Finding Pattern in Biochemical Reaction Networks

Ron Henkel[1],[2], Fabienne Lambusch[1] and Dagmar Waltemath[1]
[1]*Systems Biology and Bioinformatics, University of Rostock, Rostock, Germany*
[2]*Scientific Databases and Visualization, Heidelberg Institute for Theoretical Studies, Heidelberg, Germany*
{*ron.henkel, dagmar.waltemath, fabienne.lambusch*}*@uni-rostock.de*

Keywords: Systems Biology, Subgraph Mining, Knowledge Discovery, Graph Database, Biochemical Reaction Networks, Pattern Detection

Abstract: Biological questions today are often answered with the help of simulation models. Many of these models encode biological processes as biochemical reaction networks. The increasing amount of published models and the growing size of encoded reaction networks demand methods to analyse models. Specifically, researchers need to identify reoccurring and biologically relevant patterns. However, pattern recognition in large networks is a hard problem, and only partial solutions for very specific biological networks exist until now. In addition, while such patterns where already postulated, identifying them manually is barley feasible given a large set of complex models. This paper examines automatic methods to find reoccurring patterns in models represented as bipartite graphs. An approach is presented to find the most frequent structures within the models. Appropriate patterns were found, which occur in a major part of the 575 input models. The occurrences of the resulting structures can provide insight into the encoding of certain biological processes, evaluate the postulated structures and serve as a reasonable similarity measure for grouping models that share many common structures.

1 INTRODUCTION

Modeling has become an integral tool for research in computational biology (Finkelstein et al., 2004). In the field of Systems Biology, models are mostly distributed in standard formats, e. g., the Systems Biology Markup Language (SBML) (Hucka et al., 2010) or CellML (Lloyd et al., 2004). Model repositories such as the BioModels Database (Li et al., 2010), the CellML model repository (Yu et al., 2011), or JWS Online (Olivier and Snoep, 2004) offer to the community valuable, curated, and reusable models describing biological systems. This enables researchers to study biological systems without necessarily implementing the models from scratch, thereby saving time, effort and money.

The increasing impact of modeling for biology is reflected in the rapidly growing number and complexity of computational models of biological systems (Henkel et al., 2010; Li et al., 2010) and in the large number of computational tools for simulation, analysis, visualization, or comparison (Hucka et al., 2011). Current modeling projects such as the Virtual Physiological Humanrequire the usage of techniques for model coupling, merging, and combination at different scales. Computational support is needed to curate models (i. e., to manually validate and semantically annotate them). As models evolve over time, good management strategies are needed to ensure model exchangeability, stability and result validity, and to foster communication between project partners (Bechhofer et al., 2013; Waltemath et al., 2013; Henkel et al., 2015). Open model repositories help with model management. BioModels Database currently offers 575 curated, SBML-encoded models describing a variety of biological processes, such as cell cycle, apoptosis or mitogen-activated protein kinase, encoded as biochemical reaction networks (Juty et al., 2015).

With this rich resource of model code at hand, it is now interesting to analyse the function, structure and behavior of models (Knüpfer et al., 2013). For example, Tyson and Novák (2010) postulated common functional motifs in biochemical reaction networks. However, it remains open if and how such motifs would be encoded in model's biochemical reaction networks; or if the model encoding differs from the biological point of view. A prerequisite to validating Tyson's results, computational methods for pattern discovery in computational models are needed.

In this manuscript we describe the available data and its structure in the field. We then evaluate techniques for pattern discovery. Finally, we show that subgraph mining is a suitable method for pattern discovery and explain the revealed patterns.

2 MOTIVATION

Many of today's large biochemical reaction networks are semi-automatically being generated using data driven approaches (Smallbone et al., 2013). These networks mostly focus on network diameter and network efficiency Zhang and Zhang (2009), on the topological and dynamical properties that control the behaviour of the network (Barabasi and Oltvai, 2004), or on the degree of tolerance against errors in scale-free networks (Albert et al., 2000). While these approaches provide key figure values describing the network topology, they do not detect actual patterns. These substructures in networks are, however, necessary for modelers to determine reoccurring parts in models, or to characterise typical submodules that may help identifying biological phenomena, for example in model comparison tasks. One could, for example, ask:

- What are the common structures to encode a simple biochemical reaction?
- Does the network contain circles and how many?
- Can we find unique pattern for certain types of models (i.e. models derived from wet lab data or theoretical models encoding a postulated network to show a certain behavior)?
- Do models of the biological domain (cell cycle, apoptosis) share certain patterns?
- Can we find patterns postulated by Tyson and Novák (2010) and are they used often rather than occasionally?

Obtaining such information offers a variety of use cases. For example, determining if a model was created by a theoretical, data driven, or hybrid approach. Furthermore it would be possible to cluster models by occurrence of pattern in their networks, instead of using meta information (Alm et al., 2014), and infer an affiliation to a biological domain. Ultimately it becomes possible to calculate a similarity coefficient for models purely based on their network's structure. Combined with already existing model similarity measures (Henkel et al., 2010; Schulz et al., 2011) this will have an impact on the reuse and reproducibility of scientific results created by modeling (Waltemath et al., 2013; Henkel et al., 2015; Bechhofer et al., 2013).

To create a reasonable similarity measure for the models' networks represented as graphs, it is essential to regard the network structure as a whole, rather than treating them as a set of nodes and edges. Lakshmi and Meyyappan (2012) state that the simple pairwise comparison of nodes and edges within a network neglects its structure, whereas it is possible to respect the composition of network elements by viewing the graphs as similar, if they share many common substructures. Consequentially, the problem of detecting structural similarities within the models is defined as a frequent subgraph mining (FSM) task (Kuramochi and Karypis, 2001).

On the basis of the results, very useful applications can be implemented. For example, a function to search structural similar models could be established as well as a recommender system that suggests suitable structures, while a researcher is modelling a process.

3 STATE OF THE ART

A difficulty is that FSM algorithms require subgraph isomorphism testing - the problem to decide whether a graph is embedded in another (Lakshmi and Meyyappan, 2012). This is known as an NP-complete task (Keyvanpour and Azizani, 2012). Thus, FSM techniques rely on heuristics, prior knowledge or other particular strategies to improve the performance. A variety of FSM algorithms are available (Kuramochi and Karypis, 2001), mostly adapted and refined to serve particular purposes. When choosing an appropriate algorithm for a problem, the characteristic aspects of the methods need to be evaluated. These aspects include the type of input graph, the necessity of prior background knowledge, the need for exact or just approximate results as well as for completeness of the resulting pattern set, the available memory, and the possibility of user intervention.

FSM algorithms can be differentiated, for example, according to their input type (Keyvanpour and Azizani, 2012). Some algorithms take one large graph and find the frequent subgraphs depending on the frequency within this graph. Other algorithms take a graph set as their input and search for structures that occur in at least a certain number of graphs within the set.

The Kyoto Encyclopedia of Genes and Genomes (KEGG (Kanehisa et al., 2004)) is a widely used pathway database. KEGG already determines structural similarities of network components (Koyutürk et al., 2004; Hattori et al., 2003). Koyutürk et al. (2004) search for frequent subgraphs within a set of

metabolic pathways in the KEGG database, where the pathways are represented as directed graphs with unique node labellings. The authors state that their approach is also applicable to various other biological networks with only minor modifications at the most. They reduce the computational costs by exploiting the sparse nature of metabolic pathways and the unique node labelling. Their approach discovers common patterns of related enzyme interactions.

Hattori et al. (2003) describe an algorithm to compare chemical structures of compounds. The chemical structure is seen as a graph of atoms connected by covalent bonds as edges. The developed algorithm identifies and clusters mostly metabolic compounds.

Wong et al. (2011) find frequent occurring patterns within biological networks and investigate correlations between the functional behaviour of such patterns with their structural topology. The authors present several existing algorithms for this purpose. The algorithms are evaluated by experimental results, classified according to several characteristics, and their advantages and disadvantages are discussed.

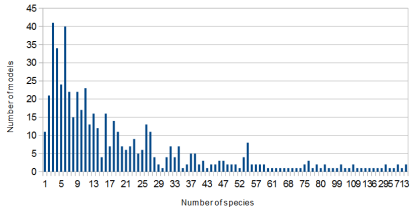

Figure 1: This figure shows the distribution of species among the models. On the x-axis the number of models containing a particular number of species (y-axis) is presented.

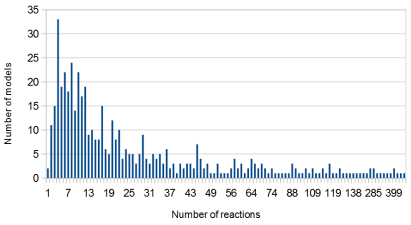

Figure 2: This figure shows the distribution of reactions among the models. On the x-axis the number of models containing a particular number of reactions (y-axis) is presented.

4 DATA SET

For our analysis, we incorporated all publicly available models from BioModels Database. The stored reaction networks are encoded in an XML-based, standard model representation format, SBML (Hucka et al., 2010). BioModels Database contains two types of models: Curated and non-curated. We here choose only models from the curated branch as those models are ensured to reproduce the results described in their accompanying publication accurately. Furthermore, curated models are syntactically validated and annotated with ontology terms according to the MIRIAM standard (Novere et al., 2005). SBML encodes biochemical reaction networks using species and reactions. A species participates in a reaction as a modifier, product or reactant. Consequently, for our analysis we have two types of nodes (labeled species and reaction) and three types of edges (labeled is_reactant, has_product, is_modifier). We here analyze Release 29 of BioModels Database (in the following referred to as R29) containing 575 curated models and, in addition, compare the results to BioModels first release containing only 30 curated models (in the following referred to as R1). First, we perform a key figure analysis. The aim of this analysis is to get quantities of the nodes for models, reactions and species as well as the edges between them. 557 out of 575 models have species and 499 have reactions, respectively. The remaining models only define species and rate rules, but do not represent a network. They are thus neglected in further analyses. Each reaction or species belongs to exactly one SBML-model. There exist 18852 reaction nodes and 16843 species nodes in total. Compared with the first release (R1), the rapid growth of models becomes obvious, as previously reported by henkel2010ranked. Data set R1 contains only 30 curated models having 736 reactions and 425 species, respectively. For R29, Figure 1 shows the distribution of species, and Figure 2 shows the distribution of reactions among the models. Most models contain less then 20 species and reactions. A noticeable accumulation of models can be found from three up to eleven species, while there are just a few models with more than 60 Species. For reactions, an accumulation of models that have three up to twelve reactions is stated. Furthermore, there are a few outliers with more than 100 reactions and species. On average, a model has 30.2 species and 37.7 reactions. For R1 (results not displayed) a model has 14.6 species and 25.4 reactions on average.

Figure 3 shows the distribution of interaction classes among the models for data set R29. An interaction class is a combination of species (reactants, products and modifier) connected to a reaction. As the figure states, most reactions have two or three participating species. The most encoded interaction class is a reaction having two species as reactants and one species as product, followed by the interaction class having a reaction with one species as reactant and one

as product. Also notable are the interaction classes for seven and more participating species, here 136 different interaction classes are encoded.

5 ALGORITHM AND METHODS

Frequent subgraph mining (FSM) is capable of detecting structural similarities of networks (Kuramochi and Karypis, 2001). One important characteristic of this approach is the candidate generation method, which mainly builds on four bases – join, extension, inductive logic programming and replacing. When generating candidate with join, the algorithm starts with small frequent substructures and then merges them into bigger structures where frequent ones in turn can be joined. Extension based methods start with frequent nodes and iteratively add one of each possible edges, while infrequent patterns often are pruned immediately and will not be observed for further extension. By using inductive logic programming, first order predicates represent the subgraphs. Keyvanpour and Azizani (2012) state that in the replacing strategy "[...] after detecting the frequent subgraph in each stage, the detected subgraph is replaced by a node in the main graph and in the next stage, the mining process continues on a new graph obtained from graph replacing."

GSpan is an extension based algorithm that takes a graph set as its input and produces all frequent connected subgraphs according to a given frequency threshold (Yan and Han, 2002). Therefore, it uses a unique minimum depth-first search code of the graphs and a lexicographic ordering on these codes. Given that, gSpan builds a search tree. As it uses the minimum depth-first search code of graphs as a canonical label, two graphs are isomorphic if and only if their code is equal. This fact transforms the task into a sequential pattern mining problem, with already existing solving algorithms for this problem. Furthermore, gSpan accelerates discovering patterns by combining candidate generation and frequency counting, while efficient pruning is performed. It also avoids false positive pruning. For example, Priyadarshini and Mishra (2010) describe an approach to graph mining using the gSpan, while Wörlein et al. (2005) evaluated its performance in comparison to the algorithms MoFa, FFSM and Gaston. For this purpose, Wörlein et al. (2005) developed a tool called the Parallel and Sequential Mining Suite (ParSeMiS). ParSeMiS is based on Java and implements algorithms such as gSpan, Gaston, and Dagma. The above mentioned advantages of gSpan, such as the use of a canonical labelling and its availability in ParSeMiS offers the opportunity to analyze the biochemical reaction networks described in the Data Set section.

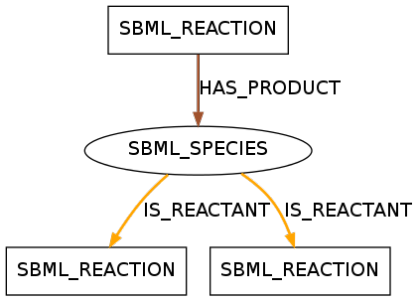

Figure 4: The displayed pattern was found in 436 models of data set R29 and 28 models of data set R1. It shows a species that takes a role as a reactant in two reactions and as a product in one reaction.

For our analysis, we furthermore build on the work published by Henkel et al. (2015). They describe a method to import models from BioModels Database into a graph database with a special focus on the encoded reaction networks. Subsequently, we build up a graph database based on Neo4J and imported models from our data sets R1 and R29 into that database. Afterwards, the reaction networks are made available to the ParSeMiS tool. We use the implemented gSpan algorithm to retrieve common subgraph patterns by model.

6 RESULTS

The aim of this work is to find common pattern in biochemical reaction networks. Using the aforementioned method, we analysed data sets R1 and R29 on a cluster node (180GB RAM, 16 Intel(R) Xeon(R) CPU X5650 @ 2.67GHz). For data set R29 we were able to identify 37 pattern in total, whith 350 being the lowest number of models that share a pattern. Identified pattern contained between one and six entities (species or reactions). The quest to identify pattern shared by less then 350 models was not successful due to memory limitations of the cluster. For the much smaller data set R1, however, we identified 190 pattern, containing between one and eleven entities. Here we were able to identify pattern shared by 20 of 30 models before limitations in memory occurred. Next to the obvious and expected pattern containing one, two or three entities (a single reaction or species or a combination of both), already pattern with four entities did not match our expectations. According to our key figure analysis (Figure 3), one would expect to find pattern containing one reaction as a center node and three species taking roles as products, reactants or modifier.

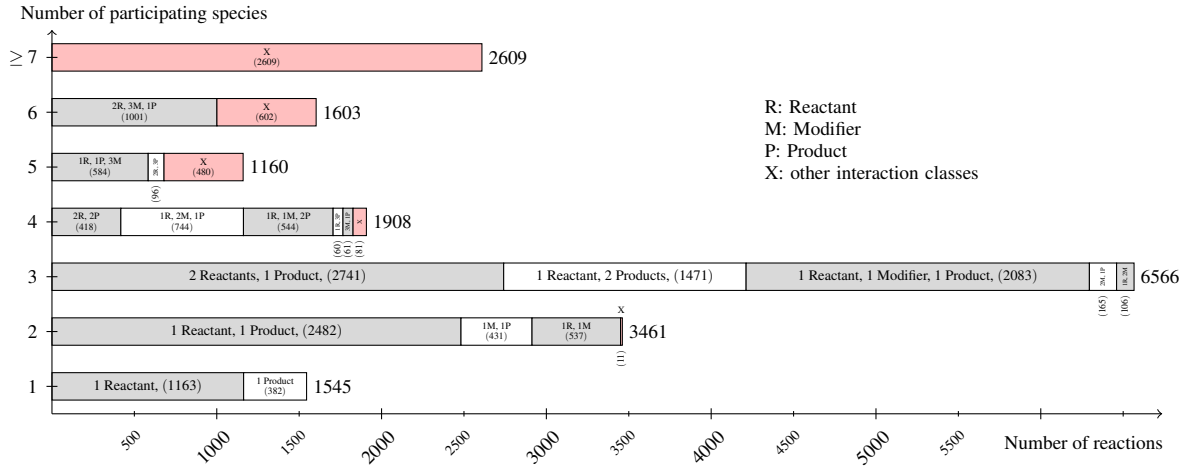

Figure 3: Listing of the node degree for reaction nodes in the data set R29 of curated models in BioModels Database. For each number of species (from 1 to 6, and more then 6) participating an an reaction, the figure lists the number of reaction nodes identified with a particular combination of its species relations (interaction class). The figure sums up smaller interaction classes displayed by X. Is becomes obvious that most reactions have two or three participating species.

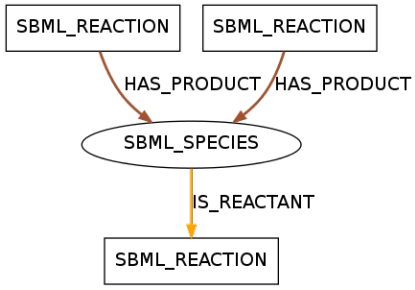

Figure 5: The displayed pattern was found in 390 models of data set R29 and in 26 models of data set R1. It shows a species that takes a role as a reactant in one reaction and as a product in two reactions.

No such pattern was found to be encoded by 350 to 575 models. Instead, patterns as the ones depicted in Figure 4 and Figure 5 were retrieved. Both examples have a species as the center node. As it is still feasible to manually list and search for all possible combinations of one reaction connected to three participating species (interaction classes with three species in Figure 3), we queried the database for those interaction classes as shown in Table 1. The data shows that the specific combination of two reactants and one product only occurs in 314 models, despite being the most encoded interaction class. We can conclude that this pattern was not found as we only retrieved pattern contained in 350 to 575 models. Same holds for all other possible interaction classes with three species.

Another interesting point is the usage of species as a modifier. Overall, species are mostly taking part in a reaction as a modifier (33209 times) compared to participation as a product (23630) or reactant (25595). However, in the 37 retrieved pattern only four of them

Table 1: This table shows the number of models containing a particular interaction class with three species taking the role of a reactant (R), product (P) or modifier (M). Values are given for Release 1 and 29 of BioModels Database. For each interaction class and release it is stated the count and percentage of models where such an interaction class was found.

Interaction Class	Release 1	Release 29
2R, 1P	18/60%	304/53%
1R, 1P, 1M	14/47%	279/49%
1R, 2P	13/43%	259/45%
1P, 2M	6/20%	223/39%
1R, 2M	6/20%	173/30%
2P, 1M	2/7%	130/23%
2R, 1M	3/10%	103/18%
3P	2/7%	83/14%
3R	3/10%	54/9%

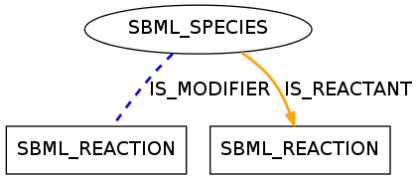

Figure 6: This pattern occurred in 351 models of data set R29 and shows a species taking part in a reaction as a reactant and a modifier.

contain a species as a modifier. One example is given in Figure 6. Apparently, all pattern are chains.

A further investigation reveals how unequally distributed the usage of modifiers among the models are. Ten models together count for 20620 modifier usages. Among those models are five derivations of the aforementioned semi-automatically created mod-

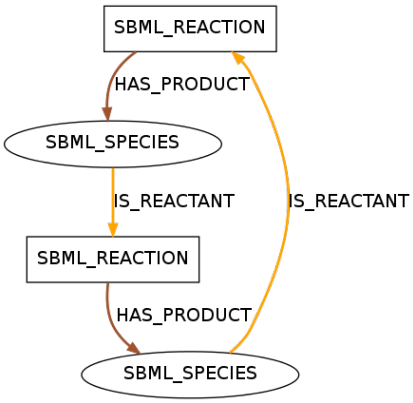

Figure 7: This pattern shows the smallest biologically meaningful circle. It is contained in 330 models of data set R29 and in 25 models of data set R1.

els by Smallbone et al. (2013). This might be a hint that modifiers are used in semi-automatically created models more often and differently.

We also expected a pattern containing a circle. Theoretically, such a pattern could be created with only one species and one reaction, if the species takes part as reactant and product. However, from a biological perspective there is no point in encoding such a construct. The next highest number of entities necessary for creating a circle is four (Figure 7). Such a construct is biologically meaningful, for example, to encode the creation and degradation of a protein, or to encode direct positive or negative feedback loops (please refer to Tyson and Novák (2010), Table 1). As we did not initially find such a pattern in R29 (350 to 575 models), we specifically searched for this pattern and identified it in 330 models.

The data set R1 is much smaller than R29. It was thus possible to identify both, more pattern and pattern containing more entities. The smallest meaningful circle (see Figure 7) was identified in 25 models of data set R1; the next possible circles containing six or eight entities was not found – even though patterns with up to 11 entities could be identified for R1 (Figure 8. Interestingly, a pattern with two circles consisting of seven entities (Figure 9) is contained in 21 models of R1. This could be a subset of motifs with 3 compounds as suggested by Tyson and Novák (2010).

7 CONCLUSION

The increasing amount of published models and the growing size of encoded reaction networks demand methods to analyse models. A number of approaches exist to compare models based on the en-

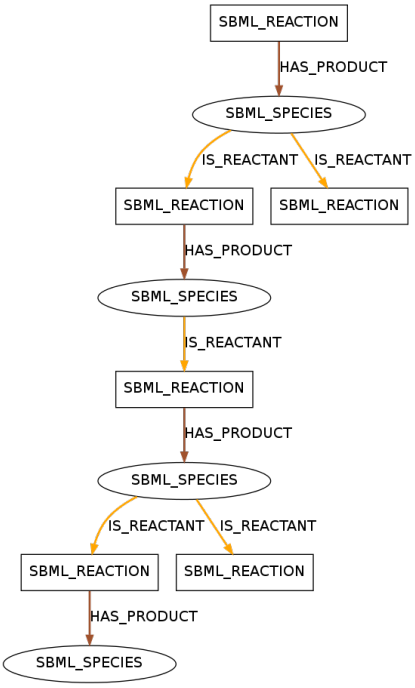

Figure 8: A pattern with ten entities containing two branches. This pattern is the biggest pattern that is not a chain.

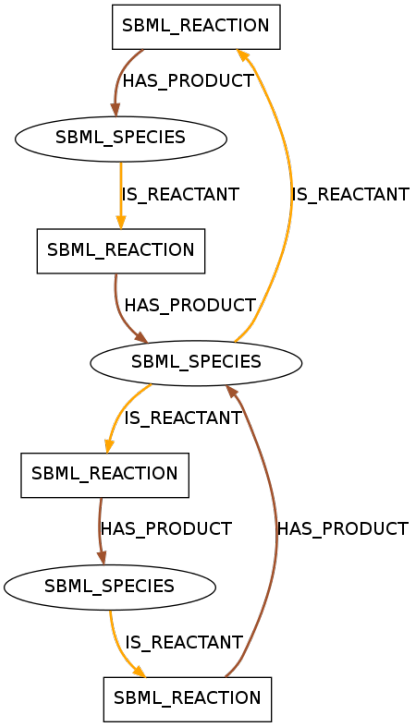

Figure 9: A pattern with seven entities containing two circles. This pattern is included in 21 models of R1.

coding format, the XML tags, or semantic annotations. We propose to add to the set of existing methods a new way of comparing models, which determines similar substructures. In this paper, we used the gSpan algorithm to analyse two data sets, the first and the latest release of models provided by BioModels Database. For the first release, we retrieved 190 patterns used in 20 to 30 models. For the latest release, we performed a key figure analysis. We then compared the identified 37 patterns (used in 350 to 575 models) and discussed the compliance and differences between the findings of the key figure analysis and the detected patterns. We found that a pure key figure analysis is not sufficient to characterize biochemical reaction networks.

We then searched for the motifs suggested by Tyson and Novák (2010). Using our algorithm, we could identify motifs with two compounds and a (presumable) subset of motifs with three compounds.

The Systems Biology Ontology (SBO) (Juty and le Novère, 2013) is an ontology representing mathematical concepts that are relevant for models. SBO thus provides terms for the functional role of a species or reaction. For example, a species that acts as a modifier can be annotated as "the modifying function is an inhibition of the reaction" (SBO:0000407). Most species and reactions in our data sets contain such annotations. The use of annotations, specifically from SBO, will enable us to identify motifs more precisely. It will also lower the computational costs of the search for submodels, because valuable semantic knowledge can be incorporated to reduce the number of potential alignments.

REFERENCES

Albert, R., Jeong, H., and Barabási, A.-L. (2000). Error and attack tolerance of complex networks. *nature*, 406(6794):378–382.

Alm, R., Waltemath, D., Wolkenhauer, O., and Henkel, R. (2014). Annotation-Based Feature Extraction from Sets of SBML Models. In *Data Integration in the Life Sciences*, pages 81—-95. Springer.

Barabasi, A.-L. and Oltvai, Z. N. (2004). Network biology: understanding the cell's functional organization. *Nature reviews genetics*, 5(2):101–113.

Bechhofer, S., Buchan, I., De Roure, D., Missier, P., Ainsworth, J., Bhagat, J., Couch, P., Cruickshank, D., Delderfield, M., Dunlop, I., et al. (2013). Why linked data is not enough for scientists. *Future Generation Computer Systems*, 29(2):599–611.

Finkelstein, A. et al. (2004). Computational challenges of systems biology. *IEEE Computer*, 37(5):26–33.

Hattori, M., Okuno, Y., Goto, S., and Kanehisa, M. (2003). Development of a chemical structure comparison method for integrated analysis of chemical and genomic information in the metabolic pathways. *J Am Chem Soc*, 125(39):11853–11865.

Henkel, R., Endler, L., Peters, A., Le Novère, N., and Waltemath, D. (2010). Ranked retrieval of computational biology models. *BMC bioinformatics*, 11(1):423.

Henkel, R., Wolkenhauer, O., and Waltemath, D. (2015). Combining computational models, semantic annotations and simulation experiments in a graph database. *Database*, 2015:bau130.

Hucka, M., Bergmann, F. T., Hoops, S., Keating, S. M., Sahle, S., Schaff, J. C., Smith, L. P., and Wilkinson, D. J. (2010). The systems biology markup language (sbml): language specification for level 3 version 1 core.

Hucka, M., Bergmann, F. T., Keating, S. M., and Smith, L. P. (2011). A profile of today's sbml-compatible software. In *e-Science Workshops (eScienceW), 2011 IEEE Seventh International Conference on*, pages 143–150. IEEE.

Juty, N., Ali, R., Glont, M., Keating, S., Rodriguez, N., Swat, M. J., Wimalaratne, S., Hermjakob, H., Le Novère, N., Laibe, C., et al. (2015). Biomodels database: Content, features, functionality, and use. *CPT: Pharmacometrics & Systems Pharmacology*, 2(4):1–14.

Juty, N. and le Novère, N. (2013). Systems biology ontology. *Encyclopedia of Systems Biology*, pages 2063–2063.

Kanehisa, M., Goto, S., Kawashima, S., Okuno, Y., and Hattori, M. (2004). The kegg resource for deciphering the genome. *Nucleic acids research*, 32(suppl 1):D277–D280.

Keyvanpour, M. R. and Azizani, F. (2012). Classification and Analysis of Frequent Subgraphs Mining Algorithms. *Journal Of Software*, 7.

Knüpfer, C., Beckstein, C., Dittrich, P., and Novère, N. L. (2013). Structure, function, and behaviour of computational models in systems biology. *BMC systems biology*, 7(1):43.

Koyutürk, M., Grama, A., and Szpankowski, W. (2004). An efficient algorithm for detecting frequent subgraphs in biological networks. In *Bioinformatics*, volume 20.

Kuramochi, M. and Karypis, G. (2001). Frequent subgraph discovery. In *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*, pages 313–320. IEEE.

Lakshmi, K. and Meyyappan, T. (2012). Frequent Subgraph Mining Algorithms - A Survey And Framework For Classification.

Li, C., Donizelli, M., Rodriguez, N., Dharuri, H., Endler, L., Chelliah, V., Li, L., He, E., Henry, A., Stefan, M. I., et al. (2010). Biomodels database: An enhanced, curated and annotated resource for published quantitative kinetic models. *BMC systems biology*, 4(1):92.

Lloyd, C. M., Halstead, M. D., and Nielsen, P. F. (2004). Cellml: its future, present and past. *Progress in biophysics and molecular biology*, 85(2):433–450.

Novere, N. L., Finney, A., Hucka, M., Bhalla, U. S., Campagne, F., Collado-Vides, J., Crampin, E. J., Halstead, M., Klipp, E., Mendes, P., et al. (2005). Minimum information requested in the annotation of biochemical models (miriam). *Nature biotechnology*, 23(12):1509–1515.

Olivier, B. G. and Snoep, J. L. (2004). Web-based kinetic modelling using jws online. *Bioinformatics*, 20(13):2143–2144.

Priyadarshini, S. and Mishra, D. (2010). An approach to graph mining using gspan algorithm. In *2010 International Conference on Computer and Communication Technology, ICCCT-2010*, pages 425–430.

Schulz, M., Krause, F., Le Novere, N., Klipp, E., and Liebermeister, W. (2011). Retrieval, alignment, and clustering of computational models based on semantic annotations. *Molecular systems biology*, 7(1):512.

Smallbone, K., Messiha, H. L., Carroll, K. M., Winder, C. L., Malys, N., Dunn, W. B., Murabito, E., Swainston, N., Dada, J. O., Khan, F., et al. (2013). A model of yeast glycolysis based on a consistent kinetic characterisation of all its enzymes. *FEBS letters*, 587(17):2832–2841.

Tyson, J. J. and Novák, B. (2010). Functional motifs in biochemical reaction networks. *Annual review of physical chemistry*, 61:219.

Waltemath, D., Henkel, R., Hälke, R., Scharm, M., and Wolkenhauer, O. (2013). Improving the reuse of computational models through version control. *Bioinformatics*, 29(6):742–748.

Wong, E., Baur, B., Quader, S., and Huang, C. H. (2011). Biological network motif detection: Principles and practice. *Briefings in Bioinformatics*, 13(2):202–215.

Wörlein, M., Meinl, T., Fischer, I., and Philippsen, M. (2005). A Quantitative Comparison of the Subgraph Miners MoFa, gSpan, FFSM, and Gaston. *9th European Conference on Principles and Practice of Knowledge Discovery in Databases*, pages 392–403.

Yan, X. Y. X. and Han, J. H. J. (2002). gSpan: graph-based substructure pattern mining. *2002 IEEE International Conference on Data Mining, 2002. Proceedings.*

Yu, T., Lloyd, C. M., Nickerson, D. P., Cooling, M. T., Miller, A. K., Garny, A., Terkildsen, J. R., Lawson, J., Britten, R. D., Hunter, P. J., et al. (2011). The physiome model repository 2. *Bioinformatics*, 27(5):743–744.

Zhang, Z. and Zhang, J. (2009). A big world inside small-world networks. *PloS one*, 4(5):e5686.