iDynoMiCS 2: Fully flexible computer modelling platform for microbial ecology

Robert J Clegg[1,2,3,*], Bastiaan J R Cockx[4], Stefan Lang[5], Barth F Smets[4], Jan-Ulrich Kreft[1,2,3],

1 Centre for Computational Biology, University of Birmingham, Edgbaston, Birmingham B12 5TT, United Kingdom
2 Institute of Microbiology and Infection, University of Birmingham, Edgbaston, Birmingham B12 5TT, United Kingdom
3 School of Biosciences, University of Birmingham, Edgbaston, Birmingham B12 5TT, United Kingdom
4 Department of Environmental Engineering, Technical University of Denmark, Bygningstorvet 115, 2800 Kgs. Lyngby, Denmark
5 Department of Bioinformatics, Friedrich Schiller University, Ernst-Abbe-Platz 2, 07743 Jena, Germany

*** r.j.clegg@bham.ac.uk**

Abstract

The source code is publicly available to download from `https://github.com/roughhawkbit/iDynoMiCS-2`.

Author Summary

Introduction

Individual-based modelling of microbial communities is becoming ever more widespread as software is developed and computational power increases [1]. Drive for more generic software: checked by more people, so more reliable [2]; easier to compare different model formulations, i.e. structural sensitivity [3]. Software based on [4–6].

The original iDynoMiCS framework [5] has seen a rise in popularity and is now used by many research groups all over the world. Half a decade later, we have initiated a rigorous overhaul of the framework. This is done to address several evident limitations of the original package: Learning how to use iDynoMiCS can be challenging and multiple 3rd party tools are required to properly use the software, simulation protocols are sensitive to typographical errors and difficult to validate, it is not possible to simulate more than a single environment, it is difficult to combine agent types, only spherical agents can be modeled, agents can only interact with a single solute grid cell and it is not possible to evaluate processes at different timescales.

To address these issues the iDynoMiCS 2 framework has been developed. This overhauled version of iDynoMiCS allows the user construct a new and unique model by combining simple building blocks that govern the processes and behavior of the agents and their environment. The user can use the default building blocks or develop own building blocks which are easily to implement because of the modular structure and

provided Java interfaces. A graphical user interface has been developed to assist the user with the model construction. This lowers the complexity for new iDynoMiCS users and reduces the changes of creating a protocol with errors.

Models

The model description follows the ODD (Overview, Design concepts, Details) protocol [7, 8]. Further details are available in Supporting Information for some parts of the model description.

Purpose

State variables and scales

Process overview and scheduling

Each compartment is self-contained. No interaction between compartments within a global time-step, all transfers are stored and happen instantly between timesteps.

Design concepts

The iDynoMiCS 2 structure inherits many concepts from previous packages [4, 5], but also builds on these.

Emergence

Modularity Aids customisability

Ease of comparison Since iDynoMiCs 2 already adopts the hybrid approach to modelling (discrete) cells in (continuous) concentration fields, it is straightforward from a design perspective to instead treat cells as a continuum. This "retrofit" allows easy comparison of the hybrid IbM approach with more traditional population-based approaches [9] for the same system. Such calibration is already used often, but implementing this feature is, to our knowledge, the first time it has been offered *within a single software package*.

Initialization

Input

Native protocol file layout, but (hopefully!) able to read in other formats such as Hucka2003 [10] and Cuellar2003 [11].

Submodels

?

1. Ferrer J, Prats C, López D. Individual-based modelling: an essential tool for microbiology. Journal of biological physics. 2008;34(1-2):19–37.
2. Joppa LN, McInerny G, Harper R, Salido L, Takeda K, O'Hara K, et al. Troubling trends in scientific software use. Science. 2013;340(6134):814–815.
3. Adamson M, Morozov AY. When can we trust our model predictions? Unearthing structural sensitivity in biological systems. In: Proc. R. Soc. A. The Royal Society; 2012. p. rspa20120500.
4. Kreft JU, Booth G, Wimpenny JW. BacSim, a simulator for individual-based modelling of bacterial colony growth. Microbiology. 1998;144(12):3275–3287.
5. Lardon LA, Merkey BV, Martins S, Dötsch A, Picioreanu C, Kreft JU, et al. iDynoMiCS: next-generation individual-based modelling of biofilms. Environmental Microbiology. 2011;13(9):2416–2434.
6. Storck T, Picioreanu C, Virdis B, Batstone D. Variable Cell Morphology Approach for Individual-Based Modeling of Microbial Communities. Biophysical Journal. 2014 may;106(9):2037–2048.
7. Grimm V, Berger U, Bastiansen F, Eliassen S, Ginot V, Giske J, et al. A standard protocol for describing individual-based and agent-based models. Ecological modelling. 2006;198(1):115–126.
8. Grimm V, Berger U, DeAngelis DL, Polhill JG, Giske J, Railsback SF. The ODD protocol: a review and first update. Ecological modelling. 2010;221(23):2760–2768.
9. Wanner O, Gujer W. A multispecies biofilm model. Biotechnology and bioengineering. 1986;28(3):314–328.
10. Hucka M, Finney A, Sauro HM, Bolouri H, Doyle JC, Kitano H, et al. The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. Bioinformatics. 2003;19(4):524–531.
11. Cuellar AA, Lloyd CM, Nielsen PF, Bullivant DP, Nickerson DP, Hunter PJ. An overview of CellML 1.1, a biological model description language. Simulation. 2003;79(12):740–747.