

SBMLsimulator: an efficient Java solver implementation for SBML

Andreas Dräger¹, Alexander Dörr¹, Roland Keller¹, Hannes Planatscher^{1,2}, Michael J. Ziller³, Max Zwießele¹, Philip Stevens¹, Akira Funahashi⁴, Akito Tabira⁴, Mike T. Cooling⁵, Nicolas Rodriguez⁶, Nicolas Le Novère⁶, Andreas Zell¹

¹Center for Bioinformatics Tuebingen (ZBIT), University of Tuebingen, Tübingen, Germany

²Naturwissenschaftlich-Medizinisches Institut, University of Tuebingen, Reutlingen, Germany

³Department of Stem Cell and Regenerative Biology, Harvard University, Cambridge, MA, United States

⁴Keio University, Graduate School of Science and Technology, Yokohama, Japan

⁵Auckland Bioengineering Institute, University of Auckland, Auckland, New Zealand

⁶EMBL European Bioinformatics Institute, Hinxton, United Kingdom

The modeling language SBML [1] (Systems Biology Markup Language) constitutes an important *de facto* standard for the exchange of biochemical network models. SBML defines a set of data structures and provides rules about how to interpret and simulate these kinds of models.

A model stored in SBML may combine an ordinary differential equation system, which is the basis for numerical simulation, with additional elements such as rules and events. These elements further influence the system. For instance, an event takes place if a certain trigger condition becomes true. Whenever this happens, event assignments may change the values of model components, such as parameter values or compartment sizes. Rules can directly assign new values to their objectives, e.g., the concentration of a reacting species.

In many cases, model parameters with uncertain values, such as kinetic constants, need to be estimated using heuristic optimization procedures in order to minimize the distance between model output and experimental measurement data.

For model building, simulation, and parameter calibration processes, dedicated software implementations are indispensable. We here present a simulator that interprets the content of such a model and predicts the dynamic behavior of the model's components. It is based on the Java™ library JSBML [2], a specifically developed data structure to read and write models from and into SBML files and to deal with their structure in memory.

The program SBMLsimulator consists of two parts: Firstly, a generic solver core library, which is completely decoupled from any graphical user interface and can hence easily be integrated as an API (Application Programming Interface) into third-party programs. The core can be downloaded at <http://sourceforge.net/projects/sbml-simulator/> under the terms of the LGPL Version 3. Secondly, a graphical and command-line user interface that provides a connection to the heuristic optimization framework EvA2 [3]. The combination of SBMLsimulator and EvA2 [3] estimates the values of all parameters with respect to given time-series of metabolite or gene expression values. The simulation core will also be integral part of the widely used program CellDesigner version 4.2 [4].

Acknowledgments

Beky Kotcon, Samantha Mesuro, Daniel Rozenfeld, Anak Yodpinyanee, Andres Perez, Eric Doi, Richard Mehlinger, Steven Ehrlich, Martin Hunt, George Tucker, Peter Scherpelz, Aaron Becker, Eric Harley, and Chris Moore, Harvey Mudd College, Claremont, California, USA.

References

- [1] Hucka, Finney, Sauro et al. The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics* 19 (4): 524–31, 2003.
- [2] Dräger, Rodriguez, Dumousseau, Dörr, Wrzodek, Le Novère, Zell, Hucka. JSBML: a flexible Java library for working with SBML, *Bioinformatics*, 2011.
- [3] Kronfeld, Planatscher, Zell. The EvA2 optimization framework. Learning and Intelligent Optimization Conference, Special Session on Software for Optimization, in *Lecture Notes in Computer Science, LNCS*, pp. 247-250, 2010. Springer-Verlag.
- [4] Funahashi, Tanimura, Morohashi, Kitano. CellDesigner: a process diagram editor for gene-regulatory and biochemical networks, *BIOSILICO*, 1:159-162, 2003.