BIOINFORMATICS APPLICATIONS NOTE Vol. 30 no. 7 2014, pages 1036–1038
doi:10.1093/bioinformatics/btt747

Systems biology

Advance Access publication December 25, 2013

Assimilating genome-scale metabolic reconstructions with modelBorgifier

John T. Sauls and Joerg M. Buescher*

BRAIN Aktiengesellschaft, Microbial Production Technologies, Platform for Quantitative Biology and Sequencing, D-64673 Zwingenberg, Germany

Associate Editor: Igor Jurisica

ABSTRACT

Motivation: Genome-scale reconstructions and models, as collections of genomic and metabolic information, provide a useful means to compare organisms. Comparison requires that models are similarly notated to pair shared components.

Result: Matching and comparison of genome-scale reconstructions and models are facilitated by modelBorgifier. It reconciles models in light of different annotation schemes, allowing diverse models to become useful for synchronous investigation.

Availability and implementation: The modelBorgifier toolbox is freely available at http://www.brain-biotech.de/downloads/modelBorgifier.zip.

Contact: jrb@brain-biotech.de

Supplementary information: Supplementary data are available at *Bioinformatics* online.

Received on July 15, 2013; revised on December 2, 2013; accepted on December 19, 2013

1 INTRODUCTION

Genomic reconstructions can help develop hypothesis-driven discovery, provide a background for high-throughput data and provide an *in silico* sandbox for metabolic engineering (Oberhardt *et al*., 2009). Because reconstructions are easily interpretable and functionally testable representations of an organism's genome and metabolism, they also provide a means to compare organisms (Oberhardt *et al*., 2011). Comparing reconstructions allows for conserved metabolic pathways, growth predictions under similar conditions and multi-organism communities to be investigated. Importantly, these studies are predicated on the stipulation that the concerned models are semantically compatible, consistently annotated and similarly notated. Ideally, reaction and metabolite entries would use identical naming methods.

The increased popularity of genome-scale reconstructions and models (GEMs) has sparked the development of markup and semantic standards. Namely, Systems Biology Markup Language (SBML) has been adopted as the *lingua franca* for model exchange (Hucka *et al*., 2003). However, this standard does not define a naming scheme for reactions and metabolites or a generally accepted reference database. Consequently, fully automated comparison of GEMs from diverse sources is currently impossible.

*To whom correspondence should be addressed.

To rectify this situation, some attempts have been made to create collections of GEMs that use a standard reaction and metabolite nomenclature. This includes the BIGG database (Schellenberger *et al*., 2010) and MetRxn (Kumar *et al*., 2012). Although both provide online databases of curated models, users cannot submit models. This precludes the spontaneous comparison of additional or novel GEMs, such as those from the model SEED (Henry *et al*., 2010).

Tools do exist to compare and to merge arbitrary SBML formatted regulatory (Randhawa *et al*., 2009; Wang *et al*., 2010) or biochemical (Krause *et al*., 2010; Schulz *et al*., 2012) models. However, these methods are predominantly aimed at combining (sub-) models into larger entities.

Here we present, modelBorgifier, a tool that is specifically aimed at comparing and combining large, curated or automatically generated, organism-specific genome-scale reconstructions. It is implemented in MATLAB and integrates seamlessly with the popular openCOBRA Toolbox (Schellenberger *et al*., 2011).

Model comparison uses both greedy and network comparison techniques to find matching reaction and metabolite entries. Merging is done in a semiautomated fashion; manual match/no match decisions are fed to machine learning algorithms to augment auto-matching. The resulting composite model provides a framework to compare models, a means to share information between them and a reaction database for *in silico* metabolic engineering. The original models can be extracted from the composition with mathematical fidelity and addended information.

2 METHODS

2.1 Model comparison and matching

Model comparison is performed via pairwise comparison of reactions based on 40 parameters. Reactions are scored via both greedy (string similarity) comparison of their annotations and the annotations of their reactants, and their local network topology. The weighted sum of the 40 parameters constitutes the similarity score for each pair of reactions or metabolites.

After reactions are scored, the user is guided with a graphical user interface (GUI, Fig. 1) to declare reactions as possessing a match or as new. Once a reaction is designated, the involved metabolites are compared and matched or declared new with an analogous GUI. Metabolites can be compared independently of reactions.

The process of matching a reaction (or metabolite) from the comparison model can be seen as a set of binary decisions between match or no match for each reaction (or metabolite) in the template model. If the

© The Author 2013. Published by Oxford University Press. All rights reserved. For Permissions, please e-mail: journals.permissions@oup.com

Downloaded from http://bioinformatics.oxfordjournals.org/ at :: on August 30, 2016

Fig. 1. Screenshot of the GUI for comparing reactions

result is 'no match' for all reactions (or metabolites) in the template model, the reaction is declared new. Based on the similarity score, reactions and metabolites can be designated automatically by setting cutoff scores for match and no match. Because declared metabolites can elucidate the proper designation for yet unconsidered reactions, auto-matching proceeds iteratively when initiated.

Sets of previously declared matches and no matches are used to optimize the weighting of the scoring parameters to improve the accuracy of auto-matching. To this end, support vector machine (SVM) learning, a linear and an exponential weighting model are available in modelBorgifier. Via machine learning, the scoring vector is tailored to emphasize only the annotation information in the models that is pertinent to comparison.

2.2 Model merging

When merging two models, modelBorgifier ensures that models can later be retrieved from the composition in a mathematically identical form and indeed carry the same flux. Reaction and metabolite matches (including those that differ only in their compartment location) share annotation information. Original model IDs are retained, but for the purposes of the composite model there is a unique ID list for reactions and metabolites.

Comparison statistics are automatically generated between source models that constitute the composition. This includes completeness of annotations, FBA results and similarity between models in terms of shared reactions and metabolites (both literal and compartment agnostic). Because the composition can contain many models, all cross similarities are calculated.

2.3 Model import and export

All models that can be imported natively into the MATLAB environment via the openCOBRA Toolbox can be read, compared and combined by modelBorgifier. Augmented reconstructions extracted from a composite model can be written to SBML. The merged model itself can be saved as a MATLAB.mat file for future comparisons or modifications. It can also be flattened and treated as a standalone model, but the ability to extract the source models is then lost.

3 IMPLEMENTATION

The modelBorgifier toolbox was created in the MATLAB programming environment. It uses the flux balance analysis functionality from the openCOBRA Toolbox for model validation and gap filling, and the contained MATLAB SBMLToolbox for model import and export. To improve weighting of scoring parameters, SVM, linear or exponential optimization are used alternatively. SVM machine learning relies on the LIBSVM library (Chang and Lin, 2013), and linear and exponential optimization depends on the MATLAB optimization toolbox. The modelBorgifier toolbox requires no installation beyond adding the folder to the MATLAB path. A manual with instructions and examples is available.

The time required to compare models is reduced by modelBorgifier as compared with purely manual method. For instance, on a standard desktop PC, it takes ~4 h to compare iBSU1103 and iJO1366. The time required for model comparison scales linearly with the number of pairwise comparisons (Supplementary Fig. S1). User oversight of the merging can generally be performed in a few hours but depends on the quality of the auto-matching.

4 CONCLUSION

The modelBorgifier toolbox is a script suite that facilitates comparing and combining openCOBRA Toolbox compatible genome-scale metabolic reconstructions. It assists in the arduous task of reconciling dissimilarly annotated models in an efficient and practical manner, combining user oversight, and smart automation. modelBorgifier works simply within COBRA, which already enjoys general ease of use, a wide function selection and an active community (http://opencobra.sourceforge.net/).

By combining models, the user can easily compare as well as share information among models. Comparing two reconstructions for the same organism can be useful for building a more complete model or one more tailored to specific investigations. Comparing reconstructions from different sources for the same organism can elucidate gaps or mistakes in either of the reconstructions. Additional datasets, such as thermodynamic data, can be mapped across multiple reconstructions. The composite model can also serve as a source database of reactions for *in silico* metabolic engineering tools (Zomorrodi *et al.*, 2012).

Funding: German Academic Exchange Service (DAAD) through a RISE professional scholarship (to J.T.S.).

Conflict of Interest: none declared.

REFERENCES

Chang,C.C. and Lin,C.J. (2013) *LIBSVM – A Library for Support Vector Machines*. http://www.csie.ntu.edu.tw/~cjlin/libsvm/.

Henry,C.S. *et al.* (2010) High-throughput generation, optimization and analysis of genome-scale metabolic models. *Nat. Biotechnol.*, **28**, 977–982.

Hucka,M. *et al.* (2003) The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics*, **19**, 524–531.

Krause,F. *et al.* (2010) Annotation and merging of SBML models with semanticSBML. *Bioinformatics*, **26**, 421–422.

Kumar,A. *et al.* (2012) MetRxn: a knowledgebase of metabolites and reactions spanning metabolic models and databases. *BMC Bioinformatics*, **13**, 6.

Oberhardt,M.A. *et al.* (2009) Applications of genome-scale metabolic reconstructions. *Mol. Syst. Biol.*, **5**, 320.

Downloaded from http://bioinformatics.oxfordjournals.org/ at :: on August 30, 2016

Oberhardt,M.A. *et al*. (2011) Reconciliation of genome-scale metabolic reconstructions for comparative systems analysis. *PLoS Comput*. *Biol*., **7**, e1001116.

Randhawa,R. *et al*. (2009) Model aggregation: a building-block approach to creating large macromolecular regulatory networks. *Bioinformatics*, **25**, 3289–3295.

Schellenberger,J. *et al*. (2010) BiGG: a biochemical genetic and genomic knowledgebase of large scale metabolic reconstructions. *BMC Bioinformatics*, **11**, 3213.

Schellenberger,J. *et al*. (2011) Quantitative prediction of cellular metabolism with constraint-based models: the COBRA Toolbox v2.0. *Nat*. *Protoc*., **6**, 1290–1307.

Schulz,M. *et al*. (2012) Propagating semantic information in biochemical network models. *BMC Bioinformatics*, **13**, 18.

Wang,Y.T. *et al*. (2010) PINT: Pathways INtegration Tool. *Nucleic Acids Res*., **38**, W124–W131.

Zomorrodi,A.R. *et al*. (2012) Mathematical optimization applications in metabolic networks. *Metab*. *Eng*., **14**, 672–686.

Downloaded from http://bioinformatics.oxfordjournals.org/ at :: on August 30, 2016