The Journal of Open Source Software

svFSI: A Multiphysics Package for Integrated Cardiac Modeling

Chi Zhu[*1], **Vijay Vedula**[†2], **Dave Parker**[3], **Nathan Wilson**[4], **Shawn Shadden**[1], **and Alison Marsden**[3]

1 University of California, Berkeley **2** Columbia University **3** Stanford University **4** University of California, Los Angeles

DOI: 10.21105/joss.04118

Software

- Review
- Repository
- Archive

Editor: Kevin M. Moerman

Reviewers:

- @chennachaos
- @JaroslavHron
- @axel-loewe

Submitted: 05 January 2022
Published: 11 February 2022

Summary

Heart disease is the number one cause of death in the US (Xu et al., 2020). Many efforts have been devoted to studying its progression, diagnosis, and treatment. During the past decade, computational modeling has made significant inroads into the research of heart disease. The heart is inherently a multiphysics system that includes electrophysiology, tissue mechanics, and blood dynamics. Its normal function starts with the propagation of electrical signals that trigger the active contraction of the heart muscle to pump blood into the circulatory system. Rooted in fundamental laws of physics such as the balance of mass, momentum, and energy, computational modeling has been instrumental in studying cardiac physiology such as left ventricular function (Mittal et al., 2015), cardiac arrhythmia (Trayanova, 2011), and blood flow in the cardiovascular system (Arzani & Shadden, 2018; Grande Gutiérrez et al., 2021). `svFSI` is the first open-source software that specializes in enabling coupled electro-mechano-hemodynamic simulations of the heart.

Statement of need

Accompanying the growing popularity of studying cardiac physiology with computational modeling, many open-source software tools that specialize in modeling one or two aspects of the multiphysics process in the heart have been developed. For example, `SimVascular` (Updegrove et al., 2017) enables patient-specific blood flow modeling by providing a complete pipeline from medical image segmentation to simulation results. `FEBio` (Maas et al., 2012) specializes in modeling large-deformation structure mechanics in biophysics with fluid-structure interaction (FSI) capability as well. `openCARP` (Vigmond et al., 2003) focuses on modeling cardiac electrophysiology. There are other general-purpose open-source software such as `LifeV` (Bertagna et al., 2017) and `FEniCS` (Logg et al., 2012), that can be flexibly adapted to simulate different physics in the heart, but significant development effort may be required for this purpose.

`svFSI` is a new multiphysics finite element solver designed specifically for computational modeling of integrative heart dynamics. As the next generation finite element solver for the `SimVascular` software, `svFSI` is capable of modeling hemodynamics, performing large-deformation FSI to capture the motion of cardiac chambers and their interaction with the blood flow, and simulating the complex excitation-contraction coupling between the intracellular ion-exchange processes and tissue contraction. To suit the diverse needs of users, our team has implemented non-Newtonian blood viscosity models, fiber-reinforced nonlinear

*Corresponding author, equal contributions to the paper
†Corresponding author, equal contributions to the paper

hyperelastic material models, and both phenomenological and biophysics-based cellular activation models, as well as interfaces to multiple iterative linear solvers and preconditioners. svFSI is also capable of resolving the inherent multi-scale phenomena in cardiac physiology such as the interaction between Purkinje fibers and cardiac muscle, and setting up closed-loop multi-scale hemodynamic simulations with physiological boundary conditions. Moreover, it is worth emphasizing that svFSI is fully compatibility with SimVascular, and thus can take advantage of the existing pipeline for medical image segmentation, anatomic model construction, meshing, boundary condition prescription, etc. svFSI is written in Fortran, taking advantage of the object-oriented programming features of Fortran 2003. The code is highly modularized for ease of interoperability and extension. We acknowledge the limited open-source compiler support for Fortran on Windows systems, and there is an ongoing effort to convert svFSI into pure C++ software.

Software architecture

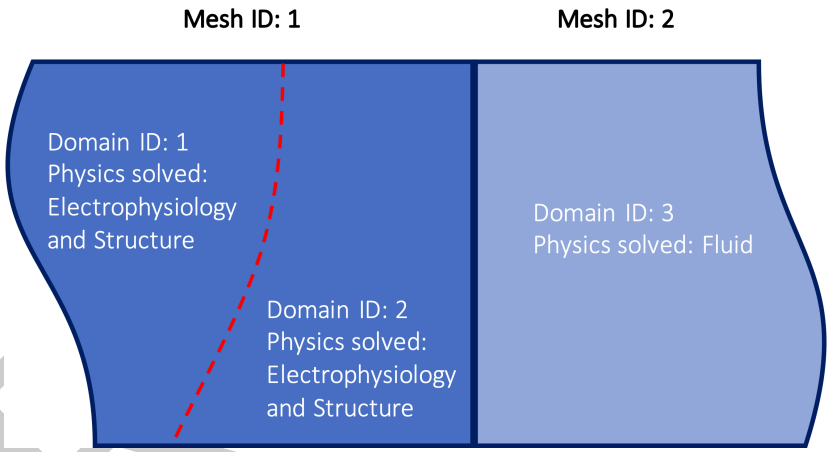

Figure 1: Illustration of a multi-mesh, multi-domain simulation configuration.

The svFSI solver defines a modeling task through three key components (objects): mesh, domain, and equation. A mesh defines a spatial discretization of the physical regions such as the Purkinje network, myocardium, or the blood volume. It can be divided into non-overlapping domains through unique domain IDs. Different physical equations, such as the Navier-Stokes (fluid) equation and the structural mechanics (struct) equation are solved on each domain with domain-specific material properties. Figure 1 illustrates how these three objects are utilized in the multiphysics modeling of cardiac mechanics. The whole computational model is composed of two meshes. Mesh 1 represents the ventricular myocardium, where both electrophysiology and nonlinear solid mechanics equations are solved. Given the material heterogeneity of the myocardium, mesh 1 can be further divided into multiple domains so that domain-specific material properties, such as electrical conductivity and material elasticity, can be assigned. Mesh 2 is the fluid region where only the Navier-Stokes equations are solved.

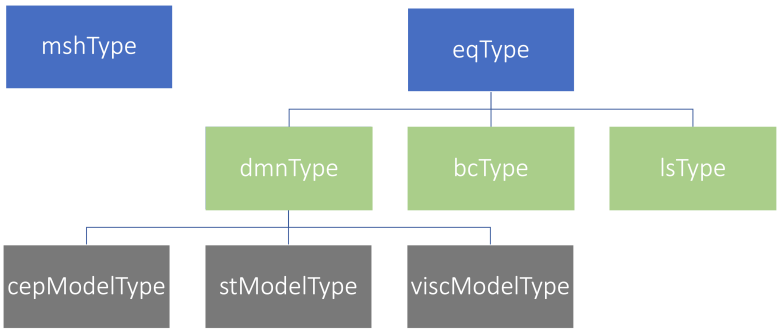

Figure 2: Class hierarchy that constitutes a simulation. Description of each class is provided in Table 1.

This highly flexible, multi-mesh, and multi-domain modeling capability is realized in `svFSI` through the class hierarchy depicted in Figure 2. The three key components, i.e. mesh, domain, and equation, correspond to classes mshType, dmnType and eqType, respectively. The material heterogeneity for electrophysiology, tissue mechanics or hemodynamics within each domain is defined through cepModelType, stModelType or viscModelType (see Table 1). Two other important classes, bcType and lsType, are also shown in Figure 2. They define the boundary conditions and settings for the linear solver that are essential to generating a well-posed simulation configuration.

Class name	Description
mshType	Defines properties of the mesh, such as the element type, number of nodes per element, and coordinates.
eqType	Defines properties of the equation, such as the physics (fluid/struct/electrophysiology), domains, linear solvers, and boundary conditions.
dmnType	Defines properties of the domain, such as the physics (fluid/structure/electrophysiology), material properties, stabilization parameters, etc.
bcType	Defines properties of the boundary conditions, such as Dirichlet or Neumann, time dependence, spatial profile, etc.
lsType	Defines properties of linear solvers.
cepModelType	Defines properties of the electrophysiology model
stModelType	Defines properties of the structure material model
viscModelType	Defines properties of the viscosity model for fluids

Table 1: Main classes defined in svFSI.

Full documentation for `svFSI` functionality is available from the `SimVascular` website at http://simvascular.github.io.

Conclusion

`svFSI` is a multiphysics finite-element solver focusing on whole heart modeling and consists of modules that can efficiently simulate hemodynamics, cardiac mechanics, cardiac electrophysiology as well as the multiphysical interactions among them. `svFSI` has been used to generate several publications (Bäumler et al., 2020; Kong & Shadden, 2020; Seo, Schiavazzi,

et al., 2020; Seo, Fleeter, et al., 2020; Vedula et al., 2017) and is employed in several active projects. We will continue to support and develop the software. New features such as methods to simulate heart valves, vascular growth and remodeling are currently under development. The software is envisioned to be applicable to a wide range of research questions in both pediatric and adult cardiovascular disease.

Acknowledgments

This work was supported by the National Science Foundation SI2-SSI 1663671 and 1663747. C.Z. also acknowledges funding from the American Heart Association award 20POST35200253. Previous contribution to the software by Dr. Mahdi Esmaily-Moghadam and Dr. Justin Tran is also acknowledged.

References

Arzani, A., & Shadden, S. C. (2018). Wall shear stress fixed points in cardiovascular fluid mechanics. *Journal of Biomechanics*, *73*, 145–152. https://doi.org/10.1016/j.jbiomech.2018.03.034

Bäumler, K., Vedula, V., Sailer, A. M., Seo, J., Chiu, P., Mistelbauer, G., Chan, F. P., Fischbein, M. P., Marsden, A. L., & Fleischmann, D. (2020). Fluid–structure interaction simulations of patient-specific aortic dissection. *Biomechanics and Modeling in Mechanobiology*, *19*(5), 1607–1628. https://doi.org/10.1007/s10237-020-01294-8

Bertagna, L., Deparis, S., Formaggia, L., Forti, D., & Veneziani, A. (2017). The LifeV library: Engineering mathematics beyond the proof of concept. *arXiv Preprint*. https://arxiv.org/abs/1710.06596

Grande Gutiérrez, N., Alber, M., Kahn, A. M., Burns, J. C., Mathew, M., McCrindle, B. W., & Marsden, A. L. (2021). Computational modeling of blood component transport related to coronary artery thrombosis in kawasaki disease. *PLOS Computational Biology*, *17*(9), e1009331. https://doi.org/10.1371/journal.pcbi.1009331

Kong, F., & Shadden, S. C. (2020). Automating Model Generation for Image-Based Cardiac Flow Simulation. *Journal of Biomechanical Engineering*, *142*(11). https://doi.org/10.1115/1.4048032

Logg, A., Mardal, K.-A., Wells, G. N., & others. (2012). *Automated solution of differential equations by the finite element method*. Springer. https://doi.org/10.1007/978-3-642-23099-8

Maas, S. A., Ellis, B. J., Ateshian, G. A., & Weiss, J. A. (2012). FEBio: Finite elements for biomechanics. *Journal of Biomechanical Engineering*, *134*(1). https://doi.org/10.1115/1.4005694

Mittal, R., Seo, J. H., Vedula, V., Choi, Y. J., Liu, H., Huang, H. H., Jain, S., Younes, L., Abraham, T., & George, R. T. (2015). Computational modeling of cardiac hemodynamics: Current status and future outlook. *Journal of Computational Physics*, *305*, 1065–1082. https://doi.org/10.1016/j.jcp.2015.11.022

Seo, J., Fleeter, C., Kahn, A. M., Marsden, A. L., & Schiavazzi, D. E. (2020). Multifidelity estimators for coronary circulation models under clinically informed data uncertainty. *International Journal for Uncertainty Quantification*, *10*(5). https://doi.org/10.1615/Int.J.UncertaintyQuantification.2020033068

Seo, J., Schiavazzi, D. E., Kahn, A. M., & Marsden, A. L. (2020). The effects of clinically-derived parametric data uncertainty in patient-specific coronary simulations with deformable walls. *International Journal for Numerical Methods in Biomedical Engineering*, *36*(8), e3351. https://doi.org/10.1002/cnm.3351

Trayanova, N. A. (2011). Whole-heart modeling : Applications to cardiac electrophysiology and electromechanics. *Circulation Research*, *108*(1), 113–128. https://doi.org/10.1161/CIRCRESAHA.110.223610

Updegrove, A., Wilson, N. M., Merkow, J., Lan, H., Marsden, A. L., & Shadden, S. C. (2017). SimVascular: An Open Source Pipeline for Cardiovascular Simulation. *Annals of Biomedical Engineering*, *45*(3), 525–541. https://doi.org/10.1007/s10439-016-1762-8

Vedula, V., Lee, J., Xu, H., Kuo, C.-C. J., Hsiai, T. K., & Marsden, A. L. (2017). A method to quantify mechanobiologic forces during zebrafish cardiac development using 4-d light sheet imaging and computational modeling. *PLoS Computational Biology*, *13*(10), e1005828. https://doi.org/10.1371/journal.pcbi.1005828

Vigmond, E. J., Hughes, M., Plank, G., & Leon, L. J. (2003). Computational tools for modeling electrical activity in cardiac tissue. *J Electrocardiol*, *36 Suppl*, 69–74. https://doi.org/10.1016/j.jelectrocard.2003.09.017

Xu, J., Murphy, S., Kochanek, K., & Arias, E. (2020). Mortality in the United States, 2018. *NCHS Data Brief No. 355*. https://www.cdc.gov/nchs/products/databriefs/db355.htm