

# sbml-diff: A Tool for Visually Comparing SBML Models in Synthetic Biology

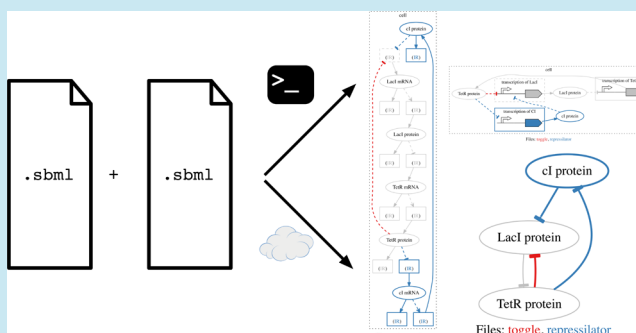
James Scott-Brown\* and Antonis Papachristodoulou

Department of Engineering Science, University of Oxford, Parks Road, Oxford OX1 3PJ, U.K.

## Supporting Information

**ABSTRACT:** We present *sbml-diff*, a tool that is able to read a model of a biochemical reaction network in SBML format and produce a range of diagrams showing different levels of detail. Each diagram type can be used to visualize a single model or to visually compare two or more models. The default view depicts species as ellipses, reactions as rectangles, rules as parallelograms, and events as diamonds. A cartoon view replaces the symbols used for reactions on the basis of the associated Systems Biology Ontology terms. An abstract view represents species as ellipses and draws edges between them to indicate whether a species increases or decreases the production or degradation of another species. *sbml-diff* is freely licensed under the three-clause BSD license and can be downloaded from <https://github.com/jamescottbrown/sbml-diff> and used as a python package called from other software, as a free-standing command-line application, or online using the form at <http://sysos.eng.ox.ac.uk/tebio/upload>

**KEYWORDS:** SBML, synthetic biology, visualization, comparison, version control



In order to predict how synthetic biological circuits will function or to determine how parameters should be tuned, it is necessary to use mathematical models. Ideally, these models would be expressed in a standardized data format that provides interoperability between software packages, so that models can be constructed and simulated using different software, the results of simulating the same model using different simulation packages can be compared, and researchers can use their preferred tools to reproduce or extend previous work that used different tools. One such standardized format is the Systems Biology Markup Language (SBML),<sup>1</sup> a free and open interchange format for computer models of biological processes based on the Extensible Markup Language (XML). An SBML model consists of species that participate in reactions. Additionally, an event may reset the values of parameters and concentrations when a particular trigger expression is satisfied. A rule may specify that the value (in the case of an AssignmentRule) or rate of change (in the case of a RateRule) of a parameter or concentration is given by a particular mathematical expression or that a particular function of parameters and/or concentrations must always equal zero (in the case of an AlgebraicRule).

The ability to compare SBML models is important, both to compare models of different systems and to compare different versions of the same model. Many other engineering fields, particularly software engineering, rely heavily on systems of version control to keep track of designs produced at each stage of an iterative design cycle. This is often accompanied by the use of file differencing (*diff*) tools to directly compare different

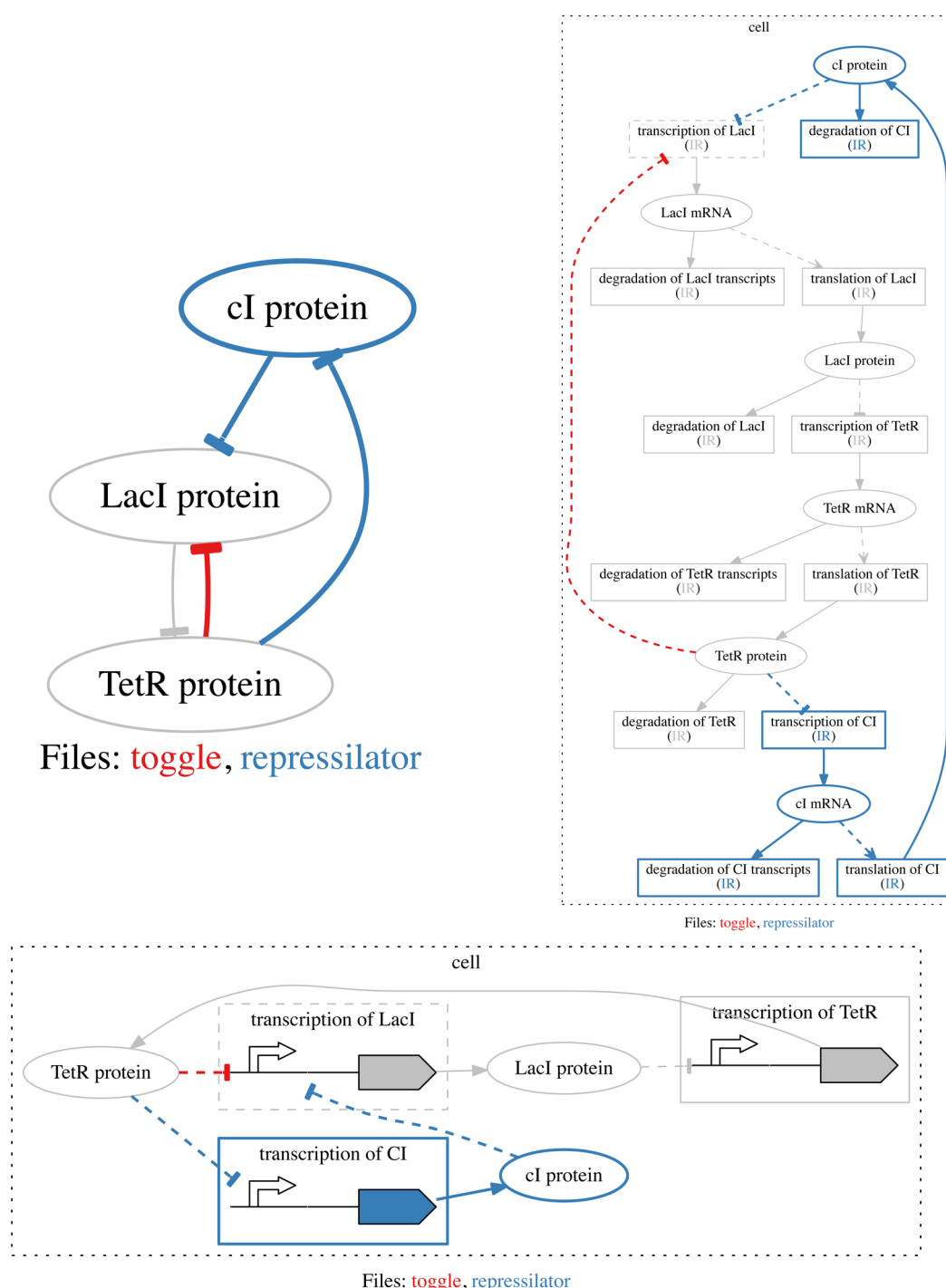
versions and identify what changes were made. However, directly comparing two models in SBML format as text is unsatisfactory: *diff* can be used, but it is difficult to spot the salient features in its output. Also, many textual changes are not significant (e.g., changes in whitespace or the ordering of elements), and if the *id* of a species is changed, this change will appear in many places (e.g., the list of reactants, list of products, and *kineticLaw* for reactions involving that species).

A visual comparison is able to put a change into context, showing not only what has changed but also how this relates to the other, unchanged, components. While it is possible to compare a set of models by simply juxtaposing independently created visualizations of each, this requires the user to “spot the differences” between diagrams in which equivalent nodes may be in different spatial positions. In this paper, we present *sbml-diff*, a tool that reads SBML models of biochemical networks and produces a range of diagrams showing different levels of detail. *sbml-diff* allows visual comparisons using a single diagram and can quickly highlight changes made between versions of a single model, make it easy to identify when a particular change was made, or check that no unintended changes were made at the same time as an intentional edit. It can also be used to check that a model has not been

**Special Issue:** IWBD 2016

**Received:** September 30, 2016

**Published:** December 30, 2016

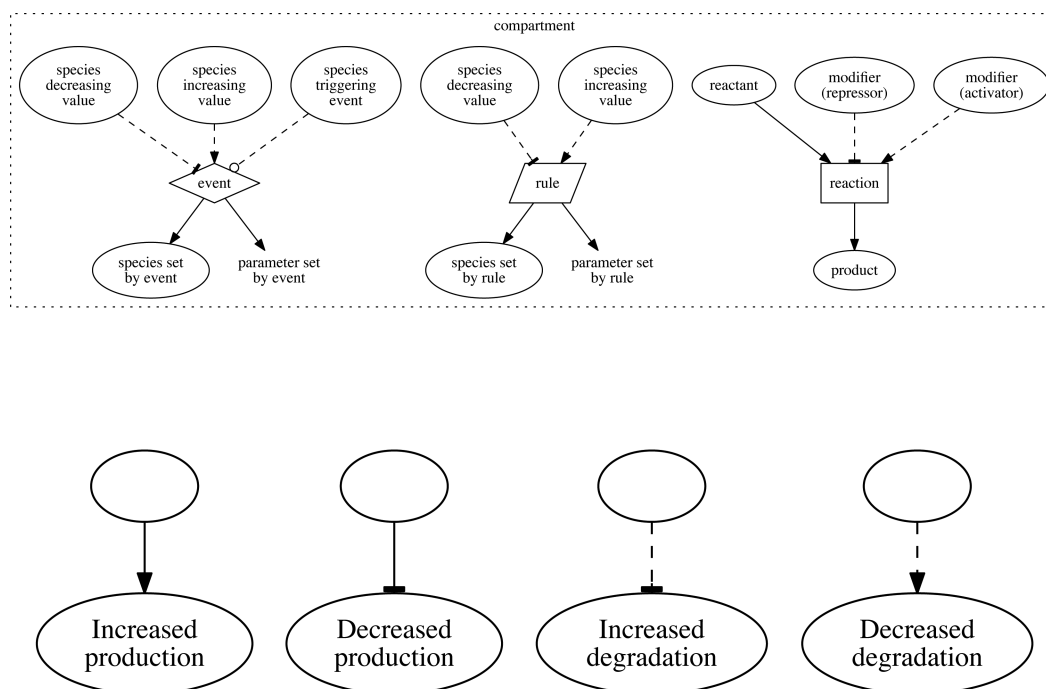


**Figure 1.** Three examples of the comparison between two models, representing the repressilator<sup>14</sup> and the toggle switch.<sup>15</sup>

accidentally altered by the process of importing and then re-exporting from another tool or converting back and forth between file formats.

Importantly, *sbml-diff* can also be used to compare models corresponding to different synthetic circuit designs and visually show the difference between them. The most developed standard for biological designs is the Synthetic Biology Open Language (SBOL),<sup>2</sup> which represents designs as being composed of components with sequences and sequence annotations. Tools exist that use SBOL designs to automatically generate the corresponding SBML models,<sup>3</sup> which could be compared using *sbml-diff*.

*sbml-diff* can also produce a diagram of a *single model*. Such a diagram is particularly useful in understanding the structure of a model produced by another researcher. Visualization can also reveal structural features that are likely to correspond to errors. For example, any parameters that are set by a rule but not used elsewhere in the model become obvious; these may indicate that the wrong parameter is used or set somewhere in the model or that the model has been changed and the parameter is no longer needed.



**Figure 2.** Meaning of each graphical element in the (top) default and (bottom) abstract views. The meaning of the lines differs between the two diagram types, but this should not cause confusion, as in the default diagram they join species and nonspecies and in the abstract diagram they join species to species.

## EXISTING TOOLS

There are some existing tools that are capable of visually representing a single SBML model as a bipartite reaction graph. These include large GUI packages with many other features (e.g., CellDesigner<sup>4</sup> and iBioSim<sup>5</sup>), for which the visualization component is difficult to script or incorporate into other tools, and a few freestanding tools that can generate DOT output (e.g., The Systems Biology Format Converter's SBML2DOT Java module<sup>6</sup> and the python library VisualizeSBML<sup>7</sup>). Several of these ignore rules and events or otherwise do not visualize all of the elements of an SBML model.

Cy3sbml<sup>8</sup> provides the ability to import an SBML file into Cytoscape,<sup>9</sup> creating a network in which each component is represented by a node. However, while a user could manipulate this network in Cytoscape to produce a diagram like those produced by sbml-diff, cy3sbml cannot produce such diagrams automatically.

BiVeS<sup>10</sup> is a tool for comparing SBML models that is intended to track changes in a model over time. It produces outputs in a variety of formats, but its visualization abilities are currently relatively limited: work is underway to produce a companion tool to produce diagrams in the Systems Biology Graphical Notation Process Description (SGBN PD) language.<sup>11</sup> This would visualize the underlying biochemistry rather than the structure of the mathematical model and thus be complementary to sbml-diff rather than competing with it.

## IMPLEMENTATION

sbml-diff reads models in SBML format and produces output in DOT format, which can be converted into an image by GraphViz<sup>12</sup> or other compatible software. It can be used as a

python package, as a freestanding command-line tool, or through a form on our Web site.

By default, elements in two models are treated as the same entity if they have the same *id* attribute. Optionally, two elements with different *id* attributes can be treated as the same entity if they have the same set of MIRIAM<sup>13</sup> annotations of type *is*: in this case, the *id* of one is changed to be the same as the other, and all of the reactions/rules/events are updated accordingly.

Coloring is used to indicate whether each node and edge are common to all models (gray), some but not all models (black), or a single model (the color specific to that model); the default colors were chosen to be distinguishable by color-blind users. A dashed node edge indicates that a component is shared between models but with differences in its attributes: a rectangle with a dashed border indicates a reaction for which not all models have the same *kineticLaw*; an ellipse with a dashed double border indicates a species for which not all models have the same *isBoundary* attribute (Figure 1).

**Default View.** In the default view, there is a direct mapping between the structure of an SBML model and the visual elements in the diagram. This view is intended to directly represent the structure of the mathematical model represented in SBML rather than the structure of the reaction network that is being modeled, which may be modeled in several different ways. For example, a reaction may be modeled in SBML using a reaction with the attribute *fast* set to *true* or by an AssignmentRule. We therefore map components directly onto symbols using the mapping shown in Figure 2.

The key SBML components are compartments (drawn as dashed rectangles), species (ellipses, drawn with a double border if *isBoundary* is *true*), parameters (labels), rules (parallelograms), reactions (rectangles), and events (diamonds).

For clarity, we depict only those parameters whose values are set by rules or events. However, the effect of every parameter can be seen if reaction nodes are labeled with their rate laws. Function definitions are not shown, as we substitute them into any expression that uses them. Units and `initialAssignment` components are not shown, as these concern the values of parameters rather than the structure of the model.

Components are joined by arrows: if a line is solid, it indicates that a species' concentration (or a parameter's value) is affected by an event, rule, or reaction; if a line is dashed, it shows that a species affects an event, rule, or reaction in some way. Normal arrowheads indicate activation, and T-shaped arrowheads indicate repression (the arrowhead direction is determined numerically—see the Supporting Information). In the case of an `AlgebraicRule`, the rule is linked to each species appearing in the corresponding mathematical expression by an edge with no arrowhead.

An option allows the user to adjust the level of detail by choosing between labeling of the reaction nodes with the corresponding reaction name, `kineticLaw`, both, or neither. If a reaction has the attribute `fast` set to `true`, this is indicated by an "F", and if a reaction has the attribute `reversible` set to `false`, this is indicated by "IR" (for `IRreversible`); when models are compared, these two markers are individually colored using the same rule as other visual elements.

**Abstract View.** The abstract view does not represent reactions using nodes but instead draws edges directly between species to indicate interactions between them. For each reaction, an edge is added from each species appearing in the `kineticLaw` to each reactant and product. Edges corresponding to a species increasing the rate of its own degradation are hidden, as these would typically be present for all species and create clutter.

These interactions are categorized into four types, distinguished visually by two styles of arrowhead (arrow or T-shaped, indicating the *sign* of the interaction) and two styles of line (solid or dashed, indicating whether the interaction affects the rate of production or degradation).

`sbml-diff` provides the option to elide a list of species (e.g., intermediate mRNAs). These species are not drawn, and if they increase the production of a second species, then the heads of arrows incident to them are moved to that species (i.e., if  $A \rightarrow B \rightarrow C$  and  $B$  is elided, then  $B$  is not drawn and instead an arrow is drawn directly from  $A$  to  $C$ ).

**Cartoon View.** The cartoon view resembles the default view but replaces some rectangular reaction nodes with symbols indicating the nature of the reaction. Specifically, any reaction annotated with a Systems Biology Ontology<sup>16</sup> term of transcription is replaced by a compound symbol containing the symbol for a promoter and a coding sequence for each product. Reactions of type translation are automatically elided unless the intermediate mRNA participates in a reaction that is not annotated as translation or degradation.

## CONCLUSION

`sbml-diff` is freely licensed under the three-clause BSD license and can be downloaded from <https://github.com/jamesscottbrown/sbml-diff> and used as a free-standing command-line application or online using the form at <http://sysos.eng.ox.ac.uk/tebio/upload>. It can also be used as a python package, allowing it to be incorporated into larger software packages, such as tools for editing and curating collections of

models, or incorporated into automated tests to ensure that other tools do not contain bugs that cause unintended changes to SBML files.

## ASSOCIATED CONTENT

### Supporting Information

The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acssynbio.6b00273.

Additional details of how arrow directions are determined (PDF)

Models used to produce Figure 1 (ZIP)

## AUTHOR INFORMATION

### Corresponding Author

\*E-mail: [james.scott-brown@eng.ox.ac.uk](mailto:james.scott-brown@eng.ox.ac.uk).

### ORCID

James Scott-Brown: 0000-0001-5642-8346

### Notes

The authors declare no competing financial interest.

## ACKNOWLEDGMENTS

J.S.-B. acknowledges funding through the EPSRC & BBSRC Centre for Doctoral Training in Synthetic Biology (EP/L016494/1) and from DSTL. A.P. acknowledges support from the EPSRC (Project EP/M002454/1).

## REFERENCES

- (1) Hucka, M., Finney, A., Sauro, H. M., Bolouri, H., Doyle, J. C., Kitano, H., et al. (2003) The systems biology markup language (SBML): A medium for representation and exchange of biochemical network models. *Bioinformatics* 19, 524–531.
- (2) Roehner, N., Beal, J., Clancy, K., Bartley, B., Misirli, G., Grünberg, R., Oberortner, E., Pocock, M., Bissell, M., Madsen, C., Nguyen, T., Zhang, M., Zhang, Z., Zundel, Z., Densmore, D., Gennari, J. H., Wipat, A., Sauro, H. M., and Myers, C. J. (2016) Sharing Structure and Function in Biological Design with SBOL 2.0. *ACS Synth. Biol.* 5, 498–506.
- (3) Roehner, N., Zhang, Z., Nguyen, T., and Myers, C. J. (2015) Generating Systems Biology Markup Language Models from the Synthetic Biology Open Language. *ACS Synth. Biol.* 4, 873–879.
- (4) Funahashi, A., Morohashi, M., Kitano, H., and Tanimura, N. (2003) CellDesigner: a process diagram editor for gene-regulatory and biochemical networks. *BIOSILICO* 1, 159–162.
- (5) Myers, C. J., Barker, N., Jones, K., Kuwahara, H., Madsen, C., and Nguyen, N.-P. D. (2009) iBioSim. *Bioinformatics* 25, 2848–2849.
- (6) Rodriguez, N., Pettit, J.-B., Dalle Pezze, P., Li, L., Henry, A., van Iersel, P. M., Jalowicki, G., Kutmon, M., Natarajan, K. N., Tolnay, D., Stefan, I. M., Evelo, C. T., and Le Novère, N. (2016) The systems biology format converter. *BMC Bioinf.* 17, 154.
- (7) Gillespie, C. S., Wilkinson, D. J., Proctor, C. J., Shanley, D. P., Boys, R. J., and Kirkwood, T. B. L. (2006) Tools for the SBML Community. *Bioinformatics* 22, 628–629.
- (8) König, M., Dräger, A., and Holzhutter, H.-G. (2012) CySBML: a Cytoscape plugin for SBML. *Bioinformatics* 28, 2402–2403.
- (9) Shannon, P. (2003) Cytoscape: A Software Environment for Integrated Models of Biomolecular Interaction Networks. *Genome Res.* 13, 2498–2504.
- (10) Scharm, M., Wolkenhauer, O., and Waltemath, D. (2016) An algorithm to detect and communicate the differences in computational models describing biological systems. *Bioinformatics* 32, 563–570.
- (11) Moodie, S., Le Novère, N., Demir, E., Mi, H., and Villéger, A. (2015) Systems Biology Graphical Notation: Process Description Language Level 1 Version 1.3. *J. Integr. Bioinf.* 12, 263.

- (12) Gansner, E. R., and North, S. C. (2000) An open graph visualization system and its applications to software engineering. *Software Pract. Exper.* 30, 1203–1233.
- (13) Laibe, C., and Le Novère, N. (2007) MIRIAM Resources: tools to generate and resolve robust cross-references in Systems Biology. *BMC Syst. Biol.* 1, 58.
- (14) Elowitz, M. B., and Leibler, S. (2000) A synthetic oscillatory network of transcriptional regulators. *Nature* 403, 335.
- (15) Gardner, T. S., Cantor, C. R., and Collins, J. J. (2000) Construction of a genetic toggle switch in *Escherichia coli*. *Nature* 403, 339.
- (16) Le Novère, N. (2006) Model storage, exchange and integration. *BMC Neurosci.* 7, S11.