

DuneCopasi: A multi-compartment reaction-diffusion simulator for systems biology

Santiago Ospina De Los Ríos  ^{1,2}, Peter Bastian¹, Liam Keegan  ³, Sven Sahle  ⁴, Dylan Vermoortele  ⁶, and Lilija Wehling  ^{4,5}

¹ Interdisciplinary Center for Scientific Computing (IWR), Heidelberg University, Germany ² Heidelberg Graduate School of Mathematical and Computational Methods for the Sciences (HGS MathComp), Heidelberg University, Germany ³ Scientific Software Center (SSC), Heidelberg University, Germany ⁴ BioQuant, Centre for Organismal Studies (COS), Heidelberg University, Germany ⁵ Institute of Pathology, University Hospital Heidelberg, Germany ⁶ Cardiovascular Imaging and Dynamics, KU Leuven, Belgium

DOI: [10.21105/joss.06836](https://doi.org/10.21105/joss.06836)

Software

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

Editor: [Prashant Jha](#) 

Reviewers:

- [@ayush9pandey](#)
- [@ajbaird](#)

Submitted: 25 April 2024

Published: 09 December 2024

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

Summary

DuneCopasi is a C++ library designed to simulate how chemical reactions and diffusion processes occur in space, which is crucial for understanding many biological systems. It allows users to model these processes in both simple and complex environments, using grids that represent one, two, or three dimensions. Optimized for modern computers, DuneCopasi can be used as a standalone command-line application, accessed through tools like Docker or a web-based terminal, as a C++ library, or integrated into a graphical interface. One such interface is the **Spatial Model Editor** (SME) (Keegan et al., 2023), which helps users create and manipulate biological reaction models. The project was initiated to provide the numerical foundation for a spatial modelling tool for biochemical reaction networks, complementing the **COnplex PAthway SImlator** (COPASI) software (Hoops et al., 2006). This development resulted from a collaboration between the COPASI team and the **Distributed and Unified Numerics Environment** (DUNE) team (Bastian et al., 2021).

Background

In the context of cell biology, computational modelling has become an essential technique for understanding and discovering biological processes. By comparing model simulations with experimental data, researchers can set up models, validate them, and test hypotheses about these processes. Traditionally, most biological systems studied this way have been spatially homogeneous. That is, the data are typically time-resolved concentrations or quantities of biochemical species, modelled using **Ordinary Differential Equations** (ODEs).

However, recent advances in live-cell imaging technology have made more detailed spatio-temporal data available, leading to a growing need for models that capture not only the time dynamics but also the spatial distribution of these biochemical species. This has prompted a shift toward spatially resolved models.

Model Problem

A natural extension of spatially homogeneous ODE models is to incorporate spatio-temporal dynamics by formulating the problem as a system of **Partial Differential Equations** (PDEs) across multiple compartments. In such models, each compartment's PDE represents the reaction-diffusion processes of biochemical species in a specific physical domain (e.g., the

cytosol), while boundary conditions between compartments (e.g., membrane fluxes) govern the interactions between them.

Specifically, our program solves the mass balance equation for the species u_{ik} in the k -th compartment Ω_k for every $k \in K$ and $i \in N_k$. Each mass balance equation is given by

$$\begin{aligned} \partial_t(\phi_{ik}u_{ik}) &= \nabla \cdot \mathbf{j}_{ik} + \mathcal{R}_{ik}(\mathbf{u}) && \text{in } \Omega_k \subset \mathbb{R}^d, \\ u_{ik} &= u_{ik}^{(0)} && \text{on } \Gamma_k^D \subseteq \partial\Omega_k, \quad \text{with } \mathbf{j}_{ik} := \sum_{j \in N_k} D_{ijk} \nabla u_{jk}. \\ \mathbf{j}_{ik} \cdot \mathbf{n}_k &= \sum_{l \in T_k} \mathcal{T}_{ikl}(\mathbf{u}) && \text{on } \partial\Omega_k \setminus \Gamma_k^D, \end{aligned}$$

Here, $\mathbf{u}_k := (u_{1k}, \dots, u_{\dim(N_k)k})$ represents the vector of species concentrations in compartment k , while the full vector of species concentrations across all compartments is denoted as $\mathbf{u} := (\mathbf{u}_1, \dots, \mathbf{u}_{\dim(K)})$. The unit outer normal vector on the boundary $\partial\Omega_k$ is represented by \mathbf{n}_k , and the set of neighbouring compartments to k is given by $T_k := \{l \in K : \partial\Omega_k \cap \overline{\Omega}_l \neq \emptyset\}$, indicating the compartments that share a boundary with Ω_k .

The reaction operator $\mathcal{R}_{ik}(\mathbf{u})$ governs the local reaction dynamics within Ω_k , while the storage terms ϕ_{ik} account for species accumulation in the compartment. Cross-diffusion terms D_{ijk} describe how species diffuse between different species within the same compartment. The non-linear transmission conditions $\mathcal{T}_{ikl}(\mathbf{u})$ represent the outflow of species u_{ik} from compartment Ω_k , where the outflow can either move to a neighbouring compartment l (if $l \neq k$) or exit the system. Likewise, Dirichlet boundary conditions $u_{ik}^{(0)}$ are imposed on the subset Γ_k^D of the boundary $\partial\Omega_k$, specifying the fixed concentrations of species on that portion of the boundary.

The parameters governing these equations are fully configurable at run-time, either through the command line or via a configuration file. By allowing full control over these parameters, users can adapt the software to simulate a wide variety of biochemical processes, from simple reactions in homogeneous environments to complex, multi-compartmental systems with intricate boundary conditions and interactions.

Capabilities

Many features of DuneCopasi have been designed and developed with the specific case of systems biology in mind. Thus, a substantial effort has been made to have a library that is interoperable with systems biology data assets and requirements needed by their practitioners. Among others, include:

- a single executable configurable at run-time,
- a run-time mathematical expression parser that understands the Systems Biology Markup Language SBML ([Hucka et al., 2003](#)) specification,
- the input of custom grid data and image files in the TIFF format for initial spatial concentrations and other parameters,
- a powerful, yet simple, boundary/transmission condition specification for each compartment to account for generic trans-membrane fluxes,
- a (non-linear) cross-diffusion specification that allows any species to cross-diffuse into other mass balance equations,
- a specification to compare results with user-defined objective functions,
- an in-place function interpolator that reduces the computational cost of evaluating common expensive reaction operations, and
- an embedded random field generator ([Kempf et al., 2023](#)) to represent statistical spatial variations on the domain.

Furthermore, DuneCopasi is a stand-alone multi-compartment reaction-diffusion solver that may easily be used for many other fields of research and engineering fitting our model problem (e.g. Figure 1).

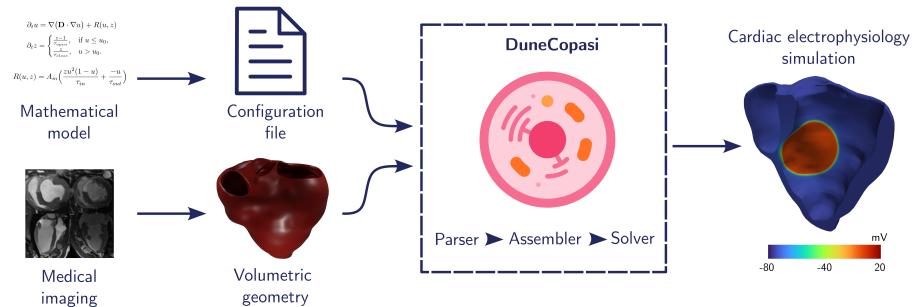


Figure 1: DuneCopasi is a highly flexible simulator that is run-time configurable using dedicated configuration files. DuneCopasi can autonomously interpret mathematical equations, assemble the computational problems and solve these equations based on the configuration file. Here we illustrate the integration of DuneCopasi within a computational electrophysiology pipeline allowing to solve the electrophysiological mono-domain equations with Mitchell-Schaefer cell model using a medical imaging derived from biventricular geometry.

Statement of Need

Finite element frameworks like DUNE (Bastian et al., 2008), Deal.II (Arndt et al., 2023), Necktar++ (Cantwell et al., 2015), or FreeFem++ (Hecht, 2012) are too generic by design and don't address the specific requirements of the computational modelling practices in systems biology out of the box. In particular, most finite element frameworks don't consider the multi-compartment systems in their design, resulting, if at all possible, in inefficient simulations or obscure tricks to force this feature. Here, we extended DUNE-PDELab (Bastian et al., 2010; Müthing, 2015) with efficient data structures especially tailored for this task.

In the space of systems biology, two well-known software packages also provide similar features to DuneCopasi, namely Morpheus (Starruß et al., 2014) and VCell (Schaff et al., 1997). Morpheus is a computational tool for multi-cellular systems which follows a cellular automata design to set rules of interactions between individual cells. It focuses on modelling approaches like cellular Potts models combined with gradients modelled by PDEs, rather than multi-compartment PDE models. Furthermore, due to its design, it falls short for moderate and big PDE computations since explicit solvers are severely limited by the required time steps. On the other hand, VCell provides a fully implicit finite volume solver on structured grids that can very well manage the multi-compartment case in addition to membrane unknowns. In comparison, our solution aims to resolve the geometry and the transmission conditions directly in the weak formulation of the problem while also being designed to provide implicit and monolithic solvers as well as tailored preconditioners for the underlying linear solvers.

Research and Use Cases

The principal purpose of our project has been to bridge the gap between systems biology and scientific computing by providing researchers with an accessible and reproducible spatial simulator. An illustration of this is the study depicted in Figure 2 (Wehling et al., 2022), where simulations with the SME (Keegan et al., 2023) aided in assessing and better understanding the mechanisms of the YAP and TAZ proteins that regulate cell proliferation in liver cells. Further examples that fit our model problem in this context have been presented by others

(Eliaš & Clairambault, 2014). Additionally, DuneCopasi stands as a versatile tool capable of accommodating diverse computational needs beyond its initial focus on systems biology. For example, the general purpose nature allows for seamless integration into electrophysiology simulations, without requiring any modifications. The emerging concept of identifying patients based on personalized cardiac electrophysiology simulations (Arevalo et al., 2016) underscores the demand for easy-to-use and efficient simulators. Figure 1 demonstrates how DuneCopasi emerges as a flexible solver poised to meet these evolving needs.

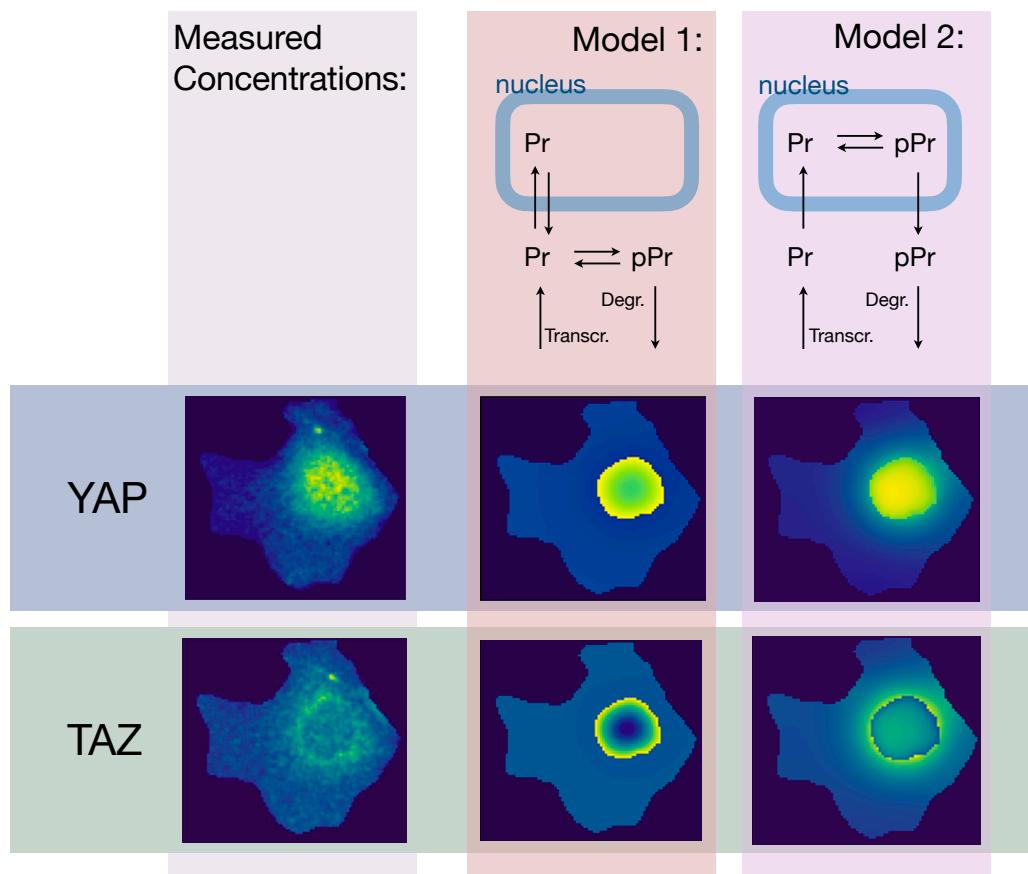


Figure 2: Mathematical modelling predicts that nuclear phosphorylation controls spatial localization of Hippo signalling pathway components YAP and TAZ. Here, we compare two model topologies - “Model 1” (canonical) and “Model 2” (alternative model) - concerning the intracellular distribution of YAP and TAZ proteins (Pr) and their phosphorylated counterparts (pPr). If the phosphorylation of YAP/TAZ takes place exclusively outside the nucleus, as shown in “Model 1”, PDE simulation indicates low spatial accordance with the experimentally measured subcellular localization of YAP/TAZ. Whereas, “Model 2” describes YAP/TAZ protein phosphorylation and dephosphorylation in the nucleus. The simulation of “Model 2” agrees with experimentally measured subcellular distribution of YAP/TAZ proteins, as reported in (Wehling et al., 2022).

Acknowledgements

We want to thank all the contributions that aided in the development and deployment of the package. Special thanks to Ursula Kummer from the COPASI team for valuable guidance. This work has been funded and supported by the German Federal Ministry of Education and Research (BMBF) FKZ 031L0158.

References

- Arevalo, H. J., Vadakkumpadan, F., Guallar, E., Jebb, A., Malamas, P., Wu, K. C., & Trayanova, N. A. (2016). Arrhythmia risk stratification of patients after myocardial infarction using personalized heart models. *Nature Communications*, 7(1), 11437. <https://doi.org/10.1038/ncomms11437>
- Arndt, D., Bangerth, W., Bergbauer, M., Feder, M., Fehling, M., Heinz, J., Heister, T., Heltai, L., Kronbichler, M., Maier, M., Munch, P., Pelteret, J.-P., Turcksin, B., Wells, D., & Zampini, S. (2023). The deal.II library, version 9.5. *Journal of Numerical Mathematics*, 31(3), 231–246. <https://doi.org/10.1515/jnma-2023-0089>
- Bastian, P., Blatt, M., Dedner, A., Dreier, N.-A., Engwer, C., Fritze, R., Gräser, C., Grüninger, C., Kempf, D., Klöfkorn, R., Ohlberger, M., & Sander, O. (2021). The dune framework: Basic concepts and recent developments. *Computers & Mathematics with Applications*, 81, 75–112. <https://doi.org/10.1016/j.camwa.2020.06.007>
- Bastian, P., Blatt, M., Dedner, A., Engwer, C., Klöfkorn, R., Ohlberger, M., & Sander, O. (2008). A generic grid interface for parallel and adaptive scientific computing. Part i: Abstract framework. *Computing*, 82, 103–119. <https://doi.org/10.1007/s00607-008-0003-x>
- Bastian, P., Heimann, F., & Marnach, S. (2010). Generic implementation of finite element methods in the Distributed and Unified Numerics Environment (DUNE). *Kybernetika*, 46(2), 294–315. dml.cz/dmlcz/140745
- Cantwell, C. D., Moxey, D., Comerford, A., Bolis, A., Rocco, G., Mengaldo, G., De Grazia, D., Yakovlev, S., Lombard, J.-E., Ekelschot, D., Jordi, B., Xu, H., Mohamied, Y., Eskilsson, C., Nelson, B., Vos, P., Biotto, C., Kirby, R. M., & Sherwin, S. J. (2015). Nektar++: An open-source spectral/hp element framework. *Computer Physics Communications*, 192, 205–219. <https://doi.org/10.1016/j.cpc.2015.02.008>
- Eliaš, J., & Clairambault, J. (2014). Reaction–diffusion systems for spatio-temporal intracellular protein networks: A beginner’s guide with two examples. *Computational and Structural Biotechnology Journal*, 10(16), 12–22. <https://doi.org/10.1016/j.csbj.2014.05.007>
- Hecht, F. (2012). New development in FreeFem++. *J. Numer. Math.*, 20(3-4), 251–265. <https://doi.org/10.1515/jnum-2012-0013>
- Hoops, S., Sahle, S., Gauges, R., Lee, C., Pahle, J., Simus, N., Singhal, M., Xu, L., Mendes, P., & Kummer, U. (2006). COPASI—a COmplex PAthway SImulator. *Bioinformatics*, 22(24), 3067–3074. <https://doi.org/10.1093/bioinformatics/btl485>
- Hucka, M., Finney, A., Sauro, H. M., Bolouri, H., Doyle, J. C., Kitano, H., Arkin, A. P., Bornstein, B. J., Bray, D., Cornish-Bowden, A., Cuellar, A. A., Dronov, S., Gilles, E. D., Ginkel, M., Gor, V., Goryanin, I. I., Hedley, W. J., Hodgman, T. C., Hofmeyr, J.-H., ... SBML Forum:, the rest of the. (2003). The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics*, 19(4), 524–531. <https://doi.org/10.1093/bioinformatics/btg015>
- Keegan, L., Andriushchenko, P., Schreiner, H., Caramizaru, H., & Patel, H. (2023). *Spatial-model-editor/spatial-model-editor: 1.5.0* (Version 1.5.0). Zenodo. <https://doi.org/10.5281/zenodo.10246531>
- Kempf, D., Klein, O., Kutri, R., Scheichl, R., & Bastian, P. (2023). Parafields: A generator for distributed, stationary gaussian processes. *Journal of Open Source Software*, 8(92), 5735. <https://doi.org/10.21105/joss.05735>
- Müthing, S. (2015). *A flexible framework for multi physics and multi domain PDE simulations* [PhD thesis, Universität Stuttgart]. <https://doi.org/10.18419/opus-3620>
- Schaff, J., Fink, C. C., Slepchenko, B., Carson, J. H., & Loew, L. M. (1997). A general

- computational framework for modeling cellular structure and function. *Biophys. J.*, *73*(3), 1135–1146. [https://doi.org/10.1016/s0006-3495\(97\)78146-3](https://doi.org/10.1016/s0006-3495(97)78146-3)
- 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. <https://doi.org/10.1093/bioinformatics/btt772>
- Wehling, L., Keegan, L., Fernández-Palanca, P., Hassan, R., Ghallab, A., Schmitt, J., Tang, Y., Le Marois, M., Roessler, S., Schirmacher, P., & others. (2022). Spatial modeling reveals nuclear phosphorylation and subcellular shuttling of YAP upon drug-induced liver injury. *eLife*, *11*, e78540. <https://doi.org/10.7554/elife.78540>