

svZeroDSolver: A modular package for lumped-parameter cardiovascular simulations

Karthik Menon^{1*}, Jakob Richter^{2*}, Martin R. Pfaller^{3*}, Jonathan Pham²,
Emilin M. Mathew², Kaitlin E. Harold², Nicholas C. Dorn², Aekaansh
Verma², and Alison L. Marsden^{2¶}

¹ Georgia Institute of Technology, Atlanta, GA, United States of America ² Stanford University,
Stanford, CA, United States of America ³ Yale University, New Haven, CT, United States of America ¶
Corresponding author * These authors contributed equally.

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

Software

- [Review](#)
- [Repository](#)
- [Archive](#)

Editor: [Open Journals](#)

Reviewers:

- [@openjournals](#)

Submitted: 01 January 1970

Published: unpublished

License

Authors of papers retain copyright
and release the work under a
Creative Commons Attribution 4.0
International License ([CC BY 4.0](https://creativecommons.org/licenses/by/4.0/))

Summary

Computational modeling of cardiovascular blood flow has emerged as a valuable tool in the diagnosis and treatment of cardiovascular disease (Menon, Hu, et al., 2024). While simulations of blood flow can be performed using high and low-fidelity techniques, lumped-parameter or zero-dimensional modeling is a widely used low-order technique in applications which require quick estimation of bulk flow quantities, such as flow and pressure at specific anatomical locations (Pfaller et al., 2024).

We introduce [svZeroDSolver](#), an efficient and modular package for performing lumped-parameter (zero-dimensional) simulations of cardiovascular blood flow. As part of the [SimVascular](#) open-source project, [svZeroDSolver](#) and [SimVascular](#) together allow users to go from medical imaging to fast zero-dimensional evaluations of patient-specific hemodynamics. [svZeroDSolver](#) is written in C++ using a modular object-oriented framework. Simply specifying a .json dictionary of lumped-parameter “blocks” – such as blood vessels, junctions between blood vessels, and boundary conditions (along with their associated parameters) – allows the code to automatically assemble and solve the governing equations corresponding to the user-specified vascular model. In addition, the package includes Python and C++ interfaces to facilitate its use with other software packages. For example, it can be integrated into Python-based optimization and uncertainty quantification applications (Lee et al., 2024; Menon, Zandoni, et al., 2024; Richter et al., 2024; Zandoni et al., 2024). It can also be interfaced with C++/Fortran software for high-fidelity cardiovascular flow simulations, where [svZeroDSolver](#) can conveniently provide physiological lumped-parameter boundary conditions (Menon et al., 2023; Menon, Khan, et al., 2024). [svZeroDSolver](#) includes an application called [svZeroDCalibrator](#) to automatically calibrate parameters of a given zero-dimensional model based on independent hemodynamic measurements or high-fidelity simulations – thus improving the accuracy of zero-dimensional models (Richter et al., 2024). It also includes graphical interfaces to interactively create lumped-parameter models for simulations, as well as to visualize the simulated anatomy and hemodynamics.

Statement of need

Non-invasive quantification of patient-specific hemodynamics via computational simulations has improved patient outcomes and reduced invasive clinical procedures in large randomized clinical trials (Taylor et al., 2023). Computational modeling is also a promising tool for non-invasive and personalized optimization of clinical treatments and surgery (Marsden, 2014).

Previous work has used several techniques to model cardiovascular blood flow, all of which

can be broadly categorized based on their level of fidelity. High-fidelity models generally involve simulations of the full three-dimensional flow field within anatomical regions of interest (Menon, Hu, et al., 2024; Updegrove et al., 2017). While these are the most accurate and informative, they are computationally expensive (each simulation can take several hours or days on hundreds of CPU cores) and, therefore, not practical in typical clinical settings or for applications such as optimization and uncertainty quantification, which often require thousands of model evaluations. On the other end of the spectrum, lumped-parameter or zero-dimensional models provide information about bulk hemodynamics, such as flow rate and pressure, at specific anatomical regions of interest. While these models are not spatially resolved, they are valuable in applications which require near real-time quantification of bulk hemodynamics, as well as those that rely on thousands of repeated model evaluations (Lee et al., 2024; Menon, Zandoni, et al., 2024; Richter et al., 2024; Zandoni et al., 2024). They are also commonly used in conjunction with high-fidelity simulations where lumped-parameter models are used as physiological boundary conditions (Menon et al., 2023; Menon, Khan, et al., 2024).

svZeroDSolver, which is a part of the SimVascular open-source project, is a new open-source software package that enables fast evaluation of zero-dimensional hemodynamics. One major challenge in zero-dimensional modeling that svZeroDSolver addresses is that different clinical applications (and individual clinical cases within the same application) often require unique anatomical arrangements of blood vessels, heart valves, etc. Moreover, distinct anatomical configurations are governed by a distinct set of governing equations. Therefore, it is common for users to implement application-specific solvers which simulate the equations governing a specific application or anatomical configuration. In contrast, the modularity of svZeroDSolver allows users to easily create arbitrary anatomical configurations by arranging a library of available “blocks”, following which the software automatically assembles the equations governing the user-specified configuration.

Another unique feature of svZeroDSolver is its ability to easily interface with other C++ and Python packages. This has been used in previous work on uncertainty quantification (Lee et al., 2024; Menon, Zandoni, et al., 2024; Richter et al., 2024; Zandoni et al., 2024) as well as in multi-scale simulations coupling three-dimensional hemodynamics with zero-dimensional representations of downstream circulation (Menon et al., 2023; Menon, Khan, et al., 2024). The C++ interface has been coupled with the high-fidelity multi-physics solver svFSIplus, which is part of the widely used SimVascular open-source software project for cardiovascular biomechanics simulations (Updegrove et al., 2017; Zhu et al., 2022). svZeroDSolver has also been integrated into the graphical user interface of the SimVascular project. This allows users to leverage the functionality in SimVascular to generate three-dimensional patient-specific anatomical models from medical images and subsequently perform patient-specific zero-dimensional simulations of blood flow by automatically converting the three-dimensional anatomy into a zero-dimensional model (Pfaller et al., 2022). The automatic conversion of arbitrary patient-specific anatomies to zero-dimensional simulations is possible due to the modular nature of svZeroDSolver. Using this pipeline, previous work has demonstrated accelerated convergence of three-dimensional simulations when using corresponding zero-dimensional simulation results as initial conditions (Pfaller et al., 2021).

In addition, svZeroDSolver includes several applications to augment its functionality. The svZeroDCalibrator application improves the accuracy of zero-dimensional models by optimizing the parameters of blood vessels to recapitulate observed hemodynamics from measurements or high-fidelity simulations. This allows users to build more accurate zero-dimensional models than those typically based purely on the anatomy of the vascular region of interest (Richter et al., 2024). The svZeroDGUI application is a web-based graphical interface that allows users to create zero-dimensional simulations by interactively dragging and dropping individual blood vessels, heart chambers, boundary conditions, connections between these blocks, etc. Another graphical application, svZeroDVisualization, is an interface to visualize the lumped-parameter structure of given anatomical models as well as the simulated hemodynamics within each block. Together, these graphical interfaces make svZeroDSolver intuitive for a wide range

95 of users, potentially expanding its use from research to instructional and clinical contexts.
 96 The functionality and accuracy of svZeroDSolver is assessed using continuous integration
 97 tests on GitHub, and has also been verified by comparing with high-fidelity three dimensional
 98 simulations (Pfaller et al., 2022). This combination of features makes svZeroDSolver uniquely
 99 applicable to a wide range of applications in cardiovascular biomechanics.

100 State of the field

101 While there are other open-source projects that provide the functionality for cardiovascular
 102 flow modeling, and specifically zero-dimensional flow modeling, svZeroDSolver has several
 103 features that distinguish it from previous work. In particular, prior packages have primarily
 104 focused on multi-physics finite element modeling for cardiovascular biomechanics (Africa et
 105 al., 2024; Arthurs, 2021; Hirschvogel, 2024; Zhu et al., 2022). Although these projects
 106 allow the implementation of simple zero-dimensional models, either as boundary conditions
 107 to three-dimensional models or as simple stand-alone zero-dimensional models, the primary
 108 focus is on the modeling of full three-dimensional fluid and tissue mechanics. Due to this,
 109 they generally lack the variety and/or modular functionality to create a broad range of user-
 110 specified zero-dimensional flow models. There are, however, packages aimed specifically
 111 at reduced-order modeling for cardiovascular flows. For example, the SimVascular project
 112 includes svOneDSolver for the purpose of one-dimensional blood flow modeling. Another
 113 popular package for one-dimensional blood flow simulations is Nektar1D (Alastruey et al.,
 114 2012). Similarly, Artery.FE implements one-dimensional blood flow modeling using the FEniCS
 115 finite element framework (Agdestein et al., 2018), the VaMpy toolkit includes a package for
 116 modeling one-dimensional blood flow using the Lax-Wendroff finite difference method (Diem
 117 & Bressloff, 2017), and openBF is a finite volume implementation of one-dimensional blood
 118 flow (Benemerito et al., 2024).

119 In the zero-dimensional modeling context, CRIMSON (Arthurs, 2021), lifex-cfd (Africa et al.,
 120 2024), and Ambit (Hirschvogel, 2024) include the ability to simulate simple zero-dimensional
 121 blood flow models. However, as mentioned above, their focus is on multi-physics simulations
 122 of cardiovascular biomechanics. Therefore, they support a limited set of stand-alone zero-
 123 dimensional models and do not feature the modularity that enables the creation of a large variety
 124 of zero-dimensional models as in svZeroDSolver. The CellML and CVSim packages include a
 125 limited set of stand-alone zero-dimensional flow models for specific anatomies/applications
 126 (Clerx et al., 2020; Heldt et al., 2010), but they do not provide the modular functionality to
 127 specify unique anatomical models. In addition, there have been other packages that use zero-
 128 dimensional modeling techniques with a focus on statistical analysis, cardiac electromechanics,
 129 or specific anatomical models (Huttary et al., 2017; Regazzoni & Quarteroni, 2021; Rosalia et
 130 al., 2021). However, these packages are either not focused on zero-dimensional modeling or
 131 use MATLAB implementations, which require software licenses and are not free to use.

132 In contrast to these existing packages, the purpose of svZeroDSolver is to provide an open-
 133 source framework specifically for simulating zero-dimensional flows in a variety of simple and
 134 complex anatomies that can be designed in a user-specific and application-specific manner
 135 – by leveraging the modular nature of the code. The unique features listed above allow
 136 the use of svZeroDSolver both as a stand-alone zero-dimensional flow solver for unique and
 137 patient-specific anatomies, as well as in conjunction with the aforementioned multi-physics
 138 solvers as boundary conditions, for parameters estimation and uncertainty quantification, or
 139 even as an instructional tool using its graphical interfaces.



Figure 1: Various zero-dimensional “blocks” included in svZeroDSolver at the time of writing.

Software details

svZeroDSolver relies on a collection of “blocks” to set up the governing equations for a given anatomical configuration. Each block is inherited from a block class, as illustrated in Figure 1, and is governed by a “local” set of equations with associated degrees of freedom. The solver parses through an input configuration .json file, which lists the blocks, their parameters, and the blocks’ connectivity, and then automatically assembles the local equations and degrees of freedom for each block into a global system of equations. The governing equations and circuit representation for each block are available in the documentation. For example, see the [documentation for a blood vessel block](#).

The zero-dimensional simulations performed by svZeroDSolver are governed by non-linear differential algebraic equations. We integrate these equations in time using the implicit generalized-alpha scheme (Jansen et al., 2000) with Newton-Raphson iterations to solve the linearized system. Under the hood, these linearized governing equations for each block are implemented as local contributions to a system of linear (matrix) equations, which are then assembled into a global linear system based on the user-specified configuration. Details on the modular implementation of the blocks, along with their governing equations, are provided in the documentation’s [Developer Guide](#). We use the [Eigen package](#) to represent and solve these sparse linear systems (Guennebaud et al., 2010). Mathematical details on this implementation are provided in the [SparseSystem](#) and [Integrator](#) classes in the documentation.

svZeroDSolver currently has implementations of different types of blood vessel blocks with

non-linear resistors to model vascular stenoses, junctions between blood vessels, a heart valve block modeled using a hyperbolic tangent function, a cardiac chamber block modeled as a time-varying capacitor and inductor, and several boundary condition blocks including simple flow, pressure and resistors blocks, windkessel boundary conditions, coronary boundary conditions that include the intramyocardial pressure experienced by coronary arteries, as well as two-sided versions of windkessel and coronary boundary conditions that allow a user to build closed-loop circulation models (H. Kim et al., 2010; H. J. Kim et al., 2009; Menon et al., 2023; Menon, Khan, et al., 2024; Mirramezani et al., 2019; Vignon-Clementel et al., 2006). The input to svZeroDSolver is a .json file which specifies the simulation parameters (number of time steps, cardiac cycles, etc.), the types of blocks to be included in the specific model, the boundary conditions, and how the blocks are connected (typically using junction blocks). Each of these blocks generally requires several parameters, which can be specified using a steady value or a list of time-varying values. The solver can either run simulations for a specified number of time steps and cardiac cycles or until the difference in mean quantities between consecutive cardiac cycles is below a given threshold.

The documentation for svZeroDSolver is automatically built on GitHub using Doxygen. It includes instructions for installation, user guides for svZeroDSolver and its various applications, as well as mathematical and graphical descriptions of each zero-dimensional block that is implemented in the solver. Examples of configuration files to run svZeroDSolver simulations using the various available blocks are in svZeroDSolver/tests/cases. The repository also includes examples demonstrating the simple API for interfacing between svZeroDSolver and external C++ software packages in svZeroDSolver/tests/test_interface. Details on creating zero-dimensional simulations from three-dimensional models using the SimVascular graphical interface are available in the SimVascular documentation.

Future development plans include functionality to specify time-varying block parameters as mathematical expressions using the exprtk package. We are also expanding the available blocks to more accurately model hemodynamics, such as by using data-driven models for pressure losses at arbitrarily shaped vascular junctions (Rubio et al., 2025). In addition, we plan to extend the svZeroDGUI application to interactively create custom zero-dimensional boundary conditions for three-dimensional simulations. The development team actively implements new features, blocks and test cases to build on the capabilities of svZeroDSolver and ensure its accuracy and speed.

Acknowledgements

This work was supported by National Science Foundation grants 1663671 and 2310909, by the National Heart, Lung, and Blood Institute of the National Institutes of Health under Award Numbers R01HL141712 and K99HL161313, and the Stanford Maternal and Child Health Institute.

References

- Africa, P. C., Fumagalli, I., Bucelli, M., Zingaro, A., Fedele, M., Dede', L., & Quarteroni, A. (2024). Lifex-cfd: An open-source computational fluid dynamics solver for cardiovascular applications. *Computer Physics Communications*, 296, 109039. <https://doi.org/10.1016/j.cpc.2023.109039>
- Agdestein, S. D., Valen-Sendstad, K., & Diem, A. K. (2018). Artery.FE: An implementation of the 1D blood flow equations in FEniCS. *Journal of Open Source Software*, 3(32), 1107. <https://doi.org/10.21105/joss.01107>
- Alastruey, J., Parker, K. H., & Sherwin, S. J. (2012). Arterial pulse wave haemodynamics. *11th International Conference on Pressure Surges*, 401–443.

- 207 Arthurs, R. A. M., Christopher J. AND Khlebnikov. (2021). CRIMSON: An open-source
208 software framework for cardiovascular integrated modelling and simulation. *PLOS Compu-*
209 *tational Biology*, 17(5), 1–21. <https://doi.org/10.1371/journal.pcbi.1008881>
- 210 Benemerito, I., Melis, A., Wehenkel, A., & Marzo, A. (2024). openBF: An open-source finite
211 volume 1D blood flow solver. *Physiological Measurement*, 45(12), 125002.
- 212 Clerx, M., Cooling, M. T., Cooper, J., Garny, A., Moyle, K., Nickerson, D. P., Nielsen, P.
213 M. F., & Sorby, H. (2020). CellML 2.0. *Journal of Integrative Bioinformatics*, 17(2–3),
214 20200021. <https://doi.org/doi:10.1515/jib-2020-0021>
- 215 Diem, A. K., & Bressloff, N. W. (2017). VaMpy: A python package to solve 1D blood flow
216 problems. *Journal of Open Research Software*. <https://doi.org/10.5334/jors.159>
- 217 Guennebaud, G., Jacob, B., & others. (2010). *Eigen v3*. <http://eigen.tuxfamily.org>.
- 218 Heldt, T., Mukkamala, R., Moody, G. B., & Mark, R. G. (2010). CVSim: An open-source
219 cardiovascular simulator for teaching and research. *The Open Pacing, Electrophysiology &*
220 *Therapy Journal*, 3, 45.
- 221 Hirschvogel, M. (2024). Ambit – a FEniCS-based cardiovascular multi-physics solver. *Journal*
222 *of Open Source Software*, 9(93), 5744. <https://doi.org/10.21105/joss.05744>
- 223 Huttary, R., Goubergrits, L., Schütte, C., & Bernhard, S. (2017). Simulation, identification
224 and statistical variation in cardiovascular analysis (SISCA) – a software framework for
225 multi-compartment lumped modeling. *Computers in Biology and Medicine*, 87, 104–123.
226 <https://doi.org/https://doi.org/10.1016/j.combiomed.2017.05.021>
- 227 Jansen, K. E., Whiting, C. H., & Hulbert, G. M. (2000). A generalized-alpha method for
228 integrating the filtered navier–stokes equations with a stabilized finite element method.
229 *Computer Methods in Applied Mechanics and Engineering*, 190(3), 305–319. [https://doi.org/10.1016/S0045-7825\(00\)00203-6](https://doi.org/10.1016/S0045-7825(00)00203-6)
- 230
231 Kim, H. J., Vignon-Clementel, I. E., Figueroa, C. A., Ladisa, J. F., Jansen, K. E., Feinstein,
232 J. A., & Taylor, C. A. (2009). On coupling a lumped parameter heart model and a
233 three-dimensional finite element aorta model. *Annals of Biomedical Engineering*, 37(11),
234 2153–2169. <https://doi.org/10.1007/s10439-009-9760-8>
- 235 Kim, H., Vignon-Clementel, I., Coogan, J., Figueroa, C., Jansen, K., & Taylor, C. (2010).
236 Patient-specific modeling of blood flow and pressure in human coronary arteries. *Annals of*
237 *Biomedical Engineering*, 38, 3195–3209. <https://doi.org/10.1007/s10439-010-0083-6>
- 238 Lee, J. D., Richter, J., Pfaller, M. R., Szafron, J. M., Menon, K., Zanoni, A., Ma, M. R.,
239 Feinstein, J. A., Kreutzer, J., Marsden, A. L., & Schiavazzi, D. E. (2024). A probabilistic
240 neural twin for treatment planning in peripheral pulmonary artery stenosis. *International*
241 *Journal for Numerical Methods in Biomedical Engineering*, 40(5), e3820. <https://doi.org/10.1002/cnm.3820>
- 242
243 Marsden, A. L. (2014). Optimization in cardiovascular modeling. *Annual Review of Fluid*
244 *Mechanics*, 46, 519–546. <https://doi.org/10.1146/ANNUREV-FLUID-010313-141341>
- 245 Menon, K., Hu, Z., & Marsden, A. L. (2024). Cardiovascular fluid dynamics: A journey
246 through our circulation. *Flow*, 4, E7. <https://doi.org/10.1017/flo.2024.5>
- 247 Menon, K., Khan, M. O., Sexton, Z. A., Richter, J., Nguyen, P. K., Malik, S. B., Boyd, J.,
248 Nieman, K., & Marsden, A. L. (2024). Personalized coronary and myocardial blood flow
249 models incorporating CT perfusion imaging and synthetic vascular trees. *Npj Imaging*,
250 2(9). <https://doi.org/10.1101/2023.08.17.23294242>
- 251 Menon, K., Seo, J., Fukazawa, R., Ogawa, S., Kahn, A. M., Burns, J. C., & Marsden, A.
252 L. (2023). Predictors of myocardial ischemia in patients with kawasaki disease: Insights
253 from patient-specific simulations of coronary hemodynamics. *Journal of Cardiovascular*

- 254 *Translational Research*, 16, 1099–1109.
- 255 Menon, K., Zanoni, A., Khan, O., Geraci, G., Nieman, K., Schiavazzi, D. E., & Marsden, A. L.
256 (2024). Personalized and uncertainty-aware coronary hemodynamics simulations: From
257 bayesian estimation to improved multi-fidelity uncertainty quantification. *arXiv*, 2409.02247.
258 <https://arxiv.org/abs/2409.02247>
- 259 Mirramezani, M., Diamond, S. L., Litt, H. I., & Shadden, S. C. (2019). Reduced Order
260 Models for Transstenotic Pressure Drop in the Coronary Arteries. *Journal of Biomechanical*
261 *Engineering*, 141(3), 31005. <https://doi.org/10.1115/1.4042184>
- 262 Pfaller, M. R., Pegolotti, L., Pham, J., Rubio, N. L., & Marsden, A. L. (2024). Chapter
263 20 - reduced-order modeling of cardiovascular hemodynamics. In T. C. Gasser, S. Avril,
264 & J. A. Elefteriades (Eds.), *Biomechanics of the aorta* (pp. 449–476). Academic Press.
265 <https://doi.org/10.1016/B978-0-323-95484-6.00016-6>
- 266 Pfaller, M. R., Pham, J., Verma, A., Pegolotti, L., Wilson, N. M., Parker, D. W., Yang, W.,
267 & Marsden, A. L. (2022). Automated generation of 0D and 1D reduced-order models of
268 patient-specific blood flow. *International Journal for Numerical Methods in Biomedical*
269 *Engineering*, 38(10). <https://doi.org/10.1002/cnm.3639>
- 270 Pfaller, M. R., Pham, J., Wilson, N. M., Parker, D. W., & Marsden, A. L. (2021). On the
271 periodicity of cardiovascular fluid dynamics simulations. *Annals of Biomedical Engineering*.
272 <https://doi.org/10.1007/s10439-021-02796-x>
- 273 Regazzoni, F., & Quarteroni, A. (2021). Accelerating the convergence to a limit cycle in 3D
274 cardiac electromechanical simulations through a data-driven 0D emulator. *Computers in Bi-*
275 *ology and Medicine*, 135, 104641. <https://doi.org/10.1016/j.combiomed.2021.104641>
- 276
- 277 Richter, J., Nitzler, J., Pegolotti, L., Menon, K., Biehler, J., Wall, W. A., Schiavazzi, D. E.,
278 Marsden, A. L., & Pfaller, M. R. (2024). Bayesian windkessel calibration using optimized
279 0D surrogate models. *arXiv*, 2404.14187.
- 280 Rosalia, L., Ozturk, C., Van Story, D., Horvath, M. A., & Roche, E. T. (2021). Object-
281 oriented lumped-parameter modeling of the cardiovascular system for physiological and
282 pathophysiological conditions. *Advanced Theory and Simulations*, 4(3), 2000216.
- 283 Rubio, N. L., Pegolotti, L., Pfaller, M. R., Darve, E. F., & Marsden, A. L. (2025). Hybrid physics-
284 based and data-driven modeling of vascular bifurcation pressure differences. *Computers in*
285 *Biology and Medicine*, 184, 109420. <https://doi.org/10.1016/j.combiomed.2024.109420>
- 286 Taylor, C. A., Petersen, K., Xiao, N., Sinclair, M., Bai, Y., Lynch, S. R., UpdePac, A.,
287 & Schaap, M. (2023). Patient-specific modeling of blood flow in the coronary arteries.
288 *Computer Methods in Applied Mechanics and Engineering*, 417, 116414. <https://doi.org/10.1016/j.cma.2023.116414>
- 289
- 290 Updegrove, A., Wilson, N. M., Merkow, J., Lan, H., Marsden, A. L., & Shadden, S. C. (2017).
291 SimVascular: An Open Source Pipeline for Cardiovascular Simulation. *Annals of Biomedical*
292 *Engineering*, 45(3), 525–541. <https://doi.org/10.1007/s10439-016-1762-8>
- 293 Vignon-Clementel, I. E., Alberto Figueroa, C., Jansen, K. E., & Taylor, C. A. (2006). Outflow
294 boundary conditions for three-dimensional finite element modeling of blood flow and
295 pressure in arteries. *Computer Methods in Applied Mechanics and Engineering*, 195(29–32),
296 3776–3796. <https://doi.org/10.1016/j.cma.2005.04.014>
- 297 Zanoni, A., Geraci, G., Salvador, M., Menon, K., Marsden, A. L., & Schiavazzi, D. E. (2024).
298 Improved multifidelity Monte Carlo estimators based on normalizing flows and dimensionality
299 reduction techniques. *Computer Methods in Applied Mechanics and Engineering*, 429,
300 117119. <https://doi.org/10.1016/j.cma.2024.117119>
- 301 Zhu, C., Vedula, V., Parker, D., Wilson, N., Shadden, S., & Marsden, A. (2022). svFSI: A

302 multiphysics package for integrated cardiac modeling. *Journal of Open Source Software*,
303 7(78), 4118. <https://doi.org/10.21105/joss.04118>

DRAFT