The Journal of Open Source Software

# svZeroDSolver: A modular package for lumped-parameter cardiovascular flow simulations

**Karthik Menon[1*], Jakob Richter[1*], Martin R. Pfaller[1*], Jonathan Pham[1], Kaitlin E. Harold[1], Dave Parker[1], and Alison L. Marsden[1¶]**

**1** Stanford University, Stanford, CA, United States of America ¶ Corresponding author * These authors contributed equally.

**DOI:** 10.xxxxxx/draft

**Software**

- Review
- Repository
- Archive

**Editor:** Open Journals

**Reviewers:**

- @openjournals

**Submitted:** 01 January 1970
**Published:** unpublished



## 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 various high and low-fidelity techniques, lumped-parameter or zero-dimensional modeling is a low-order technique that is widely used in various applications that require quick estimation of bulk flow quantities, such as flow and pressure at specific anatomical locations.

We introduce `svZeroDSolver`, an efficient and modular package for performing lumped-parameter (zero-dimensional) simulations of cardiovascular blood flow. `svZeroDSolver` is written in C++ using an object-oriented framework. It designed so that a user can simply specify a `.json` dictionary of lumped-parameter "blocks" – such as blood vessels, valves, heart chambers, junctions between blood vessels, and boundary conditions (along with their associated parameters) – which the code uses to automatically assemble and solve the governing equations corresponding to the user-specified vascular model. In addition, the package includes Python and C++ APIs to facilitate its use with other software packages. For example, it can be integrated into Python-based optimization and uncertainty quantification applications. It can also be interfaced with high-performance C++/Fortran software for high-fidelity cardiovascular flow simulations, where `svZeroDSolver` can conveniently provide lumped-parameter boundary conditions. `svZeroDSolver` also includes an application, called `svZeroDCalibrator`, to automatically calibrate parameters of a given zero-dimensional model to recapitulate independent measurements of hemodynamics at specific anatomical locations from high-fidelity simulations – thus improving the accuracy of zero-dimensional simulations.

## Statement of need

The ability to non-invasively quantify patient-specific hemodynamics via computational simulations has been shown to improve patient outcomes and reduce 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 a variety of 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 feasible for use within typical clinical settings or for applications, such as optimization, which often require thousands of 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 very valuable in applications that require near real-time quantification of bulk hemodynamics, as well as applications that rely on thousands of repeated model evaluations, such as optimization and uncertainty quantification (Lee et al., 2024; Richter et al., 2024; Zanoni et al., 2024). They are also used in conjunction with high-fidelity simulations where lumped-parameter models are used as 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., and each of these anatomical configurations is governed by a distinct set of governing equations. It is therefore 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 confguration.

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; Richter et al., 2024; Zanoni 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). In particular, the C++ interface has been succesfully coupled with high-fidelity multi-physics solvers `svSolver` and `svFSI`, which are 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).

In addition, the `svZeroDCalibrator` application within `svZeroDSolver` includes functionality to improve 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 on purely the anatomy of the vascular region of interest (Richter et al., 2024). The accuracy of `svZeroDSolver` is assessed using continuous integration tests and has also been verified by comparing with high-fidelity three dimensional simulations (Pfaller et al., 2022). This combination of features makes `svZeroDSolver` uniquely applicable to a wide range of applications in cardiovascular biomechanics.

The Journal of Open Source Software

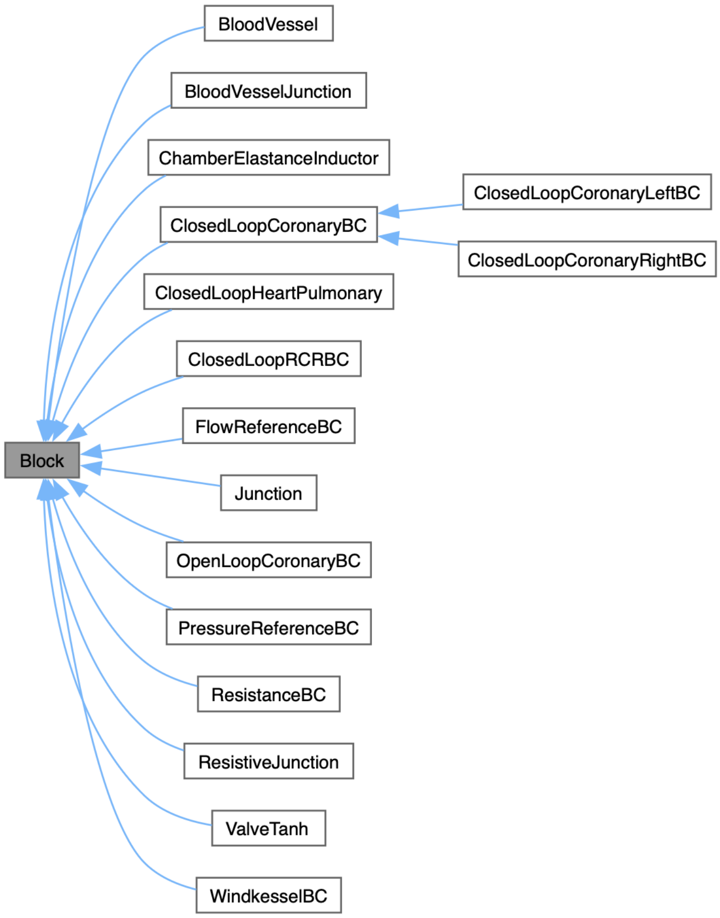


**Figure 1:** Various zero-dimensional "blocks" included in `svZeroDSolver` at the time of writing.

# 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 how they are connected to each other, 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.

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, a list of time-varying values, or a mathematical function which is parsed using exprtk. 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.

`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 different 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 closed-loop versions of windkessel and coronary boundary conditions that allow a user to build a closed-loop circulation model (H. Kim et al., 2010; H. J. Kim et al., 2009; Menon et al., 2023; Menon, Khan, et al., 2024).

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_inetrface`. Details on creating zero-dimensional simulations from three-dimensional models using the `Simvascular` graphical interface are available on the Simvascular documentation.

Future development plans include a graphical interface to create circuit representations of existing zero-dimensional models, as well as functionality to create `svZeroDSolver`-compatible `.json` configuration files directly from graphical (circuit) representations of user-specified circulatory models. 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. The development team actively implements new features, blocks and test cases to build on the capabilities of svZeroDSolver and ensure its accuracy and speed.

# Acknowledgments

This work was supported by ???. Previous contributions to the software by Dr. Aekaansh Verma are acknowledged.

# References

Guennebaud, G., Jacob, B., & others. (2010). *Eigen v3*. http://eigen.tuxfamily.org.

Jansen, K. E., Whiting, C. H., & Hulbert, G. M. (2000). A generalized-alpha method for integrating the filtered navier–stokes equations with a stabilized finite element method. *Computer Methods in Applied Mechanics and Engineering*, *190*(3), 305–319. https://doi.org/10.1016/S0045-7825(00)00203-6

Kim, H. J., Vignon-Clementel, I. E., Figueroa, C. A., Ladisa, J. F., Jansen, K. E., Feinstein, J. A., & Taylor, C. A. (2009). On coupling a lumped parameter heart model and a three-dimensional finite element aorta model. *Annals of Biomedical Engineering*, *37*(11), 2153–2169. https://doi.org/10.1007/s10439-009-9760-8

Kim, H., Vignon-Clementel, I., Coogan, J., Figueroa, C., Jansen, K., & Taylor, C. (2010). Patient-specific modeling of blood flow and pressure in human coronary arteries. *Annals of Biomedical Engineering*, *38*, 3195–3209. https://doi.org/10.1007/s10439-010-0083-6

Lee, J. D., Richter, J., Pfaller, M. R., Szafron, J. M., Menon, K., Zanoni, A., Ma, M.

R., Feinstein, J. A., Kreutzer, J., Marsden, A. L., & Schiavazzi, D. E. (2024). A probabilistic neural twin for treatment planning in peripheral pulmonary artery stenosis. *International Journal for Numerical Methods in Biomedical Engineering*, *40*(5), e3820. https://doi.org/https://doi.org/10.1002/cnm.3820

Marsden, A. L. (2014). Optimization in cardiovascular modeling. *Annual Review of Fluid Mechanics*, *46*, 519–546. https://doi.org/10.1146/ANNUREV-FLUID-010313-141341

Menon, K., Hu, Z., & Marsden, A. L. (2024). Cardiovascular fluid dynamics: A journey through our circulation. *Flow*, *4*, E7. https://doi.org/10.1017/flo.2024.5

Menon, K., Khan, M. O., Sexton, Z. A., Richter, J., Nguyen, P. K., Malik, S. B., Boyd, J., Nieman, K., & Marsden, A. L. (2024). Personalized coronary and myocardial blood flow models incorporating CT perfusion imaging and synthetic vascular trees. *Npj Imaging*, *2*(9). https://doi.org/10.1101/2023.08.17.23294242

Menon, K., Seo, J., Fukazawa, R., Ogawa, S., Kahn, A. M., Burns, J. C., & Marsden, A. L. (2023). Predictors of myocardial ischemia in patients with kawasaki disease: Insights from patient-specific simulations of coronary hemodynamics. *Journal of Cardiovascular Translational Research*, *16*, 1099–1109.

Pfaller, M. R., Pham, J., Verma, A., Pegolotti, L., Wilson, N. M., Parker, D. W., Yang, W., & Marsden, A. L. (2022). Automated generation of 0D and 1D reduced-order models of patient-specific blood flow. *International Journal for Numerical Methods in Biomedical Engineering*, *38*(10). https://doi.org/10.1002/cnm.3639

Richter, J., Nitzler, J., Pegolotti, L., Menon, K., Biehler, J., Wall, W. A., Schiavazzi, D. E., Marsden, A. L., & Pfaller, M. R. (2024). *Bayesian windkessel calibration using optimized 0D surrogate models*. https://arxiv.org/abs/2404.14187

Taylor, C. A., Petersen, K., Xiao, N., Sinclair, M., Bai, Y., Lynch, S. R., UpdePac, A., & Schaap, M. (2023). Patient-specific modeling of blood flow in the coronary arteries. *Computer Methods in Applied Mechanics and Engineering*, *417*, 116414. https://doi.org/https://doi.org/10.1016/j.cma.2023.116414

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

Zanoni, A., Geraci, G., Salvador, M., Menon, K., Marsden, A. L., & Schiavazzi, D. E. (2024). Improved multifidelity Monte Carlo estimators based on normalizing flows and dimensionality reduction techniques. *Computer Methods in Applied Mechanics and Engineering*, *429*, 117119. https://doi.org/https://doi.org/10.1016/j.cma.2024.117119

Zhu, C., Vedula, V., Parker, D., Wilson, N., Shadden, S., & Marsden, A. (2022). svFSI: A multiphysics package for integrated cardiac modeling. *Journal of Open Source Software*, *7*(78), 4118. https://doi.org/10.21105/joss.04118