Petri nets in VANTED: Simulation of Barley Seed Metabolism

Anja Hartmann[1,*], Hendrik Rohn[1], Kevin Pucknat[1], and Falk Schreiber[1,2]

[1] Leibniz Institute of Plant Genetics and Crop Plant Research (IPK), Corrensstrasse 3, 06466 Gatersleben, Germany
{hartmann,rohn,pucknat,schreibe}@ipk-gatersleben.de
[2] Martin Luther University Halle-Wittenberg, Institute of Computer Science, Von-Seckendorff-Platz 1, 06120 Halle, Germany

Abstract. Petri nets are a mathematical language, which provide a unified environment for modeling, simulation, and formal analysis of biological systems. To support the applicability of Petri nets for biological users we implemented a Petri net add-on for the widely used VANTED framework. VANTED supports Petri net reconstruction, simulation capabilities to be able to investigate dynamic system behavior, and analysis algorithms for calculating intrinsic net properties. VANTED furthermore supports advanced visualization and exploration techniques, which can be used to examine even larger Petri nets in an interactive manner. We use this framework for the simulation-based analysis of a large stoichiometric model of central barley seed metabolism and discuss problems and obstacles during this process.

Keywords: Petri net, Simulation, Analysis, VANTED, Metabolic model

1 Introduction

Metabolic network models can be analyzed using various approaches, such as topological analysis (e. g., centralities), stoichiometric analysis (e. g., Flux Balance Analysis) or kinetic modeling, each corresponding to a different level of detail and a different level of available information. Petri nets can be used in order to quantitatively model, simulate, and analyze biological systems without the need for detailed and difficult to obtain measurements, such as enzyme activities or metabolite levels. Until now several tools were released to utilize the power of Petri nets, but their focus is often on analytical approaches, such as calculating invariants, hence their applicability is limited to small networks.

We developed a Petri net add-on for VANTED [6] focusing on the needs for a metabolic modeling pipeline, which enables simulation, interactive exploration, and interpretation of various properties based on continuous and discrete place-transition nets [22]. VANTED is a network editing framework, which supports researchers in the interpretation of experimental data visualized as charts in the context of biological networks, and thus is a widely-used tool in systems biology. To benefit from the broad functionalities of VANTED and to complement other

modeling approaches, we extended VANTED for the analysis and simulation offered by Petri nets through the Petri net add-on. An example of the combination of experimental data and Petri net models is the integration of omics data represented as charts inside the Petri net nodes. This Petri net models enable the comparison of metabolic effects (places) or enzyme activities (transitions) with simulation results. VANTED can furthermore be used to simulate metabolic models by the use of stoichiometric analysis methods, such as flux balance analysis.

The second section describes the software architecture and evaluates the known Java-based Petri net tools, highlights special properties of metabolic Petri net models, and explains exploration techniques. The third section shows the application of VANTED for the simulation-based analysis of a large stoichiometric model of central barley seed metabolism and discusses problems and obstacles during this process.

2 Methods and Tool

2.1 Software Architecture

In order to evaluate tools which may be used as an extension for VANTED, a survey of Java-based Petri net tools and libraries was performed (see Table 1). The requirements were availability under an open source license, support for discrete and continuous Petri nets, simulation in single- and multiple steps, analysis of place- and transition-invariants, and calculation of reachability [23].

Table 1. Evaluation of Java-based open source Petri net tools and libraries. Licenses abbreviations: Lesser General Public License (LGPL), Berkeley Software Distribution (BSD), General Public License (GPL), Open Software License (OSL), Academic Free License (AFL), Eclipse Public License (EPL).

Name	**License**	**Petri-net Types**		**Simulation**		**Analysis**		
				Single-	**Multi-**	**Invariant**		**Reach-**
		Discrete	**Continuous**	**step**	**step**	**P**	**T**	**ability**
Jfern	LGPL	✓	✓	✓	✓			
Tortuga	LGPL	✓		✓				
HISim	BSD	✓	✓	✓	✓			
JARP	GPL	✓		✓				
PIPE2	OSL3.0	✓		✓	✓	✓	✓	✓
Petri-LLD	GPL	✓		✓				
JPetriNet	AFL	✓		✓				
Renew	LGPL	✓	✓	✓	✓			
FERN	LGPL	✓		✓	✓			
PNEditor	GPLv3	✓		✓				
TAPAAL	OSL3+BSD	✓		✓				
WoPeD	LGPL	✓		✓	✓			✓
KIT-HORUS	EPL	✓		✓	✓			

The Petri net tools and libraries JFern [9], HISim [12], and Renew [13] can process discrete and continuous Petri nets which can be simulated in single- and multiple steps, however they do not support analytical methods. Pipe2 [2] and WoPeD [11] can handle discrete Petri nets and simulate in single- and multiple steps. Pipe2 enables the computation of reachability and invariants, whereas WoPeD enables only the computation of reachability. All other tools do neither support analytical methods nor the simulation of continuous Petri nets. FERN [14] and KIT-Horus [15] can simulate discrete Petri nets in single- and multiple steps whereas Tortuga [16], Petri-LLD [17], JPetriNet [18], PNEditor [19], JARP [20], and TAPAAL [21] enable the simulation of just discrete Petri nets in single steps.

It has become apparent that no single tool was able to satisfy all requirements. Therefore we decided to incorporate two tools. JFern [9] is a compact and native Java library being able to handle object-oriented-, timed-, high-level-, and place-transition nets. It is used to perform basic Petri net operations on continuous and discrete place-transition nets. As the library does not comprise analytical methods, Pipe2 [2] is used to perform reachability, and invariant analysis. Both libraries are complemented by the capabilities of the VANTED framework itself. It provides various network importers (e. g., for SBML-, KGML-, and GML files) and direct access to network databases (e. g., MetaCrop [10], KEGG [7]). For network exchange with the Petri net community PNML [5] import and export functionality was implemented. Various image exporters enable the intuitive communication of simulation and analysis results within the scientific community. Finally, networks can be edited, semi-automatically transformed into Petri nets, layouted, and extended with other systems biology data, such as gene expression and metabolic data.

VANTEDs workflow is structured in three parts: In order to create valid Petri nets, manual editing or semi-automatic network transformations can be used in the reconstruction step. The simulation step enables to interactively follow the flow of tokens through the Petri net. Finally, complementary analytical functions to calculate intrinsic net properties can be performed in the analysis step and examined using exploration techniques. These steps are explained in detail in the following sections and can be performed using the Petri net add-on available for download under http://www.vanted.org/petrinet.

2.2 Model Reconstruction

A Petri net can be manually reconstructed with VANTED. This tedious process can be circumvented by accessing metabolic networks from files or public databases. The process of transforming such networks into syntactically valid Petri nets is supported by semi-automatic transformation functionalities which, for example, enable to assign place or transition roles to selected nodes. A user may choose to transform all selected nodes into places and unselected nodes are implicitly assigned as transitions (or vice versa). Another transformation converts hypergraphs (e. g. networks consisting of metabolites represented as nodes connected by hyperedges representing enzymes) in Petri nets by transforming

all existing nodes into places and splitting edges between these places into two arcs by inserting transition nodes. The correctness of generated Petri nets is validated before analysis and simulation is performed. VANTED examines that arcs run only from places to transitions or vice versa and checks for unconnected elements in the Petri net. Also the type of the Petri net is considered, as VANTED checks for consistent use of discrete or continuous Petri net elements by examining tokens, place-capacities and arc-weights.

Special focus was put on unique properties of metabolic Petri net reconstruction in accordance to the review of Baldan et al. [1]. The modeling of spatial properties, such as compartmentation is solved by adding suffixes to each metabolite name, thus distinguishing pathways distributed between different compartments. Multiple occurring metabolites in different reactions, such as ATP and CO_2 are represented as logical places, thereby preventing multiple edge crossings throughout the network. External metabolites for token import are modeled as source transitions (regular import) or source places (controlled import by specifying a certain amount of tokens), whereas token export is realized by creating sink transitions (export if the precondition is achieved) or sink places (traceable accumulation). In case of reversible reactions VANTED enables the use of hierarchical transitions, which are visualized as single reactions but internally split up into two distinct forward and backward reactions.

2.3 Model Simulation

Simulation of the token flow through the Petri net is an important possibility to determine dynamic properties of metabolic networks. Initially, tokens are set for all places and a user-defined number of simulation steps is performed. The marking is visualized for each step, resulting in an animation of token flow throughout the Petri net. For faster simulation results a number of steps can be performed in the background, resulting in the visualization of the final marking. For further analysis the marking of each simulation step can be exported as a text file and imported into other tools, such as MS Excel or mapped to the corresponding places in VANTED in order to examine the number of assigned tokens for each simulation step.

2.4 Model Analysis

Petri net theory provides a variety of analytical methods in order to gain information on metabolic network behavior. VANTED enables the calculation and visualization of the reachability graph, as well as place- and transition-invariants. Such methods enable the detection of structural inconsistencies, such as traps, boundedness, liveness, safeness, and deadlocks, thereby semantically checking the Petri net structure (for definitions see [23]). In order to support interpretation of analytical approaches, interactive visualization techniques were developed based on the brushing and linking concept [8]: Calculated invariants are listed in a dialog, which supports a mouse-over selection effect, resulting in a linked immediate visualization of the particular invariant in the Petri net view. Analogously,

the calculated reachability graph is visualized as an additional network and also reacts on mouse-over events, see Figure 1. Here a node mouse-over triggers the visualization of the particular marking in the Petri net view which represents a reachable state depending on the initial marking (S0 in part a). Mouse-over an arc results in the highlighting of a firing transition (t2 in part b). This interaction technique enables users to understand non-deterministic features of Petri nets and enables detailed investigation of dynamic changes of marking even in large reachability graphs.

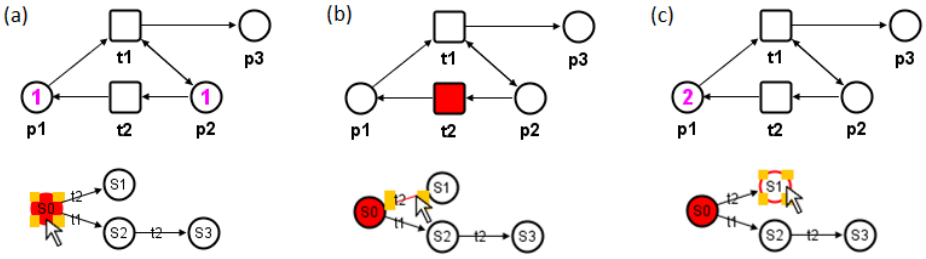

Fig. 1. Interactive visualization of the Petri net reachability analysis. (a) Initial marking is represented by the initial state S0 (marked) of the reachability graph, which is visualized in the Petri net. (b) Mouse-over an edge of the reachability graph results in a visualization of the firing transition (t2), which would lead to the next reachable state S1 (c).

3 Simulation of a Comprehensive Barley Seed Petri net

The barley seed Petri net is based on a stoichiometric model of central seed metabolism consisting of 234 metabolites and 257 reactions in different compartments, with the aim of getting a systemic understanding of barley seed storage metabolism and to study grain yield [3]. As the calculation of invariants for such large models is not feasible, the goal of this use case is to show how such large models can be converted into valid Petri nets and be investigated using the simulation capabilities of VANTED.

The stoichiometric model defined as an SBML file (see supplementary material of [3]) is imported into VANTED, layouted, and all compartmentation represented by splitting places and adding suffixes to the place names. No place-capacities were set and arc-weights (representing stoichiometry) were automatically recognized. The boundary is defined according to the goal of observing biomass accumulation by adding source transitions to allow regular import of external metabolites and sink places to trace the accumulation of produced metabolites. The starch degradation pathway was excluded from the simulation in order to prevent a drain of starch into this pathway. This is because increasing starch levels would otherwise be degraded immediately instead of being used for

biomass synthesis. An initial marking of 10 tokens for each place was set and the simulation was performed over 1000 steps. The marking gets visualized and exported for each step, animating the accumulation or depletion of metabolites over time. Figure 2 comprises important metabolites in the starch metabolism with the number of assigned tokens over these steps.

After a transient oscillation at the beginning of the simulation, starch and biomass converge into a stable marking, because no mitochondrial energy (ATP) is available (step 20). As soon as enough ATP is produced (step 100), starch is metabolized into biomass and depletes at step 165. The reason for the limitation of biomass production by lack of starch is the energy household which apparently is a deadlock: Mitochondrial ATP is not transported to the cytosol or plastid, because the export of ATP from the mitochondria demands an import of ADP from the cytosol. Instead the cytosolic ADP gets depleted very early due to the drain of other reactions. As the production of starch precursors G1P and ADPglc depends on cytosolic and plastidic ATP, starch cannot be produced.

The basic issue is the small pool of ATP and ADP compared to the large number of reactions consuming these substances in the cytosol. Such an in-balance increases the impact of non-deterministic firing of reactions being in conflict. This non-deterministic decision causes e. g. the consumption of free ADP and thereby preventing the necessary transport of ADP into mitochondria in order to export ATP. As the behavior of biochemical networks is inherently governed by stochastic laws [4], the utilization of stochastic Petri nets could resolve such dynamic conflicts.

To understand the causality of such complex interactions between different compartments, additional tests should be applied in order to solve this deadlock by adding or excluding biochemical reactions in form of transitions.

4 Conclusion and Outlook

We have shown a promising approach to simulate large metabolic models with a new Petri net add-on for the VANTED framework. Based on the broad functionality of VANTED a transformation of existing metabolic models into Petri nets and their simulation is powerful and easy to use. The barley seed model, developed for the optimization of biomass production, proved to be difficult to transfer directly into a Petri net. Typical behavior, such as non-deterministic firing of transitions and preferred firing for branches with less token uptake needs further adjustments of the metabolic model. Nevertheless, based on the results we are confident that the enhanced reconstruction and simulation process in VANTED supports users in the analysis of such problems. In the future, we plan to use other Petri net classes such as stochastic Petri nets in order to improve modeling quality of metabolic network properties.

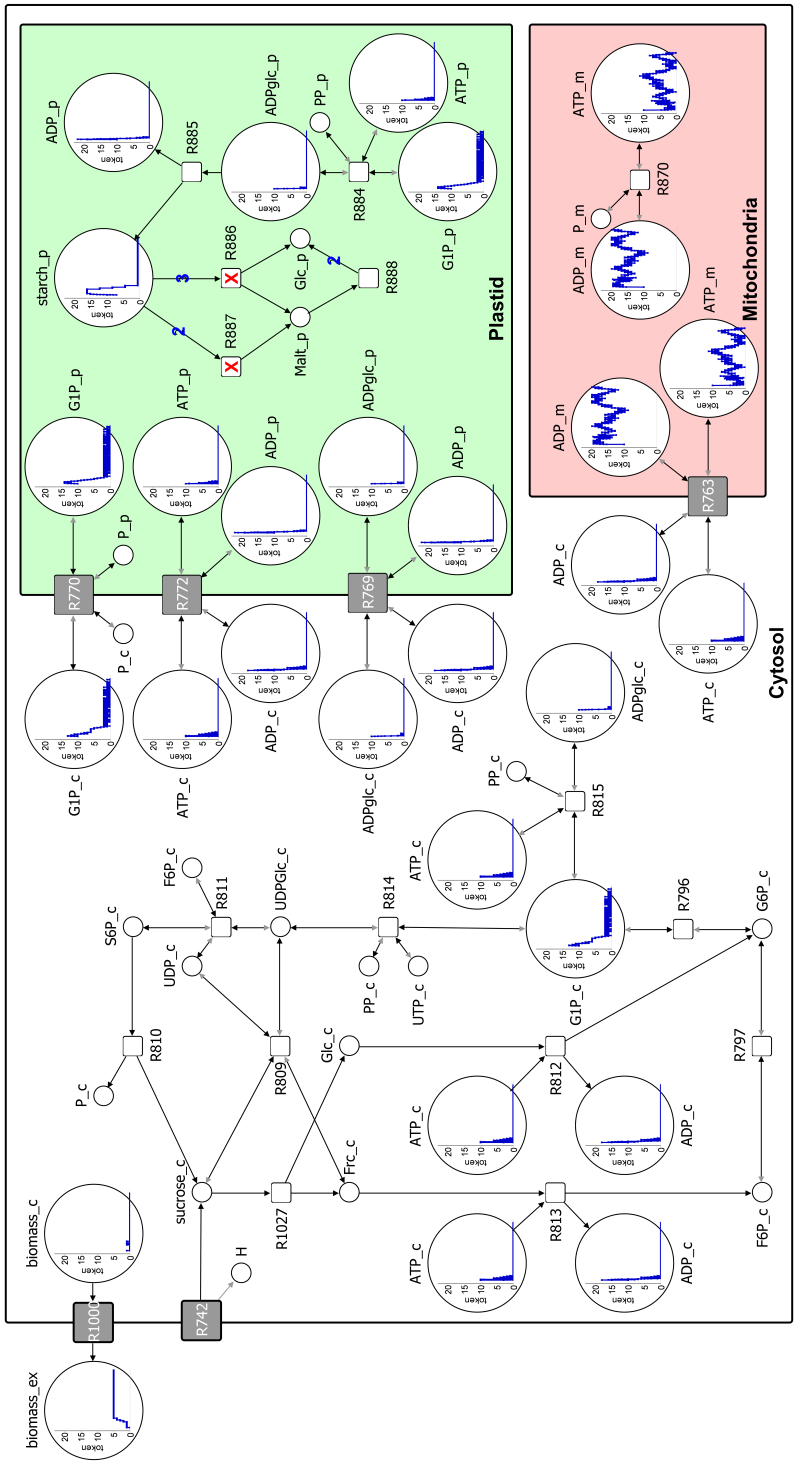

Fig. 2. Schematic overview of the condensed barley seed Petri net focusing on starch and energy metabolism. Charts depict token assignments per simulation step for important metabolites of the barley seed Petri net. The carbon source sucrose is imported into the cytosol (suffix "_c") of the cell through the source transition R742 without restrictions. Different reactions transform sucrose together with the consumption of energy (ATP) in starch precursors in the cytosol (G1P, ADPglc). The precursor ADPglc is produced either in the cytosol (R815) and afterwards transported into the plastid (R769, suffix "_p") or G1P is transported into the plastid (R770) and ADPglc produced in the plastid (R884). The last step of the starch biosynthesis (R885) and also both steps of the starch degradation (R886, R887), which are excluded from simulation as indicated by the cross, are located in the plastid. The starch precursors G1P and ADPglc located in the cytosol and plastid are consumed together with starch in order to build up biomass. Necessary energy (ATP) is produced in the respiratory chain of the mitochondria (R870) and delivered to the cytosol (R763), as well as to the plastid (R772) by exchanging ADP. In the mitochondria the energetic metabolites ATP and ADP act in an alternating manner. The first 165 steps of the simulation show a clear effect of the initial marking of 10 tokens, after which most of the metabolites converge into a stable marking.

References

1. Baldan, P., Cocco, N., Marin, A., Simeoni, M.: Petri nets for modelling metabolic pathways: a survey. Natural Computing 9, pp. 955–989 (2010)
2. Bonet, P., Llado, C., Puigjaner, R., Knottenbelt, W. J.: PIPE v2.5: A Petri Net Tool for Performance Modelling. In: 23rd Latin American Conference on Informatics (CLEI 2007)
3. Grafahrend-Belau, E., Schreiber, F., Koschützki, D., Junker, B. H.: Flux balance analysis of barley seeds: a computational approach to study systemic properties of central metabolism. Plant Physiology 1(149), pp. 585–598 (2009)
4. Heiner, M.: Petri nets for Systems and Synthetic Biology. Natural Computing 10(2), pp. 633–638 (2011)
5. Hillah, L. M., Kindler, E., Kordon, F., Petrucci, L., Treves, N.: A primer on the Petri Net Markup Language and ISO/IEC 15909-2. Petri Net Newsletter 76, pp. 9–28 (2009)
6. Junker, B. H., Klukas, C., Schreiber, F.: VANTED: A system for advanced data analysis and visualization in the context of biological networks. BMC Bioinformatics 7(109) (2006)
7. Kanehisa, M., Goto, S., Sato, Y., Furumichi, M., and Tanabe, M.: KEGG for integration and interpretation of large-scale molecular datasets. Nucleic Acids Research 40, pp. D109-D114 (2012)
8. Martin, A. R., Ward, M. O.: High dimensional brushing for interactive exploration of multivariate data. In: Proceedings on Visualization, pp. 271–278 (1995)
9. Nowostawski, M.: JFern Manual, Version 4.0.0 (2009)
10. Schreiber, F., Colmsee, C., Czauderna,T., Grafahrend-Belau, E., Hartmann, A., Junker, A., Junker, B. H., Klapperstück, M., Scholz, U., Weise, S.: MetaCrop 2.0: managing and exploring information about crop plant metabolism. Nucleic Acids Research 40, pp. D1173-D1177 (2012)
11. Eckleder, A., Freytag, T.: WoPeD A tool for teaching, analyzing and visualizing workflow nets. Petri Net Newsletter 75 (2008)
12. Amengual, A.: A computational model of attachment secure responses in the Strange Situation. Technical report, International Computer Science Institute (2009)
13. Friedrichs, F. D.: Referenznetze mit Anschriften in Scheme. PhD Thesis, University Hamburg (2007)
14. Erhard, F., Friedel, C. C., Zimmer, R.: FERN - a Java framework for stochastic simulation and evaluation of reaction networks. BMC Bioinformatics, 1(9), pp. 356 (2008)
15. Klink, S., Li, Y., Oberweis, A.: INCOME2010 - a toolset for developing process-oriented information systems based on petri nets. In: Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops, Simutools '08, pp. 14:1-14:8. ICST (2008)
16. Kuhl, F.: Tortuga, The MITRE Corporation (2004), `http://code.google.com/p/tortugades`
17. Brusey, J.: Petri(LLD) Tutorial. Department of Engineering, Cambridge University (2006)
18. Azevedo, M.: JPetriNet, Sapucai Valley University - Brazil (2004), `http://jpetrinet.sourceforge.net`
19. Riesz, M.: PNEditor, Slovak University of Technology (2010), `http://www.pneditor.org`

20. Padilha, R. S.: JARP Petri Nets Analyzer, Industrial Automation and Control Engeneering of the Federal University of Santa Catarina - Brazil (2001), `http://jarp.sourceforge.net`
21. Byg, J., Jörgensen, K. Y., Srba, J.: TAPAAL: Editor, Simulator and Verifier of Timed-Arc Petri Nets. In: Proceedings of the 7th International Symposium on Automated Technology for Verification and Analysis (ATVA '09), pp. 5799:84–5799:89, Lecture Notes in Computer Science, Springer-Verlag, Berlin-Heidelberg (2009)
22. Bernardinello, L., de Cindio, F.: A survey of basic net models and modular net classes. In: Rozenberg, Grzegorz: Advances in Petri Nets: The DEMON Project, Lecture Notes in Computer Science, pp. 609:304–351, London, 1992. Springer-Verlag
23. Murata, T.: Petri Nets: Properties, Analysis and Applications. In: Proceedings of the IEEE 77(4), pp. 541–580 (1989)