Identifying, Interpreting, and Communicating Changes in XML-encoded Models of Biological Systems

Martin Scharm[1,⋆], Olaf Wolkenhauer[1,2], and Dagmar Waltemath[1]

[1] Department of Systems Biology and Bioinformatics, University of Rostock, Rostock, Germany
[2] Stellenbosch Institute for Advanced Study (STIAS), Wallenberg Research Centre at Stellenbosch University, Stellenbosch 7600, South Africa

Background

Research in systems biology enhanced our knowledge of biological environments. Many discoveries are recorded in computational models which encode the structure of biological networks, and describe their temporal and spatial behavior. Due to tremendous efforts by the research community, the number of openly available models is numerous and still continually increasing [1]. To support the sharing of models and, thus, the reuse of research results, repositories such as the BioModels Database [2] and the Cellml Model Repository [3] collect and store models in exchangeable formats such as the Systems Biology Markup Language (SBML, [4]), or CellML [5]. Since only accessible models can be reused, such repositories are essential to guarantee transparent research.

However, model repositories to date lack sufficient mechanisms to track the updates of models in their databases [6]. Model versions often cannot be addressed unambiguously and changes occurring between versions of a model are not communicated transparently. Therefore, a framework to identify the differences between models and their versions is a fundamental requirement to compare and combine models. Only with difference detection at hand users are able to grasp a model's history and to identify errors and inconsistencies.

Results

On the poster, we reflect on the following requirements for systems that provide version control for models:

- All versions of a model must be accessible.
- Information must be available on when a model changed, how, why, and by whom.
- Changes in model versions must be made transparent to the user.

⋆ To whom correspondence should be addressed

Our current research concentrates on developing efficient and reliable difference detection for versions of models. We thereby address the abovementioned requirement that information must be available on *how* a model changed over time. Specificially, our algorithm for difference detection, BiVeS[3], is applicable to models encoded in SBML or CellML. As standard representation formats for computational models in biology use XML, BiVeS bases on an XML-diff algorithm, namely the XyDiff algorithm [7]. BiVeS identifies structures in the XML trees that both documents have in common and maps their subgraphs onto each other. The resulting mapping is then propagated into the rest of the tree, possibly leading to further mappings. That way, moved entities can be identified, as well as inserts and deletes. The algorithm is furthermore format-specific in the sense that it respects the structure of the representation formats. The major elements of the SBML Level 3 specification [8], for example, are biological entitities (species) that participate in biological processes (reactions). CellML very generally encodes biological facts as sets of interacting components. Both representation formats use semantic annotations (i. e., links to ontologies) to further describe the biological meaning of the single XML elements [9]. We use this information to further improve the mappings. The final set of of differences can be exported in both machine and human readable formats: BiVeS produces an XML-encoded patch containing all modifications which occurred between the two versions of a document (see Figure 1). Changes between model versions are

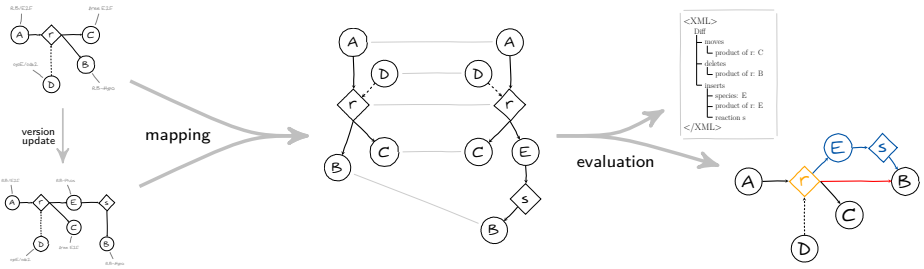

Fig. 1. A sketch of the BiVeS algorithm for difference detection.

also summarized in a report and highlighted in a graph, which comprehensively displays the updates affecting the reaction network. The algorithm is implemented in a Java library.

Gaining insights into the process of development of a particular model has the potential to increase the confidence in this model and supports the collaboration of distinct research projects dramatically. Consequently, existing model repositories can benefit from extending their software and functionalities with version control. On our poster, we show how the BiVeS library can be integrated

[3] Biomodel Version Control System, `https://sems.uni-rostock.de/projects/bives/`

with existing software: (i) BiVeS offers an API that can be used from other Java tools, (ii) a web service provides access to BiVeS via HTTP, (iii) the library can be executed directly from the command line. BiVeS is already implemented in the Functional Curation project of Chaste [10]. Furthermore, we are currently in touch with the maintainers of SEEK, a data management platform for the life sciences [11], BioModels Database, and the CellML model repository to integrate BiVeS into their infrastructures. On the poster, we demonstrate BiVeS' capabilities with our prototypic web based user interface BudHat[4]. BudHat uses BiVeS to detect changes between versions of a model stored in a database backend. Identified differences are processed and presented human readably. Changes in reaction networks, for example, are highlighted in different colors. An example is shown in Figure 2.

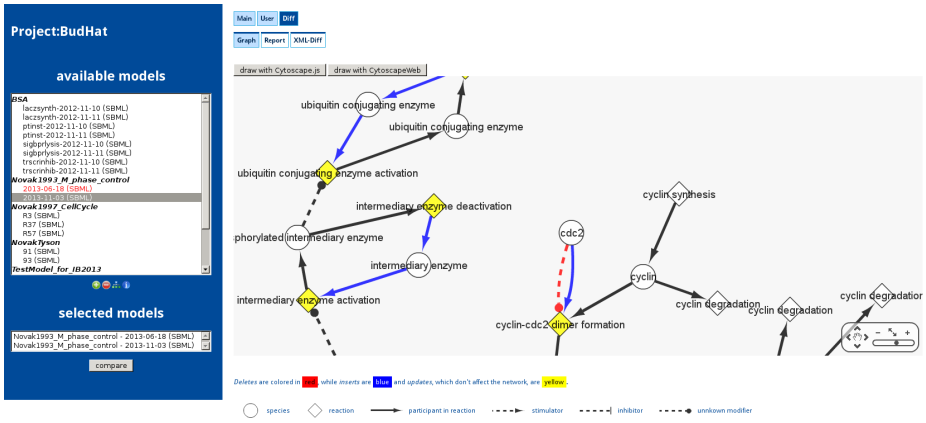

Fig. 2. Screenshot of our prototype BudHat. BudHat is an online tool that displays the differences between model versions, as computed by BiVeS, in some human readable formats (here: highlighted reaction network).

Finally, we discuss first statistics about the evolution of computational models in open repositories. We analysed models from the BioModels Database (144,253 models in SBML format) and the CellML Model Repository (600 different exposures with CellML models). Indeed, models in open repositories do change over time predominantly in two ways: First, models are modified if the representation format, used to encode the model, gets updated. These updates affect a large number of models and form a clear pattern in our visualisation of model changes. For example, all models in BioModels Database where updated when SBML replaced its own standard for links to external resources, MIRIAM, by the identifiers.org scheme [12]. Second, published models are continuously improved and corrected by model curators. We observed updates in the links pointing to terms in bio-ontologies, and to the model's network structure. For

[4] `http://budhat.sems.uni-rostock.de`

the Repressilator model[5], for example, we see that the change in network structure actually affected the simulation outcome. We also identified patterns in the CellML Model Repository, and will discuss possible reasons on our poster. With respect to performance, we used the above data sets to compare our own algorithm for difference detection against the standard Unix diff tool. Unix diff to date is the standard method to compare versions of models in open repositories. However, our results confirm that BiVeS indeed outperforms Unix' diff tool and improves the results obtained by standard XML Diff tools.

Summary

In summary, our poster introduces ongoing research in model management for computational biology, with a focus on the advantages of sophisticated model version control. We discuss in detail the requirements, show our latest research results in terms of algorithm design and tool support, and we present first statistics on the types and frequency of changes in models published in open repositories.

References

1. Henkel *et al.*: Ranked retrieval of computational biology models. *BMC bioinformatics*, 2010.
2. Li *et al.*: BioModels Database: An enhanced, curated and annotated resource for published quantitative kinetic models. *BMC Systems Biology*, 2010.
3. Lloyd *et al.*: The CellML Model Repository. *Bioinformatics*, 2008.
4. Hucka *et al.*: The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models.. *Bioinformatics* 19.4:524-531, 2003.
5. Cuellar *et al.*: An overview of CellML 1.1, a biological model description language. *Simulation* 79.12, 2003.
6. Waltemath *et al.*: Improving the reuse of computational models through version control. *Bioinformatics* 29.6:742-748, 2013.
7. Cobena *et al.*: Detecting changes in XML documents. 18th International Conference on Data Engineering, 2002.
8. Hucka *et al.*: The systems biology markup language (SBML): language specification for level 3 version 1 Core (Release 1 Candidate). *Nature proceedings*, 2010.
9. Courtot *et al.*: Controlled vocabularies and semantics in systems biology. *Molecular systems biology* 7.1, 2011.
10. Cooper *et al.*: High-throughput functional curation of cellular electrophysiology models. *Progress in Biophysics and Molecular Biology*, 2011.
11. Wolstencroft *et al.*: The SEEK: a platform for sharing data and models in systems biology. *Methods Enzymol* 500:629-655, 2011.
12. Juty *et al.*: Identifiers. org and MIRIAM Registry: community resources to provide persistent identification.. *Nucleic acids research* 40.D1:D580-D586, 2012.

[5] `http://www.ebi.ac.uk/biomodels-main/BIOMD0000000012`