PhysioFit: a software to quantify cell growth parameters and extracellular fluxes

Loïc Le Grégam[1,2], Yann Guitton[2,3], Floriant Bellvert[1,2], Stéphanie Heux[1], Fabien Jourdan[2,4], Jean-Charles Portais[1,2,5], and Pierre Millard[1,2,*]

[1]Toulouse Biotechnology Institute, Université de Toulouse, CNRS, INRAE, INSA, Toulouse, France.

[2]MetaToul-MetaboHUB, National Infrastructure of Metabolomics and Fluxomics, Toulouse, France.

[3]University of Nantes, Faculty of Pharmacy, MMS, Nantes, France.

[4]Toxalim (Research Centre in Food Toxicology), Université de Toulouse, INRAE, ENVT, INP-Purpan, UPS, Toulouse, France.

[5]RESTORE (Geroscience & Rejuvenation Center), Université de Toulouse, INSERM, CNRS, EFS, Toulouse, France.

*pierre.millard@insa-toulouse.fr

Abstract

Summary: Quantification of growth parameters and extracellular uptake and production fluxes is central in systems and synthetic biology. Fluxes can be estimated using various mathematical models by fitting time-course measurements of the concentration of cells and extracellular substrates and products. A single tool is available to calculate extracellular fluxes, but it is hardly interoperable and includes a single hard-coded growth model. We present our open-source flux calculation software, PhysioFit, which can be used with any growth model and is interoperable by design. PhysioFit includes by default the most common growth models, and additional models can be implemented by users to calculate fluxes and other growth parameters for metabolic systems or experimental setups that follow alternative kinetics. PhysioFit can be used as a Python library and includes a graphical user interface for intuitive use by end-users and a command-line interface to streamline integration into existing pipelines.

Availability and Implementation: PhysioFit is implemented in Python 3 and was tested on Windows, Unix and MacOS platforms. PhysioFit is also freely available online at https://workflow4metabolomics.org. The source code, the data and the documentation are freely distributed under GPL3 license at https://github.com/MetaSys-LISBP/PhysioFit/ and https://physiofit.readthedocs.io/.

Introduction

Quantification of microbial growth rates and extracellular uptake and production fluxes is an essential task to address fundamental and applied questions in the fields of systems and synthetic biology, biotechnology, and health (Millard *et al.*, 2023; Peiro *et al.*, 2019; Hui *et al.*, 2020). Extracellular fluxes can be estimated via mathematical models by fitting time-course measurements of the concentration of cells and extracellular substrates and products (Peiro *et al.*, 2019; Murphy and Young, 2013). Several models have been proposed to relate these dynamic measurements to fluxes and other growth parameters (such as growth rates or lag times) (Baranyi and Roberts, 1994; Nielsen and Villadsen, 1992; Poccia *et al.*, 2014). In most situations, model-based flux quantification is carried out using homemade scripts, an error-prone process that does not guarantee the robustness and reproducibility of flux studies. We found only one software, Extracellular Time-Course Analysis (ETA) (Murphy and Young, 2013), to calculate extracellular fluxes. ETA is an important step towards more robust metabolic studies, but it includes only a single hard-coded growth model, relies on a proprietary programming language and is not interoperable with upstream and/or downstream computational tools. Overall, the lack of software to calculate extracellular fluxes significantly limits the scope of applicability, throughput and reproducibility of metabolic flux studies.

Here, we present PhysioFit, a Python tool which is interoperable by design and can be extended with additional growth model to calculate extracellular fluxes and other growth parameters in any situation.

Methods and implementation

The overall process implemented in PhysioFit is shown in Figure 1. After loading the required input data containing the time-course concentrations of biomass and extracellular metabolites (provided as a *tsv* file), the user can select an appropriate growth model and processing parameters (number of Monte-Carlo iterations, initial values and bounds on model parameters, and standard deviations on measurements). Fluxes and growth parameters included in the model are then estimated by fitting the measured concentrations. All parameters p are estimated by minimizing the objective function c defined as the weighted sum of squared errors:

$$c(p) = \sum_i \left(\frac{x_i - y_i(p)}{\sigma_i} \right)^2$$

where x_i is the experimental value of data point i, with an experimental standard deviation σ_i, and $y_i(p)$ is the corresponding value simulated by the model. The objective function c is minimized with the differential evolution optimization algorithm (Storn and Price, 1997), and the best solution is refined using the L-BFGS-B method (Byrd *et al.*, 1995) to maximize convergence. The goodness-of-fit is evaluated based on a χ^2 statistical test, confidence intervals on fluxes and growth parameters are estimated using a Monte-Carlo analysis, and plots are generated for visual inspection of the fitted

profiles. Finally, PhysioFit generates *i*) a *tsv* file containing the estimated fluxes and growth parameters, *ii*) a *txt* file containing the χ^2 test results, *iii*) a *yaml* configuration file containing the run parameters that can be provided as input to ensure reproducibility, *iv*) *pdf* and *svg* plots of simulation and measurements, and v) a *log* file containing detailed information on the calculation process.

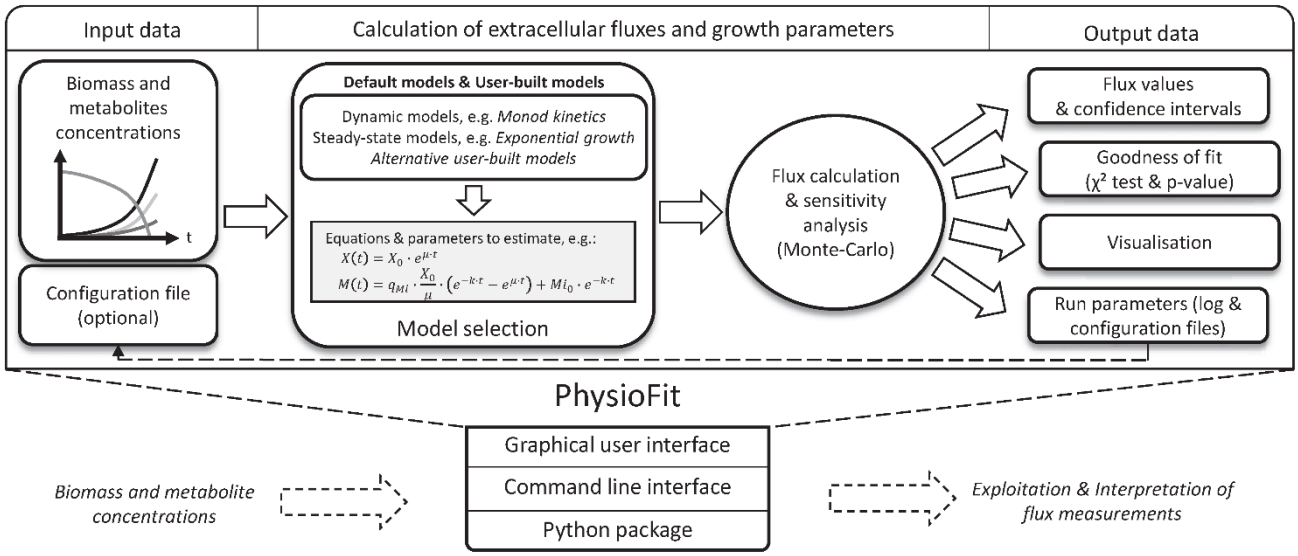


Figure 1. Overview of PhyioFit. Input data (time-course concentrations of biomass and extracellular metabolites) are fitted using a model (selected by the user), and flux calculation results are returned with associated statistics and plots. Different interfaces are available to enable utilization of PhysioFit in computational workflows.

A flux calculation model must contain equations that describe the dynamics of biomass and metabolite concentrations (used for simulations) and the list of all parameters with their default values and bounds (used for optimization). These equations are typically expressed as:

$$\frac{dX}{dt} = f(p, X, M_j)$$

$$\frac{dM_j}{dt} = g_j(p, X, M_j)$$

where X is the biomass concentration, M_j is the concentration of the metabolite j, and p is the vector of parameters. For instance, during exponential growth of *Escherichia coli* on dihydroxyacetone (DHA), a carbon source that is subject to non-enzymatic degradation (Peiro *et al.*, 2019), the functions f and g correspond to:

$$f(p, X) = \mu \cdot X$$

$$g_{DHA}(p, X, DHA) = -k \cdot DHA + X \cdot q_{DHA}$$

with parameters μ (growth rate), k (degradation constant of DHA) and q_{DHA} (DHA uptake flux).

Different steady-state and dynamic models are provided by default with PhysioFit, and additional models can be created by advanced users. A model template denoting the required structure is provided with PhysioFit, and extensive details on model construction can be found in the documentation. As detailed in Supplementary information, we have extensively validated PhysioFit by calculating fluxes from different datasets taken from the literature. Fluxes estimated by PhysioFit are in good agreement with the published flux values (maximal differences of 1%), thereby validating the software and the models implemented.

PhysioFit can be used via a Graphical User Interface, a Command-Line Interface or directly as a Python module to maximize its interoperability with other computational tools. We have implemented PhysioFit on Workflow4Metabolomics (Giacomoni *et al.*, 2015) (https://workflow4metabolomics.org), a collaborative portal for the metabolomics and fluxomics community, where it can be used as a standalone tool and be easily integrated into user-made workflows.

Conclusion

Here we present PhysioFit, the first flexible tool that allows calculating fluxes using any growth model with a user-friendly graphical interface. A command line interface and a Python package also enable integrating PhysioFit into metabolomics and fluxomics workflows. PhysioFit's modularity makes it an ideal tool for quantitative studies of any biological systems from microbial to higher cells and will benefit a wide range of fields including system and synthetic biology and biotechnology.

Acknowledgements

We thank Lindsay Peyriga and Cécilia Bergès (MetaToul-MetaboHUB, Toulouse, France) for testing the software, and Sophie Colombié and Sylvain Prigent (INRAE, Bordeaux, France) for insightful discussions.

Funding

MetaToul-MetaboHUB (Metabolomics and Fluxomics facilities, Toulouse, France, http://www.metatoul.fr) is part of MetaboHUB (http://www.metabohub.fr) funded by ANR grant MetaboHUB-ANR-11-INBS-0010.

Conflict of Interest: none declared.

References

Baranyi,J. and Roberts,T.A. (1994) A dynamic approach to predicting bacterial growth in food. *Int. J. Food Microbiol.*, **23**, 277–294.

Byrd,R.H. *et al.* (1995) A limited memory algorithm for bound constrained optimization. *SIAM J. Sci. Comput.*, **16**, 1190–1208.

Giacomoni,F. *et al.* (2015) Workflow4Metabolomics: a collaborative research infrastructure for computational metabolomics. *Bioinformatics*, **31**, 1493–1495.

Hui,S. *et al.* (2020) Quantitative fluxomics of circulating metabolites. *Cell Metab.*, **32**, 676-688.e4.

Millard,P. *et al.* (2023) Acetate is a beneficial nutrient for *E. coli* at low glycolytic flux. *EMBO J.*, e113079.

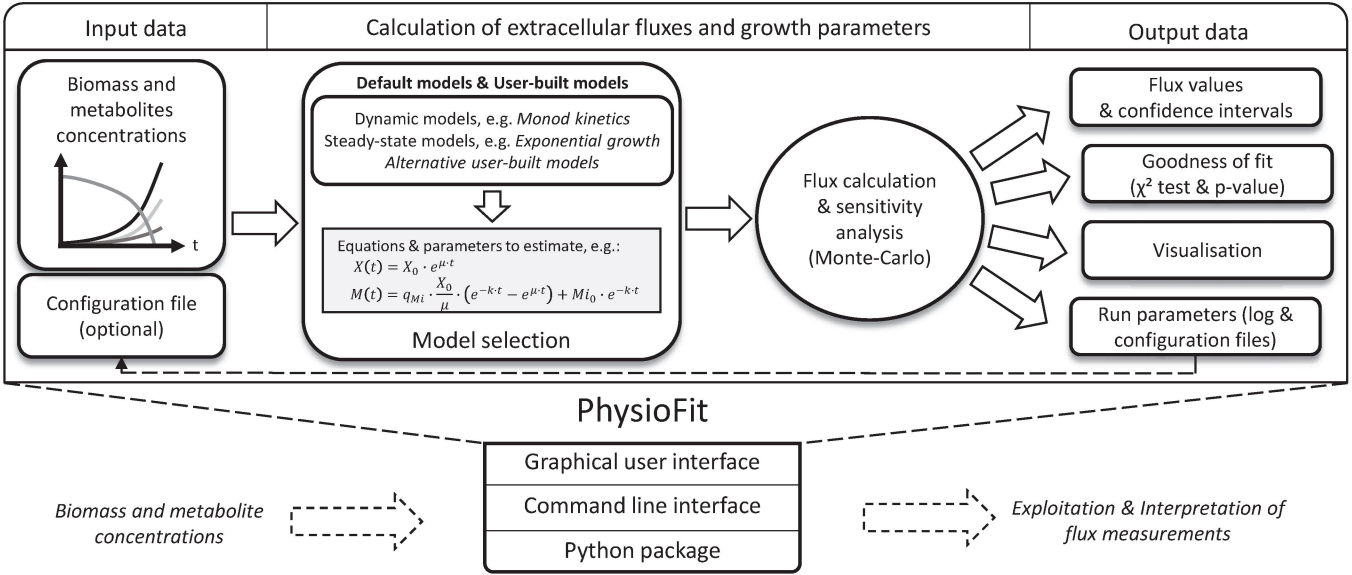
Murphy,T.A. and Young,J.D. (2013) ETA: Robust software for determination of cell specific rates from extracellular time courses. *Biotechnol. Bioeng.*, **110**, 1748–1758.

Nielsen,J. and Villadsen,J. (1992) Modelling of microbial kinetics. *Chem. Eng. Sci.*, **47**, 4225–4270.

Peiro,C. *et al.* (2019) Chemical and metabolic controls on dihydroxyacetone metabolism lead to suboptimal growth of *Escherichia coli*. *Appl. Environ. Microbiol.*, **85**, e00768-19.

Poccia,M.E. *et al.* (2014) Modeling the microbial growth of two *Escherichia coli* strains in a multi-substrate environment. *Braz. J. Chem. Eng.*, **31**, 347–354.

Storn,R. and Price,K. (1997) Differential Evolution - A simple and efficient heuristic for global optimization over continuous spaces. *J. Glob. Optim.*, **11**, 341–359.


Input data
Calculation of extracellular fluxes and growth parameters
Output data
Biomass and metabolites concentrations
t
Configuration file (optional)
Default models & User-built models
Dynamic models, e.g. Monod kinetics
Steady-state models, e.g. Exponential growth
Alternative user-built models
Equations & parameters to estimate, e.g.:
$X(t) = X_0 \cdot e^{\mu \cdot t}$
$M(t) = q_{Mi} \cdot \frac{X_0}{\mu} \cdot \left(e^{-k \cdot t} - e^{\mu \cdot t}\right) + Mi_0 \cdot e^{-k \cdot t}$
Model selection
Flux calculation & sensitivity analysis (Monte-Carlo)
Flux values & confidence intervals
Goodness of fit (χ^2 test & p-value)
Visualisation
Run parameters (log & configuration files)
PhysioFit
Graphical user interface
Command line interface
Python package
Biomass and metabolite concentrations
Exploitation & Interpretation of flux measurements