



CardioModel – New Software for Cardiac Electrophysiology Simulation

Valentin Petrov¹, Sergey Lebedev¹, Anna Pirova¹,
Evgeniy Vasilyev¹, Alexander Nikolskiy^{1,2}, Vadim Turlapov¹,
Iosif Meyerov¹, and Grigory Osipov¹(✉)

¹ Lobachevsky State University of Nizhni Novgorod, Nizhny Novgorod, Russia
valentin.s.petrov@gmail.com,
sergey.a.lebedev@gmail.com, anna.pirova@itmm.unn.ru,
eugene.unn@gmail.com, yahtingman@rambler.ru,
vadim.turlapov@gmail.com, meerov@vmk.unn.ru,
grosipov@gmail.com

² Nizhni Novgorod State Medical Academy, Nizhny Novgorod, Russia

Abstract. The rise of supercomputing technologies during the last decade has enabled significant progress towards the invention of a personal biologically relevant computer model of a human heart. In this paper we present a new code for numerical simulation of cardiac electrophysiology on supercomputers. Having constructed a personal segmented tetrahedral grid of the human heart based on a tomogram, we solve the bidomain equations of cardiac electrophysiology using the finite element method thus achieving the goal of modeling of the action potential propagation in heart. Flexible object-oriented architecture of the software allows us to expand its capabilities by using relevant cell models, preconditioners and numerical methods for solving SLAEs. The results of numerical modeling of heart under normal conditions as well as a number of simulated pathologies are in a good agreement with theoretical expectations. The software achieves at least 75% scaling efficiency on the 120 ranks on the Lobachevsky supercomputer.

Keywords: Heart simulation · Cardiac electrophysiology · Bidomain model
Finite element method · Numerical analysis · Parallel computing

1 Introduction

The unique capabilities of the mathematical modeling method combined with the growth of the performance of supercomputers open new horizons in biomedical research. One of the directions of such studies is the invention of a personal biologically relevant 3D computer model of a human heart. There are a lot of practical applications of such model, such as the extraction of new knowledge about the processes in the heart, study of the effects of drugs, generation of synthetic data for the development and tuning of novel diagnostics methods, and the construction of realistic training simulators for educational purposes. Studies in the field of numerical modeling of processes in the human heart using supercomputers are being actively pursued in

many research groups [1–12]. Current research areas include the development of methods for reconstructing a personal 3D model of heart and vessels according to computer tomography, automated segmentation of CT images, the creation of a mathematical and computer electrophysiological and electro-mechanical model of the whole human heart and its basic elements. The development of methods for the effective use of supercomputers for solving problems of numerical modeling of processes in the heart is one of the state-of-the-art topics of the current research.

In this paper, we present CardioModel – new software for numerical modeling of the human heart electrophysiology using supercomputers. The main motivation for development of this software is the abundance of open scientific problems the two main topics: (i) personalization and increasing the relevance of models of heart, (ii) increasing the efficiency of computational resources usage. Our software is based on the numerical solution of monodomain or bidomain equations [14, 15] that describe electrical activity in heart. We discretize the equations in time and space. Time discretization is performed using an explicit scheme. Space discretization is done by representing the patient’s heart tissue domain with a three-dimensional tetrahedral grid. The structure of the grid is given by a finite-element Atlas model of the human heart [28]. The Atlas model is anatomically segmented by marking the affiliation of the grid nodes to specific anatomical segments (left, right, ventricles, atria, etc.) [29]. The conduction system of heart is represented as a graph and is connected with the grid. In order to achieve personalization of the Atlas model we merge the vertices of the tetrahedral mesh with an array of the patient’s heart CT or MRI data. To increase the reliability of simulation results we employ biologically relevant cell models available from the CellML repository taking into account the segmentation of the grid to specific regions. Our software is implemented in C++ and has a flexible object-oriented architecture. That allows us easily extending the code with cell models and numerical schemes to take into account additional physical effects. We employ the MPI technology to parallelize the code on distributed memory. The ParMetis library [16] is used to distribute the grid over nodes of a supercomputer. For the SLAE solution we use iterative methods from the PETSc library [17] or the implementations of direct methods developed at Intel [18] and at UNN [19]. The project is based on the experience gained earlier in the development of the Virtual Heart software [20].

The rest of the paper is organized as follows. Section 2 provides an overview of similar works. Section 3 describes a mathematical model. The numerical scheme is formulated in Sect. 4. A parallel algorithm is given in Sect. 5. Section 6 provides an implementation overview. In Sect. 7 we present selected results of numerical simulation and discuss performance data.

2 Related Work

In 1993 the Physiome Project (<http://physiomeproject.org>) was presented aimed at “a quantitative description of the physiological dynamics and functional behavior of the intact organism” [1]. The CellML model repository [2] is one of the most important components of the project. It contains computer implementations of mathematical models of human and animal cells for use in numerical simulations.

The cardiac computational modeling software Alya Red CCM [3] is created in the Barcelona Supercomputing Center. The model underlying the code includes electrophysiological and electromechanical activity of the heart, as well as blood flow through its cavity. Electrophysiological and mechanical components are simulated on the same grid and the modeling of the blood flow is associated with the mechanics of the heart. The rabbit ventricular mesh developed in the University of Oxford [4] is used in simulations. The tool is part of the Alya software – the multiphysics simulation code [5] being developed by BSC.

Essential results of electrophysiology and electromechanics modeling of the heart are obtained in the Computational Cardiology Lab in the Johns Hopkins University [6]. The laboratory implements full heart simulation. Much attention is paid to the applications of the model: the reproduction of normal and pathological cardiac activity. Studies are underway to personalize the heart model [7].

Research centers of IBM in conjunction with the Lawrence Livermore National Laboratory carry out the Cardiod Cardiac Modeling project. The software developed by the team simulates the heart at several levels: electrophysiology of the heart tissue, electromechanics of the whole heart, and also modeling of the tissue fibers [8].

In the REO laboratory of INRIA a group of researchers is working on the modeling of electrophysiology of the heart using the model based on the description of surfaces [9]. The proposed approach allows reproducing typical heart pathologies. The group also deals with the inverse problem of electrophysiology – determining the parameters of the model from a given cardiogram, numerical modeling of cardiograms and creating simplified models of electrophysiology.

Remarkable results in numerical simulation of the heart are achieved by a team of researchers at the University of Oxford. The team develops the Chaste software aimed at solving computationally intensive problems in biology and physiology on supercomputers [13]. The part of the software, Cardiac Chaste, is capable of doing simulations of electro-mechanical processes in the heart. The software is publicly available.

Research in this direction is done in the institutes of the Russian Academy of Sciences (RAS) and in the universities. In the Institute of Numerical Mathematics of the RAS and MIPT, studies are underway to model the human cardiovascular system [10, 11]. Research on the human heart left ventricle modeling is performed in the Institute of Mathematics and Mechanics of the Ural branch of the RAS. The team proposed a new model of the left ventricle of heart which allows personalization according to MRI data. The software for simulation was also developed [12].

In general, there is a considerable interest in developing a biologically relevant personalized computer model of the human heart. One of the steps in this direction requires the development of parallel algorithms and their implementations capable of carrying out extremely computationally intensive calculations on supercomputers.

3 Model

The bidomain model is one of the most common continuous mathematical models describing the electrical activity of the heart. The model was proposed by Schmidt in 1969 [14] and was first formalized by Tang in 1978 [15]. The model assumes that the

heart tissue consists of two continuous domains separated by a cell membrane: the inner space of the cells of the cardiac tissue (intracellular domain) and the space between the cells (extracellular domain). Interaction between domains is described by the flux through the cell membrane. The description of each domain includes the relationship between the electric field obtained from the potential, the current density, and the conductivity tensor.

Next, we give a formal description of the bidomain model. Let some domain Ω be filled with cardiac tissue. The tissue is considered as a set of two continuous domains – intracellular and extracellular, interacting through the cell membrane. Let \mathbf{u} be a set of variables describing the tissue cell, φ_m and φ_e – intra- and extracellular potentials, $V = \varphi_m - \varphi_e$ – the transmembrane voltage, σ_m and σ_e – intra- and extracellular conductivity tensors, I_m^{st} and I_e^{st} – stimulus currents inside and outside the cell per unit area, χ – surface area to volume ratio, C_m – membrane capacity per unit area, I_{ion} – ion current per unit area. The functions I_{ion} and $\frac{\partial \mathbf{u}}{\partial t}$ are defined by the model of the cell. The bidomain model is then given by the following set of equations:

$$\begin{aligned} \chi(C_m \frac{\partial V}{\partial t} + I_{ion}(\mathbf{u}, V)) - \nabla \cdot (\sigma_m \nabla(V + \varphi_e)) &= I_m^{st} \\ \nabla \cdot ((\sigma_m + \sigma_e) \nabla \varphi_e + \sigma_m \nabla V) &= -I_m^{st} - I_e^{st} \\ \frac{\partial \mathbf{u}}{\partial t} &= f(\mathbf{u}, V) \\ \mathbf{n} \cdot (\sigma_m \nabla(V + \varphi_e)) &= 0; \mathbf{n} \cdot (\sigma_e \nabla \varphi_e) = 0 \end{aligned} \quad (1)$$

4 Method

We employ the following numerical scheme to get the solution of (1) in time. The bidomain equations are discretized in time and space. Time discretization is performed using an explicit scheme. Space discretization is done by dividing the heart tissue domain with a three-dimensional tetrahedral mesh; the finite element method is applied. Numerical solution of the equations is performed using the operator splitting method [24], which allows solving linear and nonlinear parts of the system (1) independently. This approach is commonly used to simulate electrophysiological activity [25–27]. According to the operator splitting method, the first equation of system (1) can be represented as follows: $\frac{\partial V}{\partial t} = (\Gamma_1 + \Gamma_2)V$, where Γ_1 and Γ_2 are nonlinear and linear differential operators, respectively. In the bidomain model $\Gamma_1 = \frac{-I_{ion}}{C_m}$, $\Gamma_2 = \frac{\nabla \cdot (\sigma_m \nabla(V + \varphi_e)) + I_m^{st}}{\chi C_m}$. Then the solution of system (1) is calculated as follows.

1. Integrate the ODE system (2) over time with a step size $\frac{\Delta t}{2}$ using computed values in time t as initial conditions:

$$\begin{cases} \frac{\partial V}{\partial t} = \Gamma_1 V \\ \frac{\partial \mathbf{u}}{\partial t} = f(\mathbf{u}, V) \end{cases} \quad (2)$$

2. Integrate the PDE $\frac{\partial V}{\partial t} = I_2 V$ over time with a step size Δt using the values computed during the step 1 as initial conditions.
3. Integrate the ODE system (2) over time with a step size $\frac{\Delta t}{2}$ using the values computed during the step 2 as initial conditions.

We employ the fourth order Runge–Kutta method to solve the ODE system, the finite element method to solve PDE system, and the Euler method for time discretization. The SLAE systems arising during the application of the finite element method are solved by means of relevant iterative methods. Additional information on the numerical integration of the bidomain equations can be obtained in the reviews [21–23].

The described approach allows us to find the solution of the bidomain equations. However, in order to perform fine tuning of the model behavior one has to adjust multiple parameter values in a large number of equations. Therefore, we developed a special scheme to simplify tuning. In this regard we represent a conduction system (CS) of the heart as a graph connecting a sinoatrial node, an atrioventricular node, left and right legs of the His bundle and Purkinje fiber. Vertices of the graph correspond to the regions of the heart, and the edges show the presence of conduction pathways up to the terminal nodes. Additional graph vertices can be added to improve the quality of the model. Each vertex of the graph is associated with the biological model of the cell from the CellML repository, the coupling coefficients with other nodes of the conduction system, and also the region of influence on other cells of the heart.

Each cell in the conduction system is an oscillator with its own natural frequency. Tuning those frequencies along with the coupling strengths (gap junction conductance) allows achieving the globally synchronous regime in the conduction system with the desired delays of the pulse propagation on different segments of the graph.

The dynamics of the conduction system is simulated in parallel with the main solver pipeline. The cells of the conduction system in heart are electrically coupled with myocardium through gap junctions. This gives a way to connect the dynamics of the conduction system with the dynamics of the excitable cells of cardiac tissue obtained in the main solver pipeline. Particularly, each time step of simulation the membrane potentials of the conduction system cells are coupled with the membrane potentials of the tissue cells with the diffusive coupling terms. Each node of the conduction system graph has a parameter R that describes the area of tissue this node corresponds to. Each tissue cell that is located inside the sphere of radius R around the conduction system node will have additional term in the equation for the rate of membrane potential: $D_{t-cs}(V_{cs} - V_t)$ where V_{cs} and V_t are membrane potentials of the conduction system cell and tissue cell respectively. Including the conduction system in the heart model allows us to fit the model behavior with theoretical expectations and to simulate typical heart pathologies. The corresponding simulation results are given in Sect. 7 below.

5 Parallel Algorithm

The parallel algorithm propagates the voltage V and the potential φ_e over time. The main numerical scheme is as follows:

1. *Data initialization step.* We read parameters from the configuration file, initialize MPI, allocate memory, read and initialize the mesh, the voltage $V(0)$, the potential $\varphi_e(0)$, the cells state \mathbf{u} , the currents I_e^{st} and I_m^{st} .
2. *Mesh partitioning step.* The mesh is partitioned by means of the parallel ParMetis software to reduce MPI communications during the main computational loop.
3. *Matrix assembly step.* We generate FEM matrices in parallel: the mass matrix and the stiffness matrix.
4. *Pre-compute step.* Each MPI process solves of the ODE system (2) with the time step $\frac{\Delta t}{2}$ by the fourth order Runge-Kutta method for the heart cells located in the mesh vertexes belonging to the process. The initial conditions are set to default values (rest states for the tissue models) or are loaded from file.
5. *Main computational loop.* During this step we propagate the voltage, potential and cells state over time by means of the Operator splitting method with a time step Δt . Every iteration of this loop is fully parallel and is as follows:
 - a. The software calculates diffusion $\nabla \cdot (\sigma_m \nabla V)$ for the voltage $V(t)$ by the finite element method. In this regard we need finding a solution of a sparse SLAE by iterative or direct method. Both approaches make it possible to find the solution of the system in parallel on the distributed memory. In most experiments we use iterative methods with appropriate preconditions, if necessary.
 - b. We compute the total current $I_{sum} = I_m^{st} + I_e^{st} + \nabla \cdot (\sigma_m \nabla V)$ in parallel.
 - c. We solve the Poisson equation $\nabla \cdot ((\sigma_m + \sigma_e) \nabla \varphi_e) = -I_{sum}$ with respect to $\varphi_e(t + dt)$ by the finite element method. We solve a sparse SLAE in parallel.
 - d. The software computes diffusion $(\sigma_m \nabla \varphi_e)$, where φ_e has been found before. We employ the finite element method and solve a sparse SLAE in parallel.
 - e. For each cell (mesh vertex) that has a coupling with the conduction system node compute additional current term: $I_{cs} = D_{t-cs}(V_{cs} - V_t)$.
 - f. Each MPI process solves ODE $\frac{\partial V}{\partial t} = \frac{\nabla \cdot (\sigma_m \nabla (V + \varphi_e)) + I_m^{st} + I_{cs}}{\chi C_m}$ with respect to the voltage V by the Euler method with the time step Δt . The results of computation at step A are used as initial conditions.
 - g. Each MPI process solves ODEs describing the dynamics of conduction system with RungeKutta 4th order approximation for time step Δt .
 - h. Each MPI process solves the ODE system (2) with the time step Δt by the fourth order Runge-Kutta method for the heart cells located in the mesh cells belonging to the process. The initial conditions are the results of calculating V at step f.
6. *Post-compute step.* Each MPI process solves the ODE system (2) with the time step $\frac{\Delta t}{2}$ by the fourth order Runge-Kutta method for the heart cells located in the mesh cells belonging to the process. The initial conditions are the results from the last step of the main computation loop.
7. *Finalization step.* The results of computations are stored, the memory is released.

6 Implementation Overview

There are many approaches for the heart activity simulation. Therefore, when designing the CardioModel software, we considered the possibility of expanding the software with new models and numerical schemes as one of the key requirements. In this regard, we make use of the object-oriented methodology to design our application. The following main subsystems and interfaces of their communication have been identified: Modeling of electrical activity of heart, Mathematical models, Configuration, Numerical schemes, Post-processing, and Graphical user interface. Thereby we can easily choose different cell models, try new methods of solving SLAEs and specific preconditioners, and add new modules for processing simulations results.

The software is implemented in the C++ programming language. Parallelization for distributed memory systems is performed using MPI technology. The ParMetis parallel library is used to distribute the mesh along the compute processes in an optimal manner. We employ distributed SLAE solvers and preconditioners from the PETSc library. The results of simulations are stored using the VTK library to ensure compatibility with the most common professional 3D visualization tools, for example, ParaView. We also implement storing the current calculations state and restarting from the previously saved checkpoint. The CMake software is used to provide the build infrastructure.

7 Numerical Results

7.1 Application

The personal three-dimensional tetrahedral finite element model of the human heart tissue is constructed by combining the vertices of a reference finite element Atlas model of a human heart with the patient's heart tissue on a tomogram (Fig. 1).

The Atlas model is anatomically segmented by marking the affiliation of the mesh nodes to specific anatomical segments (left, right ventricles, atria, etc.). This markup is then transferred to a tomogram. The fitting is performed by the Coherent Point Drift resulting in a FEM mesh of heart which is a prerequisite for the simulation of electrical activity. The conduction system of the heart is represented as a graph and is connected with the mesh. The pre-segmented Atlas model is constructed from a three-dimensional surface model of the heart as follows. Firstly, we employ the NetGen software to generate a tetrahedral grid of the heart bounded with its outer surface. Next, by means of smoothing we remove cavities, limited by the inner surface of the chambers (atria and ventricles). Finally, we make the segmentation of the finite-element mesh of the heart. At this step, the inner surfaces are expanded until they come into contact with elements of another compartment or with outer surfaces. Our software performs fast validation of the vertices of the finite element mesh.

Below we present some simulation results. First, we consider three regimes of heart: the regular mode (Fig. 2a), atrial fibrillation due to a single spiral wave (Fig. 2b), and atrial fibrillation with spiral wave brake up and the formation of spiral chaos (Fig. 2c). The snapshots are taken at specific times, corresponding to the initial,

intermediate and final part of the simulation. The color scale corresponds to a change in voltage V from -100 mV to 25 mV. In the bottom right inset one can see the wave front pattern computed with a threshold transform.

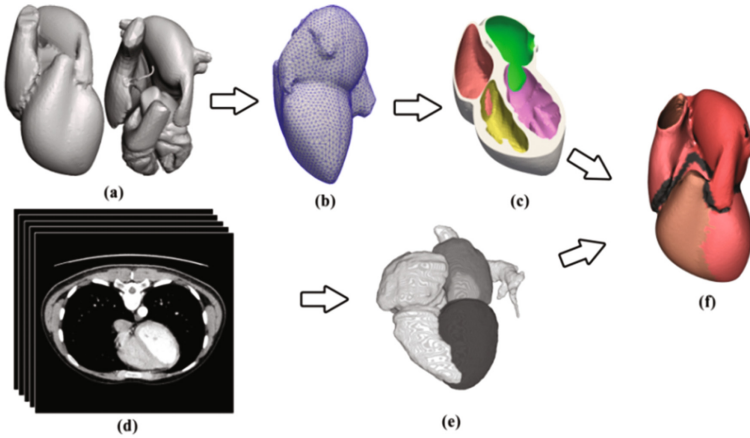


Fig. 1. The data preparation cycle. (a) The Atlas model of the heart surface. (b) The unified tetrahedral mesh of the heart. (c) The segmented mesh. (d) 3D image data. (e) The heart segmented by machine learning methods. (f) The personalized model of the human heart.

In the case of normal heart activation (Fig. 2a) one can see how the action potential originates from the sino-atrial node and propagates over atria. After that the excitation is delivered down to the ventricular apex through the Hiss bundle and eventually goes up covering left and right ventricles with a help of Purkinje fibers. The second set of results (Fig. 2b) demonstrates the simulation of atrial fibrillation which is believed to be stipulated by the spiral wave of action potential that persists in the atrial media and generates high frequency pulses. The rate of excitation may reach 500 bpm due to spiral wave curvature. At the same time AVN serves as a filter only allowing limited number of excitations to go down into the ventricles. Then (Fig. 2c) we demonstrate how the single spiral wave may become unstable and break up into a dynamic regime called spiral chaos which is also considered to be a background for atrial fibrillation. One can observe how the increase in the wave width leads to the spiral instability and to the consequent wave front break up. Multiple spiral rotors occur in the atria. They are irregular and persistent and lead to incoherent high frequency excitation.

Lastly (Fig. 3), we demonstrate the simulated electrical activity of heart under infarction conditions. The area of infarction was introduced in the left ventricle of the heart that otherwise showed a normal activation pattern. The Figure has three insets: top left illustrates the 3D image of the electrical action potential with the colormap depicting the membrane voltage (mV). The snapshot is taken at the moment of time when the ventricles finished depolarization. The area of infarction (and hence a conduction block) is clearly observed. Top right inset shows the snapshot of the dynamics of the conduction system at the moment the terminal pieces of Purkinje fiber are excited. The time series in the inset below shows the electrical activity of significant

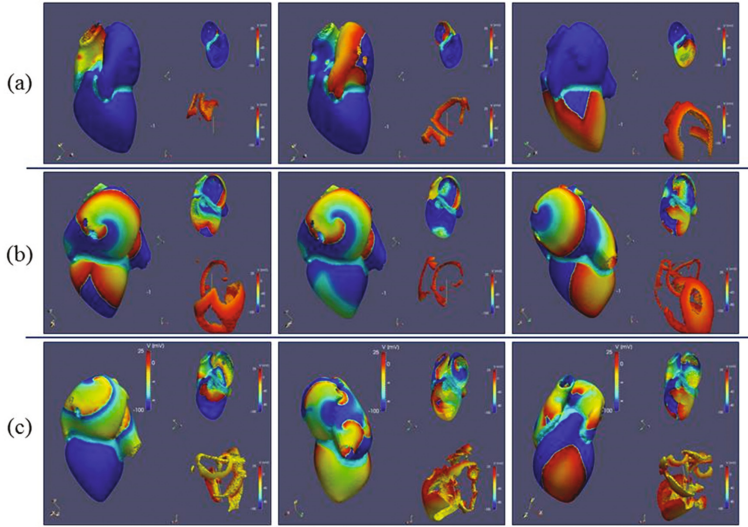


Fig. 2. Simulation results. (a) V in the regular mode. (b) Atrial fibrillation, a single spiral wave. (c) Atrial fibrillation, spiral chaos.

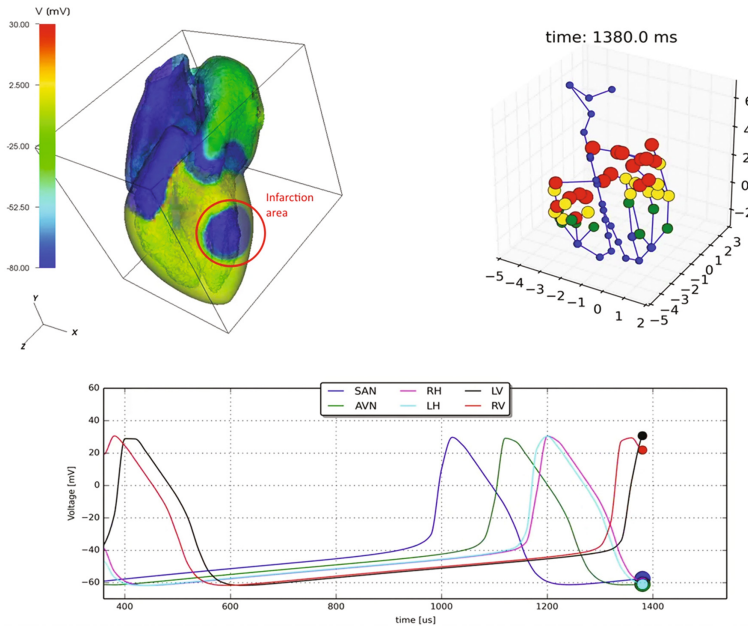


Fig. 3. Simulation results. Electrical activity of the heart under infarction conditions.

cells of the conduction system: sino-atrial node (SAN), atrio-ventricular node (AVN), terminal cells of the His bundle located at the apex of the heart (LH, RH), and last (terminal) cells of the Purkinje fiber of both ventricles (LV, RV). The results are in good agreement with theoretical expectations.

7.2 Performance and Scalability

We assess performance and scalability of the CardioModel software as follows. In the experiments we use the finite element mesh (Fig. 4) with 998 151 vertices and 5 140 810 tetrahedrons. The left and right atrium, left and right ventricles, fibrous tissue, and ventricular septum are considered as separate compartments with relevant cell models. The conduction system is represented as the graph with 55 vertices mapped to the mesh and 54 edges.

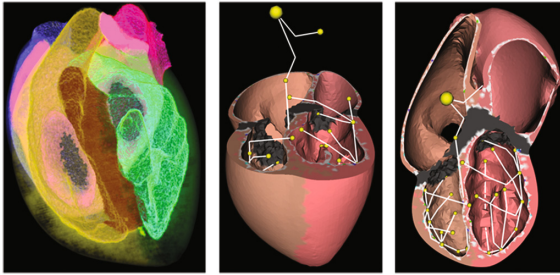


Fig. 4. The segmented 1M mesh and the conduction system.

In the experiments we use 15 nodes of the Lobachevsky supercomputer. Every node is equipped by 2 Intel Sandy Bridge E5-2660 2.2 GHz CPU (8 cores each), 64 GB RAM, OS Centos 7.2, Intel Parallel Studio 2017, PETSc 3.8, ParMetis 4.0.3.

First, we analyzed the performance of the code depending on the following factors: possible distributions of the mesh between processes to reduce MPI communications; SLAE solver selection; SLAE solver preconditioner selection; and proper selection of the starting point for the iterative methods. As expected, the use of ParMetis dramatically speeds up the calculation due to the reduction of MPI communications. The solution obtained at the previous iteration of the computational loop was chosen as a starting point for the iterative methods of the SLAE solver. This obvious strategy has made it possible to significantly speed up the convergence of methods. Large-scale computational experiments were carried out to select the most suitable method and preconditioner. As a result, it turned out that the best results for the given accuracy were shown by the combination of the Flexible Generalized Minimal Residual method (FGMRES) + GAMG preconditioner.

Table 1 shows the run time of the algorithm depending on the number of MPI processes used. The first column shows the number of processes, in the second – the run time of the ODE solver. The third and fourth columns give the time for solving the SLAE in calculating the diffusion and solving the Poisson equation, respectively. The proportion of time that has come to other stages is negligible and is shown in the sixth column. Lastly, the total calculation time is presented.

The results of the experiments show that the stage of the ODE solution scales well enough up to 120 processes. Contrary, the SLAE solution requires a significant amount of MPI communications, which leads to a decrease in the strong scaling efficiency of these stages. Overall strong scaling efficiency calculated from 8 to 64 processes is 96.2%. The further increase in the number of processes to 120 still speeds up the computations, the efficiency is 74.6% of the theoretical optimum.

Table 1. Performance results when propagating the bidomain equations. The number of time steps is set to 100. The PETSc tolerance parameter is equal to $1e-10$. Time is given in seconds.

# Processes	ODE	Diffusion solver	Poisson equation solver	Other	Total
8	24.28	24.08	76.67	4.39	129.41
16	11.82	9.87	34.22	1.49	57.40
32	6.00	5.24	17.78	1.19	30.22
64	3.09	3.04	10.03	0.65	16.82
96	2.17	3.06	9.14	0.59	14.96
120	1.76	2.32	7.07	0.42	11.57

8 Conclusion and Future Work

We presented the description of the CardioModel software, intended for numerical modeling of human heart electrophysiology on supercomputers. The software uses a tetrahedral finite-element mesh of the heart as input data. The mesh is constructed according to the results of the patient's computer tomogram and is automatically segmented. Using this mesh we integrate bidomain equations describing the propagation of an electrical signal in cardiac tissues. The paper presents the results of numerical modeling of several regimes of the heart, including the development of pathologies. Software shows acceptable (75%) scalability when using 120 MPI ranks on the mesh of ~ 1 million vertices and ~ 5 million tetrahedrons. Further developments include improving personalization of models and increasing the scalability of the software. We plan to develop custom load balancing schemes when using different cell models.

Acknowledgements. The study was supported by the Ministry of Education of Russian Federation (Contract # 02.G25.31.0157, date 01.12.2015).

References

1. Crampin, E.J., et al.: Computational physiology and the physiome project. *Exp. Physiol.* **89** (1), 1–26 (2004)
2. Lloyd, C.M., et al.: The CellML model repository. *Bioinformatics* **24**(18), 2122–2123 (2008)
3. Vázquez, M., et al.: Alya red CCM: HPC-based cardiac computational modelling. In: Klapp, J., Ruíz Chavarría, G., Medina Ovando, A., López Villa, A., Sigalotti, L. (eds.) *Selected Topics of Computational and Experimental Fluid Mechanics*. ESE, pp. 189–207. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-11487-3_11

4. Bishop, M.J., et al.: Development of an anatomically detailed MRI-derived rabbit ventricular model and assessment of its impact on simulations of electrophysiological function. *Am. J. Physiol. Heart Circ. Physiol.* **298**(2), H699–H718 (2010)
5. Vázquez, M., et al.: Alya: multiphysics engineering simulation toward exascale. *J. Comput. Sci.* **14**, 15–27 (2016)
6. Trayanova, N.A.: Whole-heart modeling. *Circ. Res.* **108**(1), 113–128 (2011)
7. Arevalo, H.J., et al.: Arrhythmia risk stratification of patients after myocardial infarction using personalized heart models. *Nat. Commun.* **7**, 11437 (2016)
8. Richards, D.F., et al.: Towards real-time simulation of cardiac electrophysiology in a human heart at high resolution. *Comput. Methods Biomech. Biomed. Eng.* **16**(7), 802–805 (2013)
9. Chapelle, D., Collin, A., Gerbeau, J.-F.: A surface-based electrophysiology model relying on asymptotic analysis and motivated by cardiac atria modeling. *Math. Models Methods Appl. Sci.* **23**(14), 2749–2776 (2013)
10. Vassilevski, Y., Danilov, A., Ivanov, Y., Simakov, S., Gamilov, T.: Personalized anatomical meshing of the human body with applications. In: Quarteroni, A. (ed.) *Modeling the Heart and the Circulatory System*. MS&A, vol. 14, pp. 221–236. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-05230-4_9
11. Danilov, A.A., et al.: Parallel software platform INMOST: a framework for numerical modeling. *Supercomput. Frontiers Innov.* **2**(4), 55–66 (2015)
12. Pravdin, S., et al.: Human heart simulation software for parallel computing systems. *Procedia Comput. Sci.* **66**, 402–411 (2015)
13. Mirams, G.R., et al.: Chaste: an open source C++ library for computational physiology and biology. *PLoS Comput. Biol.* **9**(3), e1002970 (2013)
14. Schmitt, O.H.: Biological information processing using the concept of interpenetrating domains. In: Leibovic, K.N. (ed.) *Information Processing in The Nervous System*, pp. 325–331. Springer, Heidelberg (1969). https://doi.org/10.1007/978-3-662-25549-0_18
15. Tung, L.: A bi-domain model for describing ischemic myocardial dc potentials. Massachusetts Institute of Technology (1978)
16. Karypis, G., Kumar, V.: Parallel multilevel k-way partitioning scheme for irregular graphs. *SIAM Rev.* **41**(2), 278–300 (1999)
17. Balay, S.: PETSc Users Manual, ANL-95/11–Revision 3.8. Argonne National Lab (2017)
18. Intel MKL. Sparse solver routines. <https://software.intel.com/en-us/mkl-developer-reference-fortran-sparse-solver-routines>. Accessed 1 Mar 2018
19. Lebedev, S., Akhmedzhanov, D., Kozinov, E., Meyerov, I., Pirova, A., Sysoyev, A.: Dynamic parallelization strategies for multifrontal sparse cholesky factorization. In: Malyshkin, V. (ed.) *PaCT 2015*. LNCS, vol. 9251, pp. 68–79. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-21909-7_7
20. Bastrakov, S., et al.: High performance computing in biomedical applications. *Procedia Comput. Sci.* **18**, 10–19 (2013)
21. Linge, S., et al.: Numerical solution of the bidomain equations. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* **367**(1895), 1931–1950 (2009)
22. Clayton, R.H., et al.: Models of cardiac tissue electrophysiology: progress, challenges and open questions. *Progr. Biophys. Mol. Biol.* **104**(1), 22–48 (2011)
23. Pathmanathan, P., et al.: A numerical guide to the solution of the bidomain equations of cardiac electrophysiology. *Progr. Biophys. Mol. Biol.* **102**(2), 136–155 (2010)
24. Strang, G.: On the construction and comparison of difference schemes. *SIAM J. Numer. Anal.* **5**(3), 506–517 (1968)
25. Bernabeu, M.O., et al.: Chaste: a case study of parallelisation of an open source finite-element solver with applications to computational cardiac electrophysiology simulation. *Int. J. HPC Appl.* **28**(1), 13–32 (2014)

26. Sundnes, J., Lines, G.T., Tveito, A.: An operator splitting method for solving the bidomain equations coupled to a volume conductor model for the torso. *Math. Biosci.* **194**(2), 233–248 (2005)
27. Santos, R.W., et al.: Parallel multigrid preconditioner for the cardiac bidomain model. *IEEE Trans. Biomed. Eng.* **51**(11), 1960–1968 (2004)
28. Vasiliev, E.: Generation of an atlas-based finite element model of the heart for cardiac simulation. *Int. Sci. J. Math. Model.* **4**, 207–209 (2017)
29. Lachinov, D., Belokamenskaya, A., Turlapov, V.: Precise automatic cephalometric landmark detection algorithm for CT images. In: *Proceedings of Graphicon 2017*, pp. 275–278 (2017)