COBREXA.jl: constraint-based reconstruction and exascale analysis

Miroslav Kratochvíl[1, †], Laurent Heirendt[1,4, †], St. Elmo Wilken[2], Taneli Pusa[1,3], Sylvain Arreckx[1], Alberto Noronha[3], Marvin van Aalst[2], Venkata P Satagopam[1,4], Oliver Ebenhöh[2], Reinhard Schneider[1,4], Christophe Trefois[1,4, *], and Wei Gu[1,4, *]

[1]Luxembourg Centre for Systems Biomedicine, University of Luxembourg, Campus Belval, Belvaux, Luxembourg
[2]Institute of Quantitative and Theoretical Biology, Heinrich-Heine-Universität Düsseldorf, Universitätsstraße 1, 40225 Düsseldorf, Germany.
[3]Nium S.à.r.l., 6 Avenue des hauts fourneaux, L-4362 Esch-sur-Alzette, Luxembourg
[4]ELIXIR Luxembourg, 6, avenue du Swing, Campus Belval, L-4367 Belvaux, Luxembourg.
[†]Equal contributors.
[*]To whom correspondence should be addressed.

Summary: COBREXA.jl is a Julia package for scalable, high-performance constraint-based reconstruction and analysis of very large-scale biological models. Its primary purpose is to facilitate the integration of modern high performance computing environments with the processing and analysis of large-scale metabolic models of challenging complexity. We report the architecture of the package, and demonstrate how the design promotes analysis scalability on several use-cases with multi-organism community models.
Availability and implementation: `https://doi.org/10.17881/ZKCR-BT30`.
Contact: christophe.trefois@uni.lu, wei.gu@uni.lu

1 Introduction

Understanding metabolic interactions in cells is a crucial step to investigate disease mechanisms and to discover new therapeutics (Cook and Nielsen, 2017; Apaolaza et al., 2018; Brunk et al., 2018). Constraint-Based Reconstruction and Analysis (COBRA) is a promising methodology for analyzing various metabolic processes at the organism- and community- levels (Fang, Lloyd, and Palsson, 2020). The main idea behind COBRA is to represent an organism as a constrained set of interconnected reactions and metabolites based on genomic sequencing data. This leads to a straightforward interpretation of metabolism as a constrained linear system, which enables the utilization of a wide range of well-developed analysis methods (Orth, Thiele, and Palsson, 2010).

The increasing ubiquity of genomic sequencing has led to a rapid expansion in the number and complexity of genome-scale metabolic models, e.g. the human metabolic model that has more than 80,000 reactions (Thiele et al., 2020). Recent automated reconstruction tools can generate models spanning the entire primary metabolism of both pro- and eukaryotes (Machado et al., 2018). Consequently, metabolic models are becoming considerably larger in scale than their predecessors, which is further compounded by the construction of multi-member community models. This growth implies increasing analysis complexity (see Figure S1), which in turn drives the need to develop analysis software that can accommodate this complexity. While computing the solutions to the underlying constrained optimization problems is hard to accelerate and parallelize, many analysis types can be decomposed into individual invocations of the optimizer, which may be parallelized. However, despite continued efforts (Heirendt, Thiele, and Fleming, 2017), this remains challenging due to the scalability limits of existing software implementations.

Here, we present COBREXA.jl, a package for implementing and running distributed COBRA workflows. The package is implemented in the Julia programming language (Bezanson et al., 2017), enabling facile extension with user-defined numeric-computing routines, and interoperability with many high-performance computing packages. It provides a 'batteries-included' solution for scaling analyses to make efficient use of high-performance computing (HPC) facilities, giving researchers a powerful toolkit for executing complicated high-volume workflows, such as the creation and exploration of digital metabolic twins in personalized medicine (Björnsson et al., 2020), and analysis of extensive microbial communities in ecology and biotechnology. We report the implementation architecture, and substantiate how the design accommodates future extensions and scaling of common analysis tasks.

2 Implementation and results

COBREXA.jl is an open architecture solution, providing interchangeable building blocks for implementing complicated COBRA workflows. Common analysis methods, such as flux balance, flux variability, and gene knockout analyses (Gudmundsson and Thiele, 2010), are implemented as ready-to-use functions that may be easily composed and customized. Most importantly, the building blocks are designed so that the constructed workflows can be easily separated into parallelizable analysis steps and executed on multiple computation nodes in HPC environments

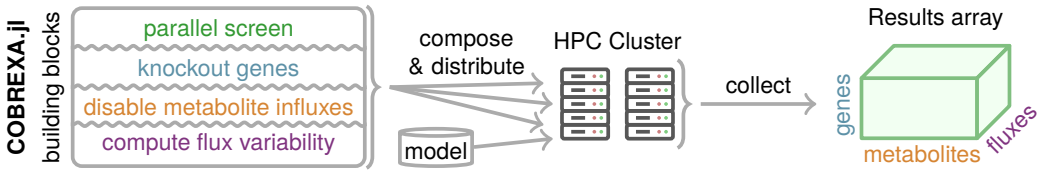

Figure 1: Schema of an example custom analysis construction that examines flux variability in many variants of a model, its distributed execution with COBREXA.jl, and collection of many results in a multi-dimensional array.

(as illustrated in Fig. 1). The concurrent execution of such workflows results in significant computational speedups, without requiring user expertise in parallel programming.

The design of COBREXA.jl distinguishes it from other COBRA implementations, which typically provide parallelization support for only a few selected methods, and no current support for parallelization of custom method variants. For example, parallel single-gene deletion analysis is commonly supported, but a variant that explores the flux variability in knockouts must be reimplemented and parallelized by the user.

A variety of model exchange and representation formats are supported, including MATLAB format (Heirendt, Arreckx, et al., 2019); object-oriented JSON format (Ebrahim et al., 2013), and SBML (Keating et al., 2020). Additionally, implementation of the workflows in Julia results in highly optimized execution of the code at the cost of minor pre-compilation overhead, which benefits large, data-heavy use cases. A detailed architecture overview is provided in Supplementary Section S1.

To evaluate the effect of the new architecture and optimizations on the performance and scalability of COBRA analyses, we benchmarked COBREXA.jl on use-cases that benefit from parallelization. We compared its performance to that obtained with COBRApy (Ebrahim et al., 2013) and COBRA Toolbox (Heirendt, Arreckx, et al., 2019), which are the widely adopted tools for running COBRA workflows. Running on a 256-CPU multi-node cluster, COBREXA.jl was able to fully utilize the available distributed computing resources and outperform the implementation of flux variability analysis in other packages by a factor of between 2× and 10×, even on relatively small models (Supplementary Table S2). We further demonstrated that COBREXA.jl is able to parallelize and distribute custom workloads by re-implementing the production envelope functionality of COBRApy; leading to speedups of over 10×, even on a single 16-core computa-

tion node (Supplementary Table S3). Consequently, we expect that the COBRA methods implemented in COBREXA.jl will enable reliable acceleration of many current and future workloads by simply adding more computing resources. The results are further discussed in Supplementary Section S3.4.

3 Conclusion

COBREXA.jl is a new package developed for large-scale distributed processing of constraint-based biological models. It differs from the other implementations of COBRA methods (Heirendt, Arreckx, et al., 2019; Ebrahim et al., 2013) by focusing on computational efficiency, and simplifies high-level construction of parallelized user-defined analysis methods. This is required for performing extensive analyses of large models, future-proof extensibility, and workload distribution that enables effective utilization of the common HPC infrastructure resources. The package thus enables fast analysis of datasets that may pose challenges for the currently available tools, such as the comprehensive human gut microbiome models.

Funding

The research leading to these results has received funding from the European Union's Horizon 2020 Programme under the PerMedCoE Project (www.permedcoe.eu), grant agreement n° 951773. This work was also partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy–EXC-2048/1–project ID 390686111 and EU's Horizon 2020 research and innovation program under the Grant Agreement 862087. The presented experiments were carried out using the HPC facilities of the University of Luxembourg (see `https://hpc.uni.lu`).

References

Apaolaza, Iñigo et al. (2018). COBRA methods and metabolic drug targets in cancer. *Mol & Cell Oncol* 5.1, e1389672.

Bezanson, Jeff et al. (2017). Julia: A fresh approach to numerical computing. *SIAM Rev* 59.1, pp. 65–98.

Björnsson, Bergthor et al. (2020). Digital twins to personalize medicine. *Genome Med* 12.1, pp. 1–4.

Brunk, Elizabeth et al. (2018). Recon3D enables a three-dimensional view of gene variation in human metabolism. *Nat Biotechnol* 36.3, p. 272.

Cook, Daniel J and Jens Nielsen (2017). Genome-scale metabolic models applied to human health and disease. *Wires Syst Biol Med* 9.6, e1393.

Ebrahim, Ali et al. (2013). COBRApy: COnstraints-Based Reconstruction and Analysis for Python. *BMC Syst Biol* 7.1, pp. 1–6.

Fang, Xin, Colton J Lloyd, and Bernhard O Palsson (2020). Reconstructing organisms in silico: genome-scale models and their emerging applications. *Nat Rev Microbiol* 18.12, pp. 731–743.

Gudmundsson, Steinn and Ines Thiele (2010). Computationally efficient flux variability analysis. *BMC Bioinformatics* 11.1, pp. 1–3.

Heirendt, Laurent, Sylvain Arreckx, et al. (2019). Creation and analysis of biochemical constraint-based models using the COBRA Toolbox v.3.0. *Nat Protoc* 14.3, pp. 639–702.

Heirendt, Laurent, Ines Thiele, and Ronan MT Fleming (2017). DistributedFBA.jl: high-level, high-performance flux balance analysis in Julia. *Bioinformatics* 33.9, pp. 1421–1423.

Keating, Sarah M et al. (2020). SBML Level 3: an extensible format for the exchange and reuse of biological models. *Mol Syst Biol* 16.8, e9110.

Machado, Daniel et al. (2018). Fast automated reconstruction of genome-scale metabolic models for microbial species and communities. *Nucleic Acids Res* 46.15, pp. 7542–7553.

Orth, Jeffrey D, Ines Thiele, and Bernhard O Palsson (2010). What is flux balance analysis? *Nat Biotechnol* 28.3, pp. 245–248.

Thiele, Ines et al. (2020). Personalized whole-body models integrate metabolism, physiology, and the gut microbiome. *Mol Syst Biol* 16.5, e8982.