Conditional flux balance analysis toolbox for python

application to research metabolism in cyclic environments

Páez-Watson, Timothy; Hernández Medina, Ricardo; Vellekoop, Loek; van Loosdrecht, Mark C.M.; Wahl, S. Aljoscha

Citation (APA)
Páez-Watson, T., Hernández Medina, R., Vellekoop, L., van Loosdrecht, M. C. M., & Wahl, S. A. (2024). Conditional flux balance analysis toolbox for python: application to research metabolism in cyclic environments. *Bioinformatics Advances*, *4*(1), Article vbae174. https://doi.org/10.1093/bioadv/vbae174

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Systems biology

Conditional flux balance analysis toolbox for python: application to research metabolism in cyclic environments

Timothy Páez-Watson [1,*], Ricardo Hernández Medina [3], Loek Vellekoop[1], Mark C.M van Loosdrecht[1], S. Aljoscha Wahl[1,2]

[1]Department of Biotechnology, Delft University of Technology, Delft 2629 HZ, The Netherlands
[2]Lehrstuhl für Bioverfahrenstechnik, Friedrich-Alexander University Erlangen-Nürnberg, Erlangen 91052, Germany
[3]Present address: Faculty of Health and Medical Sciences, University of Copenhagen, Copenhagen DK-2200, Denmark

*Corresponding author. Department of Biotechnology, Delft University of Technology, van der Massweg 9, Delft 2629 HZ, Netherlands.
E-mail: timothypaezw@gmail.com
Associate Editor: Marieke Kuijjer

Abstract

Summary: We present py_cFBA, a Python-based toolbox for conditional flux balance analysis (cFBA). Our toolbox allows for an easy implementation of cFBA models using a well-documented and modular approach and supports the generation of Systems Biology Markup Language models. The toolbox is designed to be user-friendly, versatile, and freely available to non-commercial users, serving as a valuable resource for researchers predicting metabolic behaviour with resource allocation in dynamic-cyclic environments.

Availability and implementation: Extensive documentation, installation steps, tutorials, and examples are available at https://tp-watson-python-cfba.readthedocs.io/en/. The py_cFBA python package is available at https://pypi.org/project/py-cfba/.

1 Introduction

Optimal resource allocation is a widespread theory used to study evolutionary trade-offs—inherent in metabolic processes (Molenaar et al. 2009, Flamholz et al. 2013, Vázquez-Laslop and Mankin 2014, Elsemman et al. 2022). The prevailing literature predominantly focuses on microorganisms thriving under stationary conditions (Molenaar et al. 2009, Flamholz et al. 2013, Vázquez-Laslop and Mankin 2014, Elsemman et al. 2022). While such conditions lend themselves to laboratory validation, in reality, microbial habitats in nature and most environmental biotechnology applications are far from static. Over evolutionary timescales, microorganisms have evolved diverse metabolic strategies to face a diverse array of dynamic environmental fluctuations (Bajic and Sanchez 2020, Caetano et al. 2021). These fluctuating environments and the metabolic strategies of organisms living therein can be studied with computational models. Modelling is key to establish fundamental principles governing evolutionary fitness.

Rügen et al. (2015) introduced a mathematical framework named conditional flux balance analysis (cFBA), designed to predict optimal resource allocation dynamics under fluctuating conditions. This framework has been applied to cyanobacteria (Reimers et al. 2017) and polyphosphate accumulating organisms (Páez-Watson et al. 2023). In both cases, temporal synthesis of storage polymers (e.g. glycogen, polyphosphate, and polyhydroxyalkanoates) resulted as an emergent property of resource optimization in dynamic-cyclic scenarios. Nevertheless, reports on dynamic conditions remain sparse, primarily confined to these exemplar cases.

The cFBA method integrates stoichiometric modelling, dynamic Flux Balance Analysis (dFBA) with a final optimization through the whole simulation time, and resource allocation to study metabolic dynamics within cyclic environments. As such, it serves as a potent predictive tool for unveiling optimal metabolic strategies in ecosystems such as diurnal cycles, feast–famine dynamics, and aerobic–anaerobic transitions. Given the prevalence of such environmental conditions in nature, the field of microbial ecology could benefit from the application of cFBA. To date, there is no cFBA tool based on open-source platforms. The current implementation relies on adoption of complex metabolic models in MATLAB (Rügen et al. 2015, Reimers et al. 2017) with little documentation or simple examples for its application. Here, we present an easy-to-use Python toolbox for the application of cFBA (see Table 1 for a comparison of py_cFBA with other published models). This toolbox allows users to explore the boundaries of metabolic behaviour given a stoichiometric model, enzyme capacities, and a set of environmental conditions.

2 Toolkit usage and user input

The cFBA toolkit is accessible as a Python package https://pypi.org/project/py-cfba/. It comprises a suite of functions enabling the construction of the cFBA model architecture and subsequent simulations. The required user input varies depending on the desired complexity of the analysis. For instance, to model dynamic cycling without any catalytic information, a stoichiometric matrix suffices. However, if enzyme

Table 1. A comparison of cFBA with similar methods to research dynamic metabolism.

Feature	RBApy (Bulović et al. 2019)	Cycle Sync	cFBA	py_cFBA (this research)
Method	Resource Balance Analysis (Goelzer et al. 2015)	Cycle Sync (Sarkar et al. 2019)	Conditional FBA (Rügen et al. 2015)	Conditional FBA (Rügen et al. 2015)
Language	Python	Python	MATLAB[a]	Python
Toolbox for user implementation	+	−	−	+
Comprehensive documentation, tutorials and examples	+	−	−	+
Temporal flux analysis	+	+	+	+
Enzyme constraints	+	−	+	+
Study of organisms in cyclic environments	−	+	+	+
Capture temporal use of storage polymers	−	+	+	+

[a] Requires a license for its use.

capacities are to be included, numerical constants describing the relationships between reactions and their respective catalytic efficiencies (k_{cat}) are needed. The first step in this toolkit's pipeline is to generate a basic cFBA model structure, which is encoded into Systems Biology Markup Language (SBML). Subsequently, the SBML model is parsed into a linear programming problem. Detailed instructions on model generation and SBML file creation can be found at https://tp-watson-python-cfba.readthedocs.io/en/.

3 Methods and implementation

3.1 Minimal set of constraints: unlimited catalytic activity

The underlying metabolic model in cFBA is represented by a stoichiometric matrix (S), which represents the interplay of metabolites and reactions in a metabolic network. From the participating species (metabolites enzymes and biomass components), a subset is expressed as imbalanced (M). These species—typically enzymes, ribosomes, membranes, storage polymers, and substrates—exhibit explicit concentration changes over time, which are explicitly modelled as done with dFBA (Mahadevan et al. 2002). Conversely, the remainder of metabolites ($\bar{M}$) are presumed to remain in quasi-steady state, because their turnover rate is significantly faster than that of imbalanced species (Rügen et al. 2015, Waldherr et al. 2015, Reimers et al. 2017). Users can simulate dynamic environmental changes (such as variations in light, substrate and oxygen) by constraining reactions with upper and lower bounds.

Each cFBA simulation is normalized to an initial amount of biomass [typically 1 gram dry weight (g_{DW})]. Biomass is not modelled as an independent metabolite with its corresponding biosynthesis reaction, but rather defined as the weighed sum of all components in M at each time point (all imbalanced metabolites). To normalize the initial time point of the simulation to $1\ g_{DW}$, (1) is employed (Rügen et al. 2015).

$$w^T M^{t=0} = 1 \tag{1}$$

Here, w^T represents the transpose of a matrix containing the molecular weights of each imbalanced metabolite in M. The cyclic behaviour of cFBA is achieved by enforcing identical relative amounts of imbalanced metabolites at both the beginning and end of the simulation (2) (Rügen et al. 2015).

$$M^{t_{end}} = \mu\ M^{t=0} \tag{2}$$

μ represents the balanced growth of the system. These constraints represent a quadratic programming problem, which becomes linear for each value of μ. The objective of the cFBA model is to achieve the highest multiplication factor (μ) using a binary search algorithm. Numerically-stable solvers with high numeric precision, such as Gurobi, are recommended since complex models may lead to ill-conditioned problems (Reimers et al. 2017). The implementation of this method uses the OPTLANG library in Python and the solvers supported and their limitations have been described (Sonnenschein et al. 2018).

3.2 Cellular limits and requirements on metabolites: quotas

By default, the synthesis of imbalanced metabolites is not enforced [apart from maintaining the relation in (1)]. Minimal cellular requirements can be enforced by setting quota definitions (minimal concentration constraints). For instance, Rügen et al. (2015) employed quota compounds to establish minimal thresholds for inorganic ions, cell wall constituents, lipids, DNA, and non-catalytic proteins, relative to biomass. Expanding upon the quota definitions utilized by Rügen et al. (2015) and Reimers et al. (2017), our toolkit enables users to define exact, minimum, and maximum quota constraints at any time point during the simulation. This facilitates the capture of dynamic behaviours in simulated environmental conditions.

3.3 Coupling metabolism to protein allocation: enzyme activity based on enzyme amounts

Imbalanced metabolites can also act as catalysts of specific reactions. The relation between the metabolite and the reaction it catalyses is indicated in the capacity matrices [A and B in (3)], which denote the associations between catalysed reactions and the k_{cat} values of each catalysed reaction (Rügen et al. 2015).

$$A_{cap}{}^t v_r{}^t \leq B_{cap} \cdot M^t \tag{3}$$

$$v_r{}^t \leq M_e{}^t \cdot k_{cat_e}^r \tag{4}$$

Equation (3) sets a reaction r catalysed by an imbalanced metabolite M_e to be constrained by its upper limit following the relation in (4). It is noteworthy that this defines an upper boundary to the reaction, not an exact value. Additionally, if a reversible reaction is catalysed by an imbalanced specie,

both directions of the reaction must be accounted for in the S matrix.

Standard genome-reconstructed metabolic models typically include storage polymers such as glycogen as part of biomass components. Following the approach of Ofaim et al. (2021), we allow for the explicit separation of storage metabolites from that of biomass (also referred to as *lean* biomass). This allows the independent accumulation and utilization of said polymers in various simulations irrespective of biomass composition [independent from (2)].

An illustrative example of cFBA implementation for a toy model of a minimal cell is presented in Fig. 1. The system comprises one balanced metabolite (named 'intermediate') and three imbalanced species: storage, enzymes, and biomass. The reactions for substrate uptake and biomass synthesis are catalysed by the species 'enzymes' each with a distinct k_{cat} value. The simulation incorporates a dynamic component wherein substrate is only available until 2 h in the simulation. No quota compounds are defined, and the metabolite '*biomass*' solely contributes to w^T. The cFBA simulation results in an early use of resources (substrate) into enzyme biosynthesis to reach the maximum catalytic capacity at the third time-point (1 h). After this point, the system produces biomass at a balanced rate (optimizing enzyme usage), making temporal use of storage to allow this steady rate of biomass synthesis. Variations and step-by-step examples of this model implementation are available at https://tp-watson-python-cfba.readthedocs.io/en.

4 Conclusion

The Python cFBA toolkit facilitates the study of metabolic dynamics in cyclic environments. We included clear documentation and examples for a fast familiarization to resource allocation strategies in dynamic conditions. Two considerations are of note: numerical challenges may require specialized solvers, and further developments are needed to address complex biological systems such as non-optimal balanced growth strategies or microbial communities. Notwithstanding, the toolkit represents a significant advancement in systems biology, offering researchers a powerful tool to explore metabolic behaviour in dynamic-cyclic environments.

Conflict of interest

The authors declare no conflict of interest.
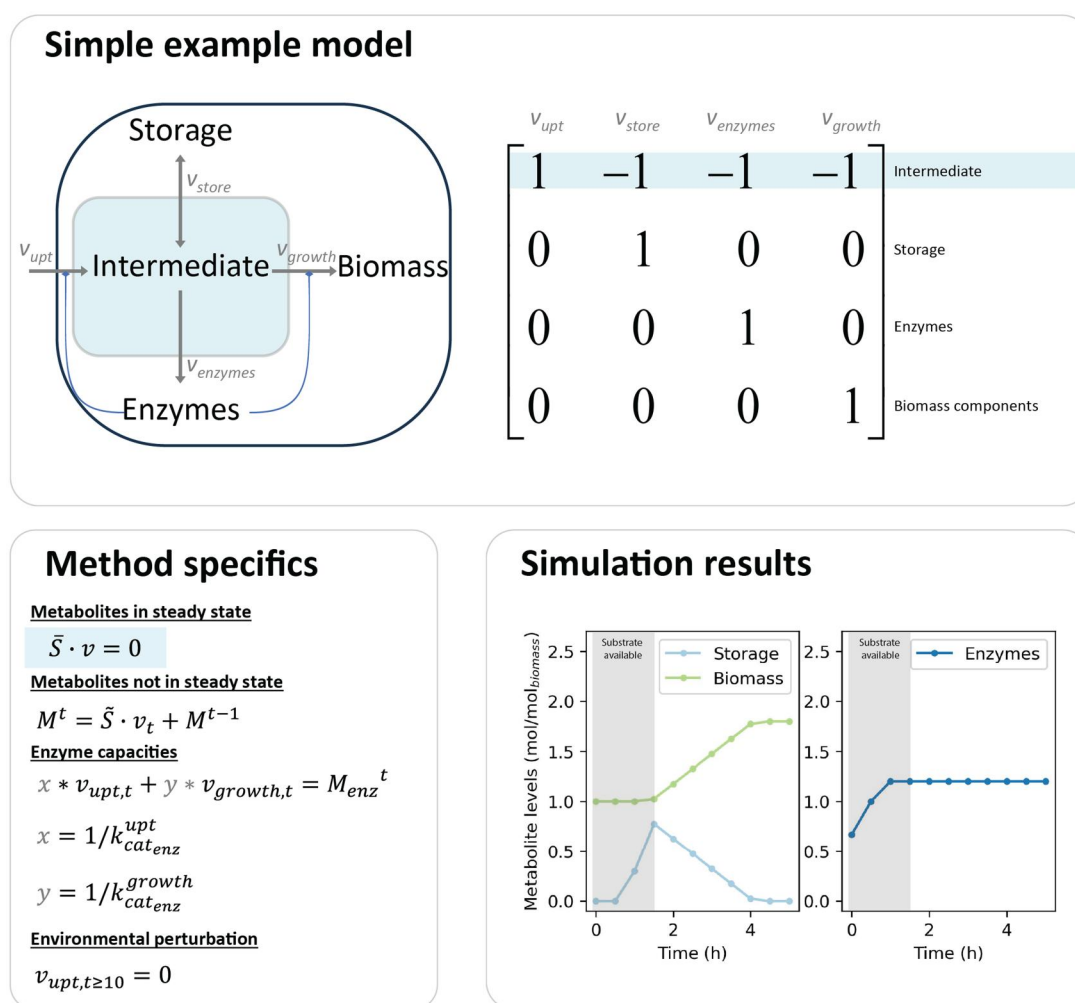
Figure 1. Basic simulation of a toy-model using the cFBA Python Toolbox. Inputs required for this model are a stoichiometric matrix, definition of balanced and imbalanced metabolites, and enzyme capacities. The simulation includes an active feed during the first 10 units of time after which there is no longer substrate simulating a feast–famine condition. Specifics and step-by-step implementation of this model in the cFBA python toolbox are available at https://tp-watson-python-cfba.readthedocs.io/en.

Funding

Data availability

Extensive documentation, installation steps, tutorials and examples are available at https://tp-watson-python-cfba.readthedocs.io/en/. The py_cFBA python package is available at https://pypi.org/project/py-cfba/.

References

Bajic D, Sanchez A. The ecology and evolution of microbial metabolic strategies. *Curr Opin Biotechnol* 2020;**62**:123–8.

Bulović A, Fischer S, Dinh M *et al.* Automated generation of bacterial resource allocation models. *Metab Eng* 2019;**55**:12–22.

Caetano R, Ispolatov Y, Doebeli M. Evolution of diversity in metabolic strategies. *Elife* 2021;**10**:e67764.

Elsemman IE, Rodriguez Prado A, Grigaitis P *et al.* Whole-cell modeling in yeast predicts compartment-specific proteome constraints that drive metabolic strategies. *Nature Commun* 2022;**13**:801.

Flamholz A, Noor E, Bar-Even A *et al.* Glycolytic strategy as a tradeoff between energy yield and protein cost. *Proc Natl Acad Sci* 2013;**110**:10039–44.

Goelzer A, Muntel J, Chubukov V *et al.* Quantitative prediction of genome-wide resource allocation in bacteria. *Metab Eng* 2015;**32**:232–43.

Mahadevan R, Edwards JS, Doyle FJ. Dynamic flux balance analysis of diauxic growth in *Escherichia coli*. *Biophys J* 2002;**83**:1331–40.

Molenaar D, Van Berlo R, De Ridder D *et al.* Shifts in growth strategies reflect tradeoffs in cellular economics. *Mol Syst Biol* 2009;**5**:323.

Ofaim S, Sulheim S, Almaas E *et al.* Dynamic allocation of carbon storage and nutrient-dependent exudation in a revised genome-scale model of *Prochlorococcus*. *Front Genet* 2021;**12**:586293.

Páez-Watson T, van Loosdrecht MC, Wahl SA. Predicting the impact of temperature on metabolic fluxes using resource allocation modelling: application to polyphosphate accumulating organisms. *Water Res* 2023;**228**:119365.

Reimers A-M, Knoop H, Bockmayr A *et al.* Cellular trade-offs and optimal resource allocation during cyanobacterial diurnal growth. *Proc Natl Acad Sci U S A* 2017;**114**:E6457–65.

Rügen M, Bockmayr A, Steuer R. Elucidating temporal resource allocation and diurnal dynamics in phototrophic metabolism using conditional FBA. *Sci Rep* 2015;**5**:15247.

Sarkar D, Mueller TJ, Liu D *et al.* A diurnal flux balance model of *Synechocystis* sp. PCC 6803 metabolism. *PLoS Comput Biol* 2019;**15**:e1006692.

Sonnenschein N, Jensen K, Cardoso JGR *et al.* biosustain/optlang (1.4.4). *Zenodo*, 2018. https://explore.openaire.eu/search/software?pid=10.5281%2Fzenodo.2222365

Vázquez-Laslop N, Mankin AS. Protein accounting in the cellular economy. *Cell* 2014;**157**:529–31.

Waldherr S, Oyarzún DA, Bockmayr A. Dynamic optimization of metabolic networks coupled with gene expression. *J Theor Biol* 2015;**365**:469–85.