{S}_{\text{act}}, \tag{8}$$

and one can show that it has the following explicit form by some straight forward tensor calculations:

$$\mathbf{S} = Ce^{Q}\mathbf{R}^{\mathbf{T}}(\mathbf{A} * [\widetilde{\mathbf{E}}])\mathbf{R} + \kappa \ln(\det \mathbf{F})\mathbf{C}^{-1} + T_{\text{act}}\mathbf{f}_0 \otimes \mathbf{f}_0, \tag{9}$$

where $\mathbf{C} = \mathbf{F}^T\mathbf{F}$ is the right Cauchy–Green tensor, $*$ denotes a component-wise multiplication operator, $\mathbf{A}$ is a coefficient matrix in the form of

$$\mathbf{A} = \begin{pmatrix} b_f & b_{fs} & b_{fs} \\ b_{fs} & b_t & b_t \\ b_{fs} & b_t & b_t \end{pmatrix}, \tag{10}$$

and $[\widetilde{\mathbf{E}}]$ represents the matrix representation of $\mathbf{E}$ relative to the local coordinate basis as described above.

Remark 1 In the literature, several hyperelasticity laws have been proposed for modeling cardiac muscles with different strain energy functions, see e.g. [3,25,33]. One popular choice is the Holzapfel-Ogden (HO) model [20] and its variants (see [17] and the references therein). Comparing to the Guccione model (4) adopted in this paper, the HO-type models involve one more characteristic direction corresponding to the cardiac fiber sheet structure, and more physical parameters, hence is more sophisticated. Since both models share a similar mathematical form as their strain energies consist of similar exponential terms, the numerical method proposed in this paper can be naturally extended to the HO-type models.

3 Methodolgy

In this section, we first present a fully implicit finite element discretization of the elastodynamic problem (1), then we discuss a highly parallel domain decomposition method for solving the nonlinear algebraic system.

3.1 Discretization

Let $\mathscr{D} = \left\{\mathbf{d} \in [H^1(\Omega)]^3 \mid \mathbf{d} = 0 \text{ on } \Gamma_d\right\}$ and $\mathscr{V} = [H^1(\Omega)]^3$, the variational form of (1) reads ([43]): Find $(\mathbf{d}, \mathbf{v}) \in \mathscr{D} \times \mathscr{V}$, such that

$$F((\mathbf{d}, \mathbf{v}), (\varphi, \psi)) = 0 \qquad \forall (\varphi, \psi) \in \mathscr{D} \times \mathscr{V}, \tag{11}$$

where

$$\begin{aligned} &F((\mathbf{d}, \mathbf{v}), (\varphi, \psi)) \\ &\equiv \rho \int_\Omega \frac{\partial \mathbf{v}}{\partial t} \cdot \varphi \mathrm{d}\mathbf{X} + \int_\Omega \mathbf{P} : \nabla\varphi \mathrm{d}\mathbf{X} - \int_{\Gamma_n} \mathbf{f}_T \cdot \varphi \mathrm{d} \\ &\quad + \int_\Omega \frac{\partial \mathbf{d}}{\partial t} \cdot \psi \mathrm{d}\mathbf{X} - \int_\Omega \mathbf{v} \cdot \psi \mathrm{d}\mathbf{X}. \end{aligned} \tag{12}$$

To discretize (11), we assume there is a quasi-uniform tetrahedral mesh T_h on Ω, and create two piecewise linear continuous finite element spaces: $\mathscr{D}_h \subset \mathscr{D}$ and $\mathscr{V}_h \subset \mathscr{V}$. By denoting the finite element solution of (3.1) as $\mathbf{u}_h \equiv (\mathbf{d}_h, \mathbf{v}_h)$, one has

$$F_h(\mathbf{u}_h, \phi_h) = 0, \qquad \forall \phi_h \in \mathscr{U}_h, \tag{13}$$

where $\mathscr{U}_h \equiv (\mathscr{D}_h, \mathscr{V}_h)$, and $\phi_h \equiv (\varphi_h, \psi_h)$ is the finite dimensional test functions.

To obtain a fully discretized form, we apply a backward Euler scheme for the time derivative in (13) [35]. Namely, by denoting $\mathbf{u}_h^n \equiv (\mathbf{d}_h^n, \mathbf{v}_h^n)$, we approximate

$$\frac{\partial \mathbf{u}_h^n}{\partial t} \approx \frac{\mathbf{u}_h^n - \mathbf{u}_h^{n-1}}{\Delta t}, \tag{14}$$

where Δt is the time step size, $t_n = n\Delta t$ is the n^{th} time step, $\mathbf{d}_h^n \equiv \mathbf{d}_h(t_n, \cdot)$ and $\mathbf{v}_h^n \equiv \mathbf{v}_h(t_n, \cdot)$.

Given an initial condition $\mathbf{u}_h^0$ at $t_0 = 0$, then at each t_n for $n = 1, 2, \ldots$, we form and solve the following finite dimensional variational problem: Find $\mathbf{u}_h^n \in \mathscr{U}_h$, such that

$$F_h^n(\mathbf{u}_h^n, \phi_h) = 0 \qquad \forall \phi_h \in \mathscr{U}_h, \tag{15}$$

where

$$\begin{aligned} &F_h^n(\mathbf{u}_h^n, \phi_h) \\ &\equiv \rho \int_\Omega \left(\mathbf{v}_h^n - \mathbf{v}_h^{n-1}\right) \cdot \varphi_h \mathrm{d}\mathbf{X} + \Delta t \int_\Omega \mathbf{P}(\mathbf{d}_h^n) : \nabla \varphi_h \mathrm{d}\mathbf{X} \\ &\quad - \Delta t \int_{\Gamma_n} \mathbf{f}_T \cdot \varphi_h \mathrm{d} \\ &\quad + \int_\Omega \left(\mathbf{d}_h^n - \mathbf{d}_h^{n-1}\right) \cdot \psi_h \mathrm{d}\mathbf{X} - \Delta t \int_\Omega \mathbf{v}_h^n \cdot \psi_h \mathrm{d}\mathbf{X}. \end{aligned} \tag{16}$$

For simplicity, we drop the scripts and write (15) as a system of nonlinear algebraic equations:

$$F(\mathbf{x}) = 0, \tag{17}$$

where $\mathbf{x}$ represents the vector of current nodal values of $\mathbf{u}_h$ to be computed.

3.2 A Newton–Krylov–Schwarz solver

The strain energy of the Guccione model depends exponentially on the strains, thus (17) is highly nonlinear. For the fast convergence and robustness, we choose the Newton-Krylov method for solving (17). That is, we start with an initial guess $\mathbf{x}_0$ of the solution, then an updated solution is computed with $\mathbf{x}_{m+1} = \mathbf{x}_m + \alpha_m \mathbf{y}_m$, where $\alpha_m \in (0, 1]$ is the line search step length [9], and $\mathbf{y}_m$ is the search direction obtained via inexactly solving the Jacobian system:

$$\|J(\mathbf{x}_m)\mathbf{y}_m + F(\mathbf{x}_m)\| \le 10^{-4}\|F(\mathbf{x}_m)\|. \tag{18}$$

Here, J represents the Jacobian matrix of F, whose assembly process will be given later. The Jacobian system (18) is solved by a restarted GMRES method with a right preconditioner B^{-1}:

$$\begin{cases} JB^{-1}\mathbf{z} = -F, \\ \mathbf{y} = B^{-1}\mathbf{z}. \end{cases} \tag{19}$$

There are many ways to construct B^{-1}. In this paper, we focus on an overlapping Schwarz method [7]. First, the finite element mesh T is partitioned into n_p non-overlapping submeshes such that $T = \cup_{i=1}^{n_p} T_i$, for T_i denoting the i^{th} subdomain. Then, each T_i is extended to T_i^δ to overlap with its neighbors, here the superscript δ denotes the number of overlapping layers. Next, we construct a restriction matrix R_i^δ that maps a global vector $\mathbf{z}$ into a sub-vector $\mathbf{z}_i^\delta$ defined on the overlapping subdomain T_i^δ:

$$\mathbf{z}_i^\delta = R_i^\delta \mathbf{z} = \mathbf{z}|_{T_i^\delta}. \tag{20}$$

A restricted additive Schwarz preconditioner B^{-1} can then be defined as

$$B^{-1} = \sum_{i=1}^{n_p} (R_i^0)^T J_i^{-1} R_i^\delta, \tag{21}$$

where J_i denotes the subdomain Jacobian matrix, and R_i^0 is the restriction matrix defined on the non-overlapping subdomain T_i. The subdomain problem corresponding to each J_i is solved approximately using incomplete LU factorization.

For clarity, we summarize the discussion above into a complete algorithm as follows:

Algorithm 1 *A fully implicit time stepping Newton–Krylov–Schwarz method for solving the elastodynamic problem* (1).

1. Set $\mathbf{x}^0$ by the initial data $\mathbf{d}^0$ and $\mathbf{v}^0$.
2. For time step $n = 1, 2, \ldots$, substitute $\mathbf{x}^{n-1}$ into (17), then solve it for $\mathbf{x}^n$ by using an inexact Newton method:
   - 2.1 Set $\bar{\mathbf{x}}_0 = \mathbf{x}^{n-1}$, $m = 0$.
   - 2.2 Compute $F(\bar{\mathbf{x}}_m)$ by (17) and its Jacobian matrix $J(\bar{\mathbf{x}}_m)$ analytically, create a RAS preconditioner $(B_m)^{-1}$ by (21), then set $\mathbf{y}_0 = 0$ and use GMRES to solve the preconditioned linear system (19) for $\mathbf{y}_m$ approximately, s.t.
     $$\|J(\bar{\mathbf{x}}_m)(B_m)^{-1}B_m\mathbf{y}_m + F(\bar{\mathbf{x}}_m)\| \le 10^{-4}\|F(\bar{\mathbf{x}}_m)\|.$$
   - 2.3 Find α_m via a line search method.
   - 2.4 Update $\bar{\mathbf{x}}_{m+1} = \bar{\mathbf{x}}_m + \alpha_m \mathbf{y}_m$.

2.5 Stop the Newton iteration if

$$\|F(\bar{\mathbf{x}}_{m+1})\| \le 10^{-6}\|F(\bar{\mathbf{x}}_0)\| \quad \text{or} \quad \|F(\bar{\mathbf{x}}_{m+1})\| \le 10^{-8},$$

set $\mathbf{x}^n = \bar{\mathbf{x}}_{m+1}$ and go to the next time step; otherwise, let $m = m + 1$ and go to step 2.2.

3.3 Assembly of the Jacobian matrix

The efficiency and the robustness of the proposed method depends on how the Jacobian matrix J is constructed. Instead of the usual approximate Jacobian, we use the exact Jacobian matrix constructed analytically. Later, we will demonstrate the advantages of the analytical Jacobian over a finite difference approximation through numerical experiments. Here, we explain some details about assembling J.

We first introduce some notations. In the following, we use $\mathbf{d}$ as the displacement function in the finite dimensional space, without ambiguity. By writing the vector function $\mathbf{d}$ in the form $\mathbf{d} = (d_1, d_2, d_3)^T$, we expand each component in terms of the finite element basis functions:

$$\mathbf{d} = \begin{pmatrix} d_1 \\ d_2 \\ d_3 \end{pmatrix} = \begin{pmatrix} d_{1,i}\Phi_i \\ d_{2,i}\Phi_i \\ d_{3,i}\Phi_i \end{pmatrix} = \mathbf{d}_i\Phi_i. \tag{22}$$

Here, the subscript i denotes the ith node of the mesh, Φ_i and $\mathbf{d}_i = (d_{1,i}, d_{2,i}, d_{3,i})^T$ represent the ith nodal basis in $\mathscr{U}_h$ and the corresponding nodal displacement vector, respectively. In this paper we assume there is a summation whenever a pair of duplicated indices appear in the same term. Similarly, we write $\mathbf{v}_i = (v_{1,i}, v_{2,i}, v_{3,i})^T$ as the ith nodal velocity.

The Jacobian matrix J can be obtained by taking partial derivatives of F, as defined in (17), with respect to each of the nodal unknowns $d_{1,i}, d_{2,i}, d_{3,i}, v_{1,i}, v_{2,i}, v_{3,i}$. More precisely, J is organized as

$$J = [J_{ij}] \quad \text{for} \quad i, j = 1, 2, \ldots, N, \tag{23}$$

where N is the total number of mesh nodes, and each J_{ij} takes the form of

$$J_{ij} = \begin{bmatrix} \dfrac{\partial F_{\mathbf{d},i}}{\partial \mathbf{d}_j} & \dfrac{\partial F_{\mathbf{d},i}}{\partial \mathbf{v}_j} \\ \dfrac{\partial F_{\mathbf{v},i}}{\partial \mathbf{d}_j} & \dfrac{\partial F_{\mathbf{v},i}}{\partial \mathbf{v}_j} \end{bmatrix}. \tag{24}$$

In (24), $F_{\mathbf{d},i}$ and $F_{\mathbf{v},i}$ are two vectors defined by

$$\begin{bmatrix} \Delta t \displaystyle\int_\Omega \mathbf{P}_{1k}\frac{\partial \Phi_i}{\partial X_k}\mathrm{d}\mathbf{X} + \rho\int_\Omega v_1\Phi_i\mathrm{d}\mathbf{X} \\ \Delta t \displaystyle\int_\Omega \mathbf{P}_{2k}\frac{\partial \Phi_i}{\partial X_k}\mathrm{d}\mathbf{X} + \rho\int_\Omega v_2\Phi_i\mathrm{d}\mathbf{X} \\ \Delta t \displaystyle\int_\Omega \mathbf{P}_{3k}\frac{\partial \Phi_i}{\partial X_k}\mathrm{d}\mathbf{X} + \rho\int_\Omega v_3\Phi_i\mathrm{d}\mathbf{X} \end{bmatrix} \tag{25}$$

and

$$\begin{bmatrix} \displaystyle\int_\Omega d_1\Phi_i\mathrm{d}\mathbf{X} - \Delta t\int_\Omega v_1\Phi_i\mathrm{d}\mathbf{X} \\ \displaystyle\int_\Omega d_2\Phi_i\mathrm{d}\mathbf{X} - \Delta t\int_\Omega v_2\Phi_i\mathrm{d}\mathbf{X} \\ \displaystyle\int_\Omega d_3\Phi_i\mathrm{d}\mathbf{X} - \Delta t\int_\Omega v_3\Phi_i\mathrm{d}\mathbf{X} \end{bmatrix}, \tag{26}$$

respectively. Note that both of the numerator ($F_{\mathbf{d},i}$ or $F_{\mathbf{v},i}$) and the denominator ($\mathbf{d}_j$ or $\mathbf{v}_j$) are vectors, thus each of these partial derivatives in (24) produces a 3-by-3 tensor.

It is straight forward to find that

$$\frac{\partial F_{\mathbf{d},i}}{\partial \mathbf{v}_j} = \rho M, \quad \frac{\partial F_{\mathbf{v},i}}{\partial \mathbf{d}_j} = M, \quad \text{and} \quad \frac{\partial F_{\mathbf{v},i}}{\partial \mathbf{v}_j} = -\Delta t M, \tag{27}$$

where M is a diagonal matrix:

$$M = \begin{bmatrix} \displaystyle\int_\Omega \Phi_i\Phi_j\mathrm{d}\mathbf{X} & 0 & 0 \\ 0 & \displaystyle\int_\Omega \Phi_i\Phi_j\mathrm{d}\mathbf{X} & 0 \\ 0 & 0 & \displaystyle\int_\Omega \Phi_i\Phi_j\mathrm{d}\mathbf{X} \end{bmatrix}. \tag{28}$$

On the other hand, calculating $\partial F_{\mathbf{d},i}/\partial \mathbf{d}_j$ needs considerably more effort. To do so, we notice that for the first term in each row of (25), one has

$$\begin{aligned} &\frac{\partial}{\partial d_{j',j}}\left(\Delta t\int_\Omega \mathbf{P}_{i'k}\frac{\partial \Phi_i}{\partial X_k}\mathrm{d}\mathbf{X}\right) \\ &\quad = \Delta t\int_\Omega \frac{\partial \mathbf{P}_{i'k}}{\partial d_{j',j}}\frac{\partial \Phi_i}{\partial X_k}\mathrm{d}\mathbf{X} \quad \text{for} \quad i', j' = 1, 2, 3. \end{aligned} \tag{29}$$

This suggests that, we first need to find the values of $\partial\Phi_i/\partial X_k$ and $\partial \mathbf{P}_{i'k}/\partial d_{j',j}$ ($k = 1, 2, 3$), at all quadrature points. Calculating the former is standard, while the latter needs the definition of $\mathbf{P}$. Namely, we calculate the partial derivative of $\mathbf{P}$ by first finding the corresponding partial derivatives of $\mathbf{F}$ and $\mathbf{S}$:

$$\frac{\partial \mathbf{P}_{i'k}}{\partial d_{j',j}} = \frac{\partial(\mathbf{FS})_{i'k}}{\partial d_{j',j}} = \frac{\partial \mathbf{F}_{i'k'}}{\partial d_{j',j}}\mathbf{S}_{k'k} + \mathbf{F}_{i'k'}\frac{\partial \mathbf{S}_{k'k}}{\partial d_{j',j}}. \tag{30}$$

Further, to evaluate $\mathbf{S}_{k'k}$ and $\partial \mathbf{S}_{k'k}/\partial d_{j',j}$ ($k' = 1, 2, 3$), a coordinate transformation needs to be performed. This is because part of the definition of $\mathbf{S}$ relies on a local coordinate system (see (9)), while the nodal unknowns $\mathbf{d}_j$ and $\mathbf{v}_j$, the finite element basis and the partial derivatives are described in the global coordinate system. To remedy the gap, at each quadrature point, we first calculate the value of $\mathbf{E}$ according to its definition (see page 2) provided that the current value of $\nabla \mathbf{d}$ has been precomputed in the global coordinate system

$$\nabla \mathbf{d} = \frac{\partial \mathbf{d}}{\partial \mathbf{X}} = \mathbf{d}_i \otimes \frac{\partial \Phi_i}{\partial \mathbf{X}}.$$

Then according to the tensor transformation (6), the value of the Green-Lagrange strain tensor in the local coordinate system is revealed, and $\mathbf{S}$ can be calculated by (9). Here $\mathbf{R}$ represents a rotation matrix for performing the global-to-local coordinate transformation, such that $\mathbf{e}_1$ of the local coordinate system aligns to $\mathbf{f}_0$. Note that although the fiber changes its direction as the deformation happens, one only needs to know its value in the reference configuration and calculates $\mathbf{R}$ once for all. In other words, $\mathbf{R}$ are not changed in time, due to the fact that the governing equation (1) is described via the Lagrangian approach, which simplifies the analysis and eases the implementation. A method proposed in [27] is adopted for calculating $\mathbf{R}$ for improving the numerical stability.

Below, we list the detailed formulae of the partial derivatives of all related tensors for calculating $\partial \mathbf{S}_{k'k}/\partial d_{j',j}$ ($k, j', k' = 1, 2, 3$ and $j = 1, 2, \ldots, N$) as a reference. In practice, they are used in a reversed order for evaluating (30) at each quadrature point and the final assembly of (29).

$$\begin{aligned}
\frac{\partial \mathbf{S}_{k'k}}{\partial d_{j',j}} &= Ce^{Q}\mathbf{R}_{mk'}\mathbf{R}_{nk}\left(\left(2(\mathbf{A} * \widetilde{\mathbf{E}}) : \frac{\partial \widetilde{\mathbf{E}}}{\partial d_{j',j}}\right)(\mathbf{A} * \widetilde{\mathbf{E}})_{mn} + \frac{\partial (\mathbf{A} * \widetilde{\mathbf{E}})_{mn}}{\partial d_{j',j}}\right) \\
&\quad + \kappa\left(\frac{1}{\det \mathbf{F}}\frac{\partial \det \mathbf{F}}{\partial d_{j',j}}\mathbf{C}^{-1}_{k'k} + \ln(\det \mathbf{F})\frac{\partial \mathbf{C}^{-1}_{k'k}}{\partial d_{j',j}}\right), \\
\frac{\partial \mathbf{E}_{mn}}{\partial d_{j',j}} &= \frac{1}{2}\frac{\partial \mathbf{C}_{mn}}{\partial d_{j',j}}, \\
\frac{\partial \det \mathbf{F}}{\partial d_{j',j}} &= \det \mathbf{F}\,\mathrm{tr}\left(\mathbf{F}^{-1}\frac{\partial \mathbf{F}}{\partial d_{j',j}}\right), \\
\frac{\partial \mathbf{C}^{-1}_{k'k}}{\partial d_{j',j}} &= -\mathbf{C}^{-1}_{k'm'}\frac{\partial \mathbf{C}_{m'n'}}{\partial d_{j',j}}\mathbf{C}^{-1}_{n'k}, \\
\frac{\partial \mathbf{C}_{mn}}{\partial d_{j',j}} &= \frac{\partial \mathbf{F}_{lm}}{\partial d_{j',j}}\mathbf{F}_{ln} + \mathbf{F}_{lm}\frac{\partial \mathbf{F}_{ln}}{\partial d_{j',j}}. \\
\frac{\partial \mathbf{F}_{lm}}{\partial d_{j',j}} &= \delta_{lj'}\frac{\partial \Phi_j}{\partial X_m},
\end{aligned}$$

where δ is the Kronecker delta, tr represents the trace operator, the dummy indices l, m, n, m', n' are all summed from 1 to 3.

Remark 2 We remark that the proposed method provides a general framework for different elasticity models, such as the Holzapfel-Ogden model [20]. The expression of the second Piola-Kirchhoff stress $\mathbf{S}$ for the specific model is needed, as well as the corresponding Jacobian matrix J. The discretization scheme, the linear and nonlinear algebraic sol-vers and the preconditioning technique can be used as is.

4 Numerical experiments

In this section, we present some numerical experiments to show the accuracy of the proposed method, its robustness with respect to various physical and numerical parameters, and the parallel performance. The software is developed using PETSc [4] (https://www.mcs.anl.gov/petsc) and the partition of the unstructured meshes are obtained using ParMetis (http://glaros.dtc.umn.edu/gkhome/metis/parmetis/overview). We first consider a well-known benchmark problem to verify our algorithm and the implementation by comparing our results with what was reported in the literature. We then apply the proposed approach to a patient-specific left ventricle, for which we use a projection method to specify the fiber distribution. In the end we show the parallel performance on a supercomputer with a large number of processor cores.

4.1 A benchmark test

To validate the proposed method, we consider a static elastic benchmark problem in [24] (Problem 3). This problem aims to find the deformation of an idealized ellipsoidal domain subject to an active stress over the body and an expansive surface loading on the interior surface. The constitutive model is described in (4) and (5), and both the geometry and the fiber distributions are defined analytically as shown in Fig. 1. In the benchmark test, the model parameters are set as: $\rho = 1.0$ g/cm^3, $C = 2 \times 10^3$ Pa, $b_f = 8.0$, $b_{fs} = 4.0$, and $b_t = 2.0$, the same values will be used for other tests, if not specified otherwise.

As mentioned in [24], 11 research groups reported results for this benchmark problem. In this paper we compare with experiments reported by the CARP group [2] that uses an unstructured tetrahedral mesh with 510,960 elements and 94,622 vertices. In our experiment, we use an unstructured tetrahedral mesh with 4,087,680 elements and 718,945 vertices.

To reach the steady state of the problem, we apply our transient algorithm with 200 pseudo time steps and $\Delta t = 0.005$. The magnitude of the active stress and the surface loading on the interior boundary are set to be 6×10^4 Pa and 1.5×10^4 Pa, respectively, and they are equally increased to the desired values during the time stepping. We carry out the experiment using different values of the incompressibility parameter κ and compare the location of the apex and the volume loss, and the results are summarized in Table 1 and Fig. 2. From Table 1, it can be seen that larger κ corresponds

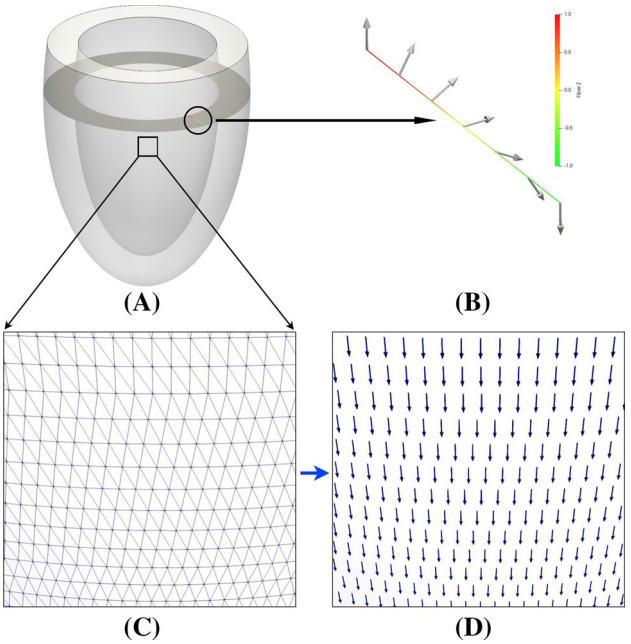

Fig. 1 **a** The geometry of the ellipsoid for the benchmark problem. **b** Fiber direction distributed along a line that penetrates the ellipsoid and lays on the XY-plane. The line is colored according to the Z-component of the unit fiber vector, which changes from -1 (epicardium) to $+1$ (endocardium). **c** and **d** are the surface mesh structure and the fiber direction in a small square area on the epicardium

Table 1 Solutions of the benchmark problem in terms of the Z-component of the displacement of the apex (Apex-Z), the volume of the deformed ellipsoid (Volume), and the loss of the volume in percentage (Vol. Loss), for different κ values. The results reported by the CARP group in [24] are listed on the second row

	Apex-Z	Volume	Vol. loss
Undeformed	0	3233	0
CARP	4.58	3186	1.45%
$\kappa = 1 \times 10^5$	5.39	2983	7.73%
$\kappa = 3 \times 10^5$	4.81	3141	2.85%
$\kappa = 5 \times 10^5$	4.65	3177	1.73%
$\kappa = 8 \times 10^5$	4.52	3192	1.38%

to smaller deformation, and we obtain the same results as in CARP's experiment when κ is 5×10^5 Pa. This shows more clearly in the contour plots of the deformed shapes at a cross section in Fig. 2. We point out that there exists a known singularity at the apex. Since the slight difference does not change the main feature of the solution, we consider our results are acceptable and this serves as a validation of the proposed algorithm.

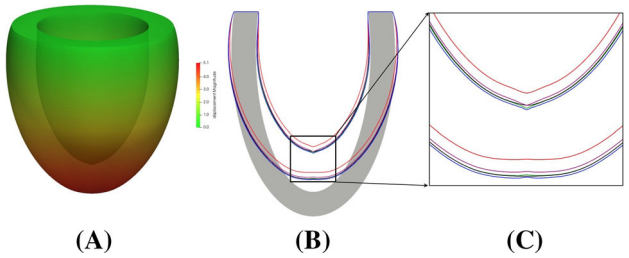

Fig. 2 **a** The benchmark solution obtained by our solver, for the case of $\kappa = 5 \times 10^5$. **b** The cross section outlines on the YZ-plane of the CARP's data (black), and of our results for $\kappa = 1 \times 10^5$ (red), $\kappa = 3 \times 10^5$ (purple), $\kappa = 5 \times 10^5$ (green) and $\kappa = 8 \times 10^5$ (blue). The gray area represents the original undeformed region. **c** Details of the outline in the area near the apex

Table 2 Three finite element meshes used in the numerical experiments

Mesh	# Vertices	# Cells	Element size (mm)
Mesh_c	104,395	496,905	0.8
Mesh_m	745,130	3,975,240	0.4
Mesh_f	5,623,243	31,801,920	0.2

4.2 A patient-specific left ventricle

In the following numerical experiments, we study the left ventricle of a patient-specific human heart for one cardiac cycle by solving the transient problem (1). By varying solver and model parameters, and taking different physical settings, we verify the convergence of the discretization scheme in both space and time, and also investigate the robustness of the proposed algorithm.

We first mention briefly the mesh generation of the left ventricle, the distribution of the fiber orientation and the simulation settings. As shown in Fig. 3a, b, the 3D geometry is reconstructed from a computer tomography (CT) image. Based on the geometry, three conformal quasi-uniform unstructured tetrahedral finite element meshes are generated, denoted as the coarse mesh (Mesh_c), the medium mesh (Mesh_m), and the fine mesh (Mesh_f), each of which has the element size on the epicardium approximately 0.8 mm, 0.4 mm, and 0.2 mm, respectively (see Table 2). Some local mesh structures on the epicardium are presented in Fig. 3d.

A heart fiber is a bundle of combined or connected neighboring cardiomyocytes that aligns locally to a common direction. The human left ventricle is composed of multiple branches of fibers as a complicated 3D coiled structure. Collectively, shrinking and extending of the fibers cause the contraction and relaxation of the ventricle muscles in the macroscopic view. As explained in the previous section, the value of the fiber orientation vector $\mathbf{f}_0$ in the reference configuration is needed for determining the constitutive relation in (9). However, the detailed *in vivo* distribution of the fiber

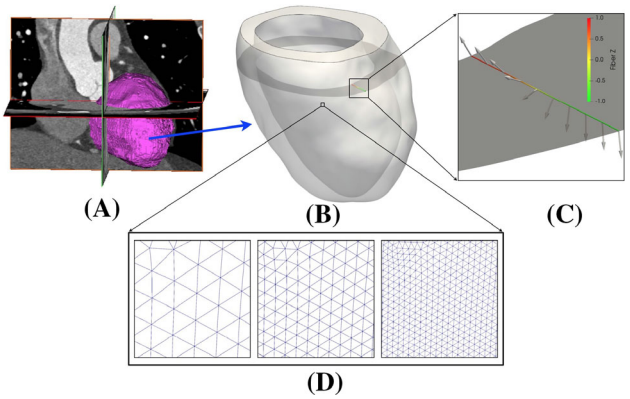

Fig. 3 **a** The CT image of a patient-specific human heart with its left ventricle highlighted in magenta. **b** The geometry of the left ventricle extracted from the CT image. **c** Fiber orientation on a line that penetrates the ventricle wall, this line lays on a horizontal slice (gray) and is colored according to the Z-component of the fiber vector. **d** Local structure in a square region on the epicardium of Mesh_c (left), Mesh_m (middle), and Mesh_f (right), respectively

orientation is difficult to get and not always available [38]. Here, inspired by the work of [42], we describe an algorithmic rule that assigns a fiber direction for each mesh point of the ventricle.

1. We first choose a point **a** on the epicardium as the apex. In our experiments, we select the point at the bottom of the epicardium $(-14.64, -5.73, -30.18)$ mm, which has the shortest distance from the endocardium.
2. For each point **p** on the epicardium and endocardium, we determine its fiber direction by its local geometrical characteristics with the following steps:
   (a) calculate the outward unit normal vector **n**, and the vector that points to the apex $\mathbf{b} = \mathbf{a} - \mathbf{p}$.
   (b) define the local circumferential vector **c** by

   $$\mathbf{c} = \mathbf{n} \times \mathbf{b},$$

   and the local longitudinal vector **l** by

   $$\mathbf{l} = \mathbf{c} \times \mathbf{n}.$$

   (c) the fiber orientation vector $\mathbf{f}_0$ at **p** is obtained by rotating **l** by a certain angle α in the local tangential plane:

   $$\mathbf{f}_0 = \mathbf{l}\cos\alpha + \mathbf{c}\sin\alpha.$$

   Usually, α is chosen differently for epicardium and endocardium points, so that the resulting directions are similar to the actual ventricular fiber distributions found from anatomic observations. In our tests, we set $\alpha = 70°$ on the epicardium and $\alpha = -110°$ on the endocardium.
3. For each of the x, y, z-component of the fiber direction, we regard its values on the epicardium and the endocardium obtained from step 2 as the Dirichlet boundary values of a Laplace equation. Then by solving the three boundary value problems we determine the fiber directions for each mesh point in the interior of the left ventricle.

Table 3 Basic settings used in the patient-specific left ventricle simulations

	Description	Default setting
ρ	Material density	1.0 g/cm^3
C	Bulk stiffness parameter	2.0×10^3 Pa
κ	Incompressible penalty parameter	1.0×10^5 Pa
b_f, b_{fs}, b_t	Strain energy weights	8.0, 4.0, 2.0
T_{act}	Active stress transient function	See Fig. 4
A	Active stress maximum magnitude	6.0×10^4 Pa
Δt	Time step size	See Table 4
Mesh	Finite element mesh	Mesh_m
Jacobian	Jacobian construction method	Analytical
ILU	ILU fill-in level	1
Overlapping	Overlapping size	1
Cores	Number of processor cores	192

In our implementation, each of the three Laplace equations is discretized by using the same finite element method on the same mesh as for the elastodynamic problem (1), and solved by using an additive Schwarz preconditioned Conjugate Gradient solver with the relative tolerance 10^{-8}. Figure 3c shows the fiber orientations along a line that penetrates the ventricle wall, obtained by using this scheme. There are other rule-based methods for fiber generations over a ventricle or heart domain, such as [5,16,40].

Both the model problem and the solution algorithm have adjustable parameters, and all the parameters have certain impact on the accuracy and efficiency of the simulation. Next we study how these factors influence the simulation results by tracking the geometrical change of the left ventricle and their impact on the performance of the numerical solver. In the rest of the paper, stress is measured in Pascal and length is measured in millimeter, and we use "Apex" to denote the displacement of the apex, "Volume" the volume of the left ventricle, "Cavity" the internal cavity, "Area" the area of the endocardium, "NI" the average number of Newton iterations per time step, "LI" the average number of GMRES iterations per Newton step, and "Time" the total compute time in second. The default values of the parameters used in the following experiments can be found in Table 3, if not mentioned otherwise.

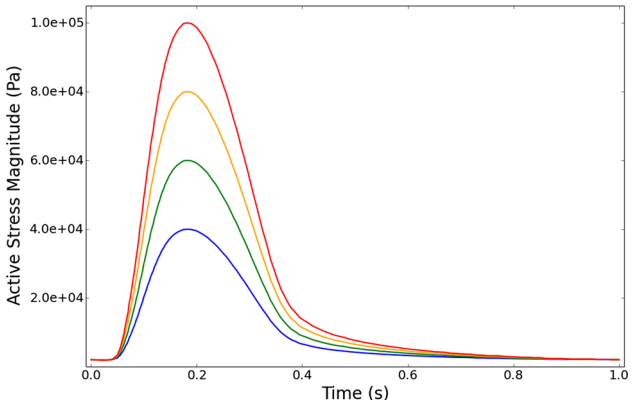

Fig. 4 The transient function of the active stress magnitude T_{act} according to the data proposed in [23]. In each curve, the maximum magnitude A is adjusted to be 4, 6, 8, 10×10^4 Pa, respectively. Here, the case of $A = 6 \times 10^4$ Pa is used as the default in our numerical experiments

Table 4 The nonuniform time step size used in this paper for the active stress magnitude transient mode shown in Fig. 4

	Stage 1	Stage 2	Stage 3	Stage 4
Time interval	[0, 0.05]	[0.05, 0.15]	[0.15, 0.4]	[0.4, 1.0]
Δt	0.01	0.0025	0.005	0.01
steps	5	40	50	60

The total number of time steps for one cardiac cycle is 155

We first consider the active stress that serves as the driving force of the elastodynamics. Customarily, one cardiac cycle can be divided into two phases: the systole phase and the diastole phase. Once the systole phase starts, the active stress in the ventricle increases rapidly to its maximum value. After that, it decreases gradually to and remains at a low level (around 10^3 Pa) until the next cycle starts. In this paper, we assume that the left ventricle experiences the same active stress T_{act} that varies with time t and satisfies the transient mode proposed in [23]. Since one cardiac cycle lasts for 1 second, the maximum magnitude of the active stress is reached at $t = 0.185$ s. Let A be the maximum value of T_a, we test several values of $A = 4, 6, 8, 10 \times 10^4$ Pa, with the corresponding transient curves of T_{act} shown in Fig. 4. To obtain better time accuracy, the cardiac cycle is divided into four stages, and different time step sizes are used in different stages, as specified in Table 4. We run the simulation from $t = 0$ till the active stress reaches the maximum magnitude A (at $t = 0.185$ s, 52 time steps) for each case, the numerical results and solver performance are reported in Table 5. As one can see, when A is larger, the average number of GMRES iterations per Newton step remains almost the same, but the average number of Newton iterations per time step becomes larger and the solver time increases. This indicates that the nonlinear algebraic system is harder to solve as A increases.

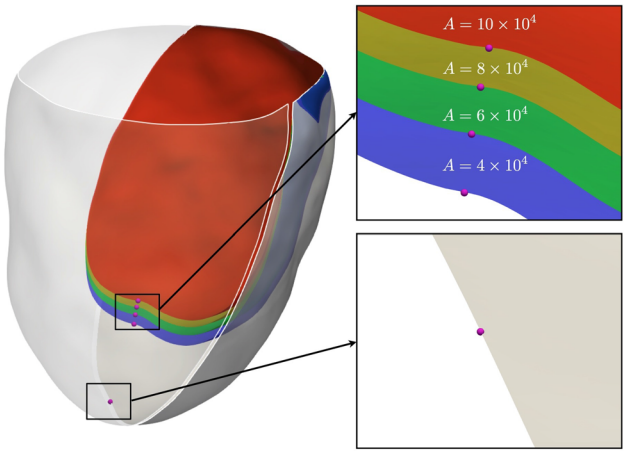

Fig. 5 Left: The epicardium (gray surface) of the undeformed left ventricle is cut through the apex with its left half removed. For the remaining right half of the epicardium, its deformed shape is shown by the blue, green, yellow and red surface for $A = 4, 6, 8, 10 \times 10^4$ Pa, respectively. Right: The bottom panel shows a neighborhood of the apex in the reference configuration, the upper panel shows the deformed shape of the neighborhood for each value of A, respectively

Further, to see how the simulation result changes for different values of A, we display the shape of the deformed epicardium. In Fig. 5, the original epicardium is plotted as the gray surface, in which its left half is removed to reveal the position of the apex (the purple ball at the bottom on the gray surface). For the remaining right half of the epicardium, when the active stress increases to the maximum magnitude as prescribed in Fig. 4, it deforms into the surface in blue ($A = 4 \times 10^4$ Pa), green ($A = 6 \times 10^4$ Pa), yellow ($A = 8 \times 10^4$ Pa) and red ($A = 10 \times 10^4$ Pa), respectively. The new location of the apex is marked by a purple ball on the corresponding surface for each case. It can be seen that all these (half) epicardial surfaces changes in a similar pattern, and a larger A leads to a larger deformation as the final position of the apex is further away from the original position. Since the fibers actively contract with slight rotation in their longitudinal direction and they all intersect at the apex, one can see a rotation in the neighborhood of the apex accordingly.

One expensive component of Algorithm 1 is the construction of the Jacobian matrix in Step 2.2. A popular approach is to approximate the matrix using a multi-colored finite difference method, MCFD ([6,15]), which is easy to program by calling the nonlinear functions several times. However, our experiments show that MCFD is very inefficient due to the many nested functions calls. Instead of MCFD, we use an analytically computed Jacobian matrix, as described in Sect. 3.3. This scheme requires a lot of programming effort but the total compute time is reduced considerably. A comparison of MCFD and the analytic Jacobian is provided in Table 6. As we can see from the table the compute time is

Table 5 Influence of the active stress maximum magnitude A (in 10^4 Pa) on the numerical simulation results and solver performance

A	Apex	Volume	Cavity	Area	NI	LI	Time
4	15.33	59,093.20	31,106.64	4647.19	2.29	198.76	1443.03
6	17.11	56,792.59	27,458.45	4326.40	2.40	201.69	1533.83
8	18.57	54,778.98	24,880.89	4092.02	2.44	206.43	1581.19
10	19.78	52,989.12	22,880.37	3906.38	2.75	204.34	1768.23

Table 6 A comparison of MCFD and analytic Jacobian

Cores	MCFD Jacobian			Analytic Jacobian		
	NI	LI	Time	NI	LI	Time
16	3	74.67	576.34	3	74.67	22.16
32	3	76.67	293.10	3	76.67	11.62
64	3	79.67	149.54	3	79.67	6.31
128	3	83.67	78.57	3	83.67	3.69

The results are for one time step using Mesh_c

Table 7 Impact of the mesh on the computed apex displacement, volume, internal cavity and endocardium area of the left ventricle

Mesh	Apex	Volume	Cavity	Area
Mesh_c	16.88	56984.80	27815.89	4343.75
Mesh_m	17.11	56792.59	27458.45	4326.40
Mesh_f	17.17	56736.28	27339.45	4329.18

reduced from 78.57 seconds to 3.69 seconds when we run the code using 128 processor cores. The number of Newton and GMRES iterations stay the same.

In the following, we determine the right mesh size for the simulation. Three meshes described in Table 2 are considered. We use 24 cores for Mesh_c, 192 cores for Mesh_m, and 1536 cores for Mesh_f. Due to the complexity of the nonlinear elastodynamic system (1) and the shape of the problem domain, the analytical solution of this problem is impossible to get. Hence we regard the result obtained on the finest mesh (Mesh_f) as the reference solution. As shown in Table 7, comparing with the reference, it is clear that the solution accuracy is improved as the mesh size becomes finer: for example, the error of the cavity is reduced form 476.44 mm^3 for Mesh_c to 119 mm^3 for Mesh_m. To further understand the proposed approach, we look at the fiber strain $E_{ff} = (\mathbf{E}\mathbf{f}_0) \cdot \mathbf{f}_0$ that is often considered as an important physiological quantity in some clinic applications. As shown in Fig. 6, the left ventricle is cut vertically, and E_{ff} is plotted on the cross section for each mesh. By comparing the results, especially in the region close to the apex at the bottom, we find that the medium mesh is able to produce almost the same E_{ff} distributions as the fine mesh, but the results from Mesh_c include some noticeable errors and artifacts.

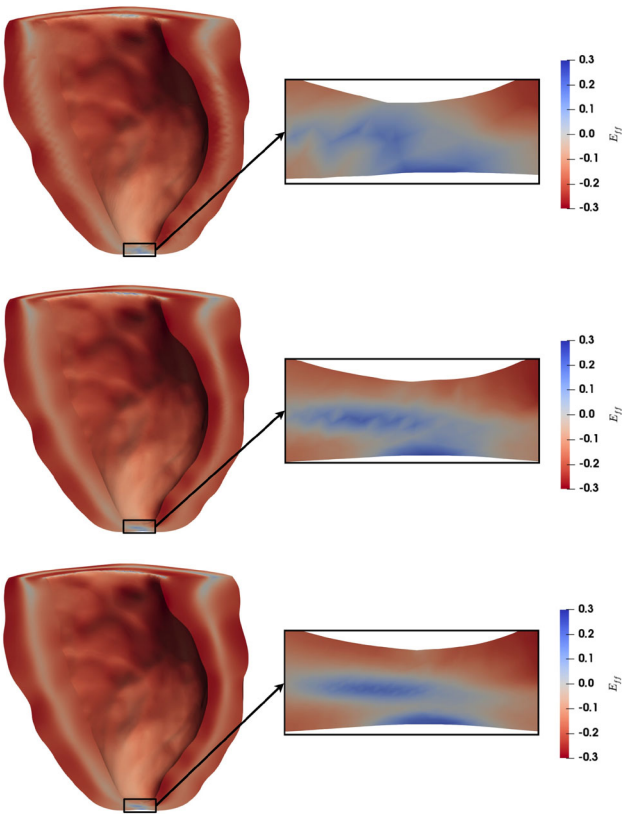

Fig. 6 Influence of the mesh size on E_{ff}, the strain in the fiber direction. The deformed left ventricle is cut vertically to reveal the distribution of E_{ff} on the cross section obtained from Mesh_c (top), Mesh_m (middle) and Mesh_f (bottom), respectively

The time step size Δt is another important parameter influencing the accuracy of the numerical simulation. In the following, we show the impact of Δt on the simulation results and the solver performance. We only run the simulation in the time interval [0 s, 0.2 s] with three values of $\Delta t = 0.1$ s, 0.01 s, and 0.001 s, respectively. The results are collected in Table 8. It can be seen that the main features of the left ventricle, such as the volume and the cavity, are not very sensitive to Δt. Hence, if one is interested in these global values then using a larger time step is reasonable. To understand the influence of Δt more comprehensively, we track the movement of the apex and three representative points on the epicardium. As shown in Fig. 7, in the subfigures, the trajectories are plotted in different colors for different choices

Table 8 Influence of the time step size Δt on the numerical results of patient specific left ventricle cardiac simulation, in terms of the displacements of the apex, the volume, the internal cavity and the endocardium area of the deformed left ventricle, at $t = 0.1$s and 0.2s, respectively

t	Δt	Apex	Volume	Cavity	Area
0.1	0.1	13.78	60,570.58	34, 077.71	4907.27
	0.01	14.05	60,542.76	34, 092.19	4901.45
	0.001	13.09	60,662.18	33, 812.34	4917.49
0.2	0.1	17.09	56,873.52	27, 582.44	4336.41
	0.01	17.08	56,874.85	27, 581.46	4336.60
	0.001	15.64	56,877.98	27, 583.91	4336.77

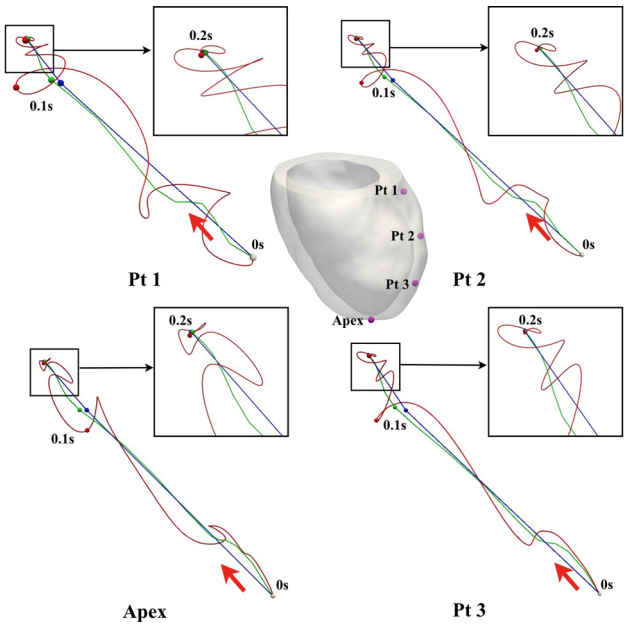

Fig. 7 3D trajectories of the apex and three monitor points for $t \in [0\text{s}, 0.2\text{s}]$. In each subfigure, the white ball at the bottom right corner denotes the original location of the apex or monitor point, which travels from bottom to top. The red, green and blue curve represents the trajectory obtained by setting $\Delta t = 0.001$s, 0.01s and 0.1s, respectively

of Δt (red for 0.001s, green for 0.01s, and blue for 0.1s). The locations of these points at $t = 0.1$s and 0.2s are also marked, as the small balls on the corresponding curves in the same color. It is noticeable that the trajectories of different monitor points share a similar geometrical shape when the same Δt is adopted, and using a relatively large Δt is sufficient to reveal the main features of the movement. These also support the results in Table 8. On the other hand, it shows that more detailed trajectory information with higher resolution can be revealed as Δt is reduced. Hence a sufficiently small Δt should be adopted if the detailed kinematical history is of interest.

Next, we show the kinetic changes of the left ventricle during the cardiac cycle. The calculation is carried out using

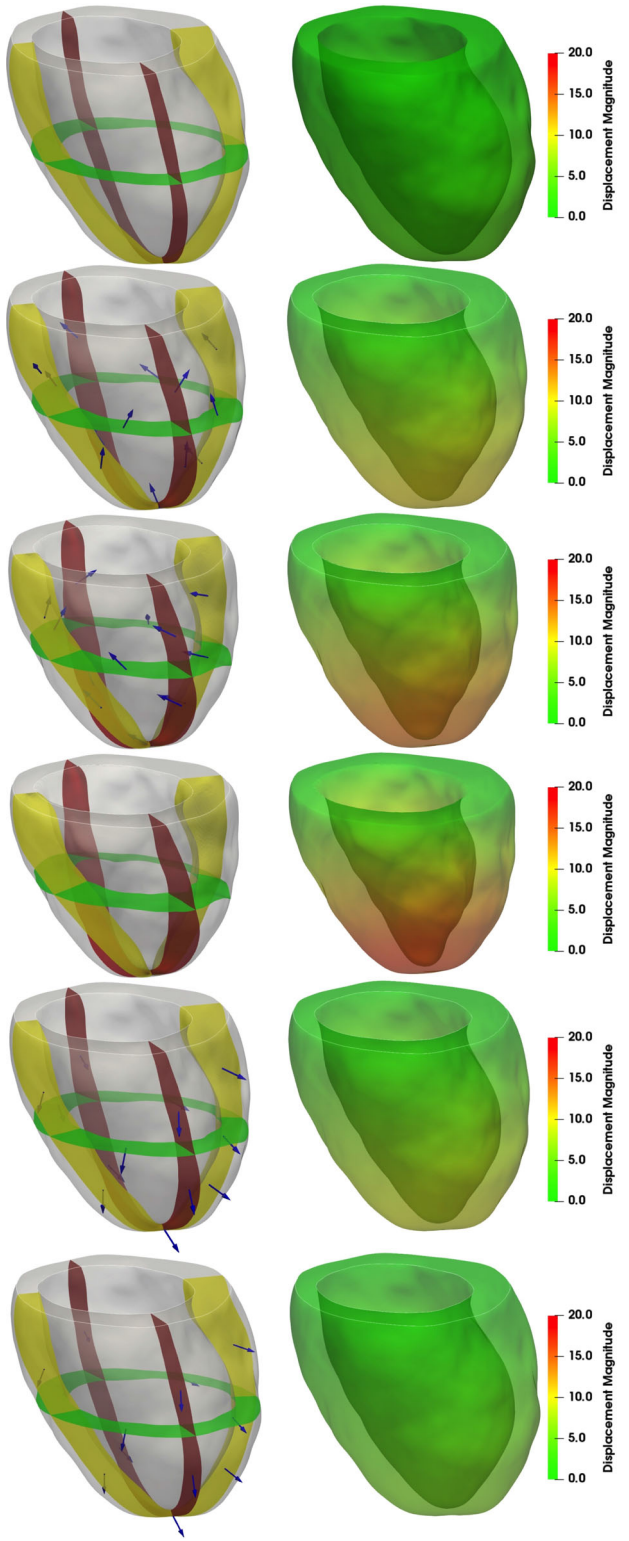

Fig. 8 Evolution of the patient-specific left ventricle during a cardiac cycle. Left column: three representative slices inside the ventricle, the blue arrow represents the moving direction at a monitor point. Right column: perspective view of the deformed left ventricle colored in the displacement magnitude. The snapshots from top to bottom are taken at $t = 0$ s, 0.075 s, 0.1 s, 0.185 s (maximum active stress state), 0.4s and 0.65 s, respectively

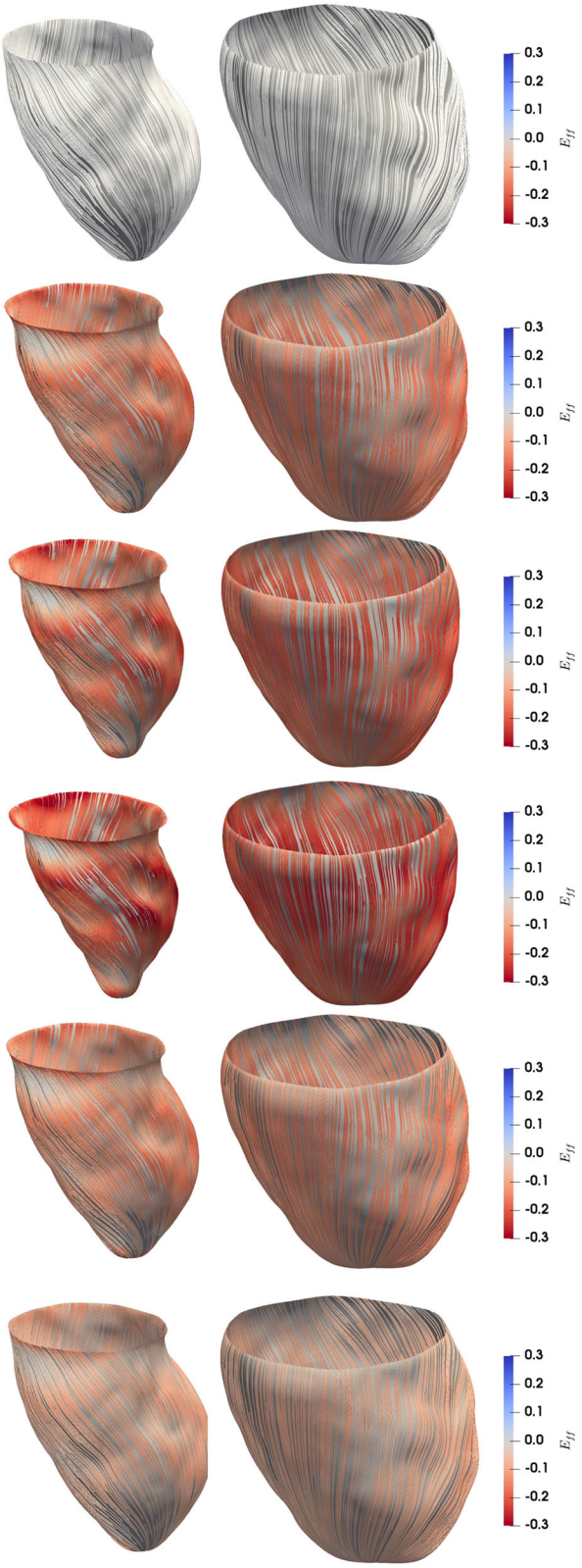

Fig. 9 Evolution of the fiber on the endocardium (left column) and the epicardium (right column) during a cardiac cycle, colored according to the strain along the fiber direction, $\mathbf{E}_{ff}$. The snapshots from top to bottom are taken at $t = 0$s, 0.075s, 0.1s, 0.185 s (maximum active stress state), 0.4s and 0.65s, respectively

Table 9 Influence of the ILU fill-in level and overlapping size on the performance of the solver

ILU	Overlap	NI	LI	Time
0	0,1,2	–	–	–
1	0	2.80	443.07	539.08
	1	2.80	278.25	437.06
	2	2.80	246.07	453.59
2	0	2.80	354.04	611.66
	1	2.80	195.04	486.63
	2	2.80	169.82	510.52

"–" indicates that GMRES fails to converge

Mesh_m. In Fig. 8, we show the deformation along three representative slices. We observe that the left ventricle first contracts due to the sustainably strengthening active stress. When the active stress reaches the maximum magnitude at $t = 0.185$ s, the fibers shrink to the extreme extent, and the velocity at each monitor point reduces to almost zero. After that, the left ventricle starts to expand as the active stress reduces, and eventually restores to its original shape. Besides, the evolutions of the fibers on the surfaces of the left ventricle are shown in Fig. 9, in which the grey surfaces represent the deformed endocardium (left column) and the epicardium (right column), and the fibers are shown as streamlines that are obtained by integrating the deformed fiber direction vector, $\mathbf{Ff}_0/\|\mathbf{Ff}_0\|$, on these surfaces. In addition, the fibers are colored according to the fiber strain E_{ff}, by which one can see that the largest shrinking extent reaches to approximately 30% at the maximum active stress state.

To study the effectiveness and efficiency of the Schwarz preconditioner, we consider two parameters: the ILU fill-in level, and the overlapping size. These two parameters play the key role in the RAS preconditioner, and affect the performance of GMRES convergence significantly. Normally, larger values of these two parameters shall reduce the number of GMRES iterations, but would require more computations and communications in each iteration. In the following, we experiment with the fill-in level 0, 1, 2, and the overlapping size 0, 1, 2, respectively. For each pair of the choices, we record the performance of NKS in Table 9. Since the performance for each time step is similar, the results in the table are for the first 10 time steps. Through the experiments, we find that the linear solver fails to converge when the ILU fill-in level is 0, and the best results in terms of the total compute time is achieved when the overlap is 1 and the level of fill-in is also 1.

Two important parameters in the elastodynamic model are the bulk stiffness C and the incompressibility penalty κ. In Table 10, we show some results using four values of C and four values of κ. As expected, a larger value of C leads to smaller deformation, and a larger value of κ corresponds to

Table 10 Influence of the bulk stiffness C (in 10^3 Pa.) and the incompressible penalty κ (in 10^5 Pa.) on the solver performance

C	κ	Apex	Volume	Cavity	Area	NI	LI	Time
0.5	0.5	18.06	58,454.49	26,766.97	4287.89	3.05	156.92	626.24
	1.0	–	–	–	–	–	–	–
	1.5	–	–	–	–	–	–	–
	2.0	–	–	–	–	–	–	–
1.0	0.5	16.60	58,368.74	30,122.41	4539.45	2.85	140.89	534.05
	1.0	15.34	61348.33	32,188.11	4748.75	2.95	307.56	996.62
	1.5	–	–	–	–	–	–	–
	2.0	–	–	–	–	–	–	–
2.0	0.5	14.48	58,462.41	34,281.68	4877.91	2.70	138.87	508.45
	1.0	13.29	61,370.64	36,317.76	5080.36	2.85	204.28	701.03
	1.5	12.83	62,480.90	37,150.73	5161.59	2.90	338.14	1063.59
	2.0	–	–	–	–	–	–	–
5.0	0.5	10.70	59,001.82	41,416.47	5492.87	2.35	147.68	465.03
	1.0	9.78	61,592.27	43,262.18	5668.96	2.40	190.90	563.56
	1.5	9.41	62,613.04	44,019.14	5740.28	2.45	245.18	665.89
	2.0	9.21	63,161.17	44,437.91	5779.37	2.65	335.23	863.93

"–" indicates that GMRES fails to converge

Table 11 Influence of the strain energy weights b_f, b_{fs}, b_t on the solution accuracy and solver performance

b_f, b_{fs}, b_t	Apex	Volume	Cavity	Area	NI	LI	Time
16, 4, 1	13.61	60,588.93	35,754.26	5024.78	2.90	219.22	751.81
8, 4, 2	13.29	61,370.64	36,317.76	5080.36	2.85	204.28	701.03
4, 4, 4	12.32	60,692.88	40,723.83	5430.51	2.80	197.29	671.98

smaller loss of the volume of the left ventricle. We remark that reducing C and/or increasing κ increase numerical difficulties, as the number of GMRES iterations increases and sometimes the linear solver fails to converge.

We also consider different values of the strain energy weights b_f, b_{fs}, b_t. By definition, these parameters determine how different normal and shear strain components contribute to the strain energy, thus characterize the stiffness of the material against shape changing in different directions relative to the fiber. Histologically, it is observed that the fiber is stiffer along its longitudinal direction than the transverse direction, hence b_f takes a larger value than b_{fs} and b_t in the default setting. Here, we consider two situations: the first is to enhance the fiber effect by increasing b_f, the other is to assume they have the same value, in which case the material is isotropic. The results in Table 11 show that the performance of the proposed method does not change significantly, indicating the robustness of the proposed method with respect to the strain energy weights.

Finally, we study the parallel scalability of the method. The mesh used in this study has about 30 million cells. In this test, the incompressibility parameter $\kappa = 1 \times 10^4$ Pa and the time step size $\Delta t = 0.004$ s, all other parameters are the same as for the previous tests. We run the experiment by using 1024, 2048, 4096, 8192 processor cores, respectively.

Table 12 Strong scalability of the numerical solver based on the first 10 time steps

Cores	NI	LI	Time	Speedup	Efficiency (%)
1024	2.7	158.9	627.6	1.00	100
2048	2.6	161.5	323.7	1.94	96.9
4096	2.6	165.9	150.2	4.18	104.5
8192	2.6	168.9	105.5	5.95	74.4

The results are summarized in Table 12 and Fig. 10, in which the parallel speedup and the efficiency are exhibited for each case. It can be seen that the proposed method has a close to linear scalability when the number of cores goes from 1024 to 4096, and maintains a parallel efficiency higher than 70% with more than 8000 cores. These results are based on the first 10 time steps.

5 Final remarks

In this paper, we investigated a domain decomposition based parallel algorithm for the numerical simulation of the left ventricle of a human heart by a hyperelastic fiber reinforced transversely isotropic model that is discretized by a

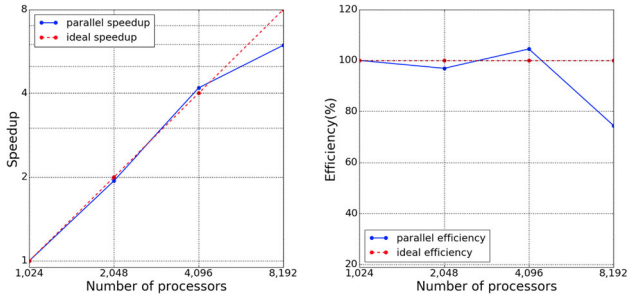

Fig. 10 Parallel speedup and efficiency of the left ventricle cardiac cycle simulation

fully implicit finite element method on a unstructured mesh. At each time step, an Newton-Krylov-Schwarz method is applied to solve the resulting nonlinear algebraic system, in which an analytically computed Jacobian matrix is employed that makes the overall method more robust and efficient. The proposed method is first verified on a benchmark problem and then applied to a patient-specific left ventricle discretized on a mesh with more than 30 million degrees of freedom. Numerical experiments show that the algorithm is robust with respect to the large deformation of the ventricle and also the material parameters, such as the bulk stiffness, the strain energy weights and the incompressible penalty, and it scales well on a supercomputer with more than 8000 processor cores.

Acknowledgements This work was partially supported by the National Key R&D Program of China 2016YFB0200601, Shenzhen grants under ZDSYS201703031711426 and JCYJ20180507182506416, and the Strategic Priority Research Program of Chinese Academy of Sciences, Grant No. XDC01040100.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Aris R (2012) Vectors, tensors and the basic equations of fluid mechanics. Courier Corporation, Chelmsford
2. Augustin CM, Neic A, Liebmann M, Prassl AJ, Niederer SA, Haase G, Plank G (2016) Anatomically accurate high resolution modeling of human whole heart electromechanics: a strongly scalable algebraic multigrid solver method for nonlinear deformation. J Comput Phys 305:622–646
3. Avazmohammadi R, Hill M, Simon M, Zhang W, Sacks M (2017) A novel constitutive model for passive right ventricular myocardium: evidence for myofiber-collagen fiber mechanical coupling. Biomech Model Mechanobiol 16(2):561–581
4. Balay S, Abhyankar S, Adams M, Brown J, Brune P, Buschelman K, Dalcin L, Dener A, Eijkhout V, Gropp W, Karpeyev D, Kaushik D, Knepley M, May D, McInnes L, Mills R, Munson T, Rupp K, Sanan P, Smith B, Zampini S, Zhang H, Zhang H (2020) PETSc Users Manual Revision 3.13, 2020
5. Bayer JD, Blake RC, Plank G, Trayanova NA (2012) A novel rule-based algorithm for assigning myocardial fiber orientation to computational heart models. Ann Biomed Eng 40(10):2243–2254
6. Bozdağ D, Catalyurek U, Gebremedhin AH, Manne F, Boman EG, Özgüner F (2005) A parallel distance-2 graph coloring algorithm for distributed memory computers. International conference on high performance computing and communications. Springer, New York, pp 796–806
7. Cai XC, Sarkis M (1999) A restricted additive Schwarz preconditioner for general sparse linear systems. SIAM J Sci Comput 21(2):792–797
8. Cai XC, Gropp WD, Keyes DE, Tidriri MD (1994) Newton–Krylov–Schwarz methods in CFD. Numerical methods for the Navier–Stokes equations. Springer, New York, pp 17–30
9. Chen R, Cai XC (2012) Parallel one-shot Lagrange–Newton–Krylov–Schwarz algorithms for shape optimization of steady incompressible flows. SIAM J Sci Comput 34(5):B584–B605
10. Chen W, Gao H, Luo X, Hill N (2016) Study of cardiovascular function using a coupled left ventricle and systemic circulation model. J Biomech 49(12):2445–2454
11. Ciarlet PG (1988) Mathematical elasticity, volume i: three-dimensional elasticity. North-Holland
12. Franzone PC, Pavarino LF, Scacchi S (2015) Parallel multilevel solvers for the cardiac electro-mechanical coupling. Appl Numer Math 95:140–153
13. Franzone PC, Pavarino LF, Scacchi S (2018) A numerical study of scalable cardiac electro-mechanical solvers on HPC architectures. Front Physiol 9:268
14. Gao H, Wang H, Berry C, Luo X, Griffith BE (2014) Quasi-static image-based immersed boundary-finite element model of left ventricle under diastolic loading. Int J Numer Meth Bio 30(11):1199–1222
15. Gebremedhin AH, Manne F, Pothen A (2005) What color is your Jacobian? Graph coloring for computing derivatives. SIAM Rev 47(4):629–705
16. Göktepe S, Kuhl E (2010) Electromechanics of the heart: a unified approach to the strongly coupled excitation-contraction problem. Comput Mech 45(2–3):227–243
17. Guan D, Ahmad F, Theobald P, Soe S, Luo X, Gao H (2019) On the AIC-based model reduction for the general Holzapfel–Ogden myocardial constitutive law. Biomech Model Mechanobiol 18(4):1213–32
18. Guccione JM, McCulloch AD, Waldman L (1991) Passive material properties of intact ventricular myocardium determined from a cylindrical model. ASME J Biomech Eng 113:42–55
19. Gurev V, Pathmanathan P, Fattebert JL, Wen HF, Magerlein J, Gray RA, Richards DF, Rice JJ (2015) A high-resolution computational model of the deforming human heart. Biomech Model Mechanobiol 14(4):829–849
20. Holzapfel GA, Ogden RW (2009) Constitutive modelling of passive myocardium: a structurally based framework for material characterization. Philos Trans R Soc A: Math Phys Eng Sci 367(1902):3445–3475
21. Hosoi A, Washio T, Okada JI, Kadooka Y, Nakajima K, Hisada T (2010) A multi-scale heart simulation on massively parallel computers. In: SC'10: Proceedings of the 2010 ACM/IEEE international conference for high performance computing, networking, storage and analysis. IEEE, pp 1–11
22. Kong F, Cai XC (2016) 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):C73–C95

23. Land S, Niederer SA, Smith NP (2011) Efficient computational methods for strongly coupled cardiac electromechanics. IEEE Trans Biomed Eng 59(5):1219–1228
24. Land S, Gurev V, Arens S, Augustin CM, Baron L, Blake R, Bradley C, Castro S, Crozier A, Favino M et al (2015) Verification of cardiac mechanics software: benchmark problems and solutions for testing active and passive material behaviour. Proc R Soc A: Math Eng Sci 471(2184):20150641
25. Langdon S, Chernecky R, Pereira C, Abdulla D, Lee J (1999) Biaxial mechanical/structural effects of equibiaxial strain during crosslinking of bovine pericardial xenograft materials. Biomaterials 20(2):137–153
26. Merodio J, Ogden R (2006) The influence of the invariant I8 on the stress-deformation and ellipticity characteristics of doubly fiber-reinforced non-linearly elastic solids. Int J Non-Linear Mech 41(4):556–563
27. Möller T, Hughes JF (1999) Efficiently building a matrix to rotate one vector to another. J Graph Tools 4(4):1–4
28. Nash MP, Hunter PJ (2000) Computational mechanics of the heart. J Elast Phys Sci Solids 61(1–3):113–141
29. Nobile F, Quarteroni A, Ruiz-Baier R (2012) An active strain electromechanical model for cardiac tissue. Int J Numer Meth Biol 28(1):52–71
30. Okada J, Washio T, Hisada T (2010) Study of efficient homogenization algorithms for nonlinear problems. Comput Mech 46(2):247–258
31. Pavarino L, Scacchi S, Zampini S (2015) Newton–Krylov-BDDC solvers for nonlinear cardiac mechanics. Comput Method Appl Mech Eng 295:562–580
32. Pavarino L, Scacchi S, Verdi C, Zampieri E, Zampini S (2017) Scalable BDDC algorithms for cardiac electromechanical coupling. Domain decomposition methods in science and engineering XXIII. Springer, New York, pp 261–268
33. Pinto J, Fung Y (1973) Mechanical properties of the heart muscle in the passive state. J Biomech 6(6):597–616
34. Reumann M, Fitch BG, Rayshubskiy A, Keller DU, Seemann G, Dossel O, Pitman MC, Rice JJ (2009) Strong scaling and speedup to 16,384 processors in cardiac electro-mechanical simulations. In: 2009 annual international conference of the IEEE Engineering in Medicine and Biology Society, pp 2795–2798
35. Riviere B (2008) Discontinuous Galerkin methods for solving elliptic and parabolic equations: theory and implementation. SIAM, Philadelphia
36. Saad Y (2003) Iterative methods for sparse linear systems. SIAM, Philadelphia
37. Santiago A, Aguado-Sierra J, Zavala-Aké M, Doste-Beltran R, Gómez S, Arís R, Cajas JC, Casoni E, Vázquez M (2018) Fully coupled fluid-electro-mechanical model of the human heart for supercomputers. Int J Numer Meth Biol 34(12):e3140
38. Scollan DF, Holmes A, Winslow R, Forder J (1998) Histological validation of myocardial microstructure obtained from diffusion tensor magnetic resonance imaging. Am J Physiol-Heart C 275(6):H2308–H2318
39. Shen JJ (2016) A structurally based viscoelastic model for passive myocardium in finite deformation. Comput Mech 58(3):491–509
40. Toussaint N, Sermesant M, Stoeck CT, Kozerke S, Batchelor PG (2010) In vivo human 3D cardiac fibre architecture: reconstruction using curvilinear interpolation of diffusion tensor images. International conference on medical image computing and computer-assisted intervention. Springer, New York, pp 418–425
41. Wang H, Luo X, Gao H, Ogden R, Griffith B, Berry C, Wang T (2013) A modified Holzapfel–Ogden law for a residually stressed finite strain model of the human left ventricle in diastole. Biomech Model Mechanobiol. https://doi.org/10.1007/s10237-013-0488-x
42. Wong J, Kuhl E (2014) Generating fibre orientation maps in human heart models using Poisson interpolation. Comput Methods Biomech Biomed Eng 17(11):1217–1226
43. Zienkiewicz OC, Taylor RL (2005) The finite element method for solid and structural mechanics. Elsevier, Amsterdam