

# GEMtractor: Extracting Views into Genome-scale Metabolic Models

Martin Scharm<sup>1</sup>, Olaf Wolkenhauer<sup>1,2</sup>, Mahdi Jalili<sup>3</sup>, and Ali Salehzadeh-Yazdi<sup>1</sup>

<sup>1</sup>Department of Systems Biology and Bioinformatics, University of Rostock, Rostock, Germany

<sup>2</sup>Stellenbosch Institute for Advanced Study, Wallenberg Research Centre at Stellenbosch University, Stellenbosch, South Africa

<sup>3</sup>Hematology, Oncology and SCT Research Center, Tehran University of Medical Sciences, Tehran, Iran

Corresponding author:

Martin Scharm<sup>1</sup>

Email address: [martin.scharm@uni-rostock.de](mailto:martin.scharm@uni-rostock.de)

## ABSTRACT

**Summary:** Computational metabolic models typically encode for graphs of species, reactions, and enzymes. Comparing genome-scale models through topological analysis of multipartite graphs is challenging. However, in many practical cases it is not necessary to compare the full networks. The GEMtractor is a web-based tool to trim models encoded in SBML. It can be used to extract subnetworks, for example focusing on reaction- and enzyme-centric views into the model.

**Availability and Implementation:** The GEMtractor is licensed under the terms of GPLv3 and developed at [github.com/binfalse/GEMtractor](https://github.com/binfalse/GEMtractor) – a public version is available at [sbi.uni-rostock.de/gemtractor](http://sbi.uni-rostock.de/gemtractor).

**Contact:** [martin.scharm@uni-rostock.de](mailto:martin.scharm@uni-rostock.de) and [ali.salehzadeh-yazdi@uni-rostock.de](mailto:ali.salehzadeh-yazdi@uni-rostock.de)

## 1 INTRODUCTION

At the most basic abstraction level, biological phenomena can be represented as mathematical graphs. Genome-scale metabolic models (GEMs) are an example, that describe the associations between genes, proteins, and reactions of an organism. Such models are typically encoded as multipartite graphs, which are partitioned into distinct subsets of vertices representing metabolites, reactions, and enzymes. Since topological analyses of large-scale multipartite graphs is challenging, one is often identifying subnetworks from a whole genome metabolic model. Extracting the reaction-centric (links of reactions) or enzyme-centric (links of enzymes) view simplifies the graph structure and shifts its perspective – from kinetic interactions to phenotypical connections. This opens opportunities for novel topological analyses of the metabolism (Lacroix *et al.*, 2008).

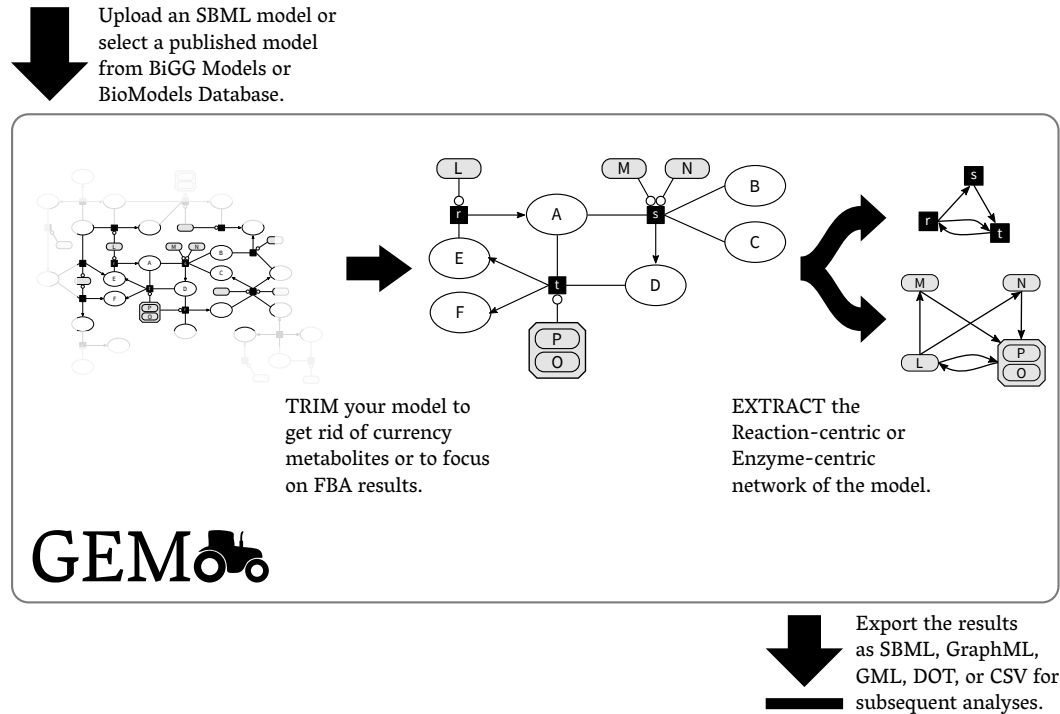
The idea of extracting and analysing the enzyme-centric network of a GEM is not new. However, Horne *et al.* (2004) is not available anymore and Asgari *et al.* (2018) apparently mistook the reaction-centric network for an enzyme-centric network.

Here we introduce the GEMtractor, a web-based tool to extract reaction-centric and enzyme-centric views of genome-scale metabolic models (GEM). GEMtractor allows for trimming of models to, for example, remove currency metabolites or to focus on context-based models. The GEMtractor is a free software and easy to install, including an increased privacy and speed because it can be deployed locally.

## 2 TECHNICAL NOTES

The GEMtractor is a Django web application, that we developed with a focus on privacy, compatibility and speed. The interactive front-end is implemented using jQuery and designed using W3CSS. Thus, it works in all modern browsers as well as on mobile devices; there is no need for registration. Figure 1 shows a typical workflow when using the GEMtractor.

The GEMtractor uses libsbml (Bornstein *et al.*, 2008) to support all models encoded in SBML. To start an analysis, the user needs to select a model. Users may upload their own models or choose a



**Figure 1.** The workflow when using the GEMtractor: A user selects a model, trims undesired entities, extracts a view into the model, and exports the results in exchangeable formats.

46 published model form BiGG (King *et al.*, 2016) or BioModels (Chelliah *et al.*, 2015). Decoding the  
47 metabolite-reaction network from the SBML is straightforward. However, understanding which enzyme  
48 combinations catalyse a certain reaction is a bit tricky. The GEMtractor supports two flavours of gene-  
49 product associations: Using the reactions' notes and proper annotations using the SBML FBC package  
50 (Olivier *et al.*, 2015). Some reactions lack both annotations – in this case, the GEMtractor assumes  
51 that there is an implicit enzyme whose identifier matches the identifier of the reaction, prefixed with  
52 `reaction_`. We call these invented enzymes *fake enzymes*.

53 As soon as the model is validly decoded, the user can start trimming entities. The front-end lists all  
54 species, reactions, enzymes, and enzyme complexes in individual tables, which can be sorted by different  
55 criteria, such as occurrences of entity names. It is then comparatively easy to remove currency metabolites.  
56 The filters can also programmatically applied using a text-field, so users may save their filters offline or  
57 paste filters obtained by their FBA results. Every change on the filters is evaluated at the front-end to  
58 check for possible inconsistencies. For example, if an enzyme is to be removed, it may be necessary to  
59 also remove all enzyme complexes, in which it participates. Such inconsistencies are highlighted and  
60 summarised at the top of the page to inform the user.

61 The user may then decide to extract a specific view from that trimmed model. In the reaction-centric  
62 network consecutive reactions are linked. That means, if a reaction  $r$  produces a metabolite that is  
63 consumed by the reaction  $s$ , there will be a link from  $r$  to  $s$ . Similarly, the enzyme-centric network links  
64 consecutive enzymes and enzyme complexes. For example, if  $r$  is catalysed by  $X$  or  $Y$  (alternatives, both  
65 can catalyse) and  $s$  is catalysed by  $M$  and  $N$  (both are necessary for the catalysis), then there will be two  
66 links: From  $X$  to ( $M$  and  $N$ ) and from  $Y$  to ( $M$  and  $N$ ). Thus, both the reaction- and the enzyme-centric  
67 network are directed, unipartite graphs.

68 Finally, the obtained graph can be exported into an exchangeable format. The GEMtractor supports  
69 SBML, DOT, GML, GraphML, and CSV as output formats. In case of SBML, it preserves as many  
70 annotations as possible. For example, the *species* in the reaction-centric network will have all the  
71 annotations of the reactions in the original metabolite-reaction network. Similarly, if the gene products  
72 were annotated using the FBC package, they will be annotated in the exported file.

73 In addition to the web browser front-end, the GEMtractor provides a decent API. Trimming and  
74 extracting can be encoded in JSON jobs and sent to the API endpoint, which will take care of the  
75 computation and immediately return the results. Client implementations in several languages are shipped  
76 with the source code. Due to Django's architecture, the heart of the GEMtractor can be used as a Python  
77 module if a network connection is undesired.

78 Even though the GEMtractor can basically handle all valid SBML models, some models cause  
79 problems. For example, the computation time increases with model size, which may cause timeouts at the  
80 web server. Similarly the maximum upload size allowed by the web server could terminate a data transfer,  
81 so not every model may be upload-able. Furthermore, some models may have invalid gene associations  
82 while being valid SBML. That is, because in SBML prior to level 3 there was no standard on encoding  
83 the gene associations – they were basically encoded in a free-text field, which is of course error-prone.  
84 This is primarily the case for models from BioModels, as those models are often encoded in SBML level  
85 2. If the GEMtractor finds a gene association but is not able to parse it, it will stop and report an error.

86 While the GEMtractor was built for GEMs, it is applicable to any SBML model. Extracting an enzyme  
87 network may be pointless, as kinetic models typically lack the gene association annotations, but you can  
88 still trim models to focus on submodels or extract the reaction-centric network for subsequent analyses.

89 Our public GEMtractor instance at [sbi.uni-rostock.de/gemtractor](http://sbi.uni-rostock.de/gemtractor) should be useful for the majority of  
90 analyses on GEMs. However, if you still need extended quotas, cache options, web server upload limits  
91 or timeouts, you are invited to install your own GEMtractor. The installation process is documented on  
92 the GEMtractor's website and in the source code. It requires a Python application server (e.g. gunicorn)  
93 to deal with the dynamic pages, and a default web server (e.g. nginx) to serve the static files. However, if  
94 you are happy with Docker, installing a GEMtractor boils down to a single command line.

95 As documentation is key to dissemination, the GEMtractor comes with an [extensive FAQ](#), proper  
96 [Python documentation](#) and [example client implementations](#) in different programming languages. Auto-  
97 matic tests cover most of its source code to prevent future programming mistakes and the GEMtractor  
98 exposes a monitoring endpoint to keep an eye on its health.

## 99 ACKNOWLEDGEMENTS

100 We would like to thank Markus Wolfien and Tom Gebhardt for creative brainstormings.

## 101 FUNDING

102 This work was supported by the German Federal Ministry of Education and Research (BMBF) as part of  
103 the ERACoBioTech project BESTER, FKZ 031 B0594B.

## 104 REFERENCES

- 105 V. Lacroix, L. Cottret *et al.* (2008) An Introduction to Metabolic Networks and Their Structural Analysis,  
106 *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, **5:4**, 594-617.
- 107 A. B. Horne, T. C. Hodgman *et al.* (2004) Constructing an enzyme-centric view of metabolism, *Bioinform-*  
108 *atics*, **20:13**, 2050–2055.
- 109 Y. Asgari, Z. Zabihinpour, and A. Masoudi-Nejad (2018) SCAN-Toolbox: Structural COBRA Add-oN  
110 (SCAN) for Analysing Large Metabolic Networks, *Current Bioinformatics*, **13:1**, 100-107.
- 111 B.J. Bornstein, S.M. Keating *et al.* (2008) LibSBML: an API library for SBML, *Bioinformatics*, **24:6**,  
112 880–881.
- 113 Z. A. King, J. Lu *et al.* (2016) BiGG Models: A platform for integrating, standardizing and sharing  
114 genome-scale models, *Nucleic Acids Research*, **44:D1**, D515–D522.
- 115 V. Chelliah, N. Juty *et al.* (2015) BioModels: ten-year anniversary, *Nucleic Acids Research*, **43:D1**,  
116 D542–D548.
- 117 B.G. Olivier and F.T. Bergmann (2015) Flux Balance Constraints, Version 2 Release 1, Available from  
118 COMBINE [identifiers.org/combine.specifications/sbml.level-3.version-1.fbc.version-2.release-1](http://identifiers.org/combine.specifications/sbml.level-3.version-1.fbc.version-2.release-1).