

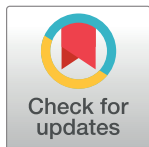
## RESEARCH ARTICLE

# CRIMSON: An open-source software framework for cardiovascular integrated modelling and simulation

Christopher J. Arthurs<sup>1\*</sup>, Rostislav Khlebnikov<sup>1</sup>, Alex Melville<sup>2</sup>, Marija Marčan<sup>1</sup>, Alberto Gomez<sup>1</sup>, Desmond Dillon-Murphy<sup>1</sup>, Federica Cuomo<sup>2</sup>, Miguel Silva Vieira<sup>1</sup>, Jonas Schollenberger<sup>2</sup>, Sabrina R. Lynch<sup>2</sup>, Christopher Tossas-Betancourt<sup>2</sup>, Kritika Iyer<sup>2</sup>, Sara Hopper<sup>2</sup>, Elizabeth Livingston<sup>2</sup>, Pouya Youssefi<sup>1</sup>, Alia Noorani<sup>1</sup>, Sabrina Ben Ahmed<sup>3</sup>, Foeke J. H. Nauta<sup>2</sup>, Theodorus M. J. van Bakel<sup>2</sup>, Yunus Ahmed<sup>2</sup>, Petrus A. J. van Bakel<sup>2</sup>, Jonathan Mynard<sup>4</sup>, Paolo Di Achille<sup>5</sup>, Hamid Gharahi<sup>2</sup>, Kevin D. Lau<sup>1</sup>, Vasilina Filonova<sup>2</sup>, Miquel Aguirre<sup>3</sup>, Nitesh Nama<sup>2</sup>, Nan Xiao<sup>1</sup>, Seungik Baek<sup>6</sup>, Krishna Garikipati<sup>2</sup>, Onkar Sahni<sup>7</sup>, David Nordsletten<sup>2</sup>, C. Alberto Figueroa<sup>2\*</sup>

**1** King's College London, London, United Kingdom, **2** University of Michigan, Ann Arbor, Michigan, United States of America, **3** Ecole des Mines de Saint-Étienne, Saint-Étienne, France, **4** Murdoch Childrens Research Institute, Melbourne, Australia, **5** Broad Institute, Cambridge, Massachusetts, United States of America, **6** Michigan State University, East Lansing, Michigan, United States of America, **7** Rensselaer Polytechnic Institute, Troy, New York, United States of America

\* [christopher.arthurs@kcl.ac.uk](mailto:christopher.arthurs@kcl.ac.uk) (CJA); [figueroa@med.umich.edu](mailto:figueroa@med.umich.edu) (CAF)



## OPEN ACCESS

**Citation:** Arthurs CJ, Khlebnikov R, Melville A, Marčan M, Gomez A, Dillon-Murphy D, et al. (2021) CRIMSON: An open-source software framework for cardiovascular integrated modelling and simulation. *PLoS Comput Biol* 17(5): e1008881. <https://doi.org/10.1371/journal.pcbi.1008881>

**Editor:** Dina Schneidman-Duhovny, Hebrew University of Jerusalem, ISRAEL

**Received:** September 4, 2020

**Accepted:** March 17, 2021

**Published:** May 10, 2021

**Copyright:** © 2021 Arthurs et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

**Data Availability Statement:** CRIMSON binaries for Microsoft Windows 10, documentation and example input files are freely available for download from [www.crimson.software](http://www.crimson.software), and the source code with compilation instructions is available on GitHub <https://github.com/carthurs/CRIMSONFlowsolver> (CRIMSON Flowsolver) under the GPL v3.0 license, and <https://github.com/carthurs/CRIMSONGUI> (CRIMSON GUI), under the AGPL v3.0 license. Support is available on the CRIMSON Google Groups forum, located at

## Abstract

In this work, we describe the CRIMSON (Cardiovascular Integrated Modelling and Simulation) software environment. CRIMSON provides a powerful, customizable and user-friendly system for performing three-dimensional and reduced-order computational haemodynamics studies via a pipeline which involves: 1) segmenting vascular structures from medical images; 2) constructing analytic arterial and venous geometric models; 3) performing finite element mesh generation; 4) designing, and 5) applying boundary conditions; 6) running incompressible Navier-Stokes simulations of blood flow with fluid-structure interaction capabilities; and 7) post-processing and visualizing the results, including velocity, pressure and wall shear stress fields. A key aim of CRIMSON is to create a software environment that makes powerful computational haemodynamics tools accessible to a wide audience, including clinicians and students, both within our research laboratories and throughout the community. The overall philosophy is to leverage best-in-class open source standards for medical image processing, parallel flow computation, geometric solid modelling, data assimilation, and mesh generation. It is actively used by researchers in Europe, North and South America, Asia, and Australia. It has been applied to numerous clinical problems; we illustrate applications of CRIMSON to real-world problems using examples ranging from pre-operative surgical planning to medical device design optimization.

<https://groups.google.com/forum/#!forum/crimson-users>. Data available from University of Michigan's Deep Blue archive. under <https://doi.org/10.7302/4vdx-ek59>.

**Funding:** CAF received support from the European Research Council (<https://erc.europa.eu/>) under the European Union's Seventh Framework Programme (FP/2007-2013) / ERC Grant Agreement n. 307532, the United States National Institutes of Health (<https://www.nih.gov/>) Grants U01 HL135842 and R01 HL105297, the American Heart Association grant 19A1ML34910010. CJA received support under the Wellcome Trust (<https://wellcome.ac.uk/>) Institutional Strategic Support Fund grant to King's College London under grant number [204823/Z/16/Z]. JM is funded by the National Health and Medical Research Council of Australia (<https://www.nhmrc.gov.au/>) and the National Heart Foundation of Australia (<https://www.heartfoundation.org.au/>). DN: EP/R003866/1 Engineering and Physical Sciences Research Council (EPSRC) <https://epsrc.ukri.org/> The work was supported by the United Kingdom Department of Health via the National Institute for Health Research (NIHR) (<https://www.nihr.ac.uk/>) comprehensive Biomedical Research Centre award to Guy's & St Thomas' NHS Foundation Trust in partnership with King's College London and King's College Hospital NHS Foundation Trust, and The Centre for Medical Engineering (CME) at King's College London. We have no further funding to declare. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

**Competing interests:** I have read the journal's policy and the authors of this manuscript have the following competing interests: CAF and CA are Members of CRIMSON Technologies LLC, a company established to support the future development of CRIMSON.

## Author summary

We provide the first full presentation in the literature of CRIMSON, the Cardiovascular Integrated Modelling and Simulation Package. CRIMSON consists of a graphical user interface desktop computer program for creating geometric models of blood vessels from medical imaging scans, specifying parameters such as the stiffness of the artery walls, the resistance of connected vessels which are not visible on the scans, and determining the appropriate parameters for all aspects of the model. CRIMSON additionally consists of the Flowsolver, a high-performance computing package which simulates the flow of blood through the models created in the graphical user interface. Whilst several packages which can simulate blood flow exist, most target general fluid simulations, and this lack of specialisation means that blood flow simulation is harder to perform, and can require *ad hoc* (and potentially scientifically-limiting) workflow decisions. CRIMSON's specialisation deals with these problems, as well as presenting a number of unique features which are unavailable elsewhere.

## Introduction

One of the revolutionary successes of twentieth century in applied mathematics is the development of the finite element method (FEM) into a reliable engineering tool. It is now routinely deployed by practitioners in fields where the problems of interest are described by systems of partial differential equations (PDEs). FEM, in combination with high-performance computing (HPC), has enabled solution of these systems of PDEs rapidly and accurately.

In the present work, FEM is applied to the incompressible Navier-Stokes equations in patient-specific vascular geometries. This enables analysis of clinically-relevant blood flow phenomena including the impact of the geometry, vascular wall properties, or prospective surgical decisions on blood pressure, wall shear stress, and mass (e.g. protein or drug) transport. These capabilities support the key applications of surgical planning, patient diagnosis, medical device design and optimisation, and basic cardiovascular disease research. Such applications are collectively called *computational haemodynamics* (CH).

A serious factor limiting widespread adoption of CH has been the arcane and unwieldy nature of existing academic software packages and workflows, coupled with the fact that no commercial package truly supports cutting-edge CH. Specifically, key aspects of the computational modelling workflow have been done in a manual fashion, resulting in cumbersome, manual workflows. This is error-prone and non-systematic. Examples of such operations are: parameter estimation, application of physiologically-relevant boundary conditions such as PC-MRI informed velocity profiles, and the ability to simulate physiologically-varying conditions.

More generally, and due in part to the patchwork nature of academic software development involving multiple researchers with differing goals over many years, simulation workflows often require using multiple pieces of software and ad-hoc scripts. One pathway to addressing this issue is specialised, problem-specific commercial enterprise. For example, HeartFlow Ltd. provides CH-as-a-service for diagnosis of coronary artery disease, taking medical imaging data and producing computational clinical reports on the hemodynamic significance of coronary lesions. In their workflow, medical image data are uploaded to the HeartFlow servers, and a report is returned to the clinician [1]. However, if CH is to be performed in a research setting using open-source tools, custom workflows must be created. Given the complexity of the software components and the interdisciplinary nature of the user base (e.g., clinicians, students,

and academics), it is important that the software provides powerful capabilities, whilst presenting an intuitive, expandable, and user-friendly graphical user interface (GUI). In this article, we describe how we address these issues in our CH software environment CRIMSON (Cardiovascular Integrated Modelling and Simulation).

CRIMSON consists of two components. The first is the CRIMSON GUI, which is a Windows application for setting up simulations via segmenting medical image data, creating finite element meshes, assigning boundary conditions, and examining results. The second component is the CRIMSON Flowsolver, which is a Message Passing Interface (MPI) parallel application, scalable to tens of thousands of cores [2], which can be run on Windows or Linux; simulations with the Flowsolver can either be run on the local machine from within the GUI, or transferred to a high-performance computer for simulation there.

The conceptual ancestor of CRIMSON is SimVascular [3], which has been widely used in numerous CH studies [4–8]. During the early design phase of CRIMSON, we determined that there was a pressing need for developing a CH software package that leveraged the strengths of SimVascular's Flowsolver with a modern UI with broad community support and best-in-class open source components for key operations of the simulation workflow (CAD, boundary condition specification, meshing, parameter estimation, etc.). For example, assigning a boundary condition directly to a surface of the 3D-rendered model was not supported by SimVascular's GUI. CRIMSON GUI was therefore created to avoid these limitations, introduce many novel features and improvements, and expand functionality, workflow robustness, and accessibility. A key decision was to utilise for the first time in CH [9] the Medical Imaging Interaction Toolkit (MITK) [10]. CRIMSON allowed the creation of novel and unique features including GUI support for a wide range of fine-tuneable boundary condition (BC) types, including direct imposition of phase-contrast magnetic resonance imaging (PC-MRI) flow data [11], rapid design and prototyping of arbitrary lumped parameter network (LPN) reduced order models, and spatially-varying vessel wall material properties, all of which are supported in a modern, user-friendly GUI. Furthermore, CRIMSON supports automatic model parameter estimation via filtering of time-resolved data [12]. The key driving philosophy is that for these powerful features to reach their maximum potential, they must be accessible to both clinicians and technical researchers. To achieve this, CRIMSON aims to present clearly-documented, logical workflows in the GUI, and to protect users from common errors. The latter goal is achieved, for example, by supporting clear graphical BC specification in the GUI, by including undo/redo stacks for operations, and by limiting the interaction that users have with text-based configuration files.

Beyond making the basic workflow as user-friendly as possible, CRIMSON has several unique features which enable novel simulation capabilities. For example, CRIMSON allows users to implement—at run-time—arbitrary rules for adjusting boundary condition parameters during simulations, based upon the simulation state. This allows modelling of cardiovascular control mechanisms, and transitional hemodynamic states—for example, between rest and exercise [13]. Further, for model parameterisation, testing, and exploring long-term behaviour, CRIMSON supports a one-click-to-enable pure-zero-dimensional mode, so that the haemodynamic influence of the boundary conditions can be explored without the computational cost of a full 3D domain. This ability for rapid exploration is invaluable for novices learning how parameters affect simulation results in simple models, and for experts exploring parameter space during the development of novel complex models. A comparison of key features between CRIMSON and some alternative packages is presented in [Table 1](#).

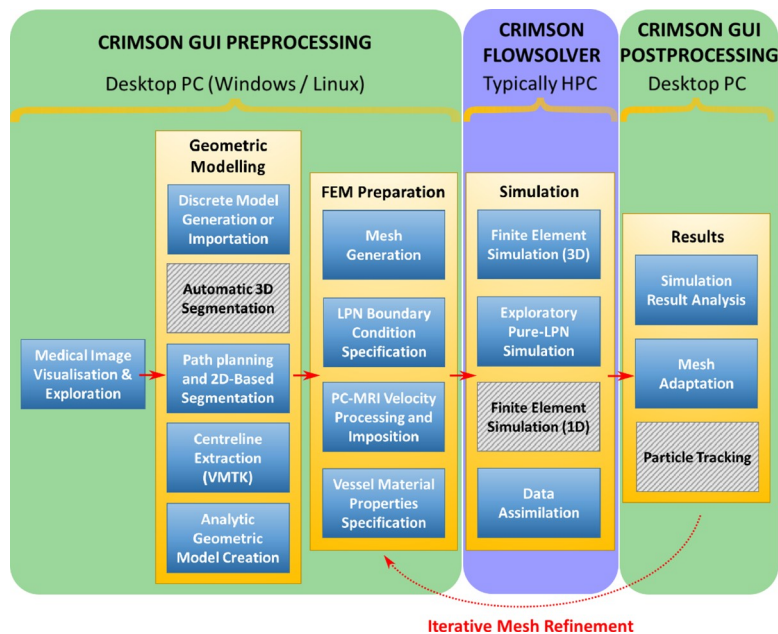
**Table 1. Comparison of selected key features for computational haemodynamics, between CRIMSON and other packages.** The information in the table is to the best of our knowledge at the time of writing. Readers are advised to make their own comparisons.

	CRIMSON	SimVascular	openFOAM	Commercial Options
Specialised for computational haemodynamics, including tools for vascular model creation	Y	Y	N	N
Fully implicit reduced-order models (e.g. Lumped Parameter Models) for boundary condition specification	Y	Y	N	N
Stabilised backflow formulation [14]	Y	Y	N	N
Stabilised formulations for scalar transport [15]	Y	N	N	N
Drag-n-drop arbitrary LPN boundary condition design [16]	Y	N	N	N
One-click Pure zero-dimensional model prototyping	Y	N	N	N
Arbitrary run-time Python scripting for cardiovascular control and transitional physiology [16]	Y	N	N	N
Direct Boundary Condition Specification from PC-MRI data [11]	Y	N	N	N
Kalman filtering tools for determining boundary condition parameters via clinical data assimilation [12]	Y	N	N	N
Fluid-structure interaction via efficient arbitrary Lagrangian-Eulerian vessel wall models [17]	Upcoming	N	N	N
Multi-species reaction-advection-diffusion problems, with run-time specification of reaction equations [15]	Upcoming	N	N	N

<https://doi.org/10.1371/journal.pcbi.1008881.t001>

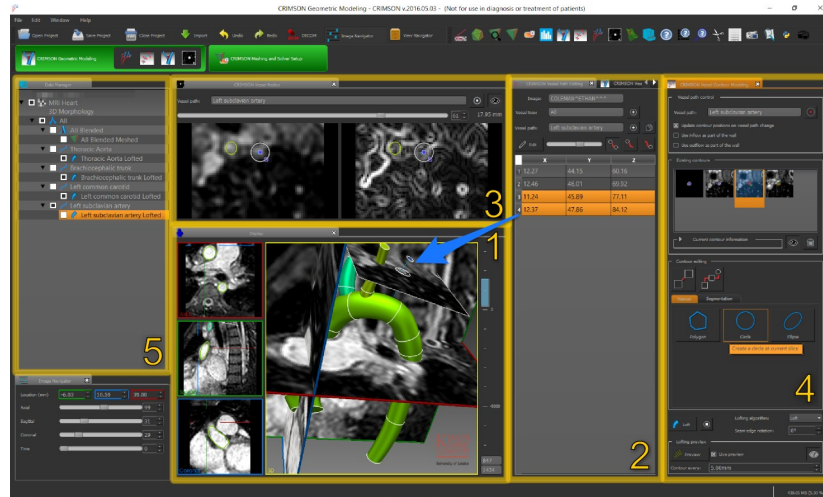
## Design and Implementation

The CRIMSON workflow, from medical imaging data to simulation results, is shown in Fig 1. The three major stages of the pipeline (i.e., Preprocessing, Flowsolver, and Postprocessing) are depicted, including some optional sub-steps in each column which may or may not be part of the process for a given CH application.



**Fig 1. The CRIMSON workflow consists of Preprocessing, Flowsolver, and Postprocessing stages represented in three different columns.** Typical hardware architecture utilized at each stage is noted. Some of the steps within each stage are optional, depending upon the complexity of the simulation. Grey hashed boxes indicate areas where the GUI support is under development. A medical image volume is required as initial input, if not importing a discrete geometric model from elsewhere. The final output is, at minimum, time-resolved pressure and velocity fields throughout the domain.

<https://doi.org/10.1371/journal.pcbi.1008881.g001>



**Fig 2. Geometric modelling aspect of the CRIMSON GUI.** (1) The image data and segmentation is viewed in three adjustable orthogonal planes and a 3D projection. (2) Centerlines are created for each vessel of interest, editable via coordinates in Panel 2, or 3D interaction in Panel 1 (blue arrow shows relationship). (3) A “re-slice” of the image, consisting of a plane perpendicular to the vessel centerline; this is used to draw two-dimensional contour of the vessels (4). Data objects (centerlines, vessel lofts and trees, etc.) are shown in the data manager (5).

<https://doi.org/10.1371/journal.pcbi.1008881.g002>

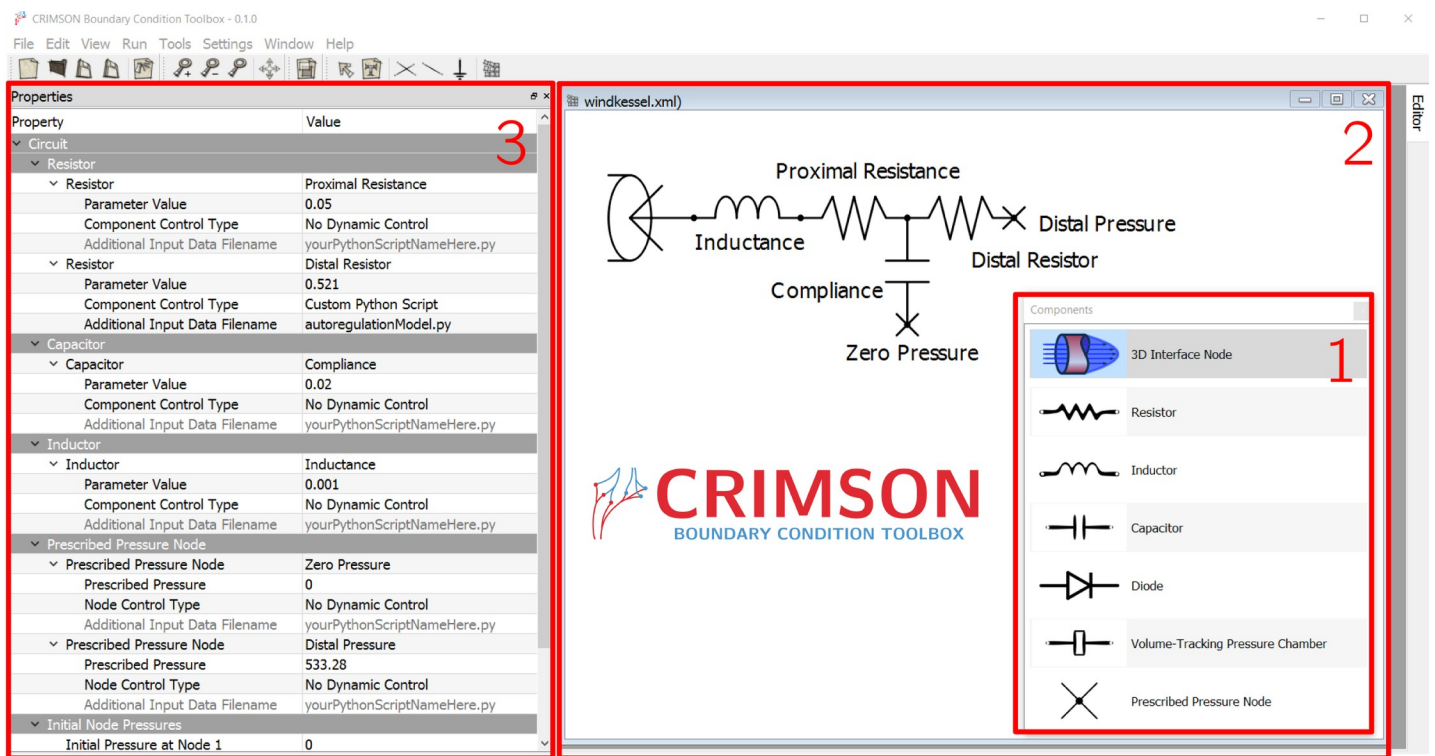
The CRIMSON GUI is built upon MITK, which provides tools to load, visualize and manipulate medical image data. A screenshot of CRIMSON during the Geometric Modelling portion of the workflow is shown in Fig 2. Here, the user loads the medical image volume, and draws one-dimensional paths approximately representing the centreline of each vessel of interest. CRIMSON then re-slices the image data using a plane perpendicular to the centreline at each point along its length (Fig 2, box 3). Upon this re-slice plane, the user generates two-dimensional contours to segment the vessel wall [18,19] using MITK’s manual and semi-automatic segmentation functions, including tools for delineating the vessel cross-section using circles, ellipses or splines. An analytic NURBS surface is then created of each vessel via lofting or sweeping operations. The vessels of interest are united using a blending operation to create a single geometric CAD model [20], which is then volume-meshed using (only) linear tetrahedral elements. Mesh generation is provided by TetGen [21]. Boundary layers can be specified. CAD models can be exported in step, iges, and brep file formats. CRIMSON also allows for direct importing of discrete surface triangulations of vascular geometries (e.g. stl format) created in external packages.

BCs are subsequently assigned, including upstream or downstream vascular reduced order models. The vessel walls can be modelled as rigid or made deformable via a coupled momentum method, making use of a linear elastic constitutive membrane model for the vessel wall [22], and spatially-varying and anisotropic mechanical properties of the arterial wall can be specified using the theory of “Small on Large” [23]. External tissue support can also be specified, modelling the influence of nearby anatomical structures on the blood vessels. BC can be Neumann, Dirichlet, or based upon the coupled multidomain method [24], which imposes a Dirichlet-to-Neumann relationship at the face. Boundaries can optionally use “backflow stabilization” [14] to prevent numerical instability during periods of flow reversal at Dirichlet-to-Neumann surfaces; this critical feature is required to avoid the use of non-physiological “flow extensions” of the 3D domain often in other software packages. Numerical simulation parameters such as time stepping, and non-linear iteration strategies are then set within the GUI. Simulation files are written and passed to the parallel incompressible Flowsolver, which uses linear

tetrahedral finite elements and a Streamline Upwind Petrov-Galerkin (SUPG) stabilized scheme for the spatial discretisation of the Navier-Stokes equations [25], and a second-order accurate generalized- $\alpha$  method [26] for time discretization. The Flowsolver scales efficiently to thousands of cores of HPC hardware, meaning that it can compute pressure and flow fields in very complex problems. Results are retrieved and can be visualized in the CRIMSON GUI. The GUI allows for field-based mesh refinement [27] and repetition of the analysis if necessary. Visualisation is also possible in third-party applications, such as Paraview [28].

The CRIMSON Flowsolver consists of a core Fortran FEM solver, based on PHASTA [29] and closely related to that of SimVascular. We have made significant additions, including: (1) the conversion of the codebase from Fortran 77 to Fortran 90; (2) the addition of a modern C++11 layer to manage extensions, including the powerful new arbitrary BC design and control system; (3) the associated Python interface for dynamically adjusting BC parameters during simulations of transitional physiology; and (4) the use of the Verdati libraries for data assimilation [30]. Additional modifications include the integration of the Google Test framework [31], and our ongoing work towards simulation of non-linear vessel wall-blood flow interactions [32,17] and flexible pipelines for simulation of systems of reaction-advection-diffusion (RAD) transport [15].

To assist with BC design, CRIMSON provides the Netlist Editor Boundary Condition Toolbox (NEBCT), shown in Fig 3. This presents a drag & drop interface for assembling LPNs, represented using electronic circuit symbols and invoking the standard analogy between fluidics and electronics. NEBCT allows tagging of components for Python-specified dynamic control during simulation and generates user-customizable Python controller class definitions,



**Fig 3. The CRIMSON Boundary Condition Toolbox.** Researchers can use this tool to design custom LPN boundary condition models by choosing and arranging components from the toolbox (1) on the workplane (2). Parameters and other component properties can be set in the Properties pane (3). Here, a depiction of a Windkessel model is being created.

<https://doi.org/10.1371/journal.pcbi.1008881.g003>

under the CRIMSON Control Systems Framework. NEBCT is based upon the QSapecNG electrical circuit simulation software [16,33].

The CRIMSON GUI embeds another Python interpreter, which enables functionalities ranging from script-based specification of the local vessel wall thickness and elastic modulus, to definition of Python “Solver Setup” modules; this permits straightforward customization of the CRIMSON GUI’s interface widgets and output files, providing a pathway to compatibility with different fluid solvers. Thus, research groups may write their own Solver Setup to generate simulation files for their own simulation software; the procedure is documented online (<https://crimsonpythonmodules.readthedocs.io/en/latest/>). For example, a Solver Setup has been written for the CHeart fluid solver [34,35]. Some additional details regarding the CRIMSON design and workflow have previously been presented [9].

A summary of the main libraries and frameworks using in CRIMSON is described in Section *Guidance for Getting Started with CRIMSON*.

## Results

We demonstrate CRIMSON using three different examples, with all necessary input files available in File 1 at <https://doi.org/10.7302/4vdx-ek59>. We present two application examples (Cases A and B) highlighting specific aspects of CRIMSON’s capabilities, followed by an example demonstrating the entire workflow (Case C). In Case A, we demonstrate some of the fundamental features of CRIMSON, including running Navier-Stokes simulation in a patient-specific aortic model. In Case B, we demonstrate segmentation of PC-MRI data to create a patient-specific aortic inflow boundary condition. In Case C, we demonstrate the simulation of dynamic adaptations of blood flow in a surgical planning application.

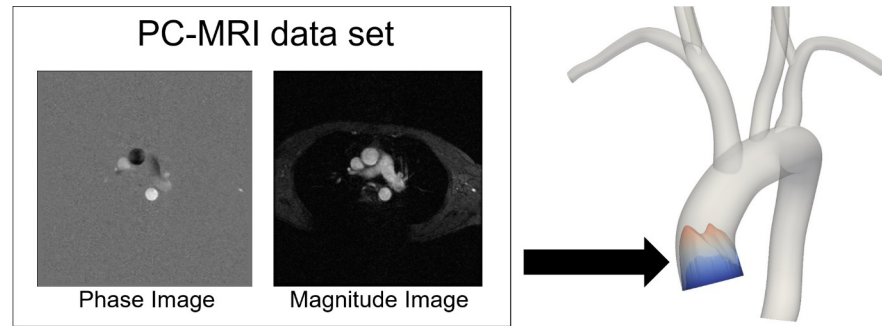
### Example Case A: STL Geometry import, Mesh generation, Netlist boundary conditions, 3D Simulation on a Desktop PC

In Example Case A, we demonstrate the simulation workflow in the case where a geometric vascular model has been created using third-party software, which may be idealised or patient-specific. CRIMSON supports this via a solid model import feature. Note that additional boundary condition data (flow rates at the various model outlets) was assumed in this case; typically, additional patient data such as blood pressure or regional flow rates would be used to parameterize the boundary conditions.

In addition to demonstrating how to work with imported geometries, this example highlights how to use the NEBCT to design simple LPN boundary conditions, how to run low-resolution 3D simulations on a desktop PC (or if the user wishes, high-resolution versions on HPC), how to refine the mesh after a simulation to focus computational effort in regions of high error in subsequent simulations, and how to visualise results within CRIMSON.

### Example Case B: Using Phase-Contrast MRI Data as a patient-specific boundary condition

A unique feature of CRIMSON is the ability to impose image-derived velocity profiles as boundary conditions. Typically, this is used at the aortic root, enabling study of the impact of valvular pathologies—which critically shape the inflow profile—on aortic haemodynamics. Fig 4 illustrates the concept. Imposing patient-specific inflow velocity profiles enables direct comparison with idealized (e.g. healthy) or hypothetical post-operative scenarios (e.g. following valve replacement procedures), without changing any of the other parameters of the system. It



**Fig 4. A PC-MRI (phase & magnitude images) plane slicing through the aortic root can be segmented and processed using CRIMSON, so that patient-specific velocity profiles can be imposed as boundary conditions on the model.**

<https://doi.org/10.1371/journal.pcbi.1008881.g004>

has been shown that CH which use idealised inflow profiles in cases where the patient's valve is diseased can lead to significantly altered simulation results [36,11,37,38].

The supplementary files in Example Case B—available in File 1 at <https://doi.org/10.7302/4vdx-ek59> - show how to use CRIMSON to segment the provided PC-MRI dataset, and impose it upon the provided model.

### Example Case C: Cardiovascular risk assessment post-liver-transplant

In Alagille syndrome patients in need of liver transplant, it is important to understand whether their cardiovascular system can withstand the stresses of the transplant surgery [39–42]. A major concern is that the heart may not be able to support the demands placed on it during *post-reperfusion syndrome* (PRS)—the result of a sudden release of vasoactive mediators into the circulation after reperfusion of the new liver. PRS is characterised by a decrease in mean arterial pressure and systemic vascular resistance, and an increase in pulmonary pressure and central venous pressure [43]. These factors have the effect of simultaneously increasing the workload on the heart, and—due to the drop in mean arterial pressure—decreasing the coronary perfusion pressure gradient. Thus, the supply of oxygen to the ventricles may become insufficient, placing the patient at risk of cardiovascular complications.

To study this, a subject-specific model of an Alagille syndrome patient was created in CRIMSON, consisting of two (pulmonary and systemic) 3D domains segmented from magnetic resonance images, and a closed-loop circulatory system created using NEBCT (Fig 5). In the present article, we illustrate the difference between the myocardial oxygen supply and demand in one region of the left ventricular myocardium; for further details see [44]. This difference was computed using a model of myocardial oxygen supply and demand [13] and implemented within a coronary boundary condition using the CRIMSON Control Systems Framework.

Fig 6 compares the imbalance between myocardial oxygen demand and supply (“myocardial hunger”) in a baseline case (case C1), calibrated to patient haemodynamic data at rest, to that same imbalance under a simulated PRS condition (case C2). No long-term mismatch is detected under baseline conditions (the long-wave oscillation is a computational artefact as the simulation reaches a fully periodic state). Conversely, the continual growth in myocardial hunger under PRS indicates that this patient may be at risk of cardiovascular complications post-transplant.

The data available in File 1 at <https://doi.org/10.7302/4vdx-ek59> describes how to run this case in 0D and visualize the results, as well as how to run in 3D, and visualize parameters such as local pressures and flows. The 0D simulation is presented here so that it is computationally



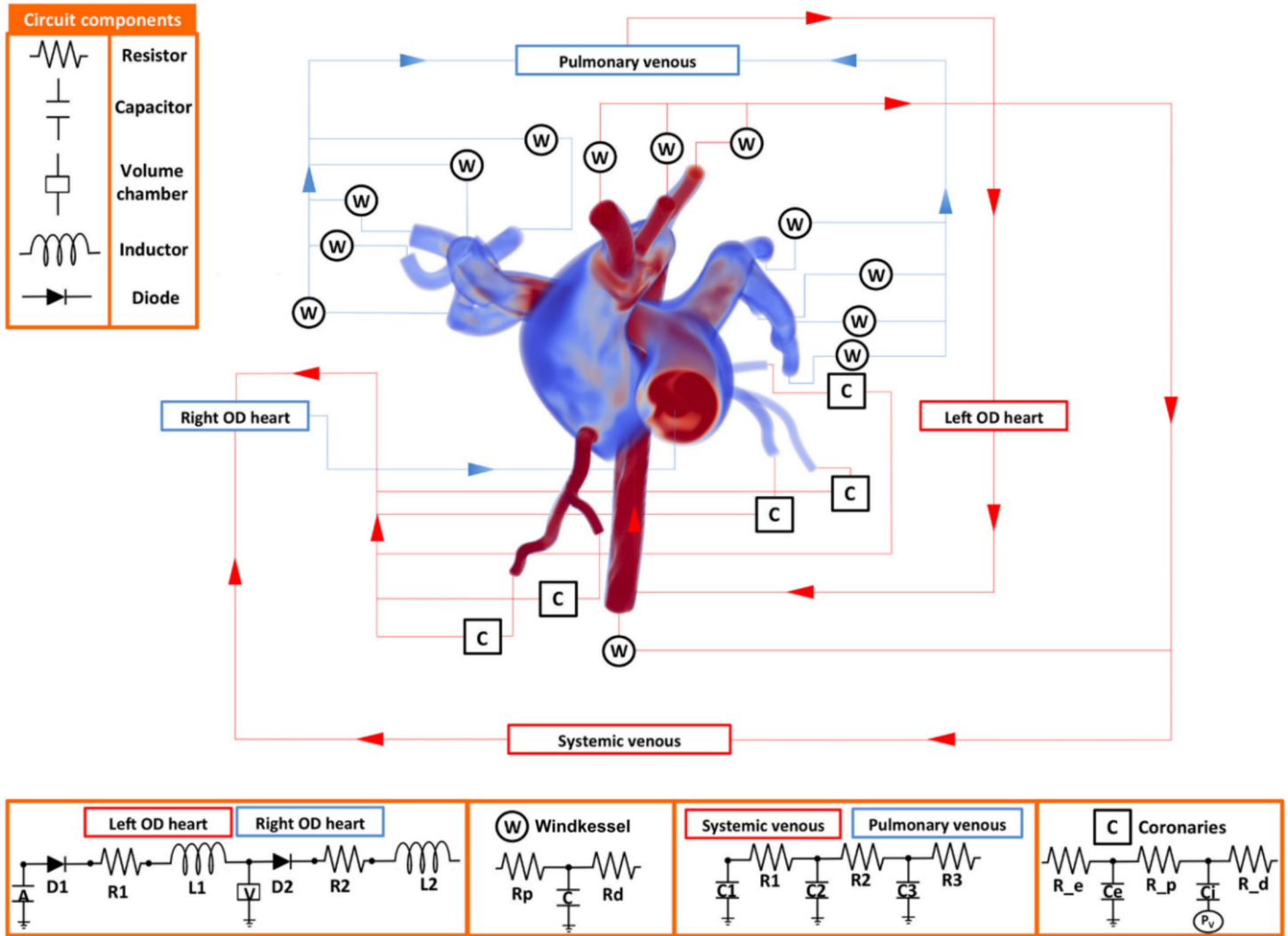


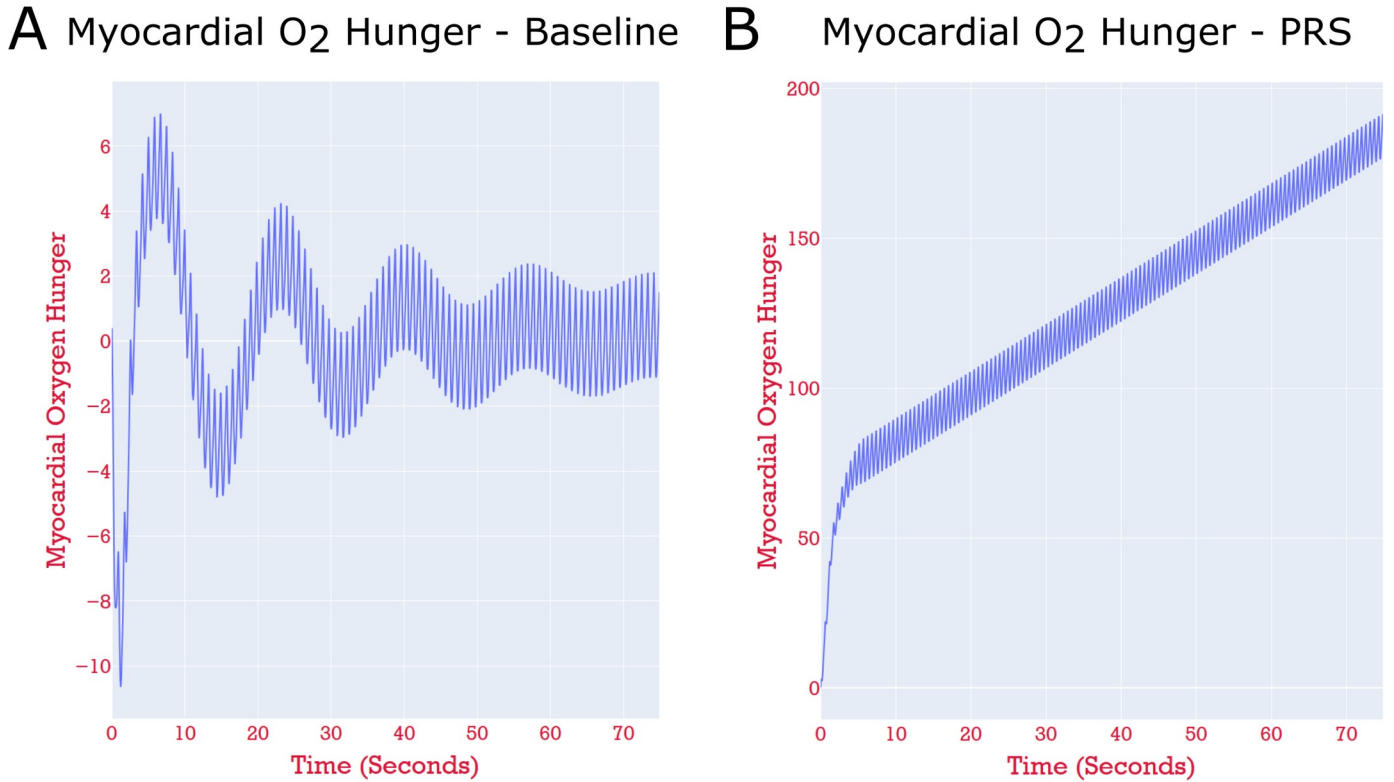
Fig 5. A complex, two-domain patient-specific 3D model with full closed-loop boundary condition system created in CRIMSON. This was used in a study of cardiovascular complications during liver transplant [44]. Image used under CC BY 4.0 license.

<https://doi.org/10.1371/journal.pcbi.1008881.g005>

tractable for most readers. Both the 0D and 3D simulations are presented in detail elsewhere [44]. The closed-loop boundary condition circuit specified in this problem is visualized in Fig 7; guidance on how to perform this visualisation is also provided. Because running this model in 3D requires around 20,000 CPU core-hours, the input files provided in File 1 at <https://doi.org/10.7302/4vdx-ek59> for Cases C1 and C2 are configured to run in a simplified 0D mode. Readers with access to HPCs can follow the instructions to enable full 3D simulation.

### Further applications of CRIMSON in the literature

Another use-case for CRIMSON is preoperative surgical planning. For example, CRIMSON was applied to a case of open surgical correction of a Fontan procedure, in a patient suffering from unilateral pulmonary arteriovenous malformations. After constructing a model reproducing the patient’s pre-operative haemodynamics, two post-operative models were created representing two possible surgical repairs. One was seen to be superior in terms of balancing hepatic venous flow between the pulmonary arteries, and the multidisciplinary surgical team

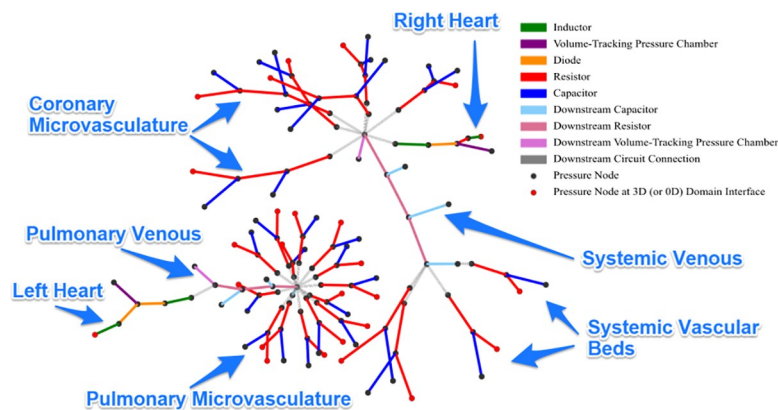


**Fig 6. The results of running a pure zero-dimensional simulation of the model shown in Fig 5.** Note the differing y-axis scales. **Panel A:** Case C1—the time evolution of the myocardial hunger in the left anterior descending (LAD) perfusion territory of the myocardium, in the patient-specific simulation under baseline resting conditions. The blood supply is such that the hunger is kept close to zero for the entire simulation. **Panel B:** Case C2—the model with parameters perturbed to simulate PRS. The hunger grows continually, indicating that the blood supply to this perfusion territory is insufficient, and thus, myocardial ischaemia.

<https://doi.org/10.1371/journal.pcbi.1008881.g006>

chose this option for the patient [45]. Post-operative clinical data agreed with the simulated predictions.

In another study, CRIMSON was used to evaluate the suitability of different designs of a medical device—vascular endografts for aortic arch repair—in terms of their impact on patient



**Fig 7. Visualisation of the Netlist boundary condition circuits shown in Fig 5.** There are two connected components; during simulation, these two are joined by the two connected components of the 3D domain—the pulmonary arteries and the systemic arteries. Pressure nodes (between components) are shown as dots. Red dots indicate points of interface with the 3D domain. Lines indicate LPN components and are coloured according to type (resistor, capacitor, etc.).

<https://doi.org/10.1371/journal.pcbi.1008881.g007>

haemodynamics. Evaluating design alternatives *in vivo* is neither practical nor ethical, so computational modelling is the only option. The results indicated that the different devices offered significantly different hemodynamic metrics in terms of blood shearing and blood flow to the brain [46].

CRIMSON has been used to systematically examine anatomical risk factors in type-B aortic dissection (TBAD) using idealized models. CRIMSON models permitted isolated changes to specific parameters present in TBAD, including vessel curvature, tear size and shape, false lumen location, and treatment state. The results indicated which combinations lead to unfavourable haemodynamic states, and agreed with clinical studies [47]. A similar study employing a patient-specific model demonstrated that dissection imposes a significant additional workload on the heart, and that the number of additional communicating tears between the true and false lumen has a substantial impact on volumetric flow and peak pressure in the true lumen [48].

In a study reproducing haemodynamics in a patient with a Blalock-Taussig (BT) shunt, CRIMSON was used to determine that the BT shunt's diameter had reduced by 22% since implantation due to neointimal growth. Because the shunt size was below the MRI resolution, CRIMSON was critical in determining this diameter change [49].

A patient-specific model involving three virtual interventions—open repair, conformable endografting, and fenestrated endografting—to repair a kinked ascending aortic graft in a patient-specific model was studied using CRIMSON [50]. The results included evaluation of a metric of platelet aggregation potential, PLAP [51,52], and indicated that surgical repair or the fenestrated endograft would most profoundly reduce thrombus formation risk.

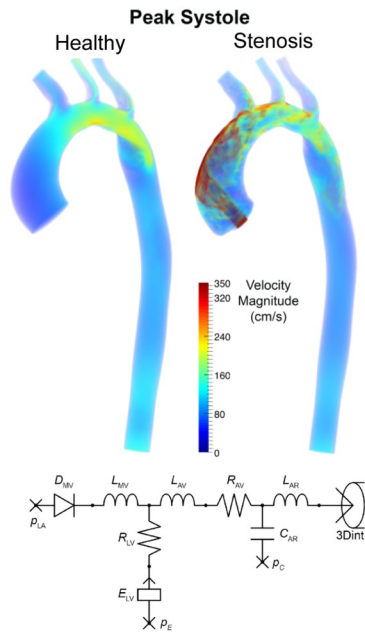
One of the key novel aspects of CRIMSON is its tools for studying cardiovascular control systems and auto-regulations. These include the role of the baroreflex system in adjusting blood pressure after a change in body posture [53], and how the coronary flow autoregulation system responds to changes in exercise intensity in a patient-specific model [13].

## Availability and future directions

We invite readers to download installers for Microsoft Windows from [www.crimson.software](http://www.crimson.software) for both CRIMSON GUI and CRIMSON Flowsolver. These are currently time-limited binaries, but users can compile their own versions from the provided source code. Alternatively, users can follow the documentation on the CRIMSON website, describing how to interface the GUI with their own fluid solver. The source code for the GUI is available on GitHub under GNU Affero General Public License v3.0 (<https://github.com/carthurs/CRIMSONGUI>), and for the Flowsolver under the GNU General Public License v3.0 (<https://github.com/carthurs/CRIMSONFlowsolver>). Both licenses further include the Commons Clause. Archived versions of the source code are available under DOIs [10.7302/679b-dw96](https://doi.org/10.7302/679b-dw96) and [10.7302/tbx5-7c07](https://doi.org/10.7302/tbx5-7c07) for the GUI and the Flowsolver, respectively. For the avoidance of confusion, a separate CRIMSON GUI called CRIMSON Pro is also available in the CRIMSONGUI GitHub repository. This provides additional interfaces for some advanced features (but does not provide those features themselves). CRIMSON Pro is not the subject of this manuscript.

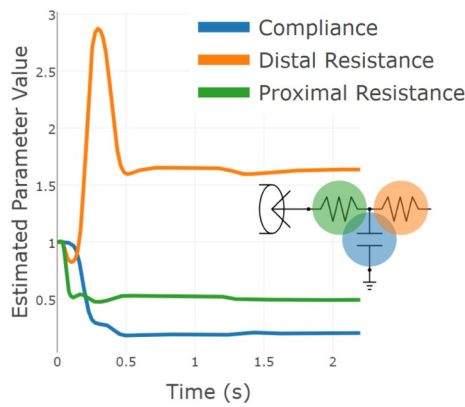
Fig 8 depicts several additional CRIMSON capabilities already available or currently under development. Panel A shows results obtained with a pre-existing custom aortic valve stenosis model, carried out using CRIMSON NEBCT and Control Systems Framework capabilities [54], which enabled straightforward implementation without requiring expert knowledge of the Flowsolver. Panel B shows results obtained using Lagrangian particle tracking analysis of haemodynamics in a “Zone 0” thoracic endovascular aortic repair [46]. The colour scale indicates PLAP (platelet activation potential), a metric used to quantify the shear rate experienced

### A Custom Valve Models

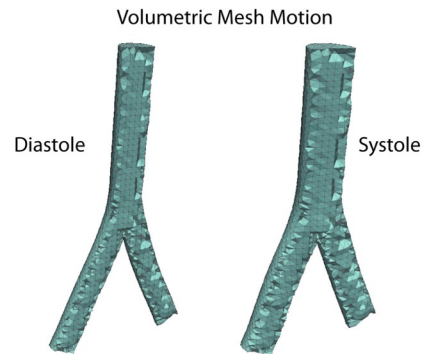


### B Lagrangian Particle Tracking

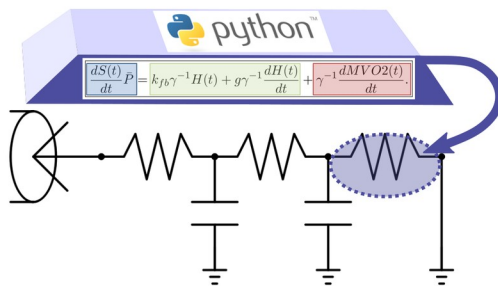
### C Kalman Filtering



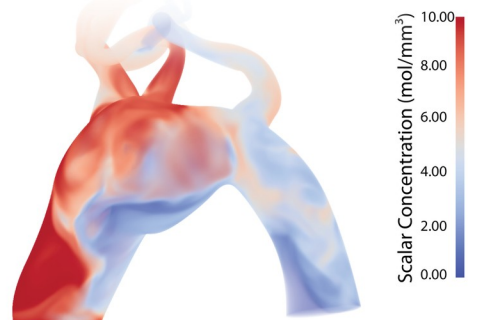
### D ALE Fluid-structure Interaction



### E Physiological Control Systems



### F Scalar Reaction & Transport



**Fig 8. Examples of advanced and upcoming CRIMSON features.** **A:** Custom aortic valve model implementation using the NEBCT and Python Control Systems Framework. **B:** Example of a Lagrangian particle tracking study of massless particle transport in the blood stream [46]. **C:** Time history of parameter convergence during Kalman filter data assimilation [12]. **D:** Arbitrary Lagrangian-Eulerian deformation of a vascular mesh between diastole and systole [17]. **E:** Support for arbitrary run-time Python code for modelling cardiovascular control systems and changes of state, by controlling component parameters during simulations [16]. **F:** Scalar reaction-advection-diffusion (RAD) problem with one species transported in the blood stream. Run-time specification of arbitrary reactions between tens of species is under current development [15].

<https://doi.org/10.1371/journal.pcbi.1008881.g008>

by massless particles advected by the flow. This index has been linked to risk of thrombus formation [50]. Panel C shows the estimation of BC model parameters during the application of a sequential Reduced-Order Unscented Kalman Filter (ROUKF). The filter is used to determine suitable parameter values in agreement with time-resolved provided patient data, such as localised pressure or flow measurements [55,30,12]. Panel D displays results of a volumetric mesh motion algorithm, which has been developed as a component of an Arbitrary Lagrangian Eulerian (ALE) based FSI framework utilizing a rotation-free nonlinear shell formulation [17]. This method will enable simulation of large vascular strains and motions, incorporating biologically-relevant constitutive models for the vessel wall (for example, nonlinear strain energy functions for soft tissue, including four fibre models), and involves a novel mathematical formulation that retains displacement-only degrees of freedom for the shell (i.e. no elements across the wall thickness) description for vessel wall, thereby ensuring minimal computational costs compared to other ALE methods. Panel E illustrates the application of a custom ODE model, composed in a few tens of lines of Python by the user, which at run-time can adjust the parameters of an attached BC component. Panel F illustrates the solution of RAD transport problems coupled with flow. Here, a single scalar with zero initial condition is considered, an advective flux is imposed at the inlet with a zero total flux condition on all walls and outlets. This allows simulation of transport under high Peclet numbers and interaction of scalar species, including proteins, drugs, contrast agents and heat, and will enable study of phenomena such as thrombogenesis [15]. CRIMSON will support arbitrary numbers of reacting species, with reaction terms specified by the end-user at run-time in Python.

A further planned enhancement is the integration of a one-dimensional (1D) incompressible Navier-Stokes solver, which enables simulating extensive vascular networks over many cardiac cycles in almost real-time, while retaining features such as pulse wave propagation phenomena [56]. This 1D flow solver could be used in combination with graph theoretic approaches to accurately and efficiently model blood flow in disease conditions [57].

Currently, there are no good models in existence for turbulent cardiovascular flow. There is a need to develop models of transitional and mild turbulence, which are the conditions occurring in some cardiovascular flows. The validation of such models would in principle have to be made using appropriate *in vitro* models, due to the challenges in acquiring *in vivo* data on transitional / complex flows.

Furthermore, there is a pressing need to develop workflows for 3D anatomical model creation that rely on machine learning methods. This would greatly reduce the current time required to create anatomical models, which currently require the use of external commercial packages (such as Mimics) or labour-intensive workflows (CRIMSON and SimVascular).

We believe that CRIMSON is an enticing prospect for CH researchers and look forward to working with the community as we continue its development. We aim to be responsive to community input, via our support mailing list, and look forward to receiving feedback from our ever-expanding user-base on how CRIMSON can best support their work.

## Guidance for getting started with CRIMSON

The version of CRIMSON GUI and CRIMSON Flowsolver current at the time of publication of this article is provided in File 1 at <https://doi.org/10.7302/4vdx-ek59>. However, a newer version may be available from [www.crimson.software](http://www.crimson.software), as suggested in each of the cases below. To install CRIMSON, first run the CRIMSON-2019.11.01 exe installer, then run the CRIMSON\_Flowsolver exe installer. The order is important. If you use the default installation paths in both, then the GUI will be able to find and run the flowsolver. If you install the GUI to a non-default location, you must change the Flowsolver's install path in the same manner.

If you wish to build from source, follow the instructions in the readme.md file in the CRIMSON Flowsolver repository <https://github.com/carthurs/CRIMSONFlowsolver>; and analogously for the GUI repository <https://github.com/carthurs/CRIMSONGUI>.

## Running example Case A

The tutorial pdf for this example case—located in File 1 at <https://doi.org/10.7302/4vdx-ek59>—will guide you through the complete process of creating a model, running the simulation and examining the results.

## File manifest

1. **ExampleCaseA\ExampleA-Meshing-BC-Simulation-visualisation-refinement.pdf**—the instructions for running this case
2. **ExampleCaseA\DataFiles\Solid\_model.stl**—the surface triangular mesh of the vascular geometry we will use
3. **ExampleCaseA\DataFiles\steady.flow**—specification file for a constant steady inflow at the aortic valve; for use with the Prescribed Velocities inflow boundary condition

## Steps to Follow:

1. Download and install CRIMSON from [www.crimson.software](http://www.crimson.software), or from the installers in this data package. If using the installers in this data package, install *CRIMSON-2019.11.01-win64-trial.exe* first (the GUI), followed by *CRIMSON\_Flowsolver\_1.4.2\_2019.11.01-installpath.exe* (the Flowsolver). If you adjust the installation path for the GUI, ensure you make the corresponding change for the flowsolver, so that the GUI can find the Flowsolver.
2. Open the document *ExampleA-Meshing-BC-Simulation-visualisation-refinement.pdf* from the ExampleCaseA available in File 1 at <https://doi.org/10.7302/4vdx-ek59>. Follow the instructions therein.

## Running example Case B

The supplementary pdf for this example—available in File 1 at <https://doi.org/10.7302/4vdx-ek59>—will guide you through segmenting and imposing a patient-specific aortic inflow velocity profile from a provide PC-MRI dataset.

## File manifest

1. **ExampleCaseB\DataFiles\Patient-PCMRI.mitk**—the CRIMSON scene save file to start from, containing a geometric model to impose the PC-MRI boundary condition upon
2. **ExampleCaseB\DataFiles\PC-MRI\_Images\Phase**—a folder containing the PC-MRI phase images throughout one cardiac cycle
3. **ExampleCaseB\DataFiles\PC-MRI\_Images\Magnitude**—a folder containing the PC-MRI magnitude images throughout one cardiac cycle
4. **ExampleCaseB\ExampleB-PC-MRI-specificVelocityProfiles.pdf**—the instructions you should follow in order to impose a PC-MRI-derived inflow boundary condition on a model of the aorta

## Steps to Follow:

1. Download and install CRIMSON from [www.crimson.software](http://www.crimson.software), or from the installers in this data package. If using the installers in this data package, install *CRIMSON-2019.11.01-win64-trial.exe* first (the GUI), followed by *CRIMSON\_Flowsolver\_1.4.2\_2019.11.01-installpath.exe* (the Flowsolver). If you adjust the installation path for the GUI, ensure you make the corresponding change for the flowsolver, so that the GUI can find the Flowsolver.
2. Open the document *ExampleB-PC-MRI-specificVelocityProfiles.pdf* from the ExampleCaseB folder available in File 1 at <https://doi.org/10.7302/4vdx-ek59>. Follow the instructions therein.

## Running example Case C

The supplementary pdf for this case—available in File 1 at <https://doi.org/10.7302/4vdx-ek59> - will guide you through running a simulation of a patient under rest conditions, and then of the same patient under post-liver-transplant conditions. Many parameters of the haemodynamics could be examined and compared between the two, but in this case we choose to examine the oxygen supply/demand balance in the myocardium under these two states, and will observe that coronary perfusion is insufficient in the post-operative simulation.

## File manifest

1. **ExampleCaseC\Mitk\_Scenes** for the Resting and PRS conditions—these are the CRIMSON saved state files
2. **ExampleCaseC\AdditionalInputFiles**—Files prepared for this specific case (such as custom CRIMSON Control Systems Toolbox Python scripts, and loop-closing circuits, which have to be made manually outside the GUI). These are for combining with the CRIMSON GUI output for each case to create complete simulation input files
3. **ExampleCaseC\ReadyToUseFiles**—folders containing simulation input files in a state which can be run immediately. These files are configured to run in purely zero dimensions, and essentially consist of a combination of the CRIMSON Solver Setup output from the MITK scenes, and the AdditionalInputFiles. You will only need these if you do not wish to

generate your own simulation files, as is one of the two options described in [ExampleC-simulating-Rest-vs-PRS-in-Alagille-patients.pdf](#)

4. **ExampleCaseC\ExampleC-simulating-Rest-vs-PRS-in-Alagille-patients.pdf**—the instructions you should follow in order to create and run the two simulation states for the patient.

### Steps to Follow:

1. Download and install CRIMSON from [www.crimson.software](http://www.crimson.software)
2. Open the document [ExampleC-simulating-Rest-vs-PRS-in-Alagille-patients.pdf](#) from the Example Case C folder available in File 1 at <https://doi.org/10.7302/4vdx-ek59>. Follow the instructions therein.

### Libraries and frameworks used in CRIMSON

The following pre-existing libraries are integral to the CRIMSON software. See [S1 Fig](#) for an overview of the structure.

The Medical Imaging Interaction Toolkit (MITK) [10] was chosen as the foundation for the CRIMSON GUI due to its established position as an open-source package for development of interactive medical image processing and ongoing development by the German Cancer Research Center. It provides functionality for working with medical image datasets, including multiple orthogonal and 3D views of the image volume, data storage and undo/redo functionality, and image processing and segmentation algorithms [58] via the Insight Toolkit (ITK), Visualization Toolkit (VTK) and Qt [59]. Additionally, it includes a Python interface for easy prototyping and MITK release 2016.11 integrates VMTK (The Vascular Modeling ToolKit) [60], a reference package for vascular segmentation and automatic centreline extraction. MITK is well-regarded, providing a foundation for other research packages including by the more general GIMIAS [61].

MITK itself is built upon the widely used visualisation tools of VTK, adopts segmentation tools from ITK, and is highly customizable due to the modular, cross-platform MITK Blue-Berry framework and CTK (*the Common Toolkit*). MITK uses the Qt application framework to manage events and connect user interface (UI) elements, the MITK DataStorage system to manage state data (The Qt Company, 2016 [59]). Qt, together with the Python Qt wrappers, allows advanced users to add, remove or modify the widgets presented by the GUI when writing custom Solver Setups (see later in this section). The functionality for lofting or sweeping to create the solid model from the segmented vessel contours, blending the intersecting vessels into a full geometry, and important filleting and Boolean operations is provided by OpenCascade [19].

Volume meshing is provided by the TetGen (Si, 2015 [21]) library; optionally, this can be replaced with the high-quality commercial Simmetrix MeshSim library [62].

*OpenCascade* [19] is an open-source C++ class library designed for rapid production of complex Computer Aided Design (CAD) applications. This class library makes it possible to define analytical models whose surfaces are parametrized by functions such as non-uniform rational basis splines (NURBS).

*Verdandi* is an open-source generic C++ library for data assimilation, developed by INRIA (Chapelle, et al., 2013 [30]). Data assimilation is a process that enables combining different sources of information to estimate the state of a dynamical system. By extension, this approach can be used to estimate parameters of a computational model if time-resolved data is available.



Verdandi was designed so that computational models implemented in C++, Fortran, or Python could be made use of Verdandi via either a C++ or a Python interface.

*QSapecNG* (Manetti, et al., 2012 [33]) is an open-source program based on Qt for drawing linear analogue circuits and performing analysis. We modified it to create a tool for designing custom circuits for BC models (the CRIMSON Netlist Editor Boundary Condition Toolbox), and outputting them in a format readable by the CRIMSON Flowsolver.

Recently a Docker container of the CRIMSON flowsolver has been made available in the American Heart Association's Precision Medicine Platform, which runs on Amazon Web Services. S1 Fig summarizes the main libraries and packages contributing to CRIMSON.

A permanently archived versions of the source code for CRIMSON GUI and CRIMSON Flowsolver, and of binary installers and example case studies described in this manuscript, have been made available using the University of Michigan's Deep Blue service [63–65].

## Supporting information

**S1 Fig. Summary of the main libraries and contributing packages in CRIMSON, and how they historically relate to one another.**

(PNG)

## Author Contributions

**Conceptualization:** Christopher J. Arthurs, Rostislav Khlebnikov, Marija Marčan, Alberto Gomez, Desmond Dillon-Murphy, Kritika Iyer, Paolo Di Achille, Hamid Gharahi, Kevin D. Lau, Vasilina Filonova, Miquel Aguirre, Nitesh Nama, Nan Xiao, Seungik Baek, Krishna Garikipati, Onkar Sahni, David Nordsletten, C. Alberto Figueroa.

**Data curation:** Christopher J. Arthurs.

**Funding acquisition:** C. Alberto Figueroa.

**Investigation:** Christopher J. Arthurs, Rostislav Khlebnikov, Marija Marčan, Alberto Gomez, Desmond Dillon-Murphy, Federica Cuomo, Miguel Silva Vieira, Jonas Schollenberger, Christopher Tossas-Betancourt, Kritika Iyer, Sara Hopper, Pouya Youssefi, Alia Noorani, Sabrina Ben Ahmed, Foeke J. H. Nauta, Theodorus M. J. van Bakel, Yunus Ahmed, Petrus A. J. van Bakel, Jonathan Mynard, Paolo Di Achille, Hamid Gharahi, Kevin D. Lau, Vasilina Filonova, Miquel Aguirre, Nitesh Nama, Nan Xiao, C. Alberto Figueroa.

**Methodology:** Christopher J. Arthurs, Alberto Gomez, Jonas Schollenberger, Sabrina R. Lynch, Paolo Di Achille, Kevin D. Lau, Vasilina Filonova, Miquel Aguirre, Nitesh Nama, Nan Xiao, C. Alberto Figueroa.

**Project administration:** C. Alberto Figueroa.

**Software:** Christopher J. Arthurs, Rostislav Khlebnikov, Alex Melville, Marija Marčan, Alberto Gomez, Desmond Dillon-Murphy, Federica Cuomo, Jonas Schollenberger, Sabrina R. Lynch, Christopher Tossas-Betancourt, Kritika Iyer, Sara Hopper, Elizabeth Livingston, Alia Noorani, Sabrina Ben Ahmed, Jonathan Mynard, Paolo Di Achille, Hamid Gharahi, Kevin D. Lau, Vasilina Filonova, Miquel Aguirre, Nitesh Nama, Nan Xiao, David Nordsletten, C. Alberto Figueroa.

**Supervision:** Christopher J. Arthurs, Seungik Baek, Krishna Garikipati, Onkar Sahni, David Nordsletten, C. Alberto Figueroa.

**Validation:** Christopher J. Arthurs, Rostislav Khlebnikov, Alex Melville, Marija Marčan, Desmond Dillon-Murphy, Federica Cuomo, Jonas Schollenberger, Sabrina R. Lynch, Kritika Iyer, Alia Noorani, Jonathan Mynard, Paolo Di Achille, Kevin D. Lau, Vasilina Filonova, Miquel Aguirre.

**Visualization:** Christopher J. Arthurs, Rostislav Khlebnikov, Alberto Gomez, Desmond Dillon-Murphy, Sabrina R. Lynch, Christopher Tossas-Betancourt, Paolo Di Achille, Kevin D. Lau.

**Writing – original draft:** Christopher J. Arthurs, C. Alberto Figueroa.

**Writing – review & editing:** Christopher J. Arthurs, Rostislav Khlebnikov, Alex Melville, Marija Marčan, Alberto Gomez, Desmond Dillon-Murphy, Federica Cuomo, Miguel Silva Vieira, Jonas Schollenberger, Sabrina R. Lynch, Christopher Tossas-Betancourt, Kritika Iyer, Sara Hopper, Elizabeth Livingston, Pouya Youssefi, Alia Noorani, Sabrina Ben Ahmed, Foeke J. H. Nauta, Theodorus M. J. van Bakel, Yunus Ahmed, Petrus A. J. van Bakel, Jonathan Mynard, Paolo Di Achille, Hamid Gharahi, Kevin D. Lau, Vasilina Filonova, Miquel Aguirre, Nitesh Nama, Nan Xiao, Seungik Baek, Krishna Garikipati, Onkar Sahni, David Nordsletten, C. Alberto Figueroa.

## References

1. Taylor CA, Fonte T A, Min JK. Computational Fluid Dynamics Applied to Cardiac Computed Tomography for Noninvasive Quantification of Fractional Flow Reserve. *JACC*. 2013; 61(22), 2233–2241. <https://doi.org/10.1016/j.jacc.2012.11.083> PMID: 23562923
2. Zhou M, Sahni O, Kim HJ, Figueroa CA, Taylor CA, Shephard MS, et al. Cardiovascular flow simulation at extreme scale. *Comput Mech*. 2010; 46, 71–82. <https://doi.org/10.1007/s00466-009-0450-z>
3. Updegrove A, Wilson NM, Mergow J, Lan H, Marsden AL, Shadden SC. SimVascular: An Open Source Pipeline for Cardiovascular Simulation. *Ann Biomed Eng*. 2016; 45(3), 525–541. Retrieved from SimTK: <https://simtk.org/projects/simvascular>. <https://doi.org/10.1007/s10439-016-1762-8> PMID: 27933407
4. Sengupta D, Kahn AM, Burns JC, Sankaran S, Shadden S. Image-based modeling of haemodynamics and coronary artery aneurysms caused by Kawasaki disease. *Biomech Model Mechanobiol*. 2012; 11(6), 915–932. <https://doi.org/10.1007/s10237-011-0361-8> PMID: 22120599
5. Marsden AL, Bernstein AJ, Reddy VM, Shadden S, Spilker R, Taylor CA, et al. Evaluation of a Novel Y-Shaped Extracardiac Fontan Baffle Using Computational Fluid Dynamics. *J Thorac Cardiovasc Surg*. 2009 137(2), 394–403. <https://doi.org/10.1016/j.jtcvs.2008.06.043> PMID: 19185159
6. Marsden AL. Simulation based planning of surgical interventions in pediatric cardiology. *Phys Fluids*. 2013; 25, 101303. <https://doi.org/10.1063/1.4825031> PMID: 24255590
7. Les AS, Shadden SC, Figueroa CA, Park JM, Tedesco MM, Herfkens RJ, et al. Quantification of haemodynamics in abdominal aortic aneurysms during rest and exercise using magnetic resonance imaging and computational fluid dynamics. *Ann Biomed Eng*. 2010; 38(4), 1288–1313. <https://doi.org/10.1007/s10439-010-9949-x> PMID: 20143263
8. Coogan JS, Humphrey JD, Figueroa CA. Computational Simulations of Hemodynamic Changes within Thoracic, Coronary, and Cerebral Arteries Following Early Wall Remodeling in Response to Distal Aortic Coarctation. *Biomech Model Mechanobiol*. 2013; Vol. 12(1), pp. 79–93. <https://doi.org/10.1007/s10237-012-0383-x> PMID: 22415052
9. Khlebnikov R, Figueroa CA. CRIMSON: Towards a Software Environment for Patient-Specific Blood Flow Simulation for Diagnosis and Treatment. *Clinical Image-Based Procedures: Translational Research in Medical Imaging*. 2016; 10–18. [https://doi.org/10.1007/978-3-319-31808-0\\_2](https://doi.org/10.1007/978-3-319-31808-0_2)
10. Wolf I, Vetter M, Wegner I, Boettger T, Nolden M, Schoebinger M, et al. The Medical Imaging Interaction Toolkit. *Med Image Anal*. 2005; 9(6), 594–604. <https://doi.org/10.1016/j.media.2005.04.005> PMID: 15896995
11. Gomez A, Marčan M, Arthurs CJ, Wright R, Youssefi P, Jahangiri M, et al. Optimal B-spline Mapping of Flow Imaging Data for Imposing Patient-specific Velocity Profiles in Computational Haemodynamics. *IEEE Trans Biomed Eng*. 2018; 66(7), 1872–1883. <https://doi.org/10.1109/TBME.2018.2880606> PMID: 30561336

12. Arthurs CJ, Xiao N, Moireau P, Schaeffter T and Figueroa CA. A Flexible Framework for Sequential Estimation of Model Parameters in Computational Haemodynamics. *Adv Model Simul Eng Sci*. 2020; 7, 48. <https://doi.org/10.1186/s40323-020-00186-x> PMID: 33282681
13. Arthurs CJ, Lau KD, Asrress KN, Redwood SR, Figueroa CA. A Mathematical Model of Coronary Blood Flow Control: Simulation of Patient-Specific Three-Dimensional haemodynamics during Exercise. *Am J Physiol Heart Circ Physiol* 2016; 310(9), H1242–H1258. <https://doi.org/10.1152/ajpheart.00517.2015> PMID: 26945076
14. Hughes TJ, Wells GN. Conservation properties for the Galerkin and stabilised forms of the advection-diffusion and incompressible Navier-Stokes Equations. *Comput Methods Appl Mech Eng*. 2005; 194 (9–11), 1141–1159. <https://doi.org/10.1016/j.cma.2004.06.034>
15. Lynch SR, Nama N, Xu Z, Arthurs CJ, Sahni O and Figueroa CA. Numerical Considerations for Advection-Diffusion Problems in Cardiovascular Haemodynamics. *Int J Numer Method Biomed Eng*. 2020; e3378. <https://doi.org/10.1002/cnm.3378> PMID: 32573092
16. Arthurs CJ, Figueroa CA. Integration of an Electrophysiologically Driven Heart Model into Three-Dimensional haemodynamics Simulation Using the CRIMSON Control Systems Framework. *Computational Biomechanics for Medicine* 2016; 155–166. Springer. [https://doi.org/10.1007/978-3-319-28329-6\\_14](https://doi.org/10.1007/978-3-319-28329-6_14)
17. Nama N, Aguirre M, Humphrey JD, Figueroa CA, A Nonlinear Rotation-free Shell Formulation with Prestressing for Vascular Biomechanics. *Scientific Reports* 2020; 10, 17528. <https://doi.org/10.1038/s41598-020-74277-5> PMID: 33067508
18. Wang KC, Dutton RW, Taylor CA. Improving geometric model construction for blood flow modeling: Geometric image segmentation and image-based model construction for computational haemodynamics. *IEEE Eng Med Biol Mag*. 1999; 18(6), 33–39. <https://doi.org/10.1109/51.805142> PMID: 10576070
19. Wilson N, Wang KD, Taylor CA. A Software Framework for Creating Patient Specific Geometric Models from Medical Imaging Data for Simulation Based Medical Planning of Vascular Surgery. *MICCAI*. 2001; pp. 449–456. [https://doi.org/10.1007/3-540-45468-3\\_54](https://doi.org/10.1007/3-540-45468-3_54)
20. Open Cascade SAS. (2017, May 9). Open Cascade. Retrieved from <https://www.opencascade.com/> (2016, July 28).
21. Si H. TetGen, a Delaunay-Based Quality Tetrahedral Mesh Generator. *ACM Trans Math Softw*. 2015; 41:2 11:1–11:36. <https://doi.org/10.1145/2629697>
22. Figueroa CA, Vignon-Clementel IE, Jansen KE, Hughes TJ, Taylor CA. A coupled momentum method for modeling blood flow in three-dimensional deformable arteries. *Comput Methods Appl Mech Eng*. 2006; 195(41–43), 5685–5706. <https://doi.org/10.1016/j.cma.2005.11.011>
23. Cuomo F, Roccabianca S, Dillon-Murphy D, Xiao N, Humphrey JD, Figueroa CA. Effects of age-associated regional changes in aortic stiffness on human hemodynamics revealed by computational modeling. *PLoS One*. 2017. <https://doi.org/10.1371/journal.pone.0173177> PMID: 28253335
24. Vignon-Clementel IE, Figueroa CA, Jansen KE, Taylor CA. Outflow boundary conditions for three-dimensional finite element modeling of blood flow and pressure in arteries. *Comput Methods Appl Mech Eng*. 2006; 195, 3776–3796. <https://doi.org/10.1016/j.cma.2005.04.014>
25. Whiting CH, Jansen KE. A stabilized finite element method for the incompressible Navier-Stokes equations using a hierarchical basis. *Int J Numer Methods Fluids*. 2001; 35(1), 93–116. [https://doi.org/10.1002/1097-0363\(20010115\)35:1<93:AID-FLD85>3.0.CO;2-G](https://doi.org/10.1002/1097-0363(20010115)35:1<93:AID-FLD85>3.0.CO;2-G)
26. Jansen KE, Whiting CH, Hulbert GM. A generalized-alpha method for integrating the filtered Navier-Stokes equations with a stabilised finite element method. *Comput Methods Appl Mech Eng*. 2000 190 (3–4), 305–319. [https://doi.org/10.1016/S0045-7825\(00\)00203-6](https://doi.org/10.1016/S0045-7825(00)00203-6)
27. Sahni O, Müller J, Jansen KE, Shephard MS, Taylor CA. Efficient anisotropic adaptive discretization of the cardiovascular system. *Comp Method Appl Mech Engng*. 2006; 195 (41–43), 5634–5655. <https://doi.org/10.1016/j.cma.2005.10.018>
28. Paraview. Retrieved from <http://www.paraview.org/> (2016, July 28).
29. Sahni O, Carothers CD, Shephard MS, Jansen KE. Strong scaling analysis of parallel, unstructured, implicit solver and the influence of the operating system interface. *Scientific Programming*. 2009; 17(3), 261–274. <https://doi.org/10.3233/SPR-2009-0287>
30. Chapelle D, Fragu M, Mallet V, Moireau P. Fundamental principles of data assimilation underlying the Verdandi library: applications to biophysical model personalization within euHeart. *Med Biol Eng Comput*. 2013; 51, 1221–1233. <https://doi.org/10.1007/s11517-012-0969-6> PMID: 23132524
31. Alphabet Inc. Google Test. Retrieved from Google Test: <https://github.com/google/googletest> (2016, July 29).
32. Le Tallec P, Mouro J. Fluid structure interaction with large structural displacements. *Comput Methods Appl Mech Eng*. 2001; 190.24–25, 3039–3067. [https://doi.org/10.1016/S0045-7825\(00\)00381-9](https://doi.org/10.1016/S0045-7825(00)00381-9)

33. Manetti S, Luchetta A, Caini M, Rossetto S. QSapecNG Project. 2012; retrieved Jul 7, 2016, from <http://qsapecng.sourceforge.net/>.
34. Lee J, Cookson A, Roy I, Kerfoot E, Asner L, Viguera G, et al. Multiphysics computational modeling in CHeart. *SIAM J Sci Comput.* 2016; 38(3), C150–C178. <https://doi.org/10.1137/15M1014097>
35. Spazzapan M, Sastry P, Dunning J, Nordsletten D, de Vecchi A. The Use of Biophysical Flow Models in the Surgical Management of Patients Affected by Chronic Thromboembolic Pulmonary Hypertension. *Front Physiol.* 2018; 9:223. <https://doi.org/10.3389/fphys.2018.00223> PMID: 29593574
36. Youssefi P, Gomez A, Arthurs CJ, Sharma R, Jahangiri M, Figueroa CA. Impact of Patient-Specific Inflow Velocity Profile on Haemodynamics of the Thoracic Aorta. *J Biomech Eng.* 2018; 140(1), 011002. <https://doi.org/10.1115/1.4037857> PMID: 28890987
37. Youssefi P, Gomez A, He T, Anderson L, Bunce N, Sharma R. Patient-specific computational fluid dynamics—assessment of aortic haemodynamics in a spectrum of aortic valve pathologies. *J Thorac Cardiovasc Surg.* 2017; 153(1), 8–20.e3. <https://doi.org/10.1016/j.jtcvs.2016.09.040> PMID: 27847162
38. Youssefi P, Sharma R, Figueroa CA, Jahangiri M. Functional assessment of thoracic aortic aneurysms—the future of risk prediction? *Br Med Bull.* 2016, 121(1), 61–71. <https://doi.org/10.1093/bmb/dw049> PMID: 27989994
39. Alagille D, Estrada A, Hadchouel M, Gautler M, Odièvre M, Dommergues JP. Syndromic paucity of interlobular bile ducts (Alagille syndrome or arteriohepatic dysplasia): Review of 80 cases. *J Pediatr.* 1987. [https://doi.org/10.1016/s0022-3476\(87\)80153-1](https://doi.org/10.1016/s0022-3476(87)80153-1) PMID: 3806290
40. Emerick KM, Rand EB, Goldmuntz E, Krantz ID, Spinner NB, Piccoli DA. Features of Alagille syndrome in 92 patients: Frequency and relation to prognosis. *Hepatology.* 1999. <https://doi.org/10.1002/hep.510290331> PMID: 10051485
41. Turnpenny PD, Ellard S. Alagille syndrome: pathogenesis, diagnosis and management. *Eur J Hum Genet.* 2012. <https://doi.org/10.1038/ejhg.2011.181> PMID: 21934706
42. Kamath BM, Baker A, Houwen R, Todorova L, Kerkar N. Systematic Review: the Epidemiology, Natural History and Burden of Alagille Syndrome. *J Pediatr Gastroenterol Nutr.* 2018. <https://doi.org/10.1097/MPG.0000000000001958> PMID: 29543694
43. Aggarwal S, Kang Y, Freeman JA, Fortunato FL, Pinsky MR. Postreperfusion syndrome: cardiovascular collapse following hepatic reperfusion during liver transplantation. *Transpl Proc.* 1987; 19(4 Suppl 3):54–55. PMID: 3303534
44. Silva Vieira M, Arthurs CJ, Hussain T, Razavi R, Figueroa CA. Patient-specific modelling of right coronary circulation vulnerability post-liver transplant in Alagille’s syndrome. *PLOS One.* 2018. <https://doi.org/10.1371/journal.pone.0205829> PMID: 30408044
45. van Bakel TMJ, Lau KD, Hirsch-Romano J, Trimarchi S, Dorfman AL, Figueroa CA. Patient-Specific Modeling of Haemodynamics: Supporting Surgical Planning in a Fontan Circulation Correction *J Cardiovasc Transl Res.* 2018. <https://doi.org/10.1007/s12265-017-9781-x> PMID: 29313269
46. van Bakel TMJ, Arthurs CJ, Herwaarden JA, Moll FL, Eagle KA, Patel HJ, et al. Hemodynamic Performance of Different Endograft Designs for Zone 0 Aortic Arch Repair: A Computational Study. *Eur J Cardiothorac Surg.* 2018. <https://doi.org/10.1093/ejcts/ezy068> PMID: 29554234
47. Ben Ahmed S, Dillon-Murphy D, Figueroa CA. Computational Study of Anatomical Risk Factors in Idealized Models of Type B Aortic Dissection. *Eur J Vasc Endovasc Surg.* 2016 52(6), 736–745. <https://doi.org/10.1016/j.ejvs.2016.07.025> PMID: 27561609
48. Dillon-Murphy D, Noorani A, Nordsletten D, Figueroa CA. Multi-modality image-based computational analysis of haemodynamics in aortic dissection. *Biomech Model Mechanobiol.* 2016; 15(4), 857–876. <https://doi.org/10.1007/s10237-015-0729-2> PMID: 26416312
49. Arthurs CJ, Agarwal P, John AV, Dorfman AL, Grifka RG, Figueroa CA. Reproducing Patient-Specific Haemodynamics in the Blalock–Taussig Circulation Using a Flexible Multi-Domain Simulation Framework: Applications for Optimal Shunt Design. *Front Pediatr.* 2017; 5 78. <https://doi.org/10.3389/fped.2017.00078> PMID: 28491863
50. Nauta FJ, Lau KD, Arthurs CJ, Eagle KA, Williams DM, Trimarchi S, et al. Computational Fluid Dynamics and Aortic Thrombus Formation Following Thoracic Endovascular Aortic Repair. *Ann Thorac Surg.* 2017; 103(6), 1914–1921. <https://doi.org/10.1016/j.athoracsur.2016.09.067> PMID: 28063468
51. Di Achille P, Tellides G, Figueroa CA and Humphrey JD. A haemodynamic predictor of intraluminal thrombus formation in abdominal aortic aneurysms. *Proc R Soc A.* 2014; 470: 20140163 <https://doi.org/10.1098/rspa.2014.0163>
52. Shadden SD and Hendabadi S. Potential fluid mechanic pathways of platelet activation. *Biomech Model Mechanobiol.* 2013; 12, 467–74 <https://doi.org/10.1007/s10237-012-0417-4> PMID: 22782543

53. Lau KD, Figueroa CA. Simulation of short-term pressure regulation during the tilt test in a coupled 3D–0D closed-loop model of the circulation. *Biomech Model Mechanobiol.* 2015; 14(4), 915–929. <https://doi.org/10.1007/s10237-014-0645-x> PMID: 25567754
54. Mynard JP, Davidson MR, Penny DJ, Smolich JJ. A simple, versatile valve model for use in lumped parameter and one-dimensional cardiovascular models. *Int J Numer Method Biomed Eng.* 2012; 28(6–7):626–41. <https://doi.org/10.1002/cnm.1466> PMID: 25364842
55. Moireau P, Bertoglio C, Xiao N, Figueroa CA, Taylor CA, Chapelle D, et al. Sequential identification of boundary support parameters in a fluid-structure vascular model using patient image data. *Biomech Model Mechanobiol.* 2013; 12(3), 475–496. <https://doi.org/10.1007/s10237-012-0418-3> PMID: 22802016
56. Xiao N, Alastruey J, Figueroa CA. A systematic comparison between 1-D and 3-D haemodynamics in compliant arterial models. *Int J Numer Method Biomed Eng.* 2014; 30(2), 204–31. <https://doi.org/10.1002/cnm.2598> PMID: 24115509
57. Banerjee R, Sagiyama K, Teichert GH, Garikipati K. A graph theoretic framework for representation, exploration and analysis on computed states of physical systems. *Comput Methods Appl Mech Eng.* 2019; 351, 501–530. <https://doi.org/10.1016/j.cma.2019.03.053>
58. Nolden M, Zelzer S, Seitel A, Wald D, Müller M, Franz AM, et al. The Medical Imaging Interaction Toolkit: challenges and advances 10 years of open-source development. *Int J Comput Assist Radiol Surg.* 2013; 8(4), 607–620. <https://doi.org/10.1007/s11548-013-0840-8> PMID: 23588509
59. The Qt Company. (2016, July 29). Retrieved from Qt: <https://www.qt.io/>.
60. Antiga L, Piccinelli M, Botti L, Ene-Iordache B, Remuzzi A, Steinman D. An Image-based Modeling Framework for Patient-specific Computational Haemodynamics. *Med Biol Eng Comput.* 2008; 46: 1097–1112. <https://doi.org/10.1007/s11517-008-0420-1> PMID: 19002516
61. Larrabide I, Omedas P, Martelli Y, Planes X, Nieber M, Moya JA, et al. GIMIAS: An Open Source Framework for Efficient Development of Research Tools and Clinical Prototypes. *Functional Imaging and Modeling of the Heart.* 2009 (pp. 417–426). Berlin: Springer. <https://doi.org/10.4321/s1130-01082009000200003> PMID: 19335045
62. Simmetrix. Retrieved from MeshSim: <http://www.simmetrix.com/products/MeshSim/MeshSim.html> (2016, July 28).
63. Arthurs C. J., Khlebnikov R., Melville A., Marcan M., Gomez A., Dillon-Murphy D. et al. CRIMSON open source project—Graphical User Interface (GUI) Source Code and Application Examples for PLOS Computational Biology. University of Michigan Deep Blue Archive. <https://doi.org/10.7302/679b-dw96>
64. Arthurs C. J., Khlebnikov R., Melville A., Marcan M., Gomez A., Dillon-Murphy D. et al. CRIMSON open source project—Flow Solver Source Code for PLOS Computational Biology. University of Michigan Deep Blue Archive. <https://doi.org/10.7302/tbx5-7c07>
65. Arthurs, C. J., Khlebnikov, R., Melville, A., Marcan, M., Gomez, A., Dillon-Murphy, D. et al. CRIMSON open source project—CRIMSON open source project—Supplementary Material for PLOS Computational Biology Publication. University of Michigan Deep Blue Archive. <https://doi.org/10.7302/4vdx-ek59>.