Chaste: Cancer, Heart and Soft Tissue Environment

Fergus R Cooper[1], Ruth E Baker[1], Miguel O Bernabeu[2], Rafel Bordas[3], Louise Bowler[3], Alfonso Bueno-Orovio[3], Helen M Byrne[1], Valentina Carapella[4], Louie Cardone-Noott[3], Jonathan Cooper[5], Sara Dutta[3], Benjamin D Evans[6, 7], Alexander G Fletcher[8, 9], James A Grogan[1], Wenxian Guo[10], Daniel G Harvey[3], Maurice Hendrix[11,12], David Kay[3], Jochen Kursawe[13], Philip K Maini[1], Beth McMillan[3], Gary R Mirams[11], James M Osborne[14], Pras Pathmanathan[15], Joe M Pitt-Francis[3], Martin Robinson[3], Blanca Rodriguez[3], Raymond J Spiteri[10], and David J Gavaghan[3]

1 Wolfson Centre for Mathematical Biology, Mathematical Institute, University of Oxford, Oxford, UK **2** Centre for Medical Informatics, Usher Institute, University of Edinburgh, Edinburgh, United Kingdom **3** Department of Computer Science, University of Oxford, Oxford, UK **4** Division of Cardiovascular Medicine, Radcliffe Department of Medicine, University of Oxford, Oxford, UK **5** Research IT Services, University College London, London, UK **6** Centre for Biomedical Modelling and Analysis, Living Systems Institute, University of Exeter, Exeter, UK **7** School of Psychological Science, University of Bristol, Bristol, UK **8** School of Mathematics & Statistics, University of Sheffield, Sheffield, UK **9** Bateson Centre, University of Sheffield, Sheffield, UK **10** Department of Computer Science, University of Saskatchewan, Canada **11** Centre for Mathematical Medicine & Biology, School of Mathematical Sciences, University of Nottingham, Nottingham, UK **12** Digital Research Service, University of Nottingham, Nottingham, UK **13** Faculty of Biology, Medicine and Health, University of Manchester, Manchester, UK **14** School of Mathematics and Statistics, University of Melbourne, Victoria, Australia **15** Office of Science and Engineering Laboratories (OSEL), Center for Devices and Radiological Health (CDRH), U.S. Food and Drug Administration (FDA), Silver Spring, MD 20993, USA

Summary

Chaste (Cancer, Heart And Soft Tissue Environment) is an open source simulation package for the numerical solution of mathematical models arising in physiology and biology.

To date, Chaste development has been driven primarily by applications that include continuum modelling of cardiac electrophysiology ('Cardiac Chaste'), discrete cell-based modelling of soft tissues ('Cell-based Chaste'), and modelling of ventilation in lungs ('Lung Chaste'). Cardiac Chaste addresses the need for a high-performance, generic, and verified simulation framework for cardiac electrophysiology that is freely available to the scientific community. Cardiac chaste provides a software package capable of realistic heart simulations that is efficient, rigorously tested, and runs on HPC platforms. Cell-based Chaste addresses the need for efficient and verified implementations of cell-based modelling frameworks, providing a set of extensible tools for simulating biological tissues. Computational modelling, along with live imaging techniques, plays an important role in understanding the processes of tissue growth and repair. A wide range of cell-based modelling frameworks have been developed that have each been successfully applied in a range of biological applications. Cell-based Chaste includes implementations of the cellular automaton model, the cellular Potts model, cell-centre models with cell representations as overlapping spheres or Voronoi tessellations, and the vertex model. Lung Chaste addresses the need for a novel, generic and efficient lung modelling software package that is both tested and verified. It aims to couple biophysically-detailed models of

airway mechanics with organ-scale ventilation models in a package that is freely available to the scientific community.

Chaste is designed to be modular and extensible, providing libraries for common scientific computing infrastructure such as linear algebra operations, finite element meshes, and ordinary and partial differential equation solvers. This infrastructure is used by libraries for specific applications, such as continuum mechanics, cardiac models, and cell-based models. The software engineering techniques used to develop Chaste are intended to ensure code quality, re-usability and reliability. Primary applications of the software include cardiac and respiratory physiology, cancer and developmental biology.

The software

Chaste is available on GitHub https://github.com/Chaste/Chaste, and the current stable release is version 2019.1. Please see the Readme.md file on the Github repository for links to the Chaste wiki and install guides.

Previous publications about Chaste have detailed the rationale for, and design principles behind, Chaste (Pitt-Francis et al., 2009), as well as the main application areas of Chaste up to 2013 (Mirams et al., 2013).

Chaste places an emphasis on reproducibility and verification and, as such, extensive automated testing is used to ensure software quality and reliability. A series of test suites must all pass before any commit is considered a release-candidate. Most testing is performed on Long Term Support (LTS) versions of Ubuntu Linux, with unit tests additionally being run on macOS.

Testing includes compilation of all libraries with GCC, Clang and Intel C++ compilers; extensive unit testing; performance profiling to identify any slowdowns over time; memory testing with valgrind; verification of code coverage; and running unit tests with different combinations of dependencies to ensure portability. The output of these tests is available at https://chaste.cs.ox.ac.uk/buildbot/.

Since 2013, Chaste has substantially changed to modernise its infrastructure and to enable new science. In terms of infrastructure, Chaste now uses a modern CMake build system, the C++14 language standard, and makes extensive use of BuildBot for continuous integration. In terms of science, Lung Chaste is entirely new and allows the use of Chaste in a new scientific domain. In Cardiac Chaste, we can now create algebraic Jacobians for CellML ODE systems, which can improve speed of simulation for cardiac action potential and tissue simulations (Cooper, Spiteri, & Mirams, 2015), and metadata annotations of CellML files have replaced manual specification of variables in configuration files. Cell-based Chaste has been overhauled to improve flexibility. Changes include hierarchies of simulation modifiers, information writers, cell-cycle models, subcellular reaction network models, and numerical methods that allow new customisation points in almost every area of all cell-based simulations. In addition, simulation output has been standardised to use VTK, a standard and powerful visualisation framework, and some cell-centre simulations now run in parallel using MPI.

Comparison with other software

Chaste provides substantial common infrastructure enabling a wide range of applications across multiple disciplines. Common elements include meshing, solving differential equations, input/output and continuum mechanics, and these form a platform for Cardiac, Cell-based and Lung Chaste.

A key goal of Chaste is to enable the implementation of many different modelling frameworks. This not only allows a user to select the most appropriate tool for their research but, importantly, enables the comparison of different modelling frameworks to better understand the

benefits and drawbacks of each (Osborne, Fletcher, Pitt-Francis, Maini, & Gavaghan, 2017). This is an explicit design goal of Chaste, which focusses on the flexibility of implementing multiple models rather than (for example) building a graphical user interface. See Table 1 for a comparison of alternatives to Chaste in specific domains, with all other software tools implementing a single modelling framework.

Software	Open Source	GUI	CA	CP	PM	VT	VM
Chaste	x		x	x	x	x	x
CompuCell3D	x	x		x			
Morpheus	x	x		x			
EPISIM		x			x		
CellSys		x			x		
PhysiCell	x				x		
Biocellion					x		
VirtualLeaf	x	x					x
EmbryoMaker	x	x			x		

Table 1: A comparison of software tools for cell-based modelling. GUI: graphical user interface. CA: cellular automata. CP: cellular Potts. PM: particle model, a cell-centre model. VT: Voronoi tessellation, a cell-centre model. VM: vertex model. References: CompuCell3D (Swat et al., 2012), Morpheus (Starruß, Back, Brusch, & Deutsch, 2014), EPISIM (Sutterlin, Kolb, Dickhaus, Jager, & Grabe, 2013), CellSys (Hoehme & Drasdo, 2010), PhysiCell (Ghaffarizadeh, Heiland, Friedman, Mumenthaler, & Macklin, 2018), Biocellion (Kang, Kahan, McDermott, Flann, & Shmulevich, 2014), VirtualLeaf (Merks, Guravage, Inzé, & Beemster, 2011), EmbryoMaker (Marin-Riera, Brun-Usan, Zimm, Välikangas, & Salazar-Ciudad, 2015).

Installation

Installation of Chaste has been greatly simplified through the development of a Docker image https://github.com/chaste/chaste-docker. Docker is a lightweight, open-source virtualisation technology for running encapsulated applications ('containers') on all major operating systems at near-native speed. This enables Chaste (including all dependencies, environment settings, convenience scripts and the latest precompiled release) to be downloaded and installed with just a single command. Isolating Chaste within a container also means that its dependencies and those installed on the user's host system can coexist without interference or version conflicts.

In addition to simplifying the set-up and execution of Chaste, importantly this also enhances its reproducibility by providing a homogeneous computational environment regardless of the underlying operating system and hardware. Not only is the Chaste source code version-controlled, but so too are the dependencies, configuration settings and environment variables used to build and run it. This means that collaborators and reviewers can easily and consistently reproduce results (to within machine precision) on any platform while developers can seamlessly migrate and scale-up their simulations from a laptop to a workstation or HPC cluster.

Example usage

Chaste has tutorials to walk users through basic functionality for each application area. Tutorial examples are bundled for each specific release version, and examples for this release are available at https://chaste.cs.ox.ac.uk/chaste/tutorials/release_2019.1.

Tutorials take the form of C++ header files that each define 'tests' in the Chaste testing infrastructure. These tests must be compiled and run to produce an output, which can be visualised using ParaView.

In the following sections we showcase a specific tutorial for each of cardiac, cell-based, and lung Chaste, with minimal commands necessary to reproduce the output shown.

Cardiac example

Here we demonstrate how to run and visualise a three-dimensional monodomain cardiac simulation. This follows the tutorial `TestMonodomain3dRabbitHeartTutorial` which simulates the result of an electrical stimulus being applied to a realistic rabbit heart geometry. Assuming that

1. Chaste has been installed on Ubuntu Linux (or is running within a Docker container),
2. the Chaste source code exists at $CHASTE_SOURCE_DIR,
3. the environment variable $CHASTE_TEST_OUTPUT is set to a valid directory,

a minimal set of commands to build and run the tutorial is as follows:

```
mkdir build && cd build
cmake $CHASTE_SOURCE_DIR
make TestMonodomain3dRabbitHeartTutorial
ctest -R TestMonodomain3dRabbitHeartTutorial
```

This will produce output in the following directory:

```
$CHASTE_TEST_OUTPUT/Monodomain3dRabbitHeart
```

To view the results evolving over time as an animation in ParaView it is necessary to post-process the results with the following command:

```
cd $CHASTE_TEST_OUTPUT/Monodomain3dRabbitHeart/vtk_output
python $CHASTE_SOURCE_DIR/python/utils/AddVtuTimeAnnotations.py \
    results.vtu annotated_results.vtu
```

To visualise the output, open the file `annotated_results.vtu` in ParaView, and select to colour by `V` (voltage).
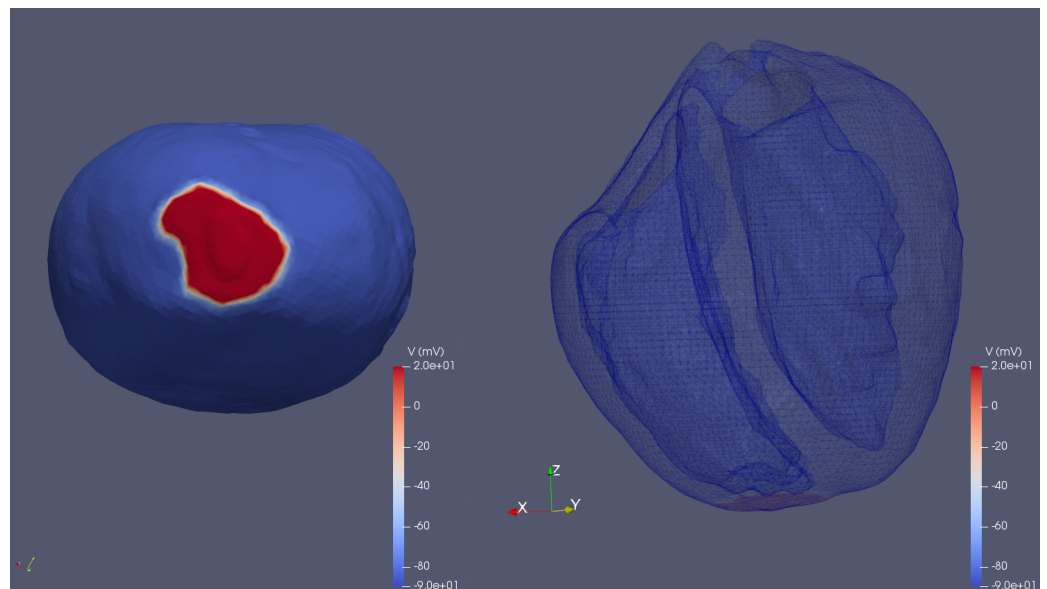
Figure 1: Trans-membrane voltage on the rabbit heart mesh at the end of the simulation. As viewed on the surface from the apex of the heart (left) and on a wireframe showing the ventricular cavities (right).

Cell-based example

Here we demonstrate how to run and visualise a cell sorting simulation using Chaste's vertex model implementation. This follows the tutorial `TestRunningDifferentialAdhesionSimulationsTutorial`. Assuming that

1. Chaste has been installed on Ubuntu Linux (or is running within a Docker container),
2. the Chaste source code exists at `$CHASTE_SOURCE_DIR`,
3. the environment variable `$CHASTE_TEST_OUTPUT` is set to a valid directory,

a minimal set of commands to build and run the tutorial is as follows:

```
mkdir build && cd build
cmake $CHASTE_SOURCE_DIR
make TestRunningDifferentialAdhesionSimulationsTutorial
ctest -R TestRunningDifferentialAdhesionSimulationsTutorial
```

This will produce output in the following directory:

```
$CHASTE_TEST_OUTPUT/TestVertexBasedDifferentialAdhesionSimulation
```

To visualise the simulation, open the file `results.pvd` in ParaView, choose to colour by 'Cell types', and display 'Surface With Edges'.
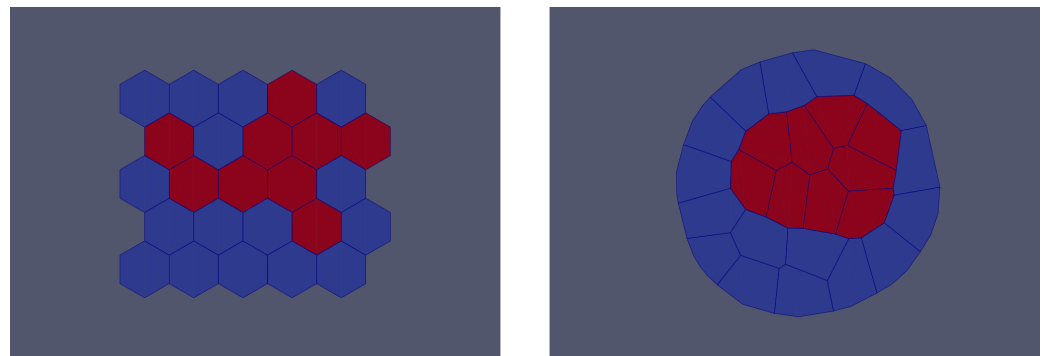
Figure 2: The initial configuration of cells (left), and the final configuration of cells after sorting has occurred (right).

Lung example

Here we demonstrate how to run and visualise the lung airway generation tutorial. This follows the tutorial `TestAirwayGenerationTutorial` which statistically generates lung airways given initial geometry segmented from a CT scan. Assuming that

1. Chaste has been installed on Ubuntu Linux (or is running within a Docker container),
2. the Chaste source code exists at `$CHASTE_SOURCE_DIR`,

3. the environment variable $CHASTE_TEST_OUTPUT is set to a valid directory,

a minimal set of commands to build and run the tutorial is as follows:

```
mkdir build && cd build
cmake $CHASTE_SOURCE_DIR
make TestAirwayGenerationTutorial
ctest -R TestAirwayGenerationTutorial
```

This will produce output in the following directory:

```
$CHASTE_TEST_OUTPUT/TestAirwayGenerationTutorial
```

To visualise the generated airway geometry, open the file `example_complete_conducting_airway.vtu` in ParaView. Application of an 'Extract Surface' filter followed by a 'Tube' filter allows the centreline and radius information to be viewed as a series of tubes.
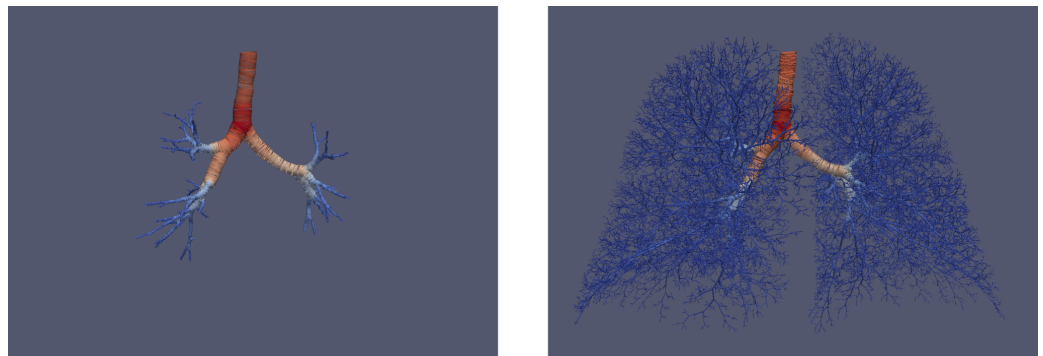
Figure 3: The initial geometry of major airways segmented from a CT scan (left), and an example of a complete generated airway tree (right).

Recent publications enabled by Chaste

Since our last publication on Chaste, over 70 peer-reviewed publications have been enabled in the areas described below.

Publications using Cardiac Chaste have included scientific studies relating to: basic mechanisms of cardiac electrophysiology and tissue structure in healthy and diseased settings; the sources and consequences of inter-subject electrophysiological variability; predicting the effects of drugs on cardiac activity, including safety assessment and the development of associated web-based tools (Cooper, Scharm, & Mirams, 2016). Other studies enabled by Cardiac Chaste have advanced technical methodologies for parameter identifiability and inference, model selection and uncertainty quantification in cardiac electrophysiology models (Johnstone et al., 2016); and for the verification and efficient numerical simulation of cardiac models (Green, Bohn, & Spiteri, 2019). The continuum-mechanics solvers in Chaste have been used for studies of cardiac electromechanics (Carapella et al., 2014).

The Cardiac Chaste code has also been applied to gastric electrophysiology, in particular focusing on the interstitial cells of Cajal network (Sathar, Trew, & Cheng, 2015).

Publications enabled by Cell-based Chaste have focused on: the cellular mechanisms and dynamics of intestinal homeostasis and carcinogenesis; the mechanisms underlying vascular tumour growth and response to therapy in the Microvessel Chaste project (Grogan et al.,

2017); the biomechanical characterization of epithelial tissue development and wound healing; the organisation and proliferation of stem and pluripotent cells in development; the spread of sexually-transmitted infections; vascular remodelling (Osborne & Bernabeu, 2018); cell-based model calibration and parameterisation (Kursawe, Baker, & Fletcher, 2018); and their efficient numerical solution (Cooper, Baker, & Fletcher, 2017).

Papers on Lung Chaste describe its use for patient-specific airway tree generation and flow modelling (Bordas et al., 2015).

Acknowledgements

References

Bordas, R., Lefevre, C., Veeckmans, B., Pitt-Francis, J., Fetita, C., Brightling, C. E., Kay, D., et al. (2015). Development and analysis of patient-based complete conducting airways models. *PlOS ONE*, *10*(12), e0144105. doi:10.1371/journal.pone.0144105

Carapella, V., Bordas, R., Pathmanathan, P., Lohezic, M., Schneider, J. E., Kohl, P., Burrage, K., et al. (2014). Quantitative study of the effect of tissue microstructure on contraction in a computational model of rat left ventricle. *PLOS ONE*, *9*(4), e92792. doi:10.1371/journal.pone.0092792

Cooper, F. R., Baker, R. E., & Fletcher, A. G. (2017). Numerical analysis of the Immersed Boundary Method for cell-based simulation. *SIAM J. Sci. Comput.*, *39*(5), 943–967. doi:10.1137/16M1092246

Cooper, J., Scharm, M., & Mirams, G. R. (2016). The cardiac electrophysiology web lab. *Biophys. J.*, *110*(2), 292–300. doi:10.1016/j.bpj.2015.12.012

Cooper, J., Spiteri, R. J., & Mirams, G. R. (2015). Cellular cardiac electrophysiology modeling with Chaste and CellML. *Front. Physiol.*, *5*, 511. doi:10.3389/fphys.2014.00511

Ghaffarizadeh, A., Heiland, R., Friedman, S. H., Mumenthaler, S. M., & Macklin, P. (2018). PhysiCell: an open source physics-based cell simulator for 3-D multicellular systems. *PLOS Comput. Biol.*, *14*(2), e1005991. doi:10.1371/journal.pcbi.1005991

Green, K. R., Bohn, T. A., & Spiteri, R. J. (2019). Direct function evaluation versus lookup tables: When to use which? *SIAM J. Sci. Comput.*, *41*(3), C194–C218. doi:10.1137/18M1201421

Grogan, J. A., Connor, A. J., Markelc, B., Muschel, R. J., Maini, P. K., Byrne, H. M., & Pitt-Francis, J. M. (2017). Microvessel chaste: an open library for spatial modeling of vascularized tissues. *Biophys. J.*, *112*(9), 1767–1772. doi:10.1016/j.bpj.2017.03.036

Hoehme, S., & Drasdo, D. (2010). A cell-based simulation software for multi-cellular systems. *Bioinformatics*, *26*(20), 2641–2. doi:10.1093/bioinformatics/btq437

Johnstone, R. H., Chang, E. T. Y., Bardenet, R., De Boer, T. P., Gavaghan, D. J., Pathmanathan, P., Clayton, R. H., et al. (2016). Uncertainty and variability in models of the cardiac action potential: can we build trustworthy models? *J. Mol. Cell. Cardiol.*, *96*, 49–62. doi:10.1016/j.yjmcc.2015.11.018

Kang, S., Kahan, S., McDermott, J., Flann, N., & Shmulevich, I. (2014). Biocellion: accelerating computer simulation of multicellular biological system models. *Bioinformatics*, *30*(21), 3101–3108. doi:10.1093/bioinformatics/btu498

Kursawe, J., Baker, R. E., & Fletcher, A. G. (2018). Approximate Bayesian computation reveals the importance of repeated measurements for parameterising cell-based models of growing tissues. *J. Theor. Biol.*, *443*, 66–81. doi:10.1016/j.jtbi.2018.01.020

Marin-Riera, M., Brun-Usan, M., Zimm, R., Välikangas, T., & Salazar-Ciudad, I. (2015). Computational modeling of development by epithelia, mesenchyme and their interactions: a unified model. *Bioinformatics*, *32*(2). doi:10.1093/bioinformatics/btv527

Merks, R. M. H., Guravage, M., Inzé, D., & Beemster, G. T. S. (2011). VirtualLeaf: an open-source framework for cell-based modeling of plant tissue growth and development. *Plant Physiol.*, *155*(2), 656–66. doi:10.1104/pp.110.167619

Mirams, G. R., Arthurs, C. J., Bernabeu, M. O., Bordas, R., Cooper, J., Corrias, A., Davit, Y., et al. (2013). Chaste: an open source C++ library for computational physiology and biology. *PLOS Comput. Biol.*, *9*, e1002970. doi:10.1371/journal.pcbi.1002970

Osborne, J. M., & Bernabeu, M. O. (2018). A fully discrete open source framework for the simulation of vascular remodelling. In *40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (pp. 4552–4555). doi:10.1109/EMBC.2018.8513223

Osborne, J. M., Fletcher, A. G., Pitt-Francis, J. M., Maini, P. K., & Gavaghan, D. J. (2017). Comparing individual-based approaches to modelling the self-organization of multicellular tissues. *PLOS Comput. Biol.*, *13*(2), 1–34. doi:10.1371/journal.pcbi.1005387

Pitt-Francis, J., Pathmanathan, P., Bernabeu, M. O., Bordas, R., Cooper, J., Fletcher, A. G., Mirams, G. R., et al. (2009). Chaste: a test-driven approach to software development for biological modelling. *Comput. Phys. Commun.*, *180*(12), 2452–2471. doi:10.1016/j.cpc.2009.07.019

Sathar, S., Trew, M. L., & Cheng, L. K. (2015). Tissue specific simulations of interstitial cells of cajal networks using unstructured meshes. In *37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (pp. 8062–8065). doi:10.1109/EMBC.2015.7320264

Starruß, J., Back, W. de, Brusch, L., & Deutsch, A. (2014). Morpheus: a user-friendly modeling environment for multiscale and multicellular systems biology. *Bioinformatics*, *30*(9), 1331–1332. doi:10.1093/bioinformatics/btt772

Sutterlin, T., Kolb, C., Dickhaus, H., Jager, D., & Grabe, N. (2013). Bridging the scales: semantic integration of quantitative SBML in graphical multi-cellular models and simulations with EPISIM and COPASI. *Bioinformatics*, *29*(2), 223–229. doi:10.1093/bioinformatics/bts659

Swat, M. H., Thomas, G. L., Belmonte, J. M., Shirinifard, A., Hmeljak, D., & Glazier, J. A. (2012). Multi-scale modeling of tissues using CompuCell3D. *Methods Cell Biol.*, *110*, 325–366. doi:10.1016/B978-0-12-388403-9.00013-8