

Hochschule Ostwestfalen-Lippe
Masterstudiengang Umweltwissenschaften
Studienrichtung Umweltingenieurwesen und Modellierung

Erweiterung der Software VANTED zum Editieren und Simulieren biologischer Petri-Netze

Masterarbeit
zur Erlangung des akademischen Grades

Master of Science

eingereicht von Kevin Pucknat
geb. am 19.08.1981 in Quedlinburg
Matrikelnummer 15188007

Hochschule Ostwestfalen-Lippe
University of Applied Sciences

Quedlinburg, den 2. Januar 2012

Gutachter Prof. Dr. Burkhard Wrenger, Hochschule OWL
Gutachter Prof. Dr. Falk Schreiber, IPK Gatersleben
Betreuer Hendrik Rohn, IPK Gatersleben

Kurzzusammenfassung

Petri-Netze sind grafische und mathematische Werkzeuge, welche eine einheitliche Umgebung für die Modellierung, formale Analyse und Simulation biologischer Systeme bieten. Mit ihrer Hilfe ist es möglich, Eigenschaften biologischer Netzwerke abzuleiten und Hypothesen zu testen, bevor aufwändige experimentelle Verfahren angewandt werden. Dazu gehört auch die Validierung von Netzwerken hinsichtlich der biologischen Aussagekraft, um ein besseres Verständnis biologischer Prozesse zu ermöglichen.

Um die Mächtigkeit von Petri-Netzen bereitzustellen, ist das Ziel dieser Arbeit die Implementierung eines Petri-Netz Add-ons für das in der biologischen Forschung angewandte VANTED Framework. Es kann diskrete und kontinuierliche Petri-Netze unterschiedlicher Komplexität verarbeiten. Die Editierung von Petri-Netzen und die Transformation von biologischen Netzwerken in Petri-Netze ist auf interaktive Weise möglich. Simulation und Analysemethoden werden durch verschiedene Visualisierungs- und Interaktionstechniken unterstützt, um Eigenschaften und Verhalten der Petri-Netze schneller erfassen zu können. Dazu gehört auch die interaktive Animation des Schaltverhaltens eines Petri-Netzes in Einzel- und Mehrfachschritten, sowie Visualisierung und Exploration der Petri-Netz-Invarianten und des Erreichbarkeitsgraphens.

Das Petri-Netz Add-on wird eingesetzt, um Lösungen verschiedener Fragestellungen bezüglich drei bekannter metabolischer Netzwerke zu verifizieren. Die Mächtigkeit und Anwendbarkeit des Petri-Netz-Ansatzes konnte demzufolge erfolgreich für realistische Szenarien aufgezeigt werden. Somit bietet diese Arbeit der biologischen Forschung ein weiteres Werkzeug zur Analyse komplexer biologischer Systeme.

Abstract

Petri nets are graphical and mathematical tools, which provide a unified environment for modeling, formal analysis and simulation of biological systems. With its help it is possible to derive properties of biological networks and to test hypotheses before elaborate experimental methods are used. This includes the validation of networks in terms of biological significance to allow a better understanding of biological processes.

To provide the power of Petri nets, the aim of this work is the implementation of a Petri net add-on for the VANTED framework used in biological research. It can handle discrete and continuous Petri nets of varying complexity. The editing of Petri nets and the transformation of biological networks into Petri nets is possible in an interactive way. Simulation and analysis are supported by different visualization and interaction techniques to capture the properties and behavior of Petri nets quickly. This includes the interactive animation of the switching behavior of a Petri net in single and multiple steps, as well as visualization and exploration of Petri net invariants and the reachability graph.

The Petri-net add-on is used to verify solutions of various questions concerning three known metabolic networks. The power and applicability of the Petri net approach could hence be demonstrated successfully for realistic scenarios. Consequently this work provides another tool in biological research for the analysis of complex biological systems.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Ziele und Vorgehensweise	2
2	Grundlagen	3
2.1	Petri-Netze und deren Grundelemente	3
2.2	Petri-Netz Klassifikation	4
2.3	Platz/Transitions-Netz Struktur	5
2.4	Platz/Transitions-Netz Dynamik	6
2.5	Analysemethoden	8
2.5.1	Strukturanalyse	10
2.5.2	Invariantenanalyse	11
2.5.3	Erreichbarkeitsanalyse	13
2.5.4	Strukturelle Petri-Netz Eigenschaften	15
2.5.5	Dynamische Petri-Netz Eigenschaften	16
3	Aufgabenstellung	19
3.1	Beschreibung der Aufgabenstellung	19
3.2	Übersicht vorhandener Programme und Bibliotheken	20
4	Implementierung	24
4.1	Petri-Netz Rekonstruktion	24
4.1.1	Editorfunktionalität	25
4.1.2	Transformationsfunktionalität	27
4.1.3	Besonderheiten der Petri Netz Repräsentation biochemischer Netzwerke	29
4.2	Petri-Netz Simulation	32
4.2.1	Selektion konzessionierter Transitionen	33
4.2.2	Simulationsmodi	34

4.3	Petri-Netz Analyse	35
4.3.1	Invariantenanalyse	36
4.3.2	Erreichbarkeitsanalyse	38
4.4	Austauschformat für Petri-Netze	40
5	Anwendung	42
5.1	Analyse des Glykolyse Petri-Netzes	43
5.2	Analyse des Sacharoseabbauweges in der Kartoffel	49
5.3	Simulation des Zentralstoffwechsels im Gerstesamen	53
6	Zusammenfassung und Ausblick	60
	Literaturverzeichnis	62

Abbildungsverzeichnis

2.1	Elemente eines P/T-Netzes	6
2.2	Petri-Netz in Maritzenschreibweise	9
2.3	Unbeschränktes Petri-Netz mit dazugehörigem Überdeckungsgraphen	14
2.4	Statischer Konflikt, Strukturelle Verklemmung und Falle	16
2.5	Dynamischer Konflikt	18
4.1	UML Aktivitätsdiagramm	25
4.2	Graphische Benutzeroberfläche Rekonstruktion	26
4.3	Ausschließen von Netzwerkknoten zur Validierung	27
4.4	Transformation der Glykolyse in ein Petri-Netz	28
4.5	Modellierung räumlicher Eigenschaften	30
4.6	Modellierung mehrfach dargestellter Metabolite	31
4.7	Modellierung externer Metabolite	32
4.8	Modellierung reversibler Reaktionen	32
4.9	Graphische Benutzeroberfläche Simulation	33
4.10	Petri-Netz Simulation	34
4.11	Graphische Benutzeroberfläche Analyse	36
4.12	Schematische Darstellung der Interaktionstechnik	38
4.13	Interaktionstechnik in der Erreichbarkeitsanalyse	40
5.1	P-Invarianten des Glykolyse Petri-Netzes	44
5.2	T-Invarianten des Glykolyse Petri-Netzes	45
5.3	Erreichbarkeitsgraph des Glykolyse Petri-Netzes	47
5.4	Eine T-Invariante des Sacharoseabbauweges in der Kartoffel	50
5.5	Markenverteilung im Sacharoseabbauweg der Kartoffel pro Simulati- onsschritt	52
5.6	Der Zentralstoffwechsel im sich entwickelnden Gerstesamen	55
5.7	Markenverteilung im Gerstesamen Petri-Netz pro Simulationsschritt .	56

5.8	Markenverteilung der Metabolite ATP_m, ADP_m, starch_p und biomass_ex pro Simulationsschritt	57
5.9	Markenverteilung im Gerstesamen Petri-Netz pro Simulationsschritt mit ausgeschaltetem Stärkeabbau	59

Tabellenverzeichnis

2.1	Beziehung zwischen Stoffwechselementen und Petri-Netz Elementen	4
2.2	Beziehung zwischen Stoffwechselweg und Petri-Netz Invarianten	13
2.3	Bestimmbare dynamische Eigenschaften des Erreichbarkeits- und Überdeckungsgraphen	15
3.1	Auflistung der im Internet vorhandenen Programme und Bibliotheken	22

1 Einleitung

1.1 Motivation

Petri-Netze wurden schon in den 60er Jahren entwickelt [36]. Carl Adam Petri, (Dissertation „Kommunikation mit Automaten“ [34]), ist der Urheber der Petri-Netz Theorie. Die Dissertation befasst sich mit der Modellierung von dynamischen Systemen. Mit Petri-Netzen ist es möglich Systeme, Netzwerke und Prozesse formal zu modellieren, diese auf ihre Funktionalität zu Überprüfen und Hypothesen zu testen, bevor die Theorie in die Praxis umgesetzt wird. Primär sind Petri-Netze entwickelt wurden um das dynamische Verhalten komplexer verteilter Systeme, die parallel, diskret und nicht-deterministisch arbeiten können, wiederzugeben. Auf Grund des definierten Formalismus von Petri-Netzen werden Systeme einheitlich grafisch darstellbar und ermöglichen somit unterschiedliche Abstraktionsgrade zu betrachten ohne die graphische Beschreibungsart zu wechseln. Der Erfolg von Petri-Netzen, ist deshalb auf die leichte Verständlichkeit aber zugleich formale Ausdrucksstärke zurückzuführen. Mit Petri-Netzen ist es möglich eine Vielzahl von Systemen unterschiedlichster Anwendungsbereiche zu modellieren und zu simulieren. Verschiedene Anwendungsgebiete für die Petri-Netz Modellierung sind zum Beispiel, das Erfassen von Arbeitsabläufen, der Ablauf paralleler Algorithmen und die Validierung biologischer Netzwerke.

Das Ziel dieser Masterarbeit, ist die Mächtigkeit der Petri-Netze für das Editieren, Analysieren und Simulieren biologischer Netzwerke, am IPK, (Leibniz-Institut für Pflanzen-genetik und Kulturpflanzenforschung) zu etablieren. Die biologischen Netzwerke sollen hinsichtlich der Validität untersucht werden, um ein besseres Verständnis für die Modellierung zu schaffen und mögliche Fehler zu korrigieren. Bestimmte Visualisierungs- und Interaktionstechniken werden die generierten Analysedaten, wie den Erreichbarkeitsgraphen und die Invarianten aufbereiten und somit die Interpretation der Ergebnisse erleichtern.

Die am IPK entwickelte Software VANTED [21] (Visualization and Analysis of Networks containing Experimental Data) unterstützt bereits das Erstellen, Editieren, Visualisieren und Analysieren biologischer Experimentaldaten. Zum Zweck der Erweiterung zusätzlicher Funktionalität, stellt VANTED eine Schnittstelle für Add-ons

bereit. Diese Schnittstelle wird für das zu entwickelnde Petri-Netz Add-on verwendet, um das Editieren, Analysieren und Simulieren biologischer Netzwerke in Form von Petri-Netzen, in VANTED, zu ermöglichen.

1.2 Ziele und Vorgehensweise

Ziel dieser Arbeit ist die Entwicklung eines Petri-Netz Add-ons für VANTED. Das Add-on soll diskrete und kontinuierliche Petri-Netze erstellen, editieren, analysieren und simulieren können. Zur Begriffsklärung werden zu Beginn in Kapitel 2 die Grundlagen der Petri-Netz Theorie erklärt und die Verbindung zu biologischen Netzwerken gezogen. Es wird sowohl der Formalismus der Petri-Netz Struktur, bestehend aus den verschiedenen Grundelementen, als auch die Dynamik von Petri-Netzen erläutert.

Da Petri-Netze schon seit den 60er Jahren des letzten Jahrhunderts existieren, wurden auf Grund dessen, bereits zahlreiche Softwareprodukte entwickelt. Eine in Java implementierte Bibliothek oder ein Programm soll deshalb die Grundlage für das zu entwickelnde Add-on sein. Eine Übersicht möglicher geeigneter Softwareprodukte wird in Kapitel 3 zusammengetragen.

Die Implementierung des Petri-Netz Add-ons ist Gegenstand des Kapitels 4. Bezogen auf die Anwendung der Petri-Netze für biologische Netzwerke wird das Erstellen und Editieren dieser, im Abschnitt Rekonstruktion erläutert. Anschließend werden die Simulation und die implementierten Analysemethoden, wie die Erreichbarkeitsanalyse und die Analyse der Platz- und Transitionsinvarianten, vorgestellt. Die Interaktionsmöglichkeiten zur Unterstützung der Interpretation von Analyseergebnissen werden anhand kleiner Beispiele dargestellt. Der Export berechneter Ergebnisse und der standardisierte Import als auch Export entwickelter Petri-Netze, zum Austausch mit anderen Programmen, wird ebenfalls beschrieben.

In Kapitel 5 wird das Petri-Netz Add-on für drei ausgewählte biologische Netzwerke angewandt. Zunächst werden die Invarianten und die Erreichbarkeit an einem kleinen Beispiel, der Glykolyse analysiert und die Ergebnisse interpretiert. Anschließend dient ein komplexeres Beispiel, der Sacharoseabbauweg in der Kartoffel, als Gegenstand der Invariantenanalyse. Die Simulation des Zentralstoffwechsels im Gerstesamen veranschaulicht die Möglichkeiten der Validierung eines komplexen Modells.

Abschließend wird in Kapitel 6 eine Zusammenfassung dieser Masterarbeit und ein Ausblick auf künftige Fragestellungen besprochen.

2 Grundlagen

Dieses Kapitel erklärt grundlegende Begriffe der Petri-Netz Theorie und bezieht diese Begriffe auf biologische Netzwerke. Zunächst werden die Grundelemente von Petri-Netze erläutert und die verschiedenen Petri-Netz Arten klassifiziert. Danach wird auf die Struktur und die Dynamik von Platz/Transitions-Netzen eingegangen. Die verschiedenen statischen und dynamischen Analysemethoden werden erläutert und abschließend die Petri-Netz Eigenschaften erklärt und beschrieben.

2.1 Petri-Netze und deren Grundelemente

Ein Petri-Netz ist ein endlicher, gerichteter, bipartiter Graph ohne isolierte Knoten [42]. Bipartit bedeutet, dass es zwei verschiedene Knotenarten gibt. In Petri-Netzen wird die eine Knotenart als Platz/Stelle bezeichnet und graphisch als Kreis oder Ellipse dargestellt. Ein Platz ist ein passives Element das Marken lagern oder speichern kann. Die zweite Knotenart heißt Transition und wird graphisch als Rechteck, Quadrat oder nur als dicker Strich dargestellt. Eine Transition ist ein aktives Element und kann Marken erzeugen, verbrauchen, transportieren oder verändern. Beide Knotenarten werden durch eine gerichtete Kante verbunden. Diese Kante drückt die Beziehung zwischen dem passiven und aktiven Element aus.

Wichtig beim Aufbau eines Petri-Netzes ist, dass immer ein passives Element mit einem aktiven Element verbunden wird. Es darf kein passives Element mit einem weiteren passiven Element verbunden sein. Dies gilt ebenfalls für die aktiven Elemente. Sollte es doch einmal vorkommen, liegt meistens ein Modellierungsfehler vor, der behoben werden muss.

Das vierte Element eines Petri-Netzes wird Marke genannt, welche graphisch nur als Punkt oder Zahl dargestellt wird. Alle Marken auf den Plätzen eines Petri-Netzes drücken den Zustand dieses Netzes aus. Ein solcher Zustand wird Markierung genannt, wobei die Anfangsmarkierung mit M_0 angegeben wird. Eine Markierung wird zumeist als ein Zeilenvektor geschrieben. Die vier beschriebenen Grundelemente Platz, Transition, Kante und Marke besitzen alle Petri-Netz Arten.

In einem Petri-Netz kann es Quell- sowie Senkenknoten (Transition und Plätze) geben. Ein Quellknoten besitzt keine eingehenden und ein Senkenknoten keine ausge-

Stoffwechsel Elemente	Petri-Netz Elemente
Metabolite	Plätze
Reaktionen, Transporter	Transitionen
Edukte	Vorplätze
Produkte	Nachplätze
Stöchiometrische Koeffizienten	Kantengewichte
Metabolitmengen	Anzahl der Marken auf Plätzen

Tabelle 2.1: Beziehung zwischen Stoffwechselelementen und Petri-Netz Elementen, in Anlehnung an [3].

henden Kanten. In Abbildung 2.2 sind die vier möglichen Quell- und Senkenknoten dargestellt, p1 ist ein Quellplatz, t2 eine Quelltransition, p7 ist ein Senkenplatz und t6 eine Senkenttransition. Diese Quell- und Senkenknoten sind ein wichtiger Aspekt bei der Modellierung eines Petri-Netzes. Ein Petri-Netz kann platz- oder transitionsberandet sein. Ein platzberandetes Netz kann keine Marken von außen erhalten oder abgeben. Ein transitionsberandetes Netz erhält Marken von außen und gibt Marken nach außen ab.

Angelehnt an die hier zu untersuchenden biologischen Netzwerke (metabolische Stoffwechselwege) werden die vier Grundelemente, wie in Tabelle 2.1 dargestellt, in Beziehung gesetzt. Ein Platz entspricht einem Metabolit (Edukt oder Produkt einer Reaktion) und eine Transition wird einer Reaktion (Enzym oder Transporter) gleichgesetzt. Verbunden durch Kanten, bilden diese beiden Grundelemente eine gerichtete Reaktion. Plätze können Marken enthalten, die als Mengen der Edukte und Produkte bezeichnet werden.

2.2 Petri-Netz Klassifikation

Der Internetauftritt „Petri Nets World“ der Universität Hamburg [18] enthält eine Übersicht der unterschiedlichen Petri-Netz Arten und deren Klassifikation. Die Klassifikation umfasst drei Ebenen, wobei sich die Einordnung der Petri-Netze in die folgenden drei Ebenen gliedert. Die erste Ebene umfasst Petri-Netze deren Plätze Marken in Form von booleschen Werten aufnehmen können. Diese booleschen Marken sind nicht unterscheidbar. Die zweite Ebene enthält Petri-Netze, deren Plätze Marken in Form von integer Werten aufnehmen können. Diese Marken sind ebenfalls nicht unterscheidbar. Die dritte und letzte Ebene umfasst Petri-Netze, die mit High-Level Werten (zum Beispiel Objekten) umgehen können. Plätze, die Marken dieser Form aufnehmen können, sind unterscheidbar. In dieser Masterarbeit werden Platz/Transitions-Netze (P/T-Netze) verwendet und diese sind der zweiten Ebene

zugeordnet. P/T-Netze verarbeiten positive integer Werte und die Marken sind nicht voneinander unterscheidbar.

2.3 Platz/Transitions-Netz Struktur

Die hier verwendeten P/T-Netze bestehen aus den Grundelementen (Platz, Transition, Kante und Marke) und zusätzlichen Platz-Kapazitäten und Kantengewichten. Alle diese Elemente verarbeiten nur positive integer Werte, da ein P/T-Netz zur zweiten Ebene der Klassifikation zugeordnet wird. Eine Platz-Kapazität ist eine Begrenzung für einen Platz. Auf einem Platz dürfen nicht mehr Marken vorhanden sein als es seine Kapazität zulässt. Es ist möglich das die Kapazität unbeschränkt ist das heißt, die Kapazität braucht nicht beachtet werden. Ein Kantengewicht wird durch die Beschriftung einer Kante mit einer positiven integer Zahl ausgedrückt. Das Kantengewicht wird auch Vielfachheit genannt [42]. Wenn eine Kante keine Beschriftung trägt, wird standardmäßig ein Gewicht von eins angenommen. In einem Petri-Netz kann ein Kantengewicht von z.B. zwei, äquivalent durch zwei Kanten dargestellt werden. Formal wird ein P/T-Netz wie folgt ausgedrückt:

Definition 1 (Platz/Transitions-Netz) Ein 6-Tupel $N = (P, T, F, K, W, M_0)$ wird Platz/Transitions-Netz genannt falls [5, 42]:

- $P = \{p_1, p_2, \dots, p_n\}$ ist eine endliche Menge von Plätzen,
- $T = \{t_1, t_2, \dots, t_m\}$ ist eine endliche Menge von Transitionen,
- $F \subseteq (P \times T) \cup (T \times P)$ ist eine Menge von Kanten (Flussrelation),
- $K: P \rightarrow N^+ \cup \{\infty\}$ ist eine Kapazitätsfunktion,
- $W: F \rightarrow N^+$ ist eine Gewichtungsfunktion,
- M_0 : Anfangsmarkierung von N , für jeden Platz $p \in P$ gilt $M_0(p) \leq K(p)$.

Bezogen auf metabolische Stoffwechselwege, sind Platz-Kapazitäten eine Begrenzung der Metabolitmenge, das heißt, die Menge eines Metaboliten darf nicht größer als die Begrenzung sein. Kantengewichte werden als stöchiometrische Koeffizienten bezeichnet, die die Anzahl der Metabolite bestimmen, die an einer Reaktion teilnehmen. In Abbildung 2.1 ist ein P/T-Netz mit Kantengewichten und einer Platz-Kapazität dargestellt.

Eine spezielle Form das P/T-Netzes ist das gewöhnliche Petri-Netz. Das gewöhnliche Petri-Netz besitzt eine unbeschränkte Kapazitätsfunktion $\forall p \in P : K(p) = \infty$ und die Gewichtsfunktion ist eins $\forall f \in F : W(f) = 1$. Falls diese

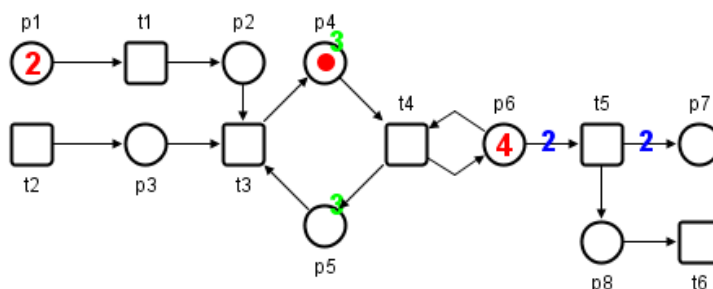


Abbildung 2.1: Ein P/T-Netz mit den vier Grundelementen (Platz, Transition, Kante und Marke) und zwei Kantengewichten (blau), zwei Platz-Kapazität (grün). Es existiert eine Schleife (self-loop) zwischen Transition t4 und Platz p6. Der Platz p1 und Transition t2 sind Quellknoten und Platz p7 und Transition t6 sind Senkenknoten.

Einschränkungen zutreffen, können die Kapazitätsfunktion und die Gewichtsfunktion weggelassen werden. Das Netz wird Formal wie folgt angegeben: 4-Tupel $N = (P, T, F, M_0)$ [5].

2.4 Platz/Transitions-Netz Dynamik

Der Vorteil eines Petri-Netzes (P/T-Netzes) ist, dass es das Verhalten eines Systems widerspiegelt. Wie schon in Abschnitt 2.1 erwähnt, sind Transitionen die aktiven Elemente. Transitionen können Marken verarbeiten und diese Verarbeitung wird Schalten genannt. Damit eine Transition schalten kann, muss diese Konzession haben (aktiviert sein). Diese Aktivierung ist an Bedingungen (Vor- und Nachbedingungen) geknüpft, die erfüllt sein müssen.

Eine Transition hat einen Vorbereich, der alle in diese Transition eingehenden Plätze umfasst (alle Vorplätze der Transition, $\forall p \in \bullet t$). Jeder Vorplatz ist eine Vorbedingung. Ebenso besitzt eine Transition einen Nachbereich ($\forall p \in t \bullet$), die Nachbedingung. Angenommen es handelt sich um ein gewöhnliches Petri-Netz, dann sind die Nachbedingungen für die Aktivierung der Transition nicht von Interesse. Gewöhnliche Petri-Netze besitzen keine Kapazitäten und somit muss der Nachbereich nicht auf die Überschreitung dieser Kapazität überprüft werden. In Abbildung 2.2 sind beispielsweise im Vorbereich der Transition t3, die Plätze p2, p3 und p5 enthalten. Im Nachbereich nur ein Platz (p4), der eine Kapazität von drei besitzt.

Übertragen auf die zu untersuchenden metabolischen Stoffwechselwege bedeutet das Schalten einer Transition, dass eine Reaktion erst dann ablaufen kann, wenn die Menge der Edukte ausreichend ist, (also sich eine entsprechende Anzahl Marken auf den Vorplätzen befinden), um die entsprechende Anzahl Produkte zu bilden. Das

Schalten einer Transition modelliert somit den Fluss einer Reaktion im metabolischen Stoffwechselweg.

Die Aktivierung einer Transition erfolgt nach einer bestimmten Regel (Schaltregel). Im Fall eines gewöhnlichen Petri-Netzes ist die schwache Schaltregel ausreichend, da keine Kapazität im Nachbereich beachtet werden muss.

Transitionsaktivierungsbedingung der schwachen Schaltregel:

$$\forall p \in \bullet t : M(p) > 0$$

Die aktivierten Transitionen verändern die Markierung des gewöhnlichen Petri-Netzes sobald sie schalten. Die Änderungen erfolgen anhand dieses Schemas [18]:

$$M'(p) = \begin{bmatrix} M(p) - 1 & \text{falls } (p \in \bullet t) \wedge (p \notin t\bullet) \\ M(p) + 1 & \text{falls } (p \in t\bullet) \wedge (p \notin \bullet t) \\ M(p) & \text{sonst} \end{bmatrix}$$

Handelt es sich um ein P/T-Netz, das nicht gewöhnlich ist, kann die schwache Schaltregel nicht angewendet werden. In diesem Fall kommt die starke Schaltregel zum Einsatz. Die starke Schaltregel beachtet die Kapazitäten. Bei dieser Regel wird der Vor- und der Nachbereich einer Transition ausgewertet. Transitionsaktivierungsbedingung der starken Schaltregel [10]:

$$\forall p \in P : W(p, t) \leq M(p) \leq K(p) - W(t, p)$$

Die starke Schaltregel aktiviert die Transition, wenn im Vorbereich genügend Marken vorhanden sind und durch das Schalten der Transition im Nachbereich weniger als, oder genau die Anzahl Marken vorhanden ist, die die Kapazität angibt. Die aktivierten Transitionen verändern die Markierung des P/T-Netzes sobald sie schalten. Die Änderungen erfolgen anhand dieses Schemas:

$$M'(p) = \begin{bmatrix} M(p) - W(p, t) & \text{falls } (p \in \bullet t) \wedge (p \notin t\bullet) \\ M(p) + W(t, p) & \text{falls } (p \in t\bullet) \wedge (p \notin \bullet t) \\ M(p) & \text{sonst} \end{bmatrix}$$

In Petri-Netzen können Nebenbedingungen auftreten. Eine Nebenbedingung ist eine Schleife (self-loop). In einer Schleife ist der Vorbereich der Transition auch gleichzeitig der Nachbereich, wie in Abbildung 2.2 zwischen Transition t_4 und Platz p_6 dargestellt ist. Petri-Netze, die mindestens eine Schleife enthalten, werden als nicht reine Petri-Netze bezeichnet. Eine Transition die in einer solchen Schleife involviert ist und die Anzahl der Marken die Kapazitätsgrenze erreicht hat, kann

aus den folgenden Gründen mit der starken Schaltregel nicht aktiviert werden. Diese Schaltregel beachtet nicht, dass bei einer Schleife Marken aus dem Vorbereich entfernt werden bevor sie dem Nachbereich hinzugefügt werden und somit die Kapazität nicht überschritten wird (symmetrische Überprüfung). Aus diesem Grund gibt es die starke Schaltregel in einer abgewandelten Form. Diese Schaltregel wird dann als starke Schaltregel mit Nebenbedingungen bezeichnet. Transitionsaktivierungsbedingung der starken Schaltregel mit Nebenbedingungen [10]:

$$\forall p \in P : W(p, t) \leq M(p) \leq K(p) - W(t, p) + W(p, t)$$

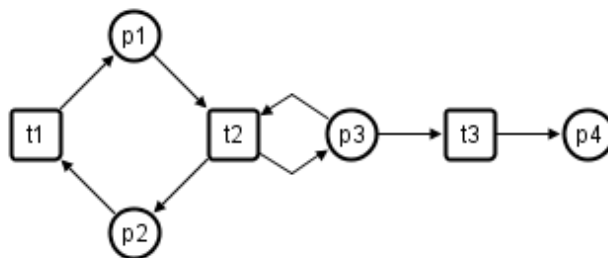
Die starke Schaltregel mit Nebenbedingungen aktiviert die Transition, wenn im Vorbereich genügend Marken vorhanden sind und nach dem Schalten die resultierende Markierung regelkonform ist. Diese Regel ist asymmetrisch, denn die Transition entfernt beim Schalten erst eine oder mehrere Marken aus dem Vorbereich und überführt diese dann erst in den Nachbereich. So ist es möglich das auch eine Schleife aktiviert wird. Die aktivierten Transitionen verändern die Markierung des P/T-Netzes sobald sie schalten, die Änderungen erfolgen anhand dieses Schemas [42]:

$$M'(p) = \left[\begin{array}{ll} M(p) - W(p, t) + W(t, p) & \text{falls } (p \in \bullet t) \wedge (p \in t \bullet) \\ M(p) - W(p, t) & \text{falls } (p \in \bullet t) \wedge (p \notin t \bullet) \\ M(p) + W(t, p) & \text{falls } (p \in t \bullet) \wedge (p \notin \bullet t) \\ M(p) & \text{sonst} \end{array} \right]$$

2.5 Analysemethoden

Eine große Stärke von Petri-Netzen ist die Simulierbarkeit von Systemen. So ist es möglich zu bestimmen, ob ein System die gewollte Verhaltensweise erfüllt. Die Simulation erfolgt anhand bestimmter Schaltregeln. Die somit entstehende Dynamik ist abhängig von der Struktur und der vorhandenen Marken im Petri-Netz. Das Simulieren ist ein zeitaufwändiger Prozess, der die Logik des Menschen benötigt um das Verhalten zu bestimmen und daraus Eigenschaften abzuleiten. Dabei sind einige Eigenschaften vom Petri-Netz durch die Struktur vorbestimmt, andere entstehen durch die vorhandenen Marken und deren Verteilung. Die Analysemethoden unterstützen den Menschen bei der Bestimmung von Petri-Netz Eigenschaften.

In den letzten Jahrzehnten sind verschiedene Analysemethoden entwickelt worden, um Eigenschaften von Petri-Netzen zu berechnen. Die Analysemethoden werden in zwei Gruppen unterteilt, statische Analysemethoden und dynamische Analysemethoden. Die Statischen Analysemethoden umfassen unter anderem die Strukturanalyse



	t1	t2	t3
p1	1	0	0
p2	0	1	0
p3	0	1	0
p4	0	0	1

(a) Vorwärtsmatrix

	t1	t2	t3
p1	0	1	0
p2	1	0	0
p3	0	1	1
p4	0	0	0

(b) Rückwärtsmatrix

	t1	t2	t3
p1	1	-1	0
p2	-1	1	0
p3	0	0	-1
p4	0	0	1

(c) Inzidenzmatrix

Abbildung 2.2: Ein Beispiel Petri-Netz bestehend aus drei Transitionen (t1 - t3) und vier Plätzen (p1 - p4) graphisch dargestellt (oben), in Matrizenschreibweise (unten).

und die Invariantenanalyse und eine der dynamischen Analysemethoden ist die Erreichbarkeitsanalyse.

Zur Erläuterung der Analysemethoden bedarf es einer Erweiterung der bereits eingeführten P/T-Netz Definition aus Abschnitt 2.3. Die Definition besagt $N = (P, T, F, K, W, M_0)$ und beinhaltet unter anderem die Flussrelation $F \subseteq (P \times T) \cup (T \times P)$. In dieser Flussrelation sind zwei Funktionen enthalten, die Input Funktion ($I = P \times T$) und die Output Funktion ($O = T \times P$), die den möglichen Markenfluss vorgeben. Der Markenfluss wird von der Input Funktion zwischen Plätzen und Transitionen beschrieben. Dementsprechend wird der Markenfluss von der Output Funktion zwischen Transitionen und Plätzen beschrieben. Aus Sicht einer Transition sind die ausgehenden Kanten (Output Funktion) die „Vorwärtskanten“ und die eingehenden Kanten (Input Funktion) die „Rückwärtskanten“.

Die Struktur eines Petri-Netzes kann als $n \times m$ Matrix dargestellt werden. Für jeden Platz (n) im Petri-Netz existiert eine Zeile in der Matrix und für jede Transition (m) existiert in der Matrix eine Spalte. In Abbildung 2.2 ist ein Petri-Netz graphisch und mit den zugehörigen (a) Vorwärts- und (b) Rückwärtsmatrizen dargestellt. Die Matrizen sind wie folgt zu interpretieren. Im Petri-Netz besitzt Transition t1 nur eine ausgehende Kante zu Platz p1, somit befindet sich in der Vorwärtsmatrix eine eins zwischen t1 und p1. Transition t2 besitzt zwei ausgehende Kanten zu p2 und p3. In

der Vorwärtsmatrix befindet sich deshalb an den Stellen t_2, p_2 und t_2, p_3 jeweils eine eins. Ebenso ist die ausgehende Kante von t_3 zu p_4 in der Vorwärtsmatrix erkennbar.

Im Gegensatz zur Vorwärtsmatrix sind in der Rückwärtsmatrix die eingehenden Kanten einer Transition erkennbar. Somit ist aus der Rückwärtsmatrix ablesbar, dass die Transition t_1 , eine eingehende Kante von Platz p_2 besitzt. Ebenso sind die eingehenden Kanten von t_2 und t_3 in der Rückwärtsmatrix ablesbar. Die hier genannte Definition von Vorwärts- und Rückwärtsmatrix aus Sicht einer Transition, wird in manchen Büchern aus der Sicht eines Platzes beschrieben. Dementsprechend wird die Struktur des Petri-Netzes umgekehrt wiedergegeben.

Eine weitere Matrix, die ebenfalls die Struktur eines Petri-Netzes beschreibt, ist die Inzidenzmatrix (C). Diese Matrix entsteht durch Subtraktion der Rückwärtsmatrix von der Vorwärtsmatrix ($C = O - I$). Anstelle von zwei Matrizen ist also eine Matrix ausreichend um die Petri-Netz Struktur zu erfassen. Jedoch gibt es eine Einschränkung, im Petri-Netz auftretende Schleifen (self-loops) sind in der Inzidenzmatrix nicht mehr sichtbar. Die Inzidenzmatrix gibt das Petri-Netz also nicht eindeutig wieder. In Abbildung 2.2 wird dieser Unterschied sichtbar. In der Vorwärtsmatrix (a) ist an Position (t_2, p_3) ein positiver Wert. Ebenso verhält es sich in der Rückwärtsmatrix (b). Jedoch ist in der Inzidenzmatrix (c) eine Null an Position (t_2, p_3) und somit die Schleife zwischen t_2 und p_3 nicht ersichtlich. Ein Petri-Netz wird nur eindeutig durch die Inzidenzmatrix beschrieben, wenn es ein reines Petri-Netz ist (keine Schleifen).

2.5.1 Strukturanalyse

Anhand der Struktur eines Petri-Netzes können strukturelle Eigenschaften (siehe Abschnitt 2.5.4) abgeleitet werden, die Rückschlüsse auf dynamische Eigenschaften (siehe Abschnitt 2.5.5) zulassen. In der Strukturanalyse braucht der Zustandsraum (state space) nicht konstruiert werden. Der Zustandsraum umfasst alle von der Anfangsmarkierung aus erreichbaren Zustände, die durch das Schalten von Transitionen erzeugt werden können. Strukturelle Eigenschaften sind zum Beispiel: gewöhnlich (ordinary), gleichartig (homogenous), bewahrend (conservative), rein (pure), statisch konfliktfrei (static conflict-free), frei von Randknoten (free of boundary nodes), Verklemmung (deadlock) und Fallen (traps) [25]. Manche Eigenschaften spiegeln sich im Namen der Petri-Netze wider. Zum Beispiel verbirgt sich die Eigenschaft gewöhnlich im Namen des gewöhnlichen Petri-Netzes. Dementsprechend erfüllt dieses Petri-Netz die „gewöhnlichen“ Eigenschaften.

2.5.2 Invariantenanalyse

Eine Invariante definiert ein verbundenes Teilnetzwerk bestehend aus den Vorplätzen, Nachplätzen, Transitionen und allen Kanten dazwischen. Invarianten können durch die Lösung eines homogenen Gleichungssystems berechnet werden ($C * x = 0$), wobei C die Inzidenzmatrix und x der Invariantenvektor ist. Solch ein homogenes Gleichungssystem kann durch das Gauss-Verfahren gelöst werden. Eine Verbesserung des Gauss-Verfahrens wird von Martinez und Silva in [30] besprochen. Eine Invariante ist nur von der Struktur eines Petri-Netzes abhängig. Dementsprechend hat die Markenbelegung (Markierung) und die Dynamik keinen Einfluss auf die Invarianten eines Petri-Netzes. Die Konstruktion des Zustandsraumes ist für die Invariantenanalyse ebenfalls nicht erforderlich. Jedoch kann die Anzahl der Invarianten exponentiell mit der Größe des Petri-Netzes wachsen [25].

Mit Hilfe der Invarianten können Aussagen über die Beschränktheit und Lebendigkeit eines Petri-Netzes erfolgen. Bezogen auf biologische Netzwerke wird die Invariantenanalyse zum Untersuchen der biologischen Plausibilität genutzt. Mit anderen Worten, ist das biologische Netzwerk valide und es wird das zu erwartende Verhalten erzielt. Es werden verschiedene Arten von Invarianten [25] unterschieden, diese sind die Platz- und die Transitionsinvarianten. Die Invarianten werden in biologischen Netzwerken genutzt, um diese zu validieren und die Invarianten für eine geeignete biologische Interpretation heranzuziehen.

Platzinvarianten Die Platzinvariante (P-Invariante) besteht aus einer Menge von Plätzen, deren gewichtete Summe der Marken beim Schalten beliebiger Transitionen konstant bleibt [26]. Diese gewichtete Markenmenge wird beginnend von der Anfangsmarkierung bei jedem Schaltvorgang aufrechterhalten. Zu den P-Invarianten kann mit Hilfe einer Formel, abhängig von der Anfangsmarkierung, eine Konstante berechnet werden. Die Formel lautet im Allgemeinen $\sum_{i=1}^k n_i * M_0(p_i) = c$, wobei k die Anzahl der Plätze, n_i die i -te P-Invariante und $M_0(p_i)$ die Anfangsmarkierung vom i -ten Platz ist.

Eine P-Invariante ist gleichzeitig eine strukturelle Verklemmung und eine Falle. Aber umgekehrt ist nicht jede Menge von Plätzen, die gleichzeitig eine Verklemmung und eine Falle ist, eine P-Invariante. So kann eine P-Invariante Aussagen über lokale Eigenschaften eines Petri-Netzes machen. Globale Eigenschaften werden jedoch erst ersichtlich, wenn das gesamte Petri-Netz von P-Invarianten überdeckt ist (im folgenden mit CPI (covered by place invariants) abgekürzt). Damit ein Petri-Netz CPI ist, muss jeder Platz zu einer P-Invariante gehören. Nur ein platzberandetes Petri-Netz kann CPI sein. Eine P-Invariante kann Aussagen über Eigenschaften, wie

Lebendigkeit und Beschränktheit über sich treffen. Ist das Petri-Netz CPI, gelten die Aussagen für das gesamte Petri-Netz. Plätze die zu einer nicht negativen P-Invariante gehören, sind beschränkt. Nicht negative P-Invariante bedeutet, alle Zahlen im Vektor der P-Invariante sind größer gleich Null. Ist ein Petri-Netz von nicht negativen P-Invarianten überdeckt, ist es ebenfalls beschränkt [25]. Damit eine P-Invariante lebendig sein kann, muss bei der Anfangsmarkierung mindestens eine Marke auf mindestens einem Platz dieser P-Invariante vorhanden sein.

Die P-Invarianten können als Erhaltungsregeln eines biologischen Netzwerks interpretiert werden, denn diese veranschaulichen eine Konservierung bestimmter Metabolite [3]. Ein Beispiel kann in Kapitel 5.1, Abbildung 5.1, betrachtet werden.

Transitionsinvarianten Eine Transitionsinvariante (T-Invariante) besteht aus einer Menge von Transitionen, deren gewichtete Summe der Vor- und Nachbereiche, identisch ist [44]. Mit anderen Worten, wenn alle Transitionen aus der Menge der T-Invariante schalten, verändert sich die Markierung des Petri-Netzes nicht. Es gibt T-Invarianten in unterschiedlichen Ausprägungen. Hier werden zwei T-Invariantentypen vorgestellt, die triviale und die minimale Invariante. Triviale T-Invarianten bestehen aus zwei Transitionen, eine für das Vorwärtsschalten und die Andere für das Rückwärtsschalten. Beim Schalten wird ein Schritt durch die „vorwärts“ Transition ausgeführt, der gleich wieder mit dem Schalten der „rückwärts“ Transition rückgängig gemacht wird. Solche trivialen T-Invarianten sind für die biologische Interpretation ohne Bedeutung [25]. Eine andere Form der T-Invariante ist die minimale T-Invariante. Eine Minimale T-Invariante darf keine Kombination von Anderen T-Invarianten sein und der größte gemeinsame Teiler ist eins. Diese minimalen T-Invarianten werden auch elementare Flussmoden genannt. Eine Erläuterung der elementaren Flussmoden kann in [40] nachgeschlagen werden.

Eine T-Invariante kann (ebenfalls wie eine P-Invariante), Aussagen über lokale Eigenschaften eines Petri-Netzes treffen. Globale Eigenschaften werden jedoch erst ersichtlich, wenn das gesamte Petri-Netz von T-Invarianten überdeckt ist (im folgenden mit CTI (covered by transition invariants) abgekürzt). Damit ein Petri-Netz CTI ist, muss jede Transition zu einer T-Invariante gehören. Ein Petri-Netz muss transitionsberandet sein, damit es CTI sein kann [26]. Wenn ein Petri-Netz mit nicht negativen T-Invarianten überdeckt ist (Definition siehe P-Invarianten), kann es lebendig und beschränkt sein.

Im Kontext biologischer Netzwerke entsprechen T-Invarianten Flussverteilungen im stabilen Zustand (steady state), dass heißt, alle Metabolite haben ein mögliches dynamisches Konzentrationsgleichgewicht erreicht. Nach Baldan et al. [3] existieren für die biologische Interpretation zwei Möglichkeiten. Einerseits stellen die Reaktionen einer

T-Invariante eine Menge von Transitionen dar, deren partiell geordnete Ausführung eine initiale Markierung reproduziert. Andererseits könnten die Reaktionen einer T-Invariante als relative Schaltrate von Transitionen interpretiert werden, die permanent und parallel auftreten und somit einen stabilen Zustand charakterisieren. Ist ein biologisches Netzwerk von T-Invarianten überdeckt, können alle enthaltenen Stoffwechselwege durch entsprechende T-Invarianten dargestellt werden. Also werden alle Reaktionen jeweils einem Stoffwechselweg (einer T-Invariante) zugeordnet. Ein Beispiel kann in Kapitel 5.1, Abbildung 5.4 betrachtet werden. Zusammenfassend werden die beiden Invariantentypen in Tabelle 2.2 mit Stoffwechselwegen in Beziehung gesetzt.

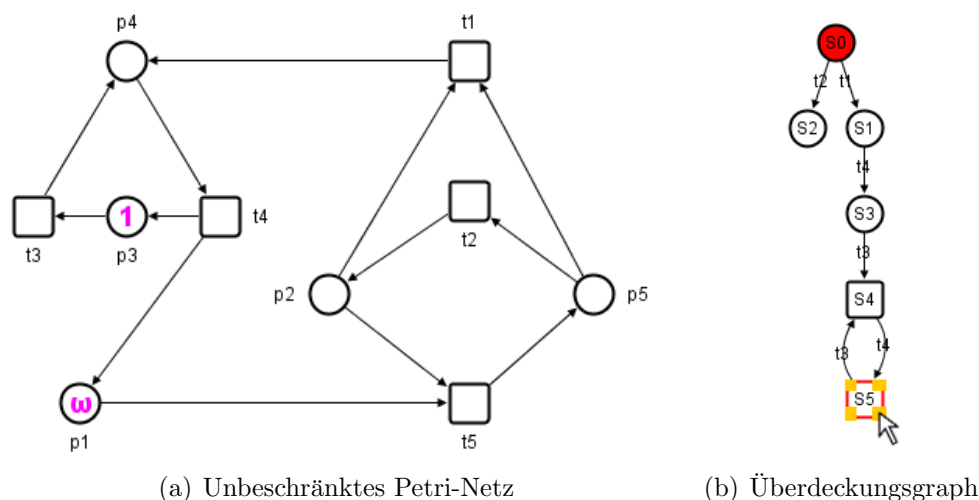
Stoffwechselweg	Petri-Netz Invarianten
Metabolit-Konservierung	P-Invariante
Flussverteilung im stabilen Zustand	T-Invariante
Elementarer Flussmodus	Minimale T-Invariante

Tabelle 2.2: Beziehung zwischen Stoffwechselweg und Petri-Netz Invarianten, in Anlehnung an [3].

2.5.3 Erreichbarkeitsanalyse

Unter der Erreichbarkeitsanalyse versteht sich die Berechnung aller erreichbaren Zustände eines Petri-Netzes, wobei überprüft wird, ob gewisse Zustände erreichbar sind. Die Ergebnisse dieser Analyse werden graphisch als Erreichbarkeitsgraph repräsentiert. Der Erreichbarkeitsgraph umfasst alle möglichen Zustände (Zustandsraum) eines Petri-Netzes, ausgehend von einer Anfangsmarkierung. Die Knoten im Erreichbarkeitsgraphen entsprechen Zuständen (mit S_0, S_1, \dots bezeichnet), also einer Markierung des Petri-Netzes, die nach einer Schaltabfolge von Transitionen erreicht wird. Die Kanten zwischen den Zuständen repräsentieren Transitionen des Petri-Netzes, die schalten um von einem Zustand in den Folgenden zu gelangen. Zyklen im Erreichbarkeitsgraphen entsprechen T-Invarianten, die bei der aktuellen Markierung, das Schalten der Transitionen dieser T-Invariante erlauben. Nur ein beschränktes Petri-Netz besitzt einen endlichen Erreichbarkeitsgraphen. Ein Beispiel kann in Kapitel 5.1, Abbildung 5.3 betrachtet werden.

Viele Petri-Netze sind nicht beschränkt, aus diesem Grund ist es möglich einen Überdeckungsgraphen zu erzeugen. Der Überdeckungsgraph führt eine pseudo Endlichkeit ein, die durch das Symbol ω dargestellt wird. Ein Platz auf dem sich das ω befindet, ist nicht beschränkt. Somit wird der Überdeckungsgraph für einen unendlichen Zustand endlich gehalten. Der Überdeckungsgraph ist eine Abstraktion des



(a) Unbeschränktes Petri-Netz

(b) Überdeckungsgraph

Abbildung 2.3: Unbeschränktes Petri-Netz (a) mit dazugehörigem Überdeckungsgraphen (b) [17]. Die Knoten des Überdeckungsgraphen entsprechen den möglichen Zuständen des Petri-Netzes und werden mit S0 - S5 bezeichnet. Die letzten beiden Zustände im rechten Zweig des Überdeckungsgraphen (eckige Knotenform), kennzeichnen unendliche Zustände um für einen unendlichen Zustand einen endlichen Überdeckungsgraphen darstellen zu können. Die Kanten entsprechen den Transitionen (t1 - t5), die schalten um von Einem zum anderen Zustand zu gelangen. Im Überdeckungsgraphen ist der Zustand S5 selektiert, um interaktiv die Markierung des Zustands im Petri-Netz zu visualisieren (Platz p1 enthält ein ω).

Erreichbarkeitsgraphen, indem einige Informationen verloren gehen. Es werden also eher Schranken als exakte Ergebnisse ermittelt [3]. In einem beschränkten Petri-Netz entspricht der Erreichbarkeitsgraph dem Überdeckungsgraphen [42].

Das Beispiel in Abbildung 2.3 (a) zeigt ein unbeschränktes Petri-Netz und (b) den zugehörigen Überdeckungsgraphen. Durch die angegebene Anfangsmarkierung M_0 stehen die Transitionen t1 und t2 im Konflikt, dieses Verhalten ist durch die Verästelung nach der Anfangsmarkierung im Überdeckungsgraphen zu erkennen. Sobald t2 schaltet kann t1 nicht mehr schalten und ist somit nicht mehr aktiv. Auf Platz p2 liegen dann zwei Marken und keine Transition kann mehr schalten. Das System befindet sich in einer Verklemmung (deadlock, siehe Abschnitt 2.5.4). Falls t1 zuerst schaltet, wird eine Marke auf p4 erzeugt und die Marken auf p2 und p5 verbraucht. Die Transition t4 ist jetzt als Einzige aktiv und kann schalten. Sobald dies geschieht, wird die Marke von p4 verbraucht und zwei Marken erzeugt, eine auf p3 und eine auf p1. Beim Schalten von t3 wird die Marke von p3 verbraucht und eine neue Marke auf p4 erzeugt. Jetzt ist wieder die Transition t4 aktiv und kann schalten. Dieses abwechselnde Schalten von t4 und t3 erzeugt immer wieder Marken auf p1. Diese unendliche Markenansammlung wird im Überdeckungsgraphen

Eigenschaften	Erreichbarkeitsgraph	Überdeckungsgraph
Erreichbarkeit	✓	
Beschränktheit	✓	✓
Lebendigkeit	✓	
Überdeckbarkeit	✓	
Reversibilität	✓	✓

Tabelle 2.3: Bestimmbare dynamische Eigenschaften von Petri-Netzen durch den Erreichbarkeits- und den Überdeckungsgraphen [17].

durch die eckige Knotenform, für die zwei letzten Zustände dargestellt. Mit Hilfe der Interaktionstechnik ist es möglich, den Zustand von S5 des Überdeckungsgraphen (b) im Petri-Netz (a) mit den jeweiligen Marken und auch ω zu betrachten.

Da der Überdeckungsgraph eine Abstraktion des Erreichbarkeitsgraphen darstellt, können mit seiner Hilfe nicht alle dynamischen Eigenschaften erhoben werden. Die Tabelle 2.3 zeigt eine Übersicht, welche dynamischen Eigenschaften mit dem Erreichbarkeitsgraphen und welche mit Überdeckungsgraphen berechnet werden können.

2.5.4 Strukturelle Petri-Netz Eigenschaften

Eine Strukturelle Eigenschaft ist ausschließlich von der Struktur des Petri-Netzes abhängig. Im folgenden werden verschiedene strukturelle Eigenschaften dargestellt.

Statischer Konflikt Ein statischer Konflikt kann nur entstehen wenn im Petri-Netz mindestens zwei Transitionen existieren, die sich einen Vorplatz teilen (siehe Abbildung 2.4 (a)). Die beiden Transitionen (t1 und t2) teilen sich den Vorplatz (p1), diese Struktur ist ein Konflikt.

Strukturelle Verklemmung Eine strukturelle Verklemmung besteht aus einer nicht leeren Menge von Plätzen $D \subseteq P$. Jede Transition die Marken in die Menge D hinzufügt, entnimmt auch Marken aus der Menge D , $\bullet D \subseteq D \bullet$ [25]. Sobald keine Marken mehr in der strukturellen Verklemmung vorhanden sind, können die Transitionen nicht mehr schalten. Mit anderen Worten, aus einer strukturellen Verklemmung werden mehr Marken entnommen als hinzugefügt (siehe Abbildung 2.4 (b)). Sobald eine strukturelle Verklemmung bei einer Markierung alle Marken verloren hat, kann diese bei allen Folgemarkierungen keine Marken mehr erhalten.

In biologischen Netzwerken könnten strukturelle Verklemmungen den Verbrauch von gespeicherten Metaboliten anzeigen. Ein Petri-Netz ohne strukturelle Verklemmungen ist lebendig [25].

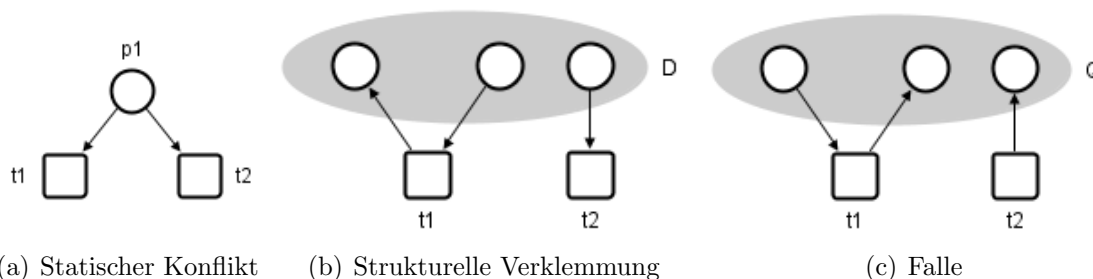


Abbildung 2.4: Statischer Konflikt, Strukturelle Verklemmung und Falle. (a) Statischer Konflikt, die Transitionen t_1 und t_2 besitzen denselben Vorplatz p_1 . (b) Strukturelle Verklemmung, die Transitionen t_1 und t_2 entnehmen Marken aus der Menge D aber nur Transition t_1 fügt eine Marke der Menge D hinzu (in Anlehnung an [17]). (c) Falle, die Transitionen t_1 und t_2 fügen der Menge Q Marken hinzu aber nur Transition t_1 entnimmt eine Marke der Menge Q (in Anlehnung an [17]).

Falle Eine Falle besteht aus einer nicht leeren Menge von Plätzen $Q \subseteq P$. Jede Transition, die Marken aus der Menge Q entnimmt, fügt die Marken der Menge Q wieder hinzu $Q \bullet \subseteq \bullet Q$ [25]. Mit anderen Worten, in eine Falle werden mehr Marken hinzugefügt als entnommen (siehe Abbildung 2.4 (c)). Dennoch können die Marken in einer Falle weniger werden, aber eine Falle kann nicht alle Marken verlieren. Wenn eine Falle bei einer Markierung eine Marke enthält, enthält die Falle wenigstens eine Marke bei allen Folgemarkierungen.

In biologischen Netzwerken könnten Fallen das Speichern von Metaboliten anzeigen.

2.5.5 Dynamische Petri-Netz Eigenschaften

Eine dynamische Eigenschaft von Petri-Netzen ist von der Anfangsmarkierung abhängig. Im Folgenden werden verschiedene dynamischen Eigenschaften dargestellt.

Lebendigkeit Eine Transition wird lebendig genannt, wenn von einer Markierung aus eine Folgemarkierung existiert, bei der die Transition wieder Konzession hat. Sind alle Transitionen in einem Petri-Netz lebendig, wird das Petri-Netz lebendig genannt. Die Lebendigkeit ist eine von vielen Systemen erwünschte Eigenschaft. Gleichzeitig ist sie eine strenge Eigenschaft, die nicht für große Petri-Netze bestimmt werden kann [31].

Es ist möglich, unter anderem, die Lebendigkeit über Transitionen zu bestimmen. Dafür werden drei Konzepte unterschieden: Lebendigkeit, Ebenen der Lebendigkeit

keit und strikte Ebenen der Lebendigkeit. Da Lebendigkeit von großen Systemen nicht bestimmt werden kann, wurden fünf Ebenen der Lebendigkeit definiert, die unterschiedlich streng sind. Diese Ebenen werden mit L0-lebendig bis L4-lebendig bezeichnet. L4-lebendig ist dabei der strengste Ebene und gleichbedeutend mit der Lebendigkeit. Sobald alle Transitionen in einem Petri-Netz eine bestimmte Ebene besitzen, erhält das gesamte Petri-Netz diese Ebene der Lebendigkeit. Das Konzept der strikten Ebenen der Lebendigkeit ist strenger als die Ebenen der Lebendigkeit, aber schwächer als die Lebendigkeit selbst. Weitere Informationen über die Lebendigkeit, Ebenen der Lebendigkeit und strikte Ebenen der Lebendigkeit können in [31] nachgeschlagen werden.

Die Lebendigkeit ist für biologische Netzwerke eine wichtige Eigenschaft. Ein biologisches Netzwerk modelliert als Petri-Netz, besitzt keine Verklemmung und läuft ohne Unterbrechung ab.

Beschränktheit Ein Petri-Netz ist beschränkt wenn eine positive Zahl k existiert, die eine Obergrenze für die Anzahl der Marken auf jedem Platz, in allen Zuständen des Netzes repräsentiert (k -beschränkt). Wenn ein Petri-Netz zu jeder Anfangsmarkierung beschränkt ist, wird es strukturell beschränkt genannt. Beschränkte Petri-Netze besitzen endlich viele erreichbare Zustände.

Petri-Netze von biologischen Netzwerken sind oftmals unbeschränkt, denn es existiert ein Austausch von Metaboliten mit der Umgebung, sodass einige Metabolite in unbeschränkten Mengen vorhanden sind. Diese Unbeschränktheit kann durch den Überdeckungsgraphen dargestellt werden. Aber dieser Graph ist weniger aussagekräftig als der Erreichbarkeitsgraph, siehe Tabelle 2.3.

Sicherheit Die Sicherheit ist ein Spezialfall der Beschränktheit. Ein Platz wird sicher (safe) genannt, wenn er höchstens eine Marke zu jedem beliebigen Zeitpunkt enthält. Ein Petri-Netz heißt sicher (1-beschränkt), wenn jeder Platz in dem Petri-Netz sicher ist.

Reversibilität Ein Petri-Netz ist reversibel wenn sein Erreichbarkeitsgraph stark zusammenhängend ist [42]. Ein Graph wird als stark zusammenhängend bezeichnet, wenn von jedem Knoten aus jeder andere erreicht werden kann, auch wenn es keinen direkten Pfad gibt.

Dynamischer Konflikt Ein dynamischer Konflikt kann nur entstehen wenn ein statischer Konflikt besteht (siehe Abschnitt 2.5.4). Sobald eine Transition einer anderen Transition die Konzession entzieht, entsteht ein dynamischer Konflikt. In

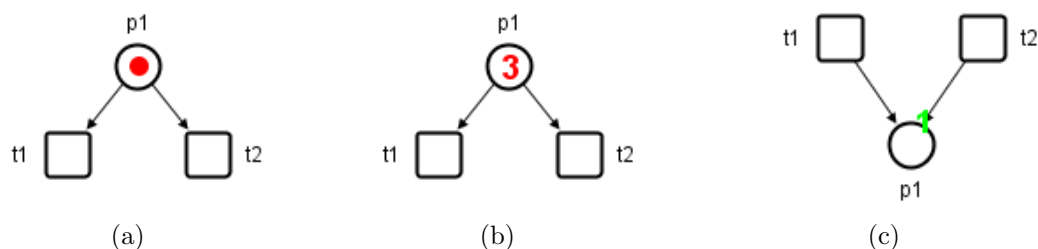


Abbildung 2.5: Dynamischer Konflikt. (a) Aus einem statischen Konflikt resultiert ein dynamischer Konflikt. Sobald eine Transition schaltet, ist keine Marke mehr vorhanden und die andere Transition kann nicht mehr Schalten. (b) Beide Transitionen können einmal Schalten ohne das eine Transition der anderen die Konzession entzieht. (c) Jede Transition könnte für sich schalten, aber sobald eine Transition geschaltet hat, ist die Kapazitätsgrenze erreicht und die andere Transition kann nicht mehr Schalten.

Abbildung 2.5 (a) ist ein dynamischer Konflikt, der den Vorbereich betrifft, dargestellt. Die Transitionen (t1 und t2) besitzen den selben Vorplatz (p1), der nur eine Marke enthält. Jede Transition könnte für sich schalten, aber sobald eine Transition geschaltet hat, kann die andere Transition nicht mehr Schalten. In Abbildung 2.5 (b) ist dasselbe Petri-Netz dargestellt, aber p1 enthält jetzt drei Marken. Bei dieser Markierung des Petri-Netzes besteht kein dynamischer Konflikt. Sobald die Transitionen t1 und t2 geschaltet haben, entsteht die Markierung in Abbildung (a) und es entsteht wieder ein dynamischer Konflikt. Ein dynamischer Konflikt kann ebenso den Nachbereich betreffen. In Abbildung 2.5 (c) ist ein dynamischer Konflikt, der den Nachbereich betrifft, dargestellt. Die Transitionen (t1 und t2) besitzen den selben Nachplatz (p1), der nur eine Kapazität von eins besitzt. Jede Transition könnte für sich schalten, aber sobald eine Transition geschaltet hat, ist die Kapazitätsgrenze erreicht und die andere Transition kann nicht mehr Schalten. Dynamische Konflikte verursachen alternative Systemverhalten, die nicht-deterministisch sind [25]. Das alternative Schaltverhalten wird ausführlich am Beispiel des Stoffwechselweges Glykolyse in Kapitel 5.1 und Abbildung 5.3 erläutert.

3 Aufgabenstellung

In diesem Kapitel wird die Aufgabenstellung detailliert beschrieben. Anschließend wird eine Übersicht bestehender Programme und Bibliotheken gegeben, die die Basis für die Simulations- und Analysemethoden des Add-ons bilden sollen.

3.1 Beschreibung der Aufgabenstellung

Im Rahmen dieser Masterarbeit soll die Software VANTED [21] erweitert werden, um biologische Netzwerke in Form von Petri-Netzen Erstellen, Simulieren und Analysieren zu können. VANTED (Visualization and Analysis of Networks containing Experimental Data) wurde am IPK (Leibniz-Institut für Pflanzen-genetik und Kulturpflanzenforschung) entwickelt und unterstützt bereits das Erstellen, Editieren, Visualisieren und Analysieren biologischer Experimentaldaten. Die ermittelten Daten können mit VANTED als Netzwerke geladen oder auf bestehende Netzwerke gemappt werden. VANTED unterstützt für den Im- und Export verschiedene Dateiformate (GML, GraphML, DOT, KGML, SBML, CSV, XLS) [24]. Des Weiteren sind zahlreiche Analysemethoden implementiert, die erstellte oder geladene Daten verarbeiten. Damit VANTED um zusätzliche Funktionalitäten erweitert werden kann, existiert eine Schnittstelle, die es ermöglicht, Add-ons einzubinden. Diese Schnittstelle wird bei der Entwicklung des Petri-Netz Add-ons genutzt, sodass die grundlegende Funktionalität von VANTED verwendet und erweitert wird.

Das Petri-Netz Add-on soll diskrete und kontinuierliche Petri-Netze verarbeiten können. Die diskreten Petri-Netze sollen nur positive Integer-Werte, kontinuierliche Petri-Netze nur positive Double-Werte besitzen. Die Petri-Netz Elemente Marken, Kantengewichte und Platz-Kapazitäten besitzen somit positive Integer-Werte in diskreten Petri-Netzen und positive Double-Werte in kontinuierlichen Petri-Netzen. Die Dynamik dieser beiden Petri-Netz Arten erfolgt identisch in diskreten Schritten. Das zu entwickelnde Petri-Netz Add-on, soll die beiden Petri-Netz Arten simulieren und analysieren können. Die Simulation soll in Einzel- und Mehrschritten ermöglicht werden. Die Analysen der Petri-Netze soll die Erreichbarkeit, Platz- und Transitionsinvarianten umfassen. Da Petri-Netze seit ca. 50 Jahren existieren und bereits viele Implementierungen vorhanden sind, sollen Quellcode offene und in Java

geschriebene Bibliotheken oder Programme benutzt werden, die die Simulation und die Analyse der Petri-Netze bereitstellen. Die zu implementierenden Analyse- und Simulationsmethoden sollen auf biologische Netzwerke verschiedener Komplexität anwendbar sein. Weiterhin soll das Petri-Netz Add-on für den Austausch mit anderen Programmen den Im- und Export von Petri-Netzen unterstützen. Ein wichtiger Aspekt für ein geeignetes Austauschformat der Petri-Netze ist ein standardisiertes Dateiformat, das von vielen Programmen unterstützt wird.

Zur Einarbeitung in die Petri-Netz Theorie wurde eine weitreichende Recherche im Internet durchgeführt. Eine Internet Seite, die umfangreiche Informationen über Petri-Netze enthält, ist „Petri Nets World“ [18]. Auf dieser Seite werden die am häufigsten verwendeten und bekanntesten Petri-Netz Arten aufgelistet. Eine Klassifikation dieser Petri-Netz Arten, deren Definition, Schaltregeln und Eigenschaften, sowie kleine Beispiele werden ausführlich dargestellt. Des Weiteren sind Referenzen zu Büchern, Veröffentlichungen und Informationen über Petri-Netz Programme hinterlegt, sodass der interessierte Leser weiterführende Informationen nachschlagen kann. Ausgehend von dieser Informationsquelle konnte die folgende Übersicht bestehender Petri-Netz Programme und Bibliotheken erstellt werden.

3.2 Übersicht vorhandener Programme und Bibliotheken

Eine Recherche soll existierende, Quellcode offene und in Java implementierte Programme und Bibliotheken erfassen, die die Analyse- und Simulationsmethoden für Petri-Netze bereitstellen. Die Auswahl einer geeigneten Implementierung als Grundlage für das zu entwickelnde Add-on ist entscheidend für die Faktoren Zeit, Realisierung und Erweiterbarkeit des Add-ons. Es ist von großem Vorteil wenn bereits eine Bibliothek oder ein Programm existiert, das einen Großteil der benötigten Anforderungen erfüllt, noch aktiv gepflegt wird und weit verbreitet ist. Außerdem sollen die Methoden zur Analyse und Simulation von Petri-Netzen in das Add-on so problemlos wie möglich integriert werden. Aus diesem Grund sind Bibliotheken den Programmen vorzuziehen, da Bibliotheken im allgemeinen gut definierte Schnittstellen besitzen und diese ohne großen Aufwand integriert werden können. Eine Dokumentation zu den Schnittstellen (API - Application Programming Interface) der Bibliothek erleichtert zusätzlich den Integrationsprozess.

Im Gegensatz dazu sind Programme im Allgemeinen nicht dafür ausgelegt in weitere Programme integriert zu werden und besitzen keine oder nur wenige Schnittstellen. Eine Dokumentation der im Programm vorhandenen Schnittstellen und Programm-

teile ist wünschenswert. Falls keine Bibliothek für den vorgesehenen Zweck existiert, besteht die Möglichkeit, Teile aus Programmen zu benutzen. Diese Teile müssen aber an die eigene Implementierung angepasst werden. Je nach Programm kann die Anpassung unterschiedlich aufwändig sein.

Im Weiteren Verlauf dieses Kapitels werden Programme und Bibliotheken nur noch als Software bezeichnet, falls es zum Verständnis notwendig sein sollte, wird konkret Programm oder Bibliothek verwendet. Die in Tabelle 3.1 aufgelistete Software umfasst nur einen Teil der im Internet vorhandenen Petri-Netz Software. Ein Grund für die eingeschränkte Auflistung ist, dass nicht mehr alle Softwareprodukte erhältlich sind, obwohl diese noch auf einigen Internetseiten verlinkt sind oder die Anforderungen nicht erfüllt werden.

Die Softwarepakete JFern, HISim und Renew können diskrete und kontinuierliche Petri-Netze verarbeiten. Die Petri-Netze können nur simuliert werden, wobei die Simulation in Einzel- und Mehrschritten möglich ist. Jedoch unterstützt keine der drei Softwarepakete Petri-Netz Analysemethoden. Von diesen drei ist nur JFern ein Framework, welches leicht in ein bestehendes Programm eingebunden werden kann. Das Programm Renew ist ein Petri-Netz Simulator für Referenznetze. Referenznetze sind objektbasierte, gefärbte Petri-Netze, die Referenzen auf Netze als Marken enthalten und über synchrone Kanäle kommunizieren können [13]. Das Programm HISim ist ein Simulator für Hybride Petri-Netze, der basierend auf dem Programm Pipe2 entwickelt wurde [1].

Pipe2 und WoPeD können nur mit diskreten Petri-Netzen umgehen. Die Simulation ist ebenfalls in Einzel- und Mehrschritten möglich. WoPeD enthält nur die in den Anforderungen gestellte Erreichbarkeitsanalyse, wohingegen Pipe2 sowohl die Erreichbarkeit als auch die Invarianten analysieren kann. Generell ist WoPeD ein umfangreiches Programm, welches eigentlich eine erweiterte Form von Petri-Netzen anwendet, die Arbeitsablaufnetze genannt werden, um Arbeits- und Geschäftsabläufe zu Visualisieren, Simulieren und Analysieren. In WoPeD werden neben der Erreichbarkeitsanalyse hauptsächlich Analysen bezüglich Arbeitsablaufnetzen unterstützt.

Alle weiteren in der Tabelle zusammengefassten Softwarepakete unterstützen keine Analysemethoden und nur eingeschränkt kontinuierliche Petri-Netze, sowie die Simulation von Petri-Netzen. FERN und KIT-Horus simulieren diskrete Petri-Netze in Einzel- und Mehrschritten. FERN ist ein Framework zur Simulation und Evaluation stochastischer Reaktionsnetzwerke und visualisiert die Simulationsergebnisse mit Gnuplot. Des Weiteren existiert FERN als Cytoscape [41] und CellDesigner [14] Plugin und kann somit in andere Anwendungen integriert werden [12]. KIT-Horus

Name	Release	Lizenz	Petri-Netz Art		Simulation			Analysemethoden		
			Diskret	Kontinuierlich	Einzel- schritt	Mehr- schritt	Invarianten	Erreich- barkeit		
									P	T
Jfern	07.04.10	LGPL	✓	✓	✓	✓				
Tortuga	01.09.08	LGPL	✓		✓					
HISim	17.07.09	BSD	✓	✓	✓					
JARP	17.07.09	GPL	✓		✓					
PIPE2	14.09.11	OSL3.0	✓		✓		✓		✓	
Petri-LLD	02.03.11	GPL	✓		✓					
JPetriNet	17.07.09	AFL	✓		✓					
Renew	28.02.09	LGPL	✓	✓	✓					
FERN	15.03.11	LGPL	✓		✓					
PNEditor	01.08.10	GPLv3	✓		✓					
TAPAAL	12.04.11	OSL3+BSD	✓		✓					
WoPeD	05.07.11	LGPL	✓		✓				✓	
KIT-HORUS	30.08.10	EPL	✓		✓					

Tabelle 3.1: Auflistung der im Internet vorhandenen Programme und Bibliotheken. Die Abkürzungen der Lizenzen lauten wie folgt: LGPL = Lesser General Public License, BSD = Berkeley Software Distribution, GPL = General Public License, OSL = Open Software License, AFL = Academic Free License, EPL = Eclipse Public License. Hinweis: Die Petri-Netz Art „Kontinuierlich“ bezieht sich nur auf kontinuierliche Werte dieser Petri-Netze, die Schritte werden weiterhin als diskret angenommen.

ist eine Programmsammlung, die die Entwicklung und Simulation prozessorientierter Informationssysteme basierend auf Petri-Netzen unterstützt [23]. Tortuga [27], PetriLLD [7], JPetriNet [2], PNEditor [37], JARP [33] und TAPAAL [8] simulieren nur diskrete Petri-Netze in Einzelschritten und werden deshalb nicht näher beschrieben.

Nach genauer Untersuchung dieser fünf Implementierungen ist die Entscheidung auf zwei Softwareprodukte gefallen. Für die Simulation wird JFern benutzt und für die Analysen werden Module aus Pipe2 verwendet. JFern wird als Framework zur Durchführung von Simulationen verwendet. Grund dafür ist, dass der Kern von JFern nur 41 Klassen umfasst (ohne graphische Oberfläche und externe Abhängigkeiten). Das Framework kann mit Objektorientierten-, High-Level-, Platz/Transitions- und zeitbehafteten Petri-Netzen umgehen. Die Petri-Netz Struktur ist in JFern von der Visualisierung getrennt. In JFern können des Weiteren verschiedene dynamische Skriptsprachen verwendet werden, wodurch sich JFern erweitern lässt. Der Einsatz der Skriptsprachen (BeanShell ¹, Kawa ² und Clojure ³) kann den Codeumfang reduzieren. Damit diese Skriptsprachen verwendet werden können, müssen die jeweiligen Pakete (*.jar) zu der jeweiligen Sprache vorhanden sein [32]. In JFern sind keine Analysemethoden implementiert. Aus diesem Grund werden für die Analyse von Petri-Netzen Module des Programms Pipe2 genutzt. Der große Vorteil von Pipe2 ist, dass es bereits die beiden gewünschten Analysemethoden, Erreichbarkeits- und Invariantenanalyse, enthält. Pipe2 ist ein sehr umfangreiches Programm und enthält verschiedene Analysemethoden, wie Strukturanalysen für Verklemmungen und Fallen und zusätzliche Zustandsraumanalysen, um zum Beispiel den kürzesten Weg zu einer Verklemmung zu berechnen [6]. Jedoch kann Pipe2 nicht mit kontinuierlichen Petri-Netzen umgehen. Damit das zu erstellende Add-on auch mit kontinuierlichen Petri-Netzen umgehen kann, wurde Pipe2 erweitert. Die Erreichbarkeitsanalyse (siehe Kapitel 2.5.3) von Pipe2 erstellt den Erreichbarkeitsgraphen mit der aktuell im Petri-Netz vorhanden Markierung. Der erstellte Erreichbarkeitsgraph kann unter anderem auf Eigenschaften wie Beschränktheit (boundedness), Verklemmung (deadlock) und Sicherheit (safeness) hin, untersucht werden. Des Weiteren kann Pipe2 einen Überdeckungsgraphen erstellen, sobald der Erreichbarkeitsgraph nicht berechenbar ist oder eine vom Nutzer angegebene Abbruchbedingung erreicht ist. Die Invariantenanalyse von Pipe2 ermöglicht Platz- und Transitionsinvarianten eines Petri-Netzes zu berechnen. Anhand dieser Invarianten ist es möglich, Rückschlüsse auf Eigenschaften des Petri-Netzes zu schließen. Mögliche Eigenschaften sind Beschränktheit (boundedness) und Lebendigkeit (liveness) des Petri-Netzes.

¹<http://www.beanshell.org/>

²<http://www.gnu.org/s/kawa/>

³<http://clojure.org/>

4 Implementierung

In diesem Kapitel wird beschrieben wie und mit welchen Hilfsmitteln die Anforderungen aus der Aufgabenstellung umgesetzt werden. Das Petri-Netz Add-on wird in drei Kernkomponenten, Rekonstruktion, Simulation und Analyse untergliedert. Die Komponente Rekonstruktion umfasst die Editier- und Transformationsfunktionalität für Netzwerke sowie Petri-Netze. Für den Austausch erstellter Petri-Netze mit anderen Programmen, enthält das Add-on zu dem ein Petri-Netz Austauschformat. Die Basis für die Simulations- und Analysekomponenten des Add-ons bilden Methoden von existierenden Implementierungen für Petri-Netze, die in einer Übersicht in Abschnitt 3.2 dieser Arbeit aufgelistet sind. Eine Kombination zweier dieser Implementierungen ergeben nun den Kern des Petri-Netz Add-ons für VANTED.

4.1 Petri-Netz Rekonstruktion

Das Petri-Netz Add-on bietet mit der Editorfunktionalität die Möglichkeit Petri-Netze neu zu erstellen oder bestehende Petri-Netze zu bearbeiten. Damit Netzwerke verschiedener Form in valide Petri-Netze transformiert werden können, bietet das Petri-Netz Add-on verschiedene Transformationsfunktionen an. Auf Grund der Besonderheit biologischer Netzwerke wird auf einige Übersetzungsregeln hingewiesen, die zu befolgen sind, um diese in valide Petri-Netze umzuwandeln. Die Funktionalität der Rekonstruktionskomponente wird in einem UML (Unified Modeling Language) Aktivitätsdiagramm [4] aus der Nutzersicht, veranschaulicht. Im UML Aktivitätsdiagramm werden die aus Nutzersicht möglichen Rekonstruktionsfunktionen des Petri-Netz Add-ons dargestellt. Der Nutzer hat drei Möglichkeiten um ein Petri-Netz zu generieren. Es kann erstellt und importiert werden oder ein Netzwerk kann importiert und anschließend in ein Petri-Netz transformiert werden. Sobald ein Petri-Netz auf einen dieser möglichen Wege generiert wurde, muss solange modifiziert werden, bis es valide ist. Die graphische Benutzeroberfläche der Rekonstruktion für Petri-Netze ist in Abbildung 4.2 dargestellt.

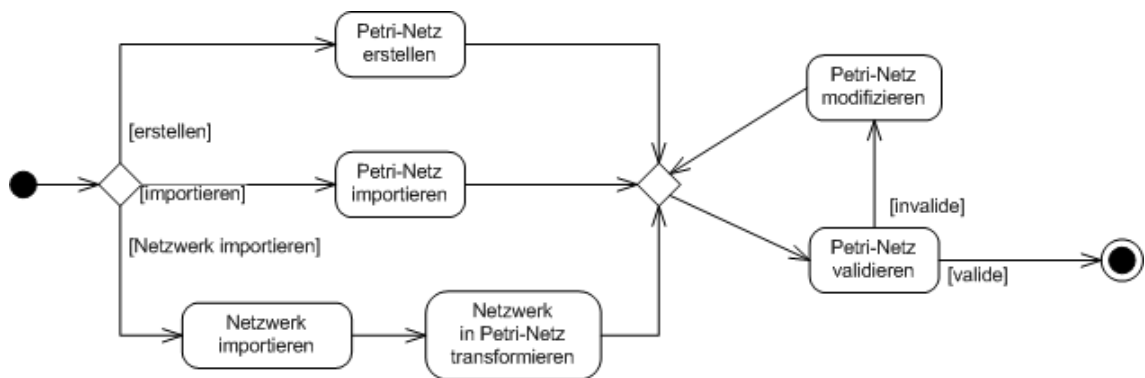


Abbildung 4.1: UML Aktivitätsdiagramm der Funktionalitäten der Rekonstruktionskomponente. Ein Petri-Netz kann erstellt und importiert werden oder ein Netzwerk kann importiert und anschließend in ein Petri-Netz transformiert werden. Ein invalides Petri-Netz muss solange modifiziert werden bis es valide ist.

4.1.1 Editorfunktionalität

Über die zur Verfügung gestellten Funktionalitäten können Petri-Netze neu erstellt und bestehende Petri-Netze modifiziert werden. Das Editieren von Petri-Netz Grundelementen, wie Transitionen und Plätzen, wird durch die Funktionalität in Abbildung 4.2 (1) bereitgestellt, wobei das Editieren von weiteren Petri-Netz Elementen, den Marken, Platz-Kapazitäten und Kantengewichten durch die Funktionalität in Abbildung 4.2 (2) unterstützt wird.

Damit Netzwerke simuliert und analysiert werden können, müssen diese in valide Petri-Netze transformiert werden. Zu beachten ist, wie im Kapitel 2.1 beschrieben, dass nur ein Platz mit einer Transition verbunden sein darf, aber nicht ein Platz mit einem anderen Platz oder eine Transition mit einer anderen Transition. Manchmal enthalten Netzwerke Knoten, die zur besseren Übersicht relevant sind, die aber gegen ein valides Petri-Netz verstoßen würden. Ein Beispiele für nicht valide Elemente ist in Abbildung 4.3 dargestellt. Diese Abbildung zeigt den Phenylalanine Abbau aus der Datenbank MetaCrop [39]. Die Reaktion ist in SBGN (Systems Biology Graphical Notation [28]) dargestellt. Ein ausgeschlossener Knoten wird durch ein rotes „X“ gekennzeichnet. Alle mit diesem Knoten verbundenen Kanten werden ebenfalls ausgeschlossen, obwohl Kanten nicht extra durch ein rotes „X“ gekennzeichnet werden. Somit kann die Funktionalität Netzwerkknoten auszuschließen, um ein valides Petri-Netz zu erhalten, realisiert werden. Ebenso ist es möglich, in einem validen Petri-Netz, Knoten für die Simulation und Analyse auszuschließen.

Entsprechend der Definition können Platz/Transitions-Netze mit positiven Integer-Werten (diskrete Werte) umgehen. Diese Definition wird für die Implementierung erweitert, sodass ein erweitertes Platz/Transitions-Netz in dieser Arbeit mit positiven

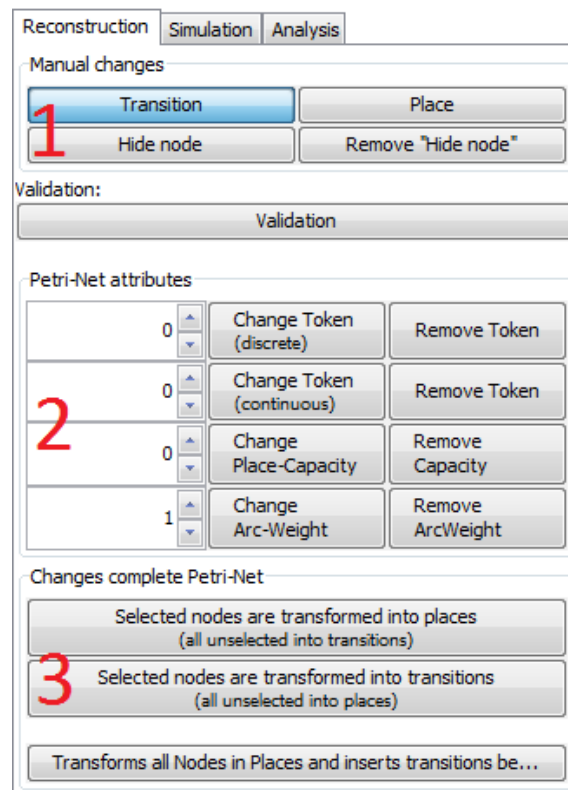


Abbildung 4.2: Die graphische Benutzeroberfläche der Rekonstruktion des Petri-Netz Add-ons. 1: Funktionalität, um Petri-Netze neu erstellen oder bestehende Petri-Netze bearbeiten zu können, 2: Funktionalität, um Marken, Platz-Kapazitäten und Kantengewichte dem Petri-Netz hinzuzufügen oder zu ändern, 3: Funktionen, um das gesamte Petri-Netz zu transformieren.

Double-Werten (kontinuierliche Werte), umgehen kann. Im weiteren Verlauf wird diese Erweiterung nicht mehr explizit erwähnt. Ein Petri-Netz kann in diesem Petri-Netz Add-on, entweder aus diskreten oder aus kontinuierlichen Elementen bestehen. Dementsprechend können Marken, Platz-Kapazitäten und Kantengewichte eines Petri-Netzes über die Funktionalität in Abbildung 4.2 (2) hinzugefügt, geändert und entfernt werden. Die folgende Auflistung zeigt in welchem Fall ein Petri-Netz als diskret oder kontinuierlich behandelt wird.

Das Petri-Netz wird zu einem diskreten Petri-Netz, sobald eine diskrete Marke hinzugefügt wird und das Petri-Netz noch keine kontinuierlichen Elemente enthält. Das Hinzufügen von kontinuierlichen Marken, kontinuierlichen Platz-Kapazitäten und/oder kontinuierlichen Kantengewichten generiert ein kontinuierliches Petri-Netz. Das Petri-Netz wird als ein leeres Netz behandelt, solange es nur diskrete Platz-Kapazitäten und/oder Kantengewichte enthält (keine Marken). Denn das Add-on kann an dieser Stelle nicht entscheiden, ob diese Elemente wirklich positive Integer-

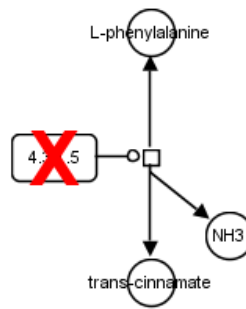


Abbildung 4.3: Phenylalanine Abbau aus der Datenbank MetaCrop [39]. Diese SBGN [28] Darstellung kann durch Ausschluss des Knotens mit der Beschriftung 4.3.1.5 in ein valides Petri-Netz überführt werden (rotes „X“).

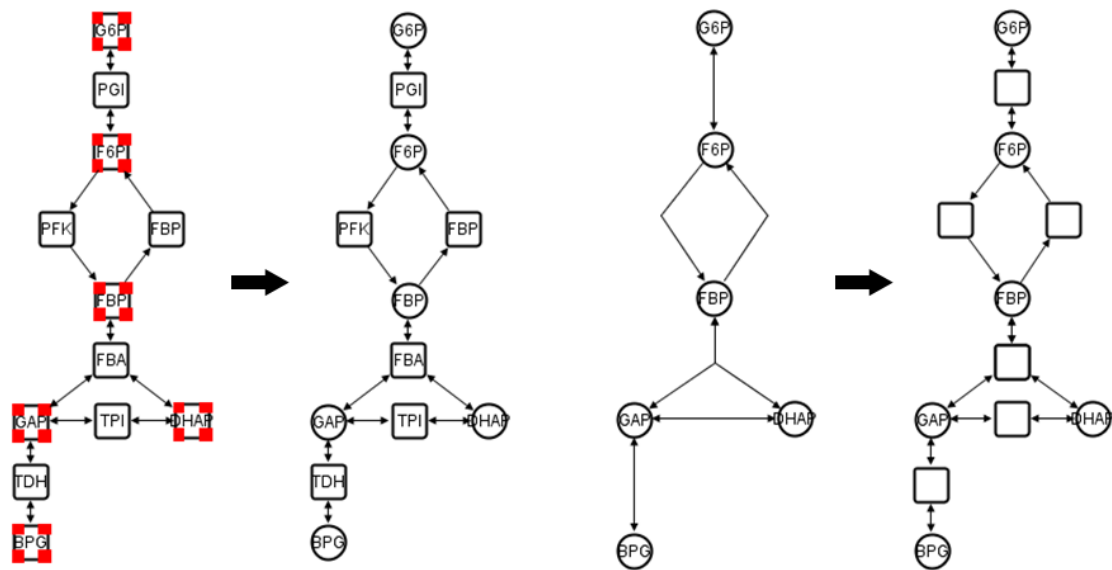
Werte oder positive Double-Werte sind. Erst mit dem Hinzufügen von, zum Beispiel diskreten Marken, wird das Petri-Netz zu einem diskreten Petri-Netz.

4.1.2 Transformationsfunktionalität

Mit dem Petri-Netz Add-on ist es möglich, Netzwerke oder Graphen zu importieren, die keine Petri-Netze sind. In öffentlichen Datenbanken werden biologische Netzwerke häufig vereinfacht dargestellt. In WikiPathways [22] zum Beispiel werden unter anderem, in manchen Netzwerken alle Knoten nur in Form eines Rechtecks dargestellt, unabhängig von deren Funktion (Metabolite, Enzyme). Laut Petri-Netz Theorie entsprechen alle rechteckigen Formen Transitionen. Damit der Nutzer solch ein Netzwerk einfach in ein Petri-Netz, bestehend aus Plätzen (Metabolite) und Transitionen (Enzyme), umwandeln kann, gibt es eine entsprechende Funktionalität, die in Abbildung 4.4 (a), am Beispiel der Glykolyse, dargestellt ist (alle Abkürzung der Metabolite und Reaktionen können der Datenbank WikiPathways entnommen werden).

Die Selektion aller Knoten, die zu Plätzen transformiert werden sollen, führt die gewünschte Transformation der Knoten durch. Auch wenn das umzuwandelnde Netzwerk aus verschiedenen Knotenformen besteht, kann diese Funktionalität verwendet werden. Bei der Selektion von Knoten, die in Plätze transformiert werden sollen, werden nicht nur diese in Plätze transformiert, sondern auch die nicht selektierten in Transitionen. Ebenso ist der umgekehrte Weg möglich, selektierte Knoten werden in Transitionen transformiert und alle nicht selektierten in Plätze (siehe Abbildung 4.2 (3)).

Eine andere Datenbank MetaCyc [9] hingegen enthält Netzwerke, die zur Vereinfachung aus nur einer Knotenform besteht. Die Metabolite werden als runde Knoten repräsentiert und die Enzyme nur als Knoten verbindende Kanten. Solche



(a) Transformation durch Selektion

(b) Automatische Transformation

Abbildung 4.4: Transformation der Glykolyse in ein Petri-Netz. (a) Selektierte Knoten werden in Plätze, alle nicht selektierten Knoten in Transitionen transformiert. (b) Alle Knoten werden in Plätze transformiert, zwischen den zwei transformierten Plätzen wird eine Transition erzeugt und alle drei Knoten werden mit zwei Kanten verbunden. Die zwei neuen gerichteten Kanten sind entsprechend der Ursprungskante gerichtet. Das Layout des transformierten Petri-Netzes wurde manuell angepasst.

Netzwerke werden als Hypergraphen bezeichnet [16], denn eine Kante kann mehr als zwei Knoten verbinden. Damit der Nutzer solch ein Netzwerk einfach in ein Petri-Netz, bestehend aus Plätzen (Metabolite) und Transitionen (Enzyme), umwandeln kann, gibt es eine weitere Funktionalität, die in Abbildung 4.4 (b), am Beispiel der Glykolyse, dargestellt ist (alle Abkürzung der Metabolite und Reaktionen können der Datenbank MetaCyc entnommen werden). Es werden alle im Netzwerk vorhandenen Knoten in Plätze umgewandelt und zwischen zwei transformierten Plätzen eine Transition erstellt. Der so entstehende Platz-Transition-Platz Verbund, ist durch zwei Kanten verbunden, die die selbe Richtung haben wie die Ursprungskante. Zu beachten ist, dass VANTED nicht mit Hypergraphen umgehen kann. Das Netzwerk in Abbildung 4.4 (c links) enthält eine Hyperkante, die die Knoten FBP mit DHAP und GAP verbindet. Diese Kante kann in VANTED nur durch zwei separate Kanten dargestellt werden, eine Kante verbindet FBP mit DHAP und eine andere FBP mit GAP (in Abbildung 4.4 (a rechts) zur Visualisierung übereinandergelegt). Bei der oben genannten Transformation dieses Netzwerkes in ein Petri-Netz, entstehen somit zwischen den separaten Kanten je eine Transition, die manuell in VANTED zu einem

Knoten zusammengeführt werden muss (sodass die Visualisierung in Abbildung 4.4 (b rechts) entsteht).

4.1.3 Besonderheiten der Petri Netz Repräsentation biochemischer Netzwerke

Damit metabolische Stoffwechselwege als Petri-Netze dargestellt werden können, gibt es einige Übersetzungsregeln zu befolgen. Abhängig von der biologischen Fragestellung an solche Netzwerke, kann es zu den folgenden Besonderheiten kommen, die auch schon von Baldan et al. [3] beschrieben wurden.

4.1.3.1 Modellierung räumlicher Eigenschaften

In einer Zelle befinden sich verschiedene Stoffwechselwege oftmals räumlich abgegrenzt (siehe Abbildung 4.5 (a)). Diese Orte werden Kompartimente genannt. Kompartimente können in Petri-Netzen nicht dargestellt werden. Eine Möglichkeit zwischen Metaboliten, in verschiedenen Kompartimenten zu unterscheiden, ist diese unterschiedlich zu bezeichnen. Bei der Rekonstruktion von metabolischen Stoffwechselwege, wird hierzu in der Regel ein bestimmtes Suffix dem Metabolitnamen hinzugefügt. Zum Beispiel kann zwischen Asparagin (abgekürzt Asn) im Peroxisomen und dem im Cytosol einer Zelle unterschieden werden, indem das Asparagin im Peroxisomen Asn_p und das im Cytosol Asn_c genannt wird (siehe Abbildung 4.5 (b)).

4.1.3.2 Modellierung mehrfach dargestellter Metabolite

Stoffwechselwege enthalten oftmals Metabolite, die in mehreren verschiedenen Reaktionen teilnehmen und zur Übersichtlichkeit mehrfach vorkommen. In der Petri-Netz Darstellung werden alle Metabolite mit gleichem Namen als ein Platz dargestellt und führen somit häufig zu sich überkreuzenden Kanten oder zumindest zur Unübersichtlichkeit. Zur besseren Visualisierung werden solche Metabolite als logische Plätze modelliert. Diese logischen Plätze werden mehrmals gezeichnet aber weisen im Petri-Netz Add-on auf genau einen Platz, um Analysen und Simulationen durchführen zu können. Zum Beispiel werden die Metabolite NADH und NAD⁺ auf Grund des häufigen Auftretens in biologischen Netzwerken mehrfach dargestellt aber im Petri-Netz existiert je nur ein Platz für die Metabolite (siehe Abbildung 4.6).

4.1.3.3 Modellierung externer Metabolite

Da eine Zelle ständig mit der Umgebung Metabolite austauscht, werden Transportprozesse benötigt, die Metabolite aus der Zelle oder in die Zelle transportieren. In

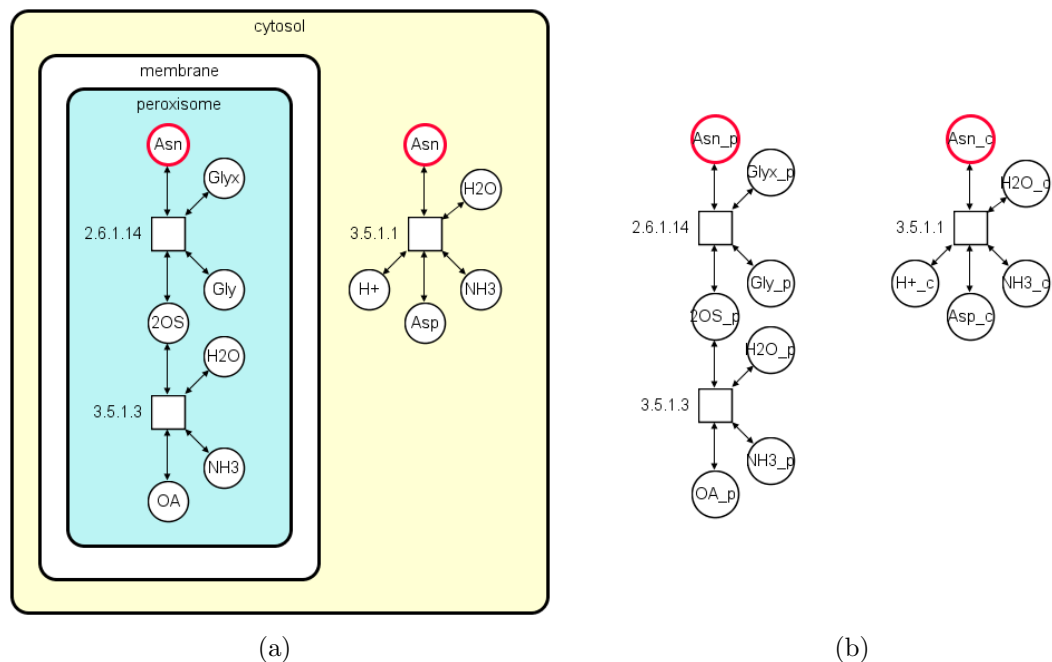


Abbildung 4.5: Modellierung räumlicher Eigenschaften am Beispiel des Stoffwechselweges Asparagin Abbau (in Anlehnung an [39]). (a) Räumliche Aufteilung des Abbauweges durch dargestellte Kompartimente erzielt. Die Kompartimente (Peroxisomen und Cytosol) enthalten jeweils den Metaboliten Asparagin (Abgekürzt Asn, rot umrahmt). (b) Räumliche Aufteilung des Abbauweges durch Suffixe an Metabolitnamen erreicht. Um Asparagin räumlich unterscheiden zu können, wird das im Peroxisomen Asn_p und das im Cytosol Asp_c genannt (rot umrahmt).

biologischen Netzwerken werden dafür gewöhnlich Reaktionen genutzt, die externe Metabolite importieren oder exportieren. In der Petri-Netz Darstellung wird die Aufnahme von Metaboliten mit Quelltransitionen/Quellplätzen und die Abgabe von Metaboliten mit Senkentransitionen/Senkenplätzen ermöglicht (siehe Abschnitt 2.1). Abhängig von der biologischen Fragestellung, gibt es verschiedene Möglichkeiten, diesen Austausch im Petri-Netz zu erzeugen.

Eine Möglichkeit ist künstliche Reaktionen zu modellieren, die Metabolite ständig aufnehmen können. Die Aufnahme von Metaboliten wird durch eine Quelltransition dargestellt, die keine Plätze als Vorbedingung besitzt (siehe Abbildung 4.7 (a)). Entsprechend der Transitionsschaltregeln besitzen eine Quelltransition ständig Konzeption und transportieren unbegrenzt die jeweiligen Metabolite in die Zelle (falls eine Platz-Kapazität im Nachbereich vorhanden ist, darf diese nicht überschritten werden). Die Abgabe von Metaboliten kann bei einer Senkentransition nur erfolgen, wenn dessen Vorbedingungen erfüllt sind, also eine entsprechende Anzahl Marken vorhanden ist und diese schaltet.

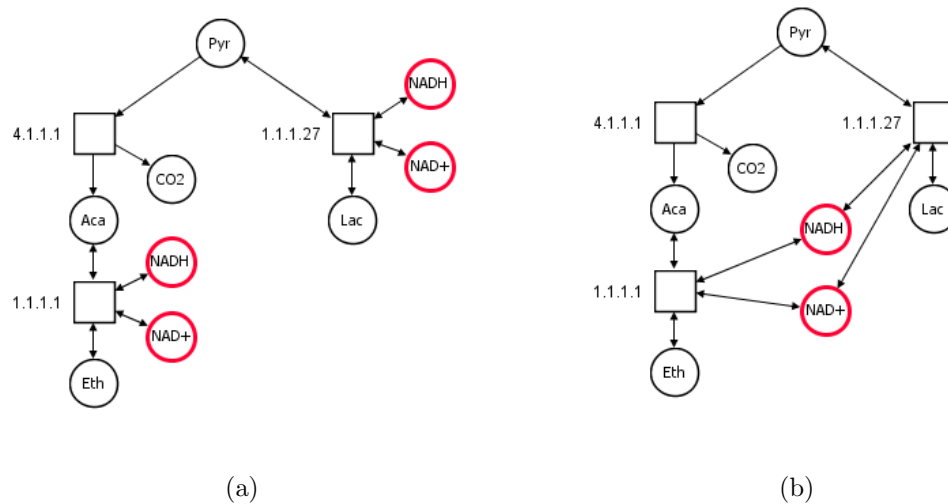


Abbildung 4.6: Modellierung mehrfach dargestellter Metabolite. Am Beispiel des Stoffwechselweges Fermentation. (a) Petri-Netz übersichtlich dargestellt mit logischen Plätzen für NADH und NAD+ (rot umrahmt). (b) Petri-Netz unübersichtlich dargestellt, da die Metabolite NADH und NAD+ mehrfach vorkommen aber nur je ein Platz existiert (rot umrahmt).

Eine andere Möglichkeit ist, die Aufnahme von Metaboliten in einer Zelle, zu steuern. Transitionen, die für die gesteuerte Aufnahme von Metaboliten genutzt werden, besitzen einen oder mehrere Quellplätze mit einer bestimmten Anzahl von Marken. Somit wird mit Hilfe der Markenanzahl eine gewisse Menge der Metabolite in die Zelle transportiert (siehe Abbildung 4.7 (b)). Werden für die Abgabe von Metaboliten Senkenplätze verwendet, kann die Anzahl externer Metabolite, im Gegensatz zu Senkentransitionen, nachvollzogen werden.

4.1.3.4 Modellierung reversibler Reaktionen

Die meisten Reaktionen einer Zelle sind reversibel. Reversibel bedeutet, es gibt eine Reaktion, die einerseits aus den Edukten Produkte bildet, aber andererseits auch in die entgegengesetzte Richtung verlaufen kann. Wird eine reversible Reaktion in einem Petri-Netz mit nur einer Transition erstellt, kann diese nur schalten, wenn alle Vorbedingungen erfüllt sind. Übertragen auf biologische Netzwerke, würde das bedeuten, dass die Edukte und Produkte gleichzeitig vorhanden sein müssen, damit die Reaktion abläuft. Damit eine reversible Reaktion entweder die Edukte in Produkte oder die Produkte in Edukte umwandelt, wird eine reversible Reaktionen durch zwei Transitionen dargestellt. Eine Transition ist die Vorwärtsrichtung und die andere ist die Rückwärtsrichtung (siehe Abbildung 4.8 (a)). Zur vereinfachten Visualisierung

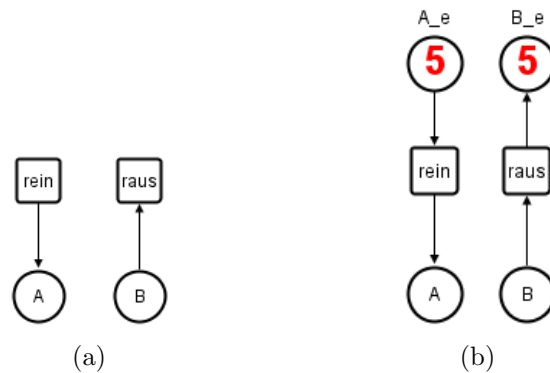


Abbildung 4.7: Modellierung externer Metabolite. (a) Unbegrenzte Aufnahme eines externen Metaboliten in das Netzwerk durch die Quelltransition. Metabolite werden aus dem Netzwerk abgegeben, sobald die Vorbedingung erfüllt ist und die Senkentransition schaltet. (b) Begrenzte Aufnahme eines externen Metaboliten durch die fünf Marken auf dem Quellplatz (A_e). Metabolite werden auf dem Senkenplatz (B_e) gesammelt und die Anzahl der externen Metabolite (fünf) kann nachvollzogen werden.

und Übersichtlichkeit werden die zwei Vorwärts- und Rückwärts-Transitionen als eine hierarchische Transition (reversible Reaktion) dargestellt (siehe [Abbildung 4.8 \(b\)](#)). Im Petri-Netz Add-on weist diese hierarchische Transition auf die Vorwärts- und Rückwärts-Transitionen.

4.2 Petri-Netz Simulation

Die graphische Benutzeroberfläche der Simulation von Petri-Netzen ist in [Abbildung 4.9](#) dargestellt. Die Simulation umfasst die Dynamik der Marken. Der in der



Abbildung 4.8: Modellierung reversibler Reaktionen. (a) Eine reversible Reaktion aufgespalten in eine Vorwärts- und Rückwärts-Transition in der Petri-Netz Darstellung. (b) Die gleiche reversible Reaktion als hierarchische Transition visualisiert.

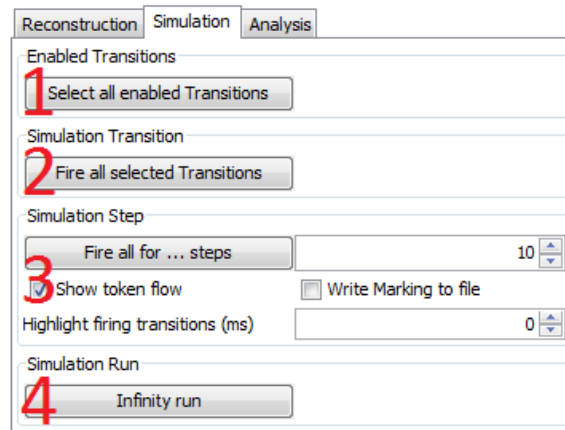


Abbildung 4.9: Die graphische Benutzeroberfläche der Simulation des Petri-Netz Add-ons. 1: alle aktiven Transitionen werden automatisch selektiert, 2: alle selektierten Transitionen werden geschaltet, 3: ermöglicht eine Schrittweise Simulation mit verschiedenen Parametern, 4: ermöglicht eine unendliche Simulation.

Simulation entstehende Markenfluss kann beobachtet werden und Aufschluss über das simulierte System geben. Bevor ein Petri-Netz simuliert wird, wird es implizit einer Validierung unterzogen. Eine Simulation kann nur mit einem validen Petri-Netz vollzogen werden. Falls ein Fehler vorliegt, werden alle betroffenen Elemente selektiert, sodass der Nutzer den fehlerhaften Bereich schnell finden kann. Diese Hervorhebung der invaliden Elemente ist in großen Systemen hilfreich und kann wertvolle Zeit sparen.

In Abbildung 4.10 ist der Simulationsvorgang an einem kleinen Petri-Netz dargestellt. Das Petri-Netz besteht aus drei Plätzen ($p_1 - p_3$) und drei Transitionen ($t_1 - t_3$). Die Simulationsschritte werden in der Reihenfolge (a - e) repräsentiert. In (a) ist die Anfangsmarkierung des Petri-Netzes dargestellt, wobei p_1 zwei Marken enthält. In diesem Petri-Netz Add-on wird eine Marke, auf einem Platz, als roter Punkt angezeigt. Ist die Anzahl der Marken ungleich eins wird der Wert der Marke(n) als Zahl angezeigt. (b) zeigt, dass t_1 schaltet, wobei eine Marke von p_1 verbraucht und auf p_2 erzeugt wird (c). Im nächsten Schritt schalten t_1 und t_2 (d). Dadurch entsteht eine neue Markierung (e), p_2 und p_3 besitzen eine Marke.

Die graphische Benutzeroberfläche ist in mehrere Simulationsmodi aufgeteilt, um den Markenfluss in einem Petri-Netz zu simulieren, die im Folgenden erläutert werden.

4.2.1 Selektion konzessionierter Transitionen

Es gibt die Möglichkeit alle Transitionen die Konzession haben, automatisch zu selektieren (siehe Abbildung 4.9 (1)). Diese Selektion gibt den Überblick, welche Transitionen im nächsten Simulationsschritt schalten könnten. In großen Netzwerken

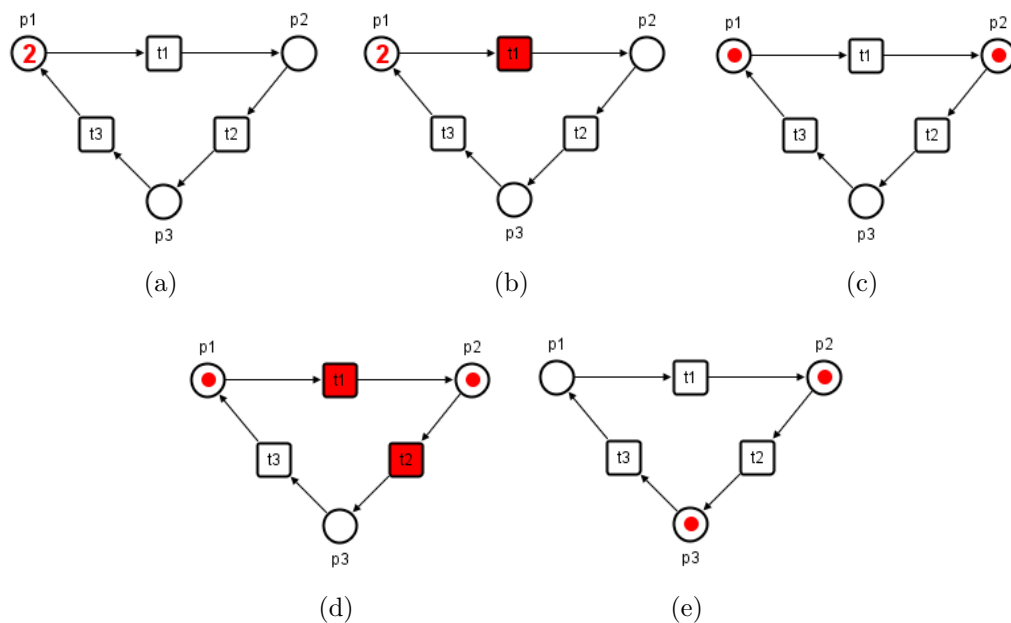


Abbildung 4.10: Die Simulation eines Beispiel Petri-Netzes mit der Anfangsmarkierung (a). Das Schalten der Transitionen (b, d) erzeugt die Folgemarkierungen (c, e).

müssen die aktiven Transitionen somit nicht selbständig aufgesucht werden, denn logische Plätze erschweren das Auffinden dieser Transitionen. Logische Plätze verbessern zwar das Layout von Petri-Netzen aber erschweren das Verfolgen von Reaktionszusammenhängen (siehe Abschnitt 4.1.3.2). Zu beachten ist, dass mehrere Transitionen in Konkurrenz zueinander stehen können und das Schalten einer Transition einer weiteren (konkurrierenden) Transition, die Konzession entzieht (Konflikt). In diesem Fall kann die Transition, die die Konzession verloren hat, nicht mehr schalten. Dieser Konflikt kann bei allen folgenden Simulationsmodi auftreten.

4.2.2 Simulationsmodi

Im Petri-Netz Add-on gibt es drei verschiedene Möglichkeiten Simulationen durchzuführen. Ein Simulationsmodus führt einen Schaltvorgang für alle selektierten Transitionen aus die Konzession haben, und visualisiert die neu entstandene Markierung (siehe Abbildung 4.9 (2)).

Der Simulationsmodus „Schrittweise Simulation“, umfasst mehrere Parameter, die Einfluss auf die Simulation nehmen (siehe Abbildung 4.9 (3)). Die Schrittzahl für die Simulation kann angegeben werden, wobei ein Schritt das Schalten aller Transitionen umfasst, die zur Zeit Konzession haben. Des Weiteren ist es möglich den Markenfluss zu visualisieren. Nach jedem Schritt wird die aktuelle Markierung angezeigt und wirkt

wie eine Animation des Markenflusses. Zusätzlich können die jeweils schaltenden Transitionen farblich hervorgehoben werden. Wenn eine Transition Konzession hat und schaltet, wird diese rot hinterlegt. Die Dauer der farblich Hervorhebung ist ebenfalls einstellbar, somit kann auch in großen Netzwerken beobachtet werden, welche Transitionen geschaltet haben. Die Visualisierung des Markenflusses und die farbliche Hervorhebung der schaltenden Transitionen kann bei großen Petri-Netzen die Simulation erheblich verlangsamen. Damit dennoch der Markenfluss nachvollziehbar bleibt und auch später ausgewertet werden kann, gibt es die Möglichkeit, die in jedem Schritt entstehende Markierung, in eine Textdatei auszugeben. Für jeden Platz wird die Anzahl der Marken pro Simulationsschritt aufgezeichnet und kann in, zum Beispiel Tabellenkalkulationsprogrammen, detailliert untersucht werden (siehe Abbildung 5).

Eine weitere Funktionalität ist die Simulation eines Petri-Netzes im „unendlichen“ Modus (siehe Abbildung 4.9 (4)). Dieser Modus ist der performanteste und kann gewählt werden, wenn nur die Markenverteilung nach einer großen Schrittzahl im Vordergrund steht. Sobald die Simulation im „unendlichen“ Modus gestartet ist, wird ein Schrittzähler mit dem aktuell vollzogenen Schritt angezeigt. Im Petri-Netz wird bis zur Visualisierung des Ergebnisses kein Markenfluss oder schaltende Transitionen angezeigt. Falls im Verlauf der Simulation keine Transition mehr aktiv ist, stoppt der Simulator die Simulation automatisch und visualisiert die entstandene Markenverteilung. Dieses automatische Beenden der Simulation und auch die Möglichkeit die Simulation nach einer bestimmten Schrittzahl selbst zu stoppen, wird ebenfalls im Simulationsmodus „Schrittweise Simulation“ unterstützt.

4.3 Petri-Netz Analyse

Die graphische Benutzeroberfläche der Analyse von Petri-Netzen ist in Abbildung 4.11 dargestellt. Im Petri-Netz Add-on kann ein Petri-Netz auf Invarianten und Erreichbarkeit analysiert werden. Die Invariantenanalyse umfasst die Platz- und Transitionsinvarianten, welche Informationen über die Beschränktheit und Lebendigkeit des Petri-Netzes zulassen. Die Erreichbarkeitsanalyse errechnet alle möglichen Zustände des Petri-Netzes abhängig von der Anfangsmarkierung und erstellt daraus einen Erreichbarkeitsgraphen oder einen Überdeckungsgraphen. Der Überdeckungsgraph wird berechnet wenn das Petri-Netz nicht beschränkt ist oder das Abbruchkriterium erreicht ist. Bevor ein Petri-Netz analysiert werden kann, wird es zuvor implizit validiert. Es können nur valide Petri-Netze analysiert werden.

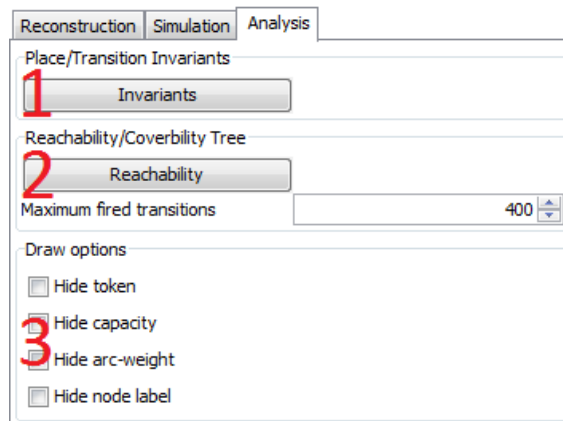


Abbildung 4.11: Die graphische Benutzeroberfläche der Analyse des Petri-Netz Add-ons. 1: Berechnung der Platz- und Transitions-Invarianten, 2: Berechnung des Erreichbarkeitsgraphen, nach Erreichen des Abbruchkriteriums (Angabe der Maximalen zu schaltenden Transitionen) wird der Überdeckungsgraph berechnet, 3: Darstellungsparameter ermöglichen Attribute des Petri-Netzes ein- und auszublenden.

4.3.1 Invariantenanalyse

Die Invariantenanalyse ist nur von der Struktur eines Petri-Netzes abhängig. Dementsprechend hat die Anfangsmarkierung und die Dynamik keinen Einfluss auf die Invarianten eines Petri-Netzes. Weitere Informationen zu Invarianten sind im Kapitel 2.5.2 erklärt.

Die Invariantenanalyse untersucht zunächst das Petri-Netz auf Platzinvarianten (P-Invarianten). Wenn das Petri-Netz P-Invarianten besitzt, werden diese in einer Tabelle aufgelistet und zusätzlich als P-Invarianten Gleichungen dargestellt. Die Gleichungen der P-Invarianten geben Konstanten der P-Invarianten an, die abhängig von der Anfangsmarkierung des Petri-Netzes sind. Befindet sich eine Marke in einer der berechneten P-Invariante, kann eine konkrete Konstante für die P-Invariante bestimmt werden. Diese gibt ein festes Markenverhältnis innerhalb dieser P-Invariante an. Anschließend wird überprüft ob das Petri-Netz von P-Invarianten überdeckt ist. Ist das Petri-Netz von positiven P-Invarianten überdeckt (CPI), kann die Eigenschaft Beschränktheit, abgeleitet werden. Die Aussage: „Das Petri-Netz ist von positiven P-Invarianten überdeckt (CPI) und somit beschränkt“, wird bestimmt. Ist das Petri-Netz nicht CPI, wird die folgende Aussage bestimmt: „Das Petri-Netz ist nicht von positiven P-Invarianten überdeckt (CPI) und somit kann die Beschränktheit nicht nachgewiesen werden“. Des Weiteren untersucht die Invariantenanalyse das Petri-Netz auf Transitionsinvarianten (T-Invarianten). Wenn das Petri-Netz T-Invarianten besitzt, werden diese in einer Tabelle dargestellt und es wird geprüft, ob das Petri-Netz von T-Invarianten überdeckt ist. Ist das Petri-Netz von positiven T-Invarianten

überdeckt (CTI), können die Eigenschaften Beschränktheit und Lebendigkeit abgeleitet werden. Die Aussage: „Das Petri-Netz ist von positiven T-Invarianten überdeckt (CTI) und könnte beschränkt und lebendig sein“, wird bestimmt. Ist das Petri-Netz nicht CTI wird die folgende Aussage bestimmt: „Das Petri-Netz ist nicht von positiven T-Invarianten überdeckt (CTI) und somit kann die Beschränktheit und Lebendigkeit nicht nachgewiesen werden“. Ist die Invariantenanalyse (siehe Abbildung 4.11 (1)) beendet, werden die Ergebnisse in einem Dialog repräsentiert. Der Dialog enthält einen Reiter für die P-Invarianten und einen Reiter für die T-Invarianten. Der Dialog für die P-Invarianten ist in Kapitel 5 Abbildung 5.1 am Beispiel des Glykolyse Petri-Netzes dargestellt. Die Ergebnisse der Invariantenanalyse werden in einer Tabelle zusammengefasst. Im Kontext des Petri-Netzes können die Invarianten intuitiver ausgewertet werden. Eine Interaktionstechnik, die im Folgenden beschrieben wird, soll die Verbindung zwischen Petri-Netz und Invariante realisieren.

Interaktionstechnik Aus der Menge der berechneten Invarianten soll genau eine Invariante herausgefiltert und im Petri-Netz detailliert untersucht werden können. Eine interaktive Visualisierung der Invarianten auf dem zugehörigen Petri-Netz, soll den Nutzer bei der Auswertung der Ergebnisse der Invariantenanalyse unterstützen. Die Interaktion zwischen dem Petri-Netz und den Invarianten wird durch die Interaktionstechnik Linking [11] und Brushing [29] erreicht. In Abbildung 4.12 wird diese Interaktionstechnik schematisch repräsentiert.

Das Linking verbindet die Daten aus der Invariantentabelle mit der graphischen Darstellung des Petri-Netzes. Wie in Abbildung 4.12 dargestellt, werden die Einträge in den Spalten der Tabelle, mit den Petri-Netz Elementen verbunden. Zum Beispiel, wird die eins in der Spalte mit dem Bezeichner c der Tabelle, mit der Transition mit dem Bezeichner c im Petri-Netz verbunden. Das Brushing wird zur Hervorhebung ausgewählter Daten verwendet. Es gibt verschiedene Möglichkeiten das Hervorheben zu realisieren. Im Petri-Netz Add-on wird je eine Invariante aus der Tabelle rot hervorgehoben, wenn sich der Mauszeiger über dieser befindet. Die Daten aus der Zeile dieser Tabelle werden dann auf die jeweiligen Knoten des Petri-Netzes visualisiert. In Abbildung 4.12 (a) wird zum Beispiel mit dem Mauszeiger die erste Zeile der Tabelle selektiert und somit die erste T-Invariant im Petri Netz visualisiert. Die Transitionen a , c und e erhalten, wie bereits in der Tabelle ersichtlich, eine eins. Wird die zweite Zeile der Tabelle selektiert (Abbildung 4.12 (b)), erhalten die Transitionen a , b und d je eine eins und die zweite T-Invariante wird visualisiert.

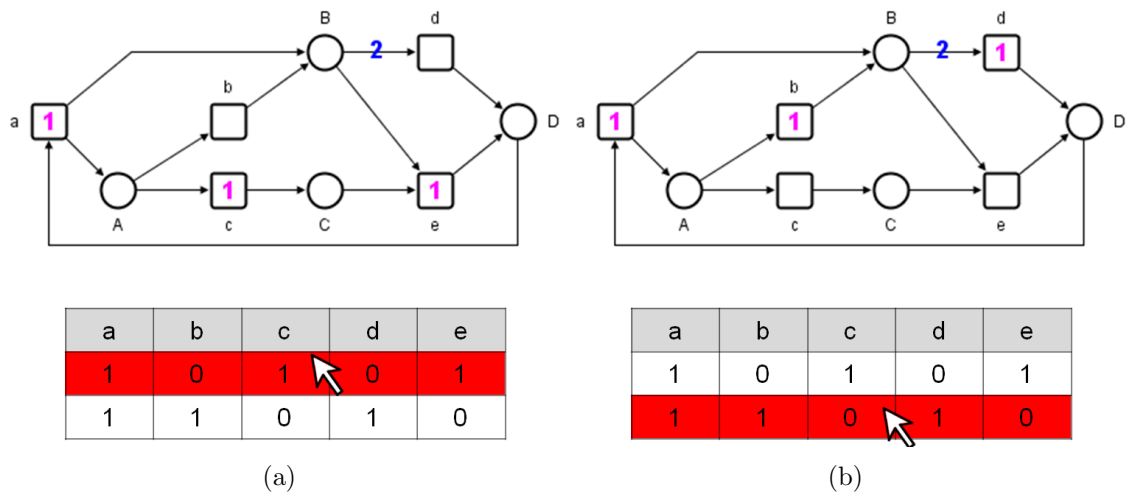


Abbildung 4.12: Schematische Darstellung der Interaktionstechnik. Die Interaktionstechnik (Brushing, Linking) wird an zwei T-Invarianten des Beispiel Petri-Netzes vorgestellt (in Anlehnung an [26]). (a) Die erste berechnete T-Invariante aus der Tabelle (unten) wird durch das bewegen des Mauszeigers über die erste Zeile im Petri-Netz visualisiert (oben). (b) Die zweite berechnete T-Invariante aus der Tabelle (unten) wird durch das bewegen des Mauszeigers über die zweite Zeile im Petri-Netz visualisiert (oben).

4.3.2 Erreichbarkeitsanalyse

Die Erreichbarkeitsanalyse konstruiert den Erreichbarkeitsgraphen. Die Knoten im Erreichbarkeitsgraphen entsprechen Zuständen (mit S_0, S_1, \dots bezeichnet), also einer Markierung des Petri-Netzes, die nach einer Schaltabfolge von Transitionen erreicht wird. Die Kanten zwischen den Zuständen repräsentieren Transitionen des Petri-Netzes, die schalten, um von einem Zustand in den Folgenden zu gelangen. Der Erreichbarkeitsgraph kann unter Umständen nicht für jedes Petri-Netz berechnet werden. Es kann vorkommen, dass ein Petri-Netz unendlich viele Zustände besitzt (abhängig von Markierung und Struktur des Petri-Netzes). Bei solchen Petri-Netzen wird der Erreichbarkeitsgraph unendlich groß und kann somit nicht berechnet werden. Aus diesem Grund kann anstelle des Erreichbarkeitsgraphen der Überdeckungsgraph erstellt werden. Weitere Informationen zur Erreichbarkeitsanalyse sind in Kapitel 2.5.3 erläutert.

In der graphischen Benutzeroberfläche (siehe Abbildung 4.11 (2)) ist es möglich eine Obergrenze für Transitionen, die schalten dürfen, einzustellen. Nachdem diese Grenze erreicht ist, wird der Überdeckungsgraph erstellt. In großen Petri-Netzen kann auch die Berechnung des Überdeckungsgraphen mit erheblichem Zeitaufwand verbunden sein. Sobald die Berechnung des Erreichbarkeits- oder des Überdeckungsgraphen beendet ist, wird dieser in einem neuen Fenster dargestellt.

Der Erreichbarkeits- oder der Überdeckungsgraph wird im Folgenden nur noch mit Erreichbarkeitsgraph bezeichnet. Auch in der Erreichbarkeitsanalyse wird die Interaktionstechnik Bruching und Linking verwendet, um den erstellten Erreichbarkeitsgraphen mit der graphischen Repräsentation des Petri-Netzes zu verbinden und Daten hervorzuheben (siehe Abschnitt 4.3.1). Sobald sich der Mauszeiger über einem Zustand im Erreichbarkeitsgraphen befindet, wird die Markierung die den entsprechenden Zustand widerspiegelt, im Petri-Netz dargestellt. Somit erhält jeder Platz im Petri-Netz eine entsprechende Anzahl Marken. Befindet sich der Mauszeiger über einer Kante im Erreichbarkeitsgraphen, wird die dazugehörige Transition, die schaltet um in den nächsten Zustand zu gelangen, im Petri-Netz rot hervorgehoben. Mit Hilfe der Interaktionstechnik kann von einem Zustand, über eine Kante in den folgenden Zustand, die Simulation eines Petri-Netzes, Schritt für Schritt nachvollzogen werden. In Kapitel 5 Abbildung 5.3 ist dazu am Beispiel des Glykolyse Petri-Netzes der Erreichbarkeitsgraph dargestellt und erläutert. Ein schematisches Beispiel, anhand dessen die Interaktion zwischen Erreichbarkeitsgraph und Petri-Netz visualisiert wird, ist in Abbildung 4.13 abgebildet. Der Anfangszustand „S0“ des Erreichbarkeitsgraphen, welcher die Anfangsmarkierung des Petri-Netzes widerspiegelt, wird rot hervorgehoben (a). Navigiert der Nutzer mit dem Mauszeiger im Erreichbarkeitsgraphen über einen Knoten, wird die jeweilige Markierung dieses Zustandes im Petri-Netz angezeigt (a, c). Ebenso verhält es sich beim Navigieren über eine Kante im Erreichbarkeitsgraphen. Sobald sich der Mauszeiger über einer Kante befindet, wird die Transition, die schaltet (b), um in den nächsten Zustand S1 zu gelangen, im Petri-Netz rot markiert. Der durch das Schalten erreichte nächste Zustand S1 (c) kann wiederum durch das Navigieren des Mauszeigers über den entsprechenden Zustandsknoten S1 im Petri-Netz betrachtet werden. Auf diese Weise ist es möglich das Verhalten des Petri-Netzes, den Übergang von einem Zustand in einen anderen Zustand, visuell nachzuvollziehen.

Sobald der Erreichbarkeitsgraph berechnet ist und visualisiert wird, werden die im Petri-Netz vorhandenen Marken implizit ausgeblendet. Somit wird eine bessere Sicht auf die Knoten Beschriftung (falls vorhanden) erlaubt. Es gibt die Möglichkeit die ausgeblendete Markierung wieder einzublenden (siehe Abbildung 4.11 (3)). Des Weiteren können zur Übersichtlichkeit die Attribute Platz-Kapazitäten, Kantengewichte und die Knoten Beschriftungen ein- oder ausgeblendet werden.

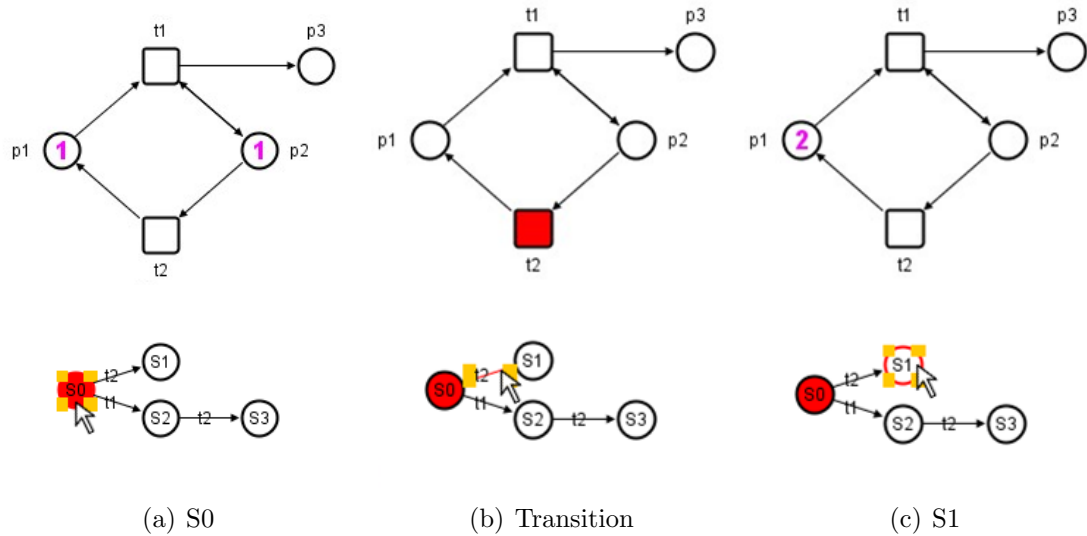


Abbildung 4.13: Interaktionstechnik in der Erreichbarkeitsanalyse. Die Interaktionstechnik (Brushing, Linking) wird an Zuständen S0 (a) und S1 (c), sowie einer Kante (b) des aus dem Beispiel Petri-Netz (in Anlehnung an [36]) errechneten Erreichbarkeitsgraphen vorgestellt. (a) Befindet sich der Mauszeiger über dem Anfangszustand S0 im Erreichbarkeitsgraphen (rot hervorgehoben) wird die Markierung für diesen Zustand im Petri-Netz dargestellt, zwei Plätze (p1, p2) erhalten je eine Marke. (b) Befindet sich der Mauszeiger über einer Kante im Erreichbarkeitsgraphen wird die Transition (t2) im Petri-Netz rot hervorgehoben, die schaltet um von Zustand S0 in den Zustand S1 zu gelangen. (c) Befindet sich der Mauszeiger über dem Zustand S1 im Erreichbarkeitsgraphen wird die Markierung für diesen Zustand im Petri-Netz dargestellt, ein Platz (p1) erhält zwei Marken.

4.4 Austauschformat für Petri-Netze

Das Format in dem die Petri-Netze gespeichert werden, ist ein essentieller Bestandteil des zu erstellenden Add-ons. Ein wichtiger Aspekt ist, dass die mit dem Add-on erstellten Petri-Netze zwischen verschiedenen Programmen ausgetauscht, sowie bereits existierende Petri-Netze importiert werden können. Es existiert eine Vielzahl von unterschiedlichen Petri-Netz Arten (siehe Abschnitt 2.2), die am Besten mit ein und demselben Format ausgetauscht werden sollen. Der Vorteil mit einem Austauschformat alle Petri-Netz Arten zu unterstützen, sichert die Erweiterbarkeit des Add-ons für andere Petri-Netz Arten. Des Weiteren soll das Format die erweiterten Platz/Transitions-Netze (P/T-Netze) importieren und exportieren können (siehe Abschnitt 4). Diese Anforderungen an das Petri-Netz Format sollen erfüllt werden um die Kompatibilität beim Austausch sicherzustellen.

Nach einer Internetrecherche ist die Wahl auf das PNML (Petri Net Markup Language) Format gefallen. PNML basiert auf XML, ist standardisiert und weit

verbreitet. Die Spezifikation von PNML umfasst drei Teile, von denen bisher erst zwei Teile standardisiert sind. Der erste Teil von PNML wurde im Jahr 2004 standardisiert und trägt die ISO (International Standard Organisation) Bezeichnung ISO/IEC 15909-1 ¹. Die Standardisierung des zweiten Teils (ISO/IEC 15909-2) wurde 2011 abgeschlossen ² [19]. Im ersten Teil des Standards wird das semantische Modell eines Petri-Netzes definiert. Es beschreibt drei Petri-Netz Arten: das High-Level Petri-Netz, das Symmetrische Petri-Netz und das Platz/Transitions Petri-Netz. Der Hauptaugenmerk von Teil zwei liegt auf dem Austausch der drei Petri-Netz Arten, die in Teil eins definiert wurden. In dem noch nicht standardisierten Teil drei, werden nach Aussage von PNML.org, zukünftig die Grundlagen für die Standardkompatibilität von nutzerdefinierte Erweiterungen festgelegt.

In PNML werden allgemeine und spezifische Petri-Netz Eigenschaften unterschieden. Die allgemeinen Petri-Netz Eigenschaften sind unabhängig von den verschiedenen Petri-Netz Arten. Allgemeine Eigenschaften sind zum Beispiel die Grundelemente (Plätze, Transitionen und Kanten), die eine bestimmte graphische Repräsentation besitzen. Die spezifischen Eigenschaften enthalten anwendungsspezifische Informationen. Programme, die diese PNML Dateien erstellen oder laden, interpretieren diese zusätzlichen Informationen. Diese Informationen können von anderen Programmen ignoriert werden [43].

Das entwickelte Petri-Netz Add-on benutzt PNML um alle Grundelemente (Plätze, Transitionen, Kanten und Marken) und P/T-Netz Elemente (Kapazitäten und Kantengewichte) zu speichern. Außerdem wird auch das Layout in PNML abgebildet, sodass beim erneuten Laden des Petri-Netzes sich alle Elemente an der ursprünglichen Position mit der ursprünglichen Größe befinden. Im Petri-Netz Add-on werden ebenfalls spezifische Eigenschaften genutzt, um die erweiterten P/T-Netze mit anderen Programmen austauschen zu können. Beim Erstellen von erweiterten P/T-Netzen werden die folgenden Eigenschaften beachtet. Die Double-Werte werden als spezifische Eigenschaft für das Petri-Netz Add-on gespeichert und zusätzlich wird ein gerundeter Double-Wert als allgemeine Eigenschaft in der PNML Datei gespeichert. Somit können auch andere Programme, die P/T-Netze verarbeiten können, diese erweiterten P/T-Netze laden (gerundete Double-Werte), aber auch das Petri-Netz Add-on, welches dann die Double-Werte aus den spezifischen Eigenschaften anzeigt.

¹http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=38225

²http://www.iso.org/iso/catalogue_detail.htm?csnumber=43538

5 Anwendung

In den letzten 15 Jahren sind vielfältige wissenschaftliche Bestrebungen zur Repräsentation und Analyse von biologischen Netzwerken, mit Hilfe von Petri-Netzen, unternommen worden [3]. Die ersten Publikationen zur Anwendung von Petri-Netz-Modellen metabolischer Netzwerke stammen von Reddy et al. [35] und Hofestädt [20]. Petri-Netze stellen einen adäquat definierten Formalismus dar, dem es gelingt, solche komplexen Netzwerke biochemischer Reaktionen zu modellieren, und dienen als Basis für Analysen und Simulationen. Auf Grund der intuitiven graphischen Repräsentation dieser Netzwerke, unterstützen Petri-Netze das bessere Verständnis dieser biochemischen Prozesse.

Im Folgenden werden die Analysen und Simulationen von Petri-Netzen, anhand drei sich in ihrer Komplexität unterscheidende metabolische Netzwerke, angewandt. Zunächst wird die Berechnung der P- und T-Invarianten, sowie des Erreichbarkeitsgraphen an einem einzelnen Stoffwechselweg, der Vorbereitungsphase der Glykolyse aus MetaCrop [39], durchgeführt. Dieser Stoffwechselweg eignet sich für diese Analysen, denn der Erreichbarkeitsgraph kann nur an kleinen Petri-Netzen berechnet werden, aber auch die Invarianten geben Aufschluss über einige Eigenschaften. Anhand eines anschließenden komplexeren Beispiels, werden die P- und T-Invarianten des Petri-Netzes analysiert und eine Simulation durchgeführt, die die Ergebnisse der Invariantenanalyse bestätigt. Der komplexere Anwendungsfall stellt den bisher detailliert untersuchten Sacharoseabbauweg in der Kartoffel dar [25]. Bei dieser Größe des Stoffwechselweges können die Invarianten noch berechnet werden und dienen anstelle des Erreichbarkeitsgraphen der Analyse der Petri-Netz Eigenschaften. Das Ziel ist es hier, die bisherigen Ergebnisse aus anderen Studien zu bestätigen. Abschließend wird mit Hilfe der Petri-Netz Simulation der Zentralstoffwechsel im Gerstesamen untersucht [15], wobei die Frage, ob die Petri-Netz Simulation die Bildung von Biomasse wieder geben kann, beantwortet wird. Auf Grund der räumlichen Abgrenzung der Stoffwechselwege im Gerstesamenmodell und die somit notwendigen Transporter, von denen die meisten reversibel sind und einige Stoffwechselwege redundant in zwei Kompartimenten vorkommen, wird für dieses Petri-Netz eine Simulation und keine vollständige Analyse durchgeführt. Die Erreichbarkeitsgraph- als auch die Invarian-

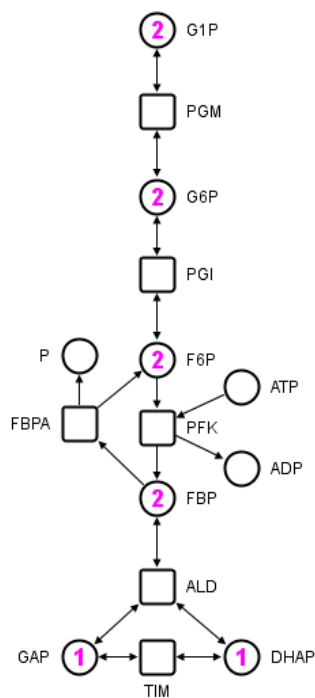
tenanalyse sind bei diesem Beispiel nicht mehr berechenbar, aber die Simulation kann hier zur Validierung des Modells herangezogen werden.

5.1 Analyse des Glykolyse Petri-Netzes

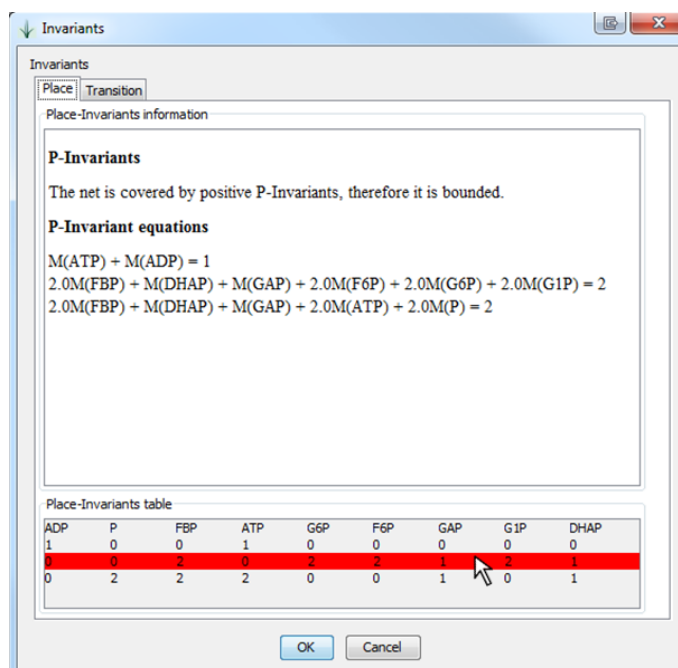
Die Glykolyse ist ein zentraler Vorgang im Energiestoffwechsel, der Energie in Form von ATP bildet. Die entstehende Energie wird für weitere Stoffwechselwege benötigt. Die Glykolyse lässt sich in zwei Phasen unterteilen. Die erste Phase wird als Vorbereitungsphase bezeichnet, bei der zunächst Energie in Form von ATP investiert wird. In der zweiten Phase wird dann erst ausreichend Energie gebildet. Für die Anwendung des Petri-Netz Add-ons wird auf Grund der Übersichtlichkeit nur die Vorbereitungsphase der pflanzlichen Glykolyse (basierend auf [39]) untersucht, siehe Abbildung 5.1 (a). Alle beteiligten Metabolite und Reaktionen können in der Datenbank MetaCrop unter dem Stoffwechselweg: Glycolysis, Gluconeogenesis nachgesehen werden.

Invariantenanalyse des Glykolyse Petri-Netzes Zunächst wird die Invariantenanalyse mit dem Petri-Netz Add-on durchgeführt, um Aussagen zu strukturellen Eigenschaften, wie P- und T-Invarianten treffen zu können. Die Eigenschaften Lebendigkeit und Beschränktheit des Glykolyse Petri-Netzes werden mit Hilfe der Invarianten veranschaulicht.

Zu Beginn sollen die P-Invarianten die Konservierung bestimmter Metabolite darstellen und es wird überprüft, ob das Petri-Netz von P-Invarianten überdeckt ist. Es sind drei P-Invarianten im Petri-Netz der Glykolyse berechnet wurden, siehe Abbildung 5.1 (b). Diese drei P-Invarianten überdecken das Petri-Netz und somit ist es beschränkt. In diesem Petri-Netz bleiben alle Marken erhalten, das heißt, die Reaktionen wandeln Metabolite ineinander um, solange genügend Marken auf den Plätzen enthalten sind. Es werden keine Marken hinzukommen oder abgegeben. Das Glykolyse Petri-Netz ist in der Lage alle Reaktionen mit einer Anfangsmarkierung zu schalten, wenn mindestens eine Marke auf je einem Platz der folgenden P-Invarianten enthalten ist. Die erste P-Invariante stellt die Konservierung der Summe von ATP und ADP fest. Dies ist ein Paar allgegenwärtiger Metabolite die ineinander umgewandelt werden, wenn diese durch eine Reaktion produziert oder verbraucht werden. Die zweite P-Invariante, die auch in der Abbildung 5.1 (a) visualisiert ist, ist die weitreichendste im Petri-Netz. Diese P-Invariante besteht aus allen primären Metaboliten der Glykolyse. Als primäre Metabolite werden in diesem Zusammenhang jene bezeichnet, die die Reaktionen der Glykolyse charakterisieren und somit von besonderer Bedeutung sind. Im stabilen Zustand (steady state)



(a)



(b)

Abbildung 5.1: P-Invarianten des Glykolyse Petri-Netzes. (a) Vorbereitungsphase der Glykolyse visualisiert eine der analysierten P-Invarianten (magenta Marken in einigen Plätzen). (b) Dialog zur Invariantenanalyse mit angezeigtem P-Invarianten-Reiter. Die Eigenschaft Beschränktheit und die Gleichungen der drei P-Invarianten werden dargestellt. Eine Tabelle fasst die drei P-Invarianten zusammen, wobei die zweite P-Invariante zur Visualisierung im Petri-Netz selektiert ist.

sollte eine P-Invariante existieren, die die gesamte Menge der primären Metabolite zusammenfasst [45]. Es ist möglich G1P über die Abfolge von Reaktionen in die zwei Metabolite GAP und DHAP umzuwandeln. Diese P-Invariante fasst somit den Hauptweg der Glykolyse zusammen. Die dritte P-Invariante stellt die Konservierung eines zusätzlichen Phosphats dar. ATP liefert ein Phosphat, das von einem Metaboliten zum anderen übertragen und schließlich durch die Reaktion FBPA erneut freigesetzt wird. Diese Übertragung des Phosphats ändert die Summe der phosphorylierten Metabolite und des freien Phosphats nicht und bleibt somit erhalten.

Die T-Invariantenanalyse des Glykolyse Petri-Netzes soll die Flussverteilungen im stabilen Zustand (steady state) widerspiegeln. Es wird überprüft, ob das geordnete Ausführen aller Reaktionen einer T-Invariante, die Anfangsmarkierung reproduziert, sodass in einem von T-Invarianten überdeckten Petri-Netz die Eigenschaften

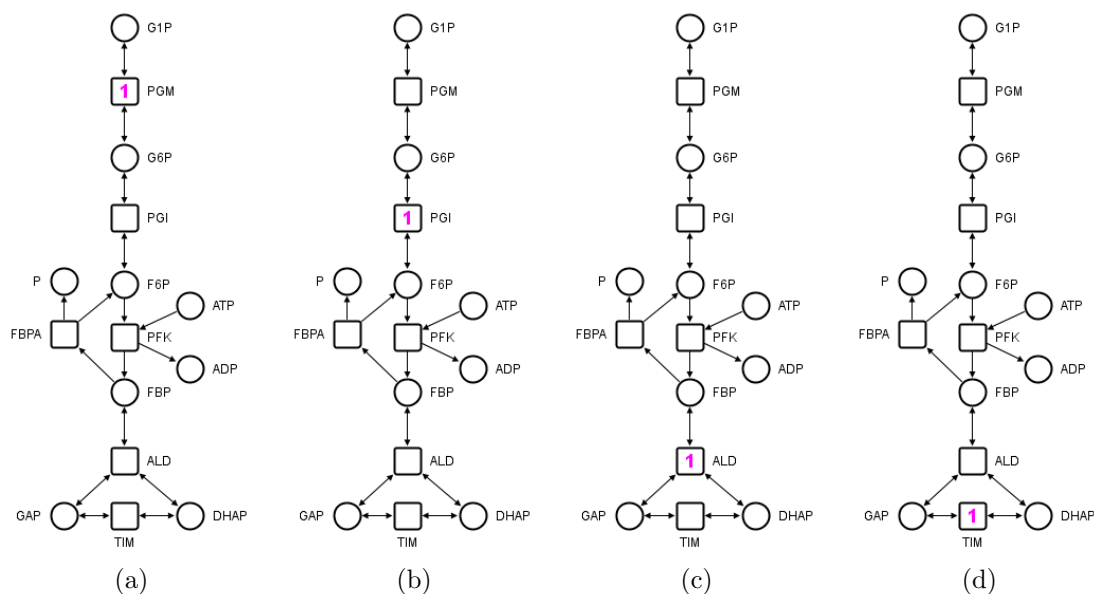


Abbildung 5.2: T-Invarianten des Glykolyse Petri-Netzes. Die Vorbereitungsphase der Glykolyse visualisiert alle vier analysierten trivialen T-Invarianten (magenta Marken in den Transitionen (a) PGM, (b) PGI, (c) ALD und (d) TIM).

Beschränktheit und Lebendigkeit abgeleitet werden können. Die Lebendigkeit veranschaulicht, dass jede Transition in einem erstellten Petri-Netz, nach einer Abfolge von Markierungen erneut schalten kann, also jede Reaktion in der Glykolyse immer wieder Metabolite umwandeln kann. Die Analyse der T-Invarianten im Petri-Netz der Glykolyse ergibt vier triviale T-Invarianten, siehe Abbildung 5.2 (a - d).

Diese vier T-Invarianten überdecken nicht das Petri-Netz und somit kann keine direkte Aussage über die Beschränktheit und Lebendigkeit gemacht werden. Alle berechneten T-Invarianten stellen triviale T-Invarianten dar und sind für die biologische Interpretation unbedeutend. Es handelt sich lediglich um die reversiblen Reaktionen PGM, PGI, ALD und TIM. Jede dieser Reaktionen wird zur trivialen T-Invariante, da die Vorwärtsrichtung jeder dieser Reaktionen eine Marke überträgt und die Rückwärtsrichtung dieser Reaktion die ursprüngliche Markierung wieder herstellt. Diese trivialen T-Invarianten veranschaulichen nur das je ein Metabolit durch eine Reaktion immer wieder in einen anderen umgewandelt werden kann, also nur diese vier Reaktionen immer wieder ablaufen können, aber der gesamte Stoffwechselweg nicht. Dementsprechend kann die Glykolyse unabhängig von der Anfangsmarkierung, nicht immer wieder aus G1P die Metabolite GAP und DHAP bilden. Der Stoffwechselweg verbraucht also nicht immer wieder ATP und stoppt wenn die Ziel Metabolite generiert wurden.

Erreichbarkeitsanalyse des Glykolyse Petri-Netzes Abschließend wird das Glykolyse Petri-Netz, hinsichtlich der dynamischen Eigenschaften analysiert. Die Erreichbarkeitsanalyse soll alle erreichbaren Zustände des Glykolyse Petri-Netzes mit der gegebenen Anfangsmarkierung berechnen. Somit sollen präzisere Aussagen als die der Invariantenanalyse getroffen werden. Dazu berechnet das Petri-Netz Add-on den Erreichbarkeitsgraphen, der auf Grund der Eigenschaft Beschränktheit des Petri-Netzes endlich ist und somit berechnet werden kann, siehe Abbildung 5.3.

Der Erreichbarkeitsgraph besteht aus 10 Zuständen (S0 - S9). Die zustandsverbindenden Kanten entsprechen den Transitionen im Petri-Netz, die schalten um von einem Zustand in den Folgenden zu gelangen. Jeder Zustand entspricht einer einzigartigen Markierung des Petri-Netzes, die nach einer Schaltabfolge von Transitionen, erreicht wird. Der Zustand S3 wird zum Beispiel erreicht, wenn von der Anfangsmarkierung aus (Zustand S0) zunächst die Transition PGM schaltet. Somit wird der Zustand S1 erreicht und von dort aus schaltet nun Transition PGI, um in Zustand S2 zu gelangen. Damit der Zustand S3 erreicht wird, schaltet anschließend Transition PFK. Die Marke von G1P ist durch die Schaltfolge der Transitionen auf den Platz FBP gelangt. Ebenfalls ist durch das Schalten von PFK die Marke von Platz ATP auf Platz ADP gelangt. Die biochemische Umwandlung von G1P zu FBP hat unter Verbrauch von ATP stattgefunden. Diese energieverbrauchende Reaktion kann mit der Anfangsmarkierung nur einmal schalten und somit nur einmal Energie verbrauchen. Von Zustand S3 aus teilt sich der Erreichbarkeitsgraph auf. Diese Verzweigung im Erreichbarkeitsgraphen veranschaulicht einen Konflikt im Petri-Netz. Enthält der Platz FBP eine Marke kann entweder die Transition ALD oder FBPA schalten und verschiedene Folgezustände erzeugen. Das Petri-Netz schaltet bei solchen Konflikten nicht-deterministisch entweder die eine oder die andere Transition. Dementsprechend können mit dieser Anfangsmarkierung entweder mit dem Ablauf einer Reaktion (ALD) aus FBP die Metabolite GAP und DHAP produziert werden oder gar nicht. Wenn die Reaktion FBPA aus FBP die Metabolite P und F6P produziert, kann die Reaktion PFK nicht mehr erfolgen. Es ist keine Marke mehr auf Platz ATP und somit kann die Transition nicht schalten, weil die Vorbedingung nicht erfüllt ist. Sobald die Reaktion FBPA die Umwandlung von FBP nach P und F6P vollzieht, sind nur noch die Rückwärtsreaktionen PGI und PGM möglich. Wenn ALD geschaltet hat, kann nun Transition TIM schalten und durch Vorwärts- bzw. Rückwärtsreaktion in die Zustände S7 und S8 gelangen. Weil die Reaktion ALD aus einer Marke auf Platz FBP zwei Marken, je eine auf Platz GAP und DHAP produziert, enthält nach dem anschließenden schalten von TIM entweder GAP oder DHAP zwei Marken.

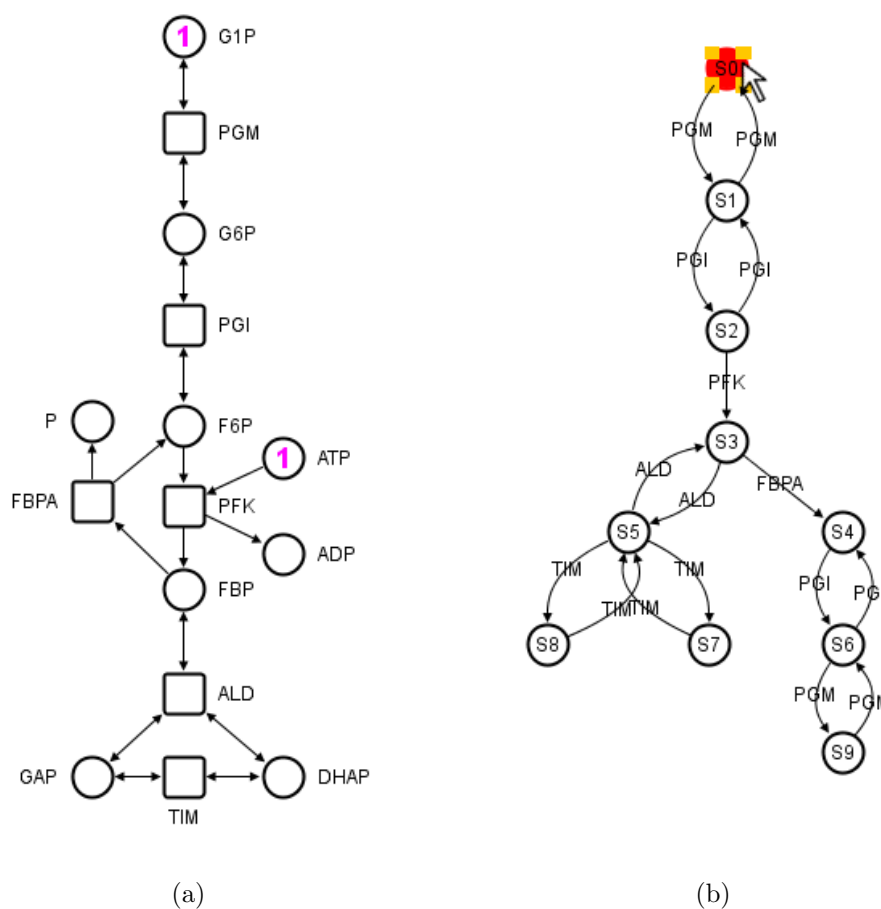


Abbildung 5.3: Erreichbarkeitsgraph des Glykolyse Petri-Netzes. (a) Vorbereitungsphase der Glykolyse visualisiert die Anfangsmarkierung (magenta Marken in den Plätzen G1P und ATP). (b) Erreichbarkeitsgraph, wobei der Anfangszustand S0 selektiert ist, der im Petri-Netz (a) die Anfangsmarkierung interaktiv visualisiert. Der Erreichbarkeitsgraph besteht aus 10 Zuständen (S0 - S9) und die zustandsverbindenden Kanten entsprechen den Transitionen im Petri-Netz, die schalten um von einem Zustand in den folgenden zu gelangen. Der Erreichbarkeitsgraph verzweigt sich in Zustand S3, dass heißt mit dieser Markierung im Petri-Netz ist es möglich entweder die Reaktion ALD oder die Reaktion FBPA auszuführen. In diesem Konflikt schaltet das Petri-Netz nicht-deterministisch entweder die eine oder die andere Transition und gelangt in verschiedene Folgezustände.

Ist jeweils eine Marke in GAP und DHAP, kann auch wieder die Rückwärtsreaktion der ALD erfolgen und aus zwei Marken wird wieder eine auf Platz FBP und Zustand S3 wäre erneut erreicht.

Die Reaktionen PGM, PGI, ALD und TIM sind reversibel und können somit im Petri-Netz vorwärts und rückwärts schalten. Im Erreichbarkeitsgraphen wird die Reversibilität durch Zyklen zwischen Zuständen sichtbar. Diese Zyklen entsprechen

auch den oben genannten T-Invarianten, die bei der aktuellen Markierung das Schalten in die Vorwärts- als auch in die Rückwärtsrichtung erlauben. Jede dieser Reaktionen veranschaulicht auch wieder einen Konflikt im Petri-Netz. Ist eine Marke auf einem Platz enthalten, der mit mindestens einer reversiblen Reaktion verbunden ist, schaltet das Petri-Netz nicht-deterministisch entweder die eine Reaktion oder die Andere. Enthält zum Beispiel Platz G6P eine Marke, kann entweder die Rückreaktion von Transition PGM ablaufen oder die Hinreaktion von PGI. Die Reaktionen PFK und FBPA sind irreversibel und werden im Erreichbarkeitsgraphen als eine Kante in eine Richtung dargestellt. Es ist in diesem Glykolyse Petri-Netz durch den Nicht-Determinismus also nicht sicher, ob die Metabolite GAP und DHAP produziert werden.

Der Erreichbarkeitsgraph erlaubt Aussagen über die folgenden dynamischen Petri-Netz Eigenschaften zu treffen. Die P-Invariantenanalyse hat bereits die Beschränktheit des Glykolyse Petri-Netzes ausgesagt. Jedoch konnte die T-Invarianten Analyse keine Aussagen zur Lebendigkeit und Beschränktheit des Netzes treffen. Mit Hilfe des Erreichbarkeitsgraphen wird die Beschränktheit des Glykolyse Petri-Netzes eindeutig nachgewiesen. Es ist sogar 2-beschränkt, denn bedingt durch die Anfangsmarkierung ist die maximal zu erreichende Markenanzahl auf den Plätzen zwei. Nur die Plätze GAP und DHAP können zwei Marken enthalten, alle weiteren Plätze nur eine Marke. Auf Grund dessen ist das Glykolyse Petri-Netz auch nicht sicher. Das Glykolyse Petri-Netz ist auch nicht lebendig, denn es gibt mindestens eine Transition, die bei einer bestimmten Markierung potentiell nicht schalten kann. Zum Beispiel kann die Transition PFK mit der Anfangsmarkierung und allen daraus erreichbaren Markierungen nur ein einziges Mal schalten. Der mit dieser Transition verbundene Platz ATP besitzt nur eine Marke, die beim Schalten von PFK verbraucht wird und nicht wieder produziert werden kann. Da der Erreichbarkeitsgraph nicht stark zusammenhängend ist, geht auch die fehlende Reversibilität des Petri-Netzes hervor, denn wenn PFK geschaltet hat, kann der Anfangszustand nicht wieder erreicht werden. Von der Anfangsmarkierung aus existiert keine tote Transition, die nie schaltet. Aber hat PFK geschaltet, kann diese nicht wieder schalten und ist somit tot. Dieses Petri-Netz ist nicht frei von Konflikten. Die Transitionen TIM und ALD teilen sich zwei Vorplätze und besitzen somit einen strukturellen Konflikt, der sich außerdem auch dynamisch auswirkt. Wenn zum Beispiel GAP und DHAP je eine Marke haben, könnte entweder TIM oder ALD schalten und wenn eine davon schaltet, wird der anderen Transition die Konzession entzogen. Zusammengefasst ist das Glykolyse Petri-Netz 2-beschränkt, nicht lebendig und nicht reversibel.

5.2 Analyse des Sacharoseabbauweges in der Kartoffel

Der Sacharoseabbauweg in der Kartoffel (beschrieben in [25]), wird in diesem Abschnitt als komplexeres Petri-Netz verwendet. Ziel ist es die Invarianten des Petri-Netzes zu analysieren und somit die bisherigen Ergebnisse aus anderen Studien zu bestätigen. Das metabolische Netzwerk repräsentiert als Petri-Netz, in Abbildung 5.4 dargestellt und zeigt den Hauptkohlenstofffluss.

Das Petri-Netz wird nun mit dem Petri-Netz Add-on analysiert, um die Ergebnisse von Koch et al. [25] zu bestätigen. Zunächst ist die Invariantenanalyse mit dem Petri-Netz Add-on durchgeführt wurden, um Aussagen zu strukturellen Eigenschaften wie P- und T-Invarianten treffen zu können. Zur Wahrung der Übersichtlichkeit wird jeweils nur eine der P- bzw. T-Invarianten abgebildet. Wie in Koch et al. [25] sind mit dem Petri-Netz Add-on ebenfalls die drei folgenden P-Invarianten berechnet wurden.

1. P-Invariante: UDPglc, UTP, UDP
2. P-Invariante: ATP, AMP, ADP
3. P-Invariante: G6P, F6P, G1P, UTP, 2 ATP, ADP, S6P, P_i, 2 PP

Diese drei P-Invarianten überdecken das Petri-Netz nicht und somit kann auf Grund dieser Analyse kein Rückschluss auf die Beschränktheit des Kartoffel Petri-Netzes gezogen werden. Die ersten beiden P-Invarianten geben an, dass zum einen die Menge von ATP, AMP und ADP und zum anderen auch die Menge von UDPglc, UTP und UDP konserviert ist. Das Kartoffel Petri-Netz enthält keine Quell- und Senkentransitionen, die ATP und UTP mit der Umgebung austauschen. Deshalb ist die Summe dieser Metabolite nicht veränderlich. Die beiden P-Invarianten validieren diese Aussage. Die dritte P-Invariante enthält die Menge aller Metabolite, die ein Phosphat bereitstellen und das Phosphat selbst. Das Phosphat wird von Metabolit zu Metabolit übertragen. Das Petri-Netz enthält keine Quell- und Senkentransitionen für Phosphat, sodass die Summe der phosphorylierten Metabolite und des Phosphates unverändert bleibt.

Die Analyse der T-Invarianten des Sacharoseabbauweges der Kartoffel von Koch et al. [25] ergibt 19 T-Invarianten, bestehend aus sieben trivialen, welche auf Grund der reversiblen Reaktionen (SPS, UGPase, SuSy, PGM, NDPkin, AdK und PGI) entstehen und 12 nicht-trivialen T-Invarianten. Diese 12 nicht-trivialen Invarianten repräsentieren eine Kombination von Teilnetzwerken.

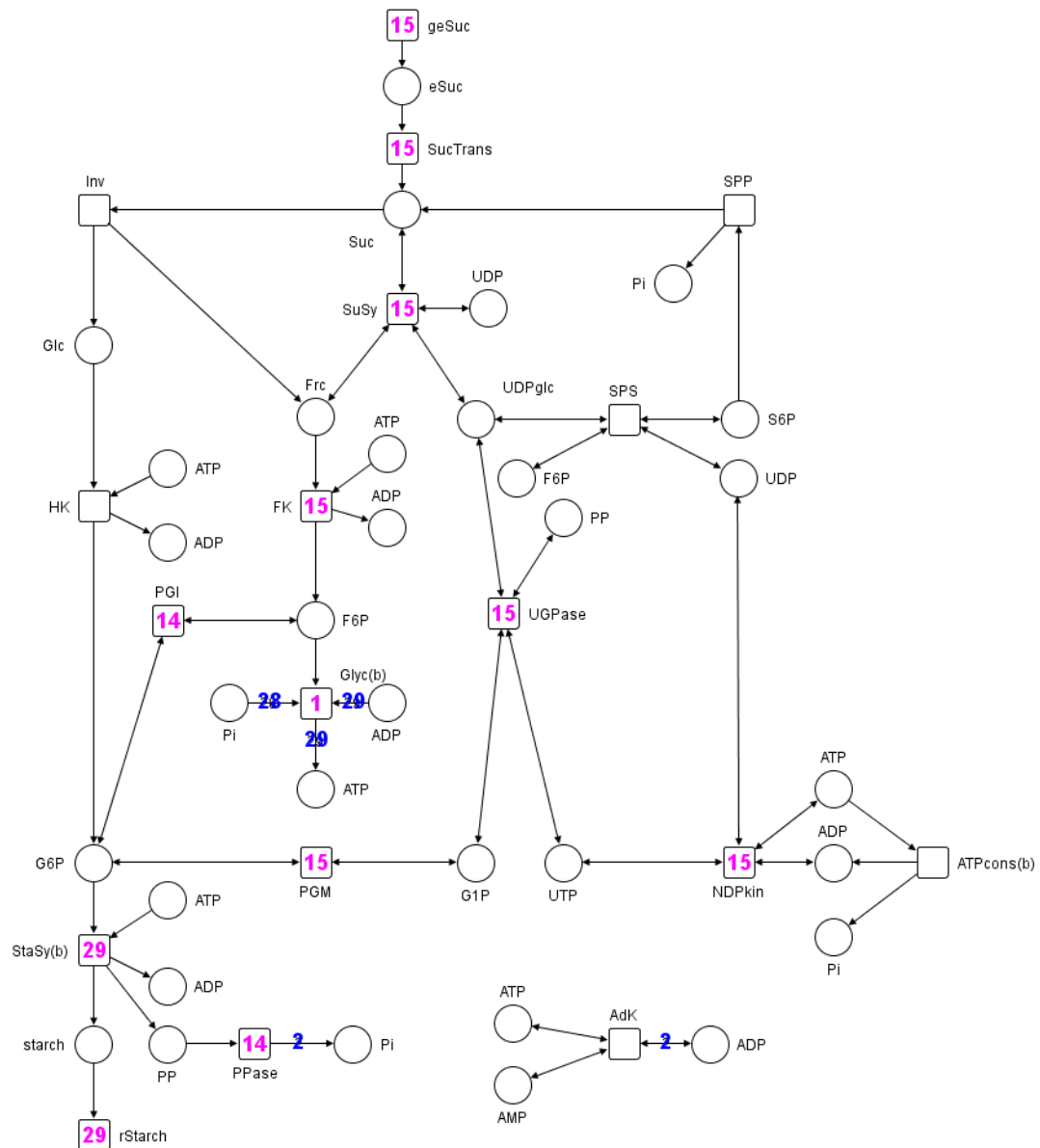


Abbildung 5.4: Das Petri-Netz besteht aus 17 Plätzen und 25 Transitionen, von denen sind sieben hierarchische Transitionen, drei zusammengefasste Prozesse und zwei künstliche Transitionen zur Aufnahme (Quelltransition) von Saccharose (geSuc) und Abgabe (Senkentransition) von Stärke (rStarch, weitere Informationen in Koch et al. [25]). Es wird eine der 19 T-Invarianten (magenta Marken in einigen Transitionen) dargestellt. Diese T-Invariante beschreibt den Hauptfluss des Saccharoseabbauweges in der Kartoffel.

Die Invarianten unterscheiden sich hauptsächlich in der ersten Reaktion, in der die aufgenommene Saccharose (Suc) entweder durch Inv oder SuSy gespalten wird und letztlich in Stärke (starch) eingebaut wird. In Abbildung 5.4 wird die T-Invariante 10 (siehe Koch et al. [25]), der Hauptfluss des Saccharoseabbauweges in der Kartoffel

repräsentiert. Die biologische Interpretation dieser und der restlichen T-Invarianten, würde in dieser Arbeit zu weit führen und kann in Koch et al. [25] nachgelesen werden.

Die T-Invariantenanalyse für dieses Petri-Netz konnte mit dem Petri-Netz Add-on nicht nachvollzogen werden. Auf Grund von Fehlfunktionen mit den verwendeten Methoden aus Pipe2, ist es nicht möglich, die T-Invarianten des Kartoffel Petri-Netzes zu berechnen. Auch die Kontrollanalyse des Kartoffel Petri-Netzes direkt in Pipe2, lieferte kein Ergebnis. Auffällig ist, dass Petri-Netze, die nur wenige reversible Reaktionen besitzen, keine Probleme mit der T-Invariantenanalyse auslösen. Wenn die reversiblen Reaktionen im Kartoffel Petri-Netz reduziert werden, können T-Invarianten berechnet werden. Meist hat es ausgereicht aus einer reversiblen eine irreversible Reaktion zu bilden und schon konnten Ergebnisse erzielt werden. Also je mehr reversible Reaktionen in einem Petri-Netz vorhanden sind, desto weniger wahrscheinlich ist es, dass Pipe2 eine Lösung berechnen kann. Beim Debuggen des Programms konnte festgestellt werden, dass Matrizen berechnet werden, die sich Zwischenergebnisse ohne Abbruchkriterium austauschen. Deshalb beendet Pipe2 die Analyse nicht und kommt schlussendlich zu keiner Lösung in Bezug auf die T-Invarianten. Zum Zweck der Visualisierung der T-Invarianten wurde das Programm INA [38] zur Berechnung verwendet. Die berechneten T-Invarianten wurden anschließend mit dem Petri-Netz Add-on visualisiert.

Die Simulation des Saccharoseabbauweges der Kartoffel wird nun durchgeführt, um die Ergebnisse der Invariantenanalyse zu bestätigen. Das Petri-Netz besitzt für die Aufnahme der Saccharose eine Quelltransition (rSuc), die es erlaubt unbegrenzt Saccharose (suc) aufzunehmen. Die Abgabe der Stärke (starch) wird durch eine Senkentransition (rStarch) modelliert. Auf diese Weise ist es bei der Aufzeichnung der Anzahl der Marken, die sich pro Simulationsschritt auf dem Platz Stärke befindet nicht möglich die Bildung von Stärke nachzuvollziehen. Immer wenn sich eine Marke auf diesem Platz befindet, wird die Marke auf Grund der Senkentransition aus dem Modell abgegeben. Damit die Bildung der Stärke beobachtet werden kann, wird die Senkentransition (rStarch) von der Simulation ausgeschlossen. Beginnend mit der vorgegebenen Anfangsmarkierung aus Koch et al. [25] wird die Simulation des Saccharoseabbauweges der Kartoffel durchgeführt und die Anzahl der Marken pro Platz für jeden Simulationsschritt als Diagramm dargestellt (siehe Abbildung 5.5).

Anhand des Simulationsergebnisses wird deutlich, dass sowohl die Reaktion Inv als auch die Reaktion SuSy an der Bildung von Stärke beteiligt sind. In den ersten Schritten der Simulation werden die beide Metabolite Glc und Frc gebildet und durch

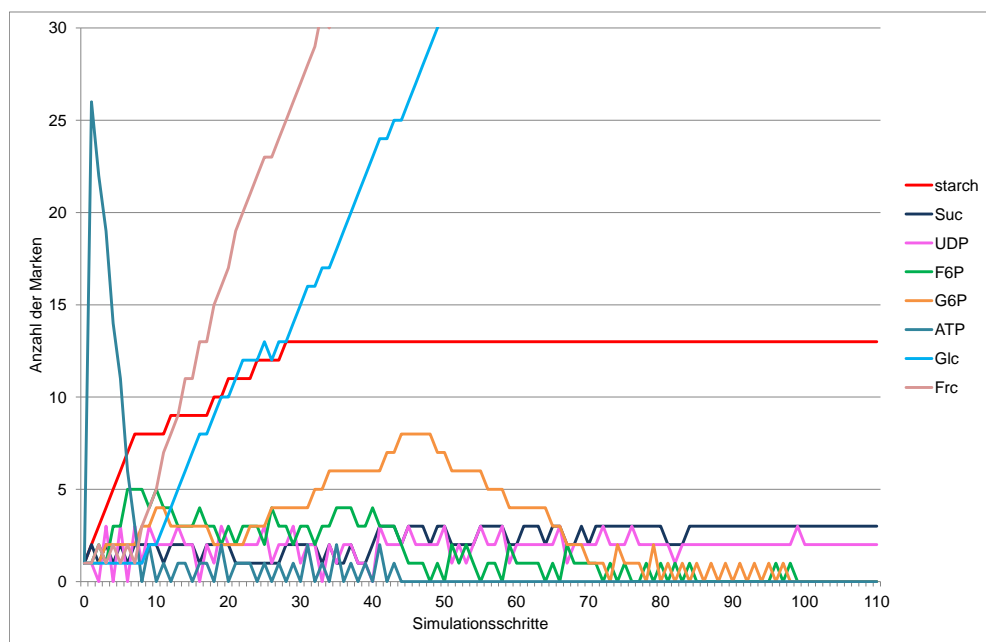


Abbildung 5.5: Markenverteilung im Saccharoseabbauweg der Kartoffel pro Simulationsschritt. Es sind acht von 17 Metaboliten als Kurven dargestellt. Die Stärkebildung wird nach 28 Schritten beendet und ab Simulationsschritt 100 findet keine weitere Veränderung mehr statt.

die Folgereaktionen (HK und FK) wieder verbraucht, die dann die Metabolite F6P und G6P produzieren. Der Metabolit F6P wird dann ebenfalls in G6P umgewandelt. Edukt für die Reaktion $StSys(b)$, die Stärke bildet, ist unter anderen G6P. Das nicht-deterministische Schalten der Transitionen an Konfliktpositionen, wie an Platz Suc, F6P und auch G6P dieses Teilnetzwerkes verursachen, dass verschiedene Transitionen die Marken dieser Plätze erzeugen oder auch verbrauchen. Somit erklären sich die ständigen Wechsel zwischen aufsteigenden und absteigenden Kurvenverläufen der Metabolite im Diagramm und auch die verschiedenen T-Invarianten. Die T-Invarianten werden nach Koch et al. [25] zum Beispiel in zwei Gruppen aufgeteilt, die Invarianten, die die Saccharose durch die Reaktion Inv oder durch die Reaktion $SuSy$ spalten, um am Ende des Teilnetzwerkes mit der gleichen Reaktion ($StaSys(b)$) Stärke zu bilden. Die Konflikte bewirken außerdem, dass der Überdeckungsgraph (nicht Erreichbarkeitsgraph da dieses Petri-Netz auf Grund der Quelltransition nicht beschränkt ist) für dieses Petri-Netz sehr viele Zustände hat und somit nicht in angemessener Zeit berechnet werden kann.

Nach sieben Simulationsschritten steigen die Kurven der Metabolite Frc und Glc an. Sacharose wird durch die unbegrenzte Aufnahme stetig in Frc und Glc umgewandelt. Diese beiden Metabolite werden nicht mehr verbraucht, denn es ist unzureichend ATP im Modell vorhanden, sodass die Folgetransitionen (HK und FK), die ATP benötigen nicht mehr schalten und die Ansammlung der beiden Metabolite verursachen. Der Metabolit G6P wird dann nur noch durch den Ablauf der Reaktionen UGPase und PGM gebildet, sodass die Stärke mit dem verbleibenden ATP und G6P bis zu Schritt 28 noch gebildet werden kann. Danach ist die Transition StaSy(b) nicht mehr lebendig und die Stärkebildung stoppt.

Mit Hilfe der Simulation des Sacharoseabbauweges der Kartoffel sind also Aussagen hinsichtlich des Systemverhaltens möglich, die ebenfalls durch die Invariantenanalyse gezogen werden können. Jedoch bedarf es bei der Auswertung der Simulationsergebnisse mehr Aufmerksamkeit und Interpretation des Nutzers als bei der Betrachtung der Invarianten- oder der Erreichbarkeitsanalyse, wobei die Ergebnisse aufbereitet werden.

5.3 Simulation des Zentralstoffwechsels im Gerstesamen

Das gewählte metabolische Modell vom Gerstesamen entspricht dem Zentralstoffwechsel. Dieses komplexe Modell wurde von Grafahrend-Belau et al. erstellt und mit Hilfe von Optimierungsalgorithmen untersucht [15]. Das Modell besteht aus 234 Metaboliten und 257 Reaktionen. Von den 257 Reaktionen sind 192 biochemische Reaktionen in verschiedenen Kompartimenten, die durch 65 Transportprozesse verbunden werden. Das Modell ist entwickelt worden, um die Maximierung von Biomasse zu simulieren. Die sich ergebende Frage hinsichtlich der Petri-Netz Simulation ist die folgende: Kann mit der Petri-Netz Simulation des Gerstesamen Modells die Bildung von Biomasse erreicht werden?

Zu Beginn ist das Modell in ein Petri-Netz Modell umzuwandeln. Die aktuelle Struktur des Modells entspricht bereits einem Petri-Netz und konnte somit ohne Probleme übernommen werden. Zur Lösung der Fragestellung bedarf es dennoch folgender Vorüberlegungen. Abhängig vom Ziel der Simulation muß über die Berandung des Petri-Netzes entschieden werden. Es gibt drei Berandungsmöglichkeiten: pure Platzberandung, pure Transitionsberandung oder eine Kombination dessen. Der Vorteil der Platzberandung für die Lösung der Fragestellung ist, dass die Menge der durch das Petri-Netz gebildeten Biomasse angesammelt und somit das Ziel der Simulation gleich sichtbar wird. Jedoch würde die Aufnahme von Metaboliten zur

Biomassebildung dargestellt, als Quellplatz den Nachteil haben, dass die minimale Menge von Metaboliten bekannt sein müsste, damit die Simulation des Petri-Netzes nicht auf Grund von Metabolitmangel stoppt. Für die Modellierung der Aufnahme von Metaboliten ist somit die Transitionsberandung geeignet, denn dann können Metabolite unbegrenzt in das Petri-Netz aufgenommen werden. Die Struktur, die sich somit für das Petri-Netz Modell des Gerstesamens ergibt, ist unbeschränkt. Diese Unbeschränktheit führt dazu, dass die Berechnung des Erreichbarkeitsgraphen nicht möglich ist.

Nachdem die Struktur des Petri-Netzes vermutlich der Lösung der Fragestellung genügt, ist die Anfangsmarkierung als Grundlage für die Simulation festzulegen. Zu Beginn muss das Petri-Netz eine Anfangsmarkierung erhalten, damit die Simulation erfolgen kann. Es ist nicht ausreichend dass Metabolite aufgenommen werden, denn viele Reaktionen laufen nur ab (Transitionen schalten nur), wenn alle Edukte einer Reaktion vorhanden sind (also die Vorbedingung erfüllt ist). Deshalb sollten auch diese Metabolite zu Beginn der Simulation in ausreichender Menge zur Verfügung stehen. Damit die Anfangsmarkierung nicht eine Bevorzugung bestimmter Metabolite erzeugt, es also einen Metaboliten in größerer Menge gibt als einen Anderen (mehr Marken auf einem Platz sind), wird eine Anfangsmarkierung von 10 Marken je Platz festgelegt (außer für die Biomasse, um deren ansteigen nachvollziehen zu können), siehe Abbildung 5.6.

Dieses Gerstesamen Petri-Netz wird für 1000 Schritte simuliert und die Markenverteilung pro Simulationsschritt für jeden Metaboliten aufgezeichnet. Das daraus resultierende Diagramm (siehe Abbildung 5.7) enthält zur Übersichtlichkeit nur die 26 Metabolite, die zur Bildung von Biomasse relevant sind. Die Kurvenverläufe der Metabolite sind zum Teil sehr unterschiedlich. Von der Anfangsmarkierung aus (Simulationsschritt 0, außer der Biomasse (`biomass_c`, `biomass_ex`) beginnen alle dargestellten Metabolite mit 10 Marken) steigen die Kurven entweder vorerst an oder steigen ab. Ein Beispiel für das Aufsteigen eines Metaboliten über alle Simulationsschritte hinweg ist `Asp_c`, das im Petri-Netz Modell eine Ansammlung darstellt. Der Metabolit `Pro_p` hingegen wird zunächst angesammelt und dann abgebaut, bis dieser verbraucht ist und nicht wieder neu gebildet wird. Ebenso verhält es sich mit der Saccharose (`sucrose_c`), die letztlich nicht verbraucht wird, sondern auf Grund der Quellknoteneigenschaft immer wieder in das Modell aufgenommen wird und somit konstant bis zum Ende der Simulation bleibt. Der konstante Kurvenverlauf kann entweder bedeuten, dass der Metabolit produziert und zur gleichen Menge wieder verbraucht wird, oder es findet keine Umwandlung durch Reaktionen statt.

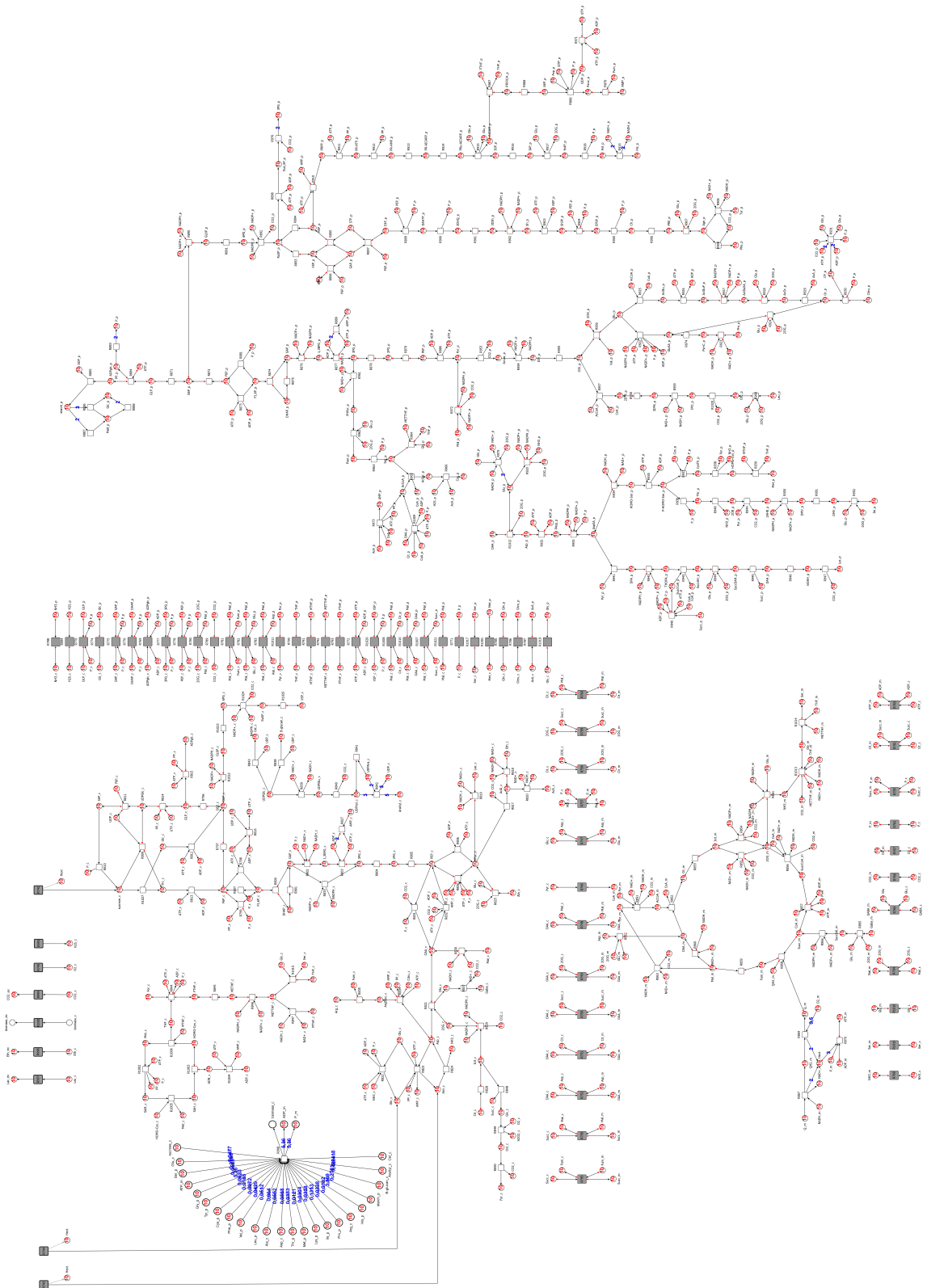


Abbildung 5.6: Petri-Netz Modell des Zentralstoffwechsel im sich entwickelnden Gerstsaamen. Das Petri-Netz besteht aus 242 Plätzen und 257 Transitionen von denen sind 144 hierarchische Transitionen und neun künstliche Transitionen zur Aufnahme (Quelltransition) bzw. Abgabe (Senkenplätze) von Metaboliten (weitere Informationen in Grafahrend-Belau et al. [15]). Die Markierung auf den Plätzen entspricht der Anfangsmarkierung (rote Werte).

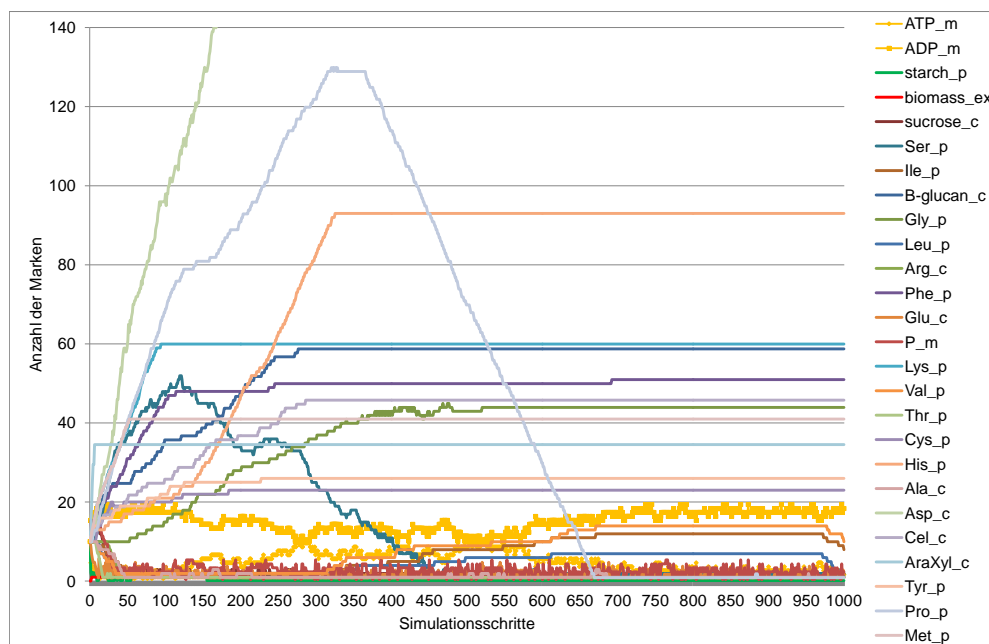


Abbildung 5.7: Markenverteilung im Gerstesamen Petri-Netz pro Simulationsschritt. Es werden von 242 Metaboliten nur die 26 Metabolite dargestellt, die zur Biomassebildung relevant sind.

Die Metabolite P_m und ADP_m werden auch zunächst angesammelt, dann aber wieder verbraucht, um anschließend wieder angesammelt zu werden. Diese Kurven verlaufen fluktuierend bis zum Ende der Simulation. Die Kurve des Metaboliten ATP_m verläuft genau entgegengesetzt zu ADP_m. Zunächst wird ATP_m verbraucht, dann wieder angesammelt und wieder verbraucht. Insgesamt verläuft die Kurve entgegengesetzt fluktuierend zu ADP_m bis zum Ende der Simulation. Dieser gegenläufige Kurvenverlauf entspricht einer P-Invariante, bestehend aus ATP_m und ADP_m. Die Summe von ATP_m und ADP_m ist konserviert über die Simulation hinweg. Je mehr ATP_m im Modell ist, desto weniger ADP_m ist zur gleichen Zeit im Modell und umgekehrt. Der Metabolit Stärke (starch_p) hingegen wird schnell verbraucht und auch im Wechsel angesammelt und wieder verbraucht, aber nach 155 Simulationsschritten nicht mehr gebildet. Die zu bildende Biomasse (biomass_ex) entsteht auch nur im ersten Schritt der Simulation und zwar genau dann, wenn alle Metabolite, die zur Biomassebildung relevant sind, in ausreichender Menge vorhanden sind, wie in [Abbildung 5.7](#) in den ersten Simulationsschritten dargestellt ist. In [Abbildung 5.8](#)

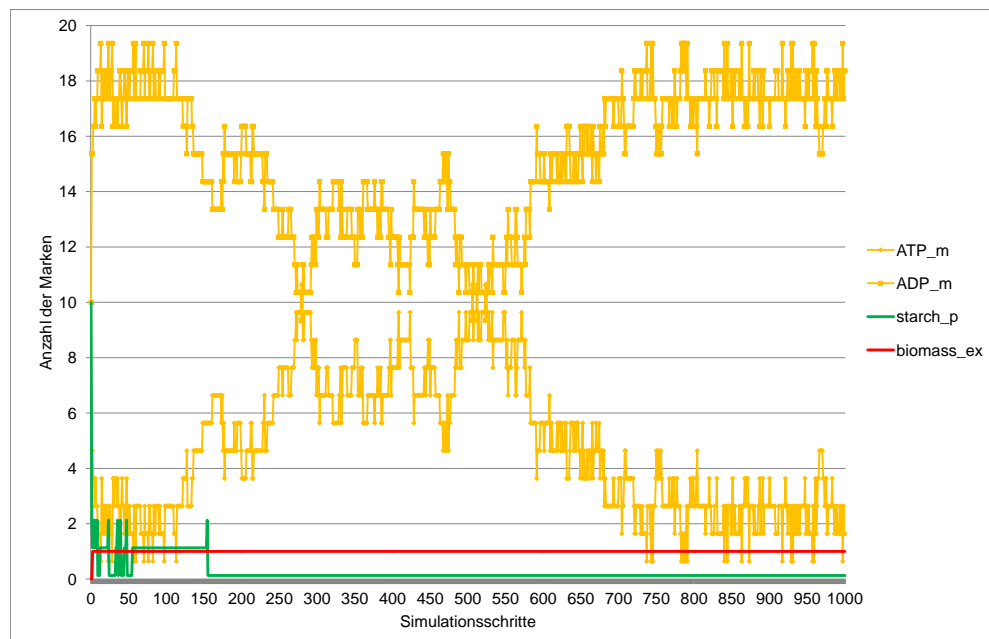


Abbildung 5.8: Markenverteilung der Metabolite ATP_m, ADP_m, starch_p und biomass_ex pro Simulationsschritt. Die Biomasse (biomass_ex) wird nur zu Beginn der Simulation gebildet. Die Kurven der Metabolite ATP_m und ADP_m verlaufen entgegengesetzt. Die Stärke (starch_p) wird nach dem Verbrauch der Anfangsmarkierung (10 Marken) nur maximal zwei Marken enthalten.

werden die Kurvenverläufe der Metabolite ADP_m, ATP_m, Stärke und der Biomasse deutlich sichtbar.

Die Diagramme stellen das Problem der Biomassebildung jedoch nicht eindeutig dar. Jeder Schritt der Simulation kann im Gerstesamen Modell durch das Petri-Netz Add-on visualisiert werden. Mit dem Fokus auf die Biomasse Reaktion (R998) wird deutlich, dass eine unzureichende Menge Stärke, die zur Bildung der Biomasse pro Simulationsschritt notwendig ist, das Problem darstellt. Die anderen Biomasse relevanten Metabolite werden angesammelt oder deren gebildete Menge ist ausreichend um die Biomasse zu bilden.

Das Problem könnte die Anfangsmarkierung sein und deshalb werden Simulationen mit verschiedenen Anfangsmarkierungen durchgeführt (nicht dargestellt). Die Anfangsmarkierungen von 10, 20, 20, 40, 50 und 60 Marken pro Metabolit führen zu einer Biomassebildung von eins, zwei, vier, fünf und acht Marken. Dabei stoppt die Biomassebildung schon zu Beginn der Simulationen wenn die Anfangsmarkierung

verbraucht ist. Die Biomasse wird durch die steigende Anfangsmarkierung zwar mehr, aber sobald die gewisse Anzahl Metabolite für die Biomasse verbraucht ist, ist die Menge Stärke wieder zu gering um weitere Biomasse zu bilden. Also unabhängig von der Anfangsmarkierung kann über die Simulationsschritte hinweg nicht mehr Biomasse gebildet werden.

Eine nähere Betrachtung der Simulation führt zu dem Schluss, dass die gleichzeitig modellierten Auf- und Abbauwege der Stärke im Gerstesamen Petri-Netz, das Problem darstellen. Der Abbau der Stärke wird durchgeführt, sobald zwei Marken auf dem Platz Stärke enthalten sind. Somit wird die Markenanzahl von 3,869 für die Biomassebildung nach dem Verbrauch der Anfangsmarkierung nicht wieder erreicht. Genau diese maximale Bildung von zwei Marken für Stärke konnte bereits im Diagramm in Abbildung 5.8 beobachtet werden.

Wenn der Abbau der Stärke nicht statt findet, sollte es möglich sein Biomasse zu bilden. Für diesen Zweck wird eine Manipulation des Petri-Netzes im Petri-Netz Add-on verwendet, das Ausschließen von Transitionen. Zwei Reaktionen, die die Stärke abbauen werden für die Simulation des Gerstesamen Modells, ausgeschlossen. Mit einer Anfangsmarkierung von 10 Marken für jeden Platz im Gerstesamen Modell (außer für die Biomasse) werden erneut 1000 Simulationsschritte durchgeführt. Das Simulationsergebnis ist für eine ausgewählte Anzahl von Metaboliten in Abbildung 5.9 dargestellt. Zunächst wird durch die Kurvenverläufe der Biomasse (`biomass_ex`) und auch der Stärke (`starch_p`) im Diagramm sichtbar, dass es bis zum Simulationsschritt 165 möglich ist genügend Stärke aufzubauen, um die Bildung der Biomasse fünf mal durchzuführen. Erst nach 100 Simulationsschritten wird genügend ATP_m (5.36 Marken für die Biomasse) gebildet, damit die Biomassebildung erneut ablaufen kann. Danach wird keine Stärke mehr aufgebaut, um für die Biomassebildung bereit zu stehen und somit stoppt die Biomassebildung erneut.

Es stellte sich heraus, dass es nicht einfach möglich ist ein bestehendes Modell, das zur Optimierung der Biomassebildung erstellt wurde in ein Petri-Netz Modell umzuformen und die Bildung der Biomasse durch die Simulation eines Petri-Netzes nachzuvollziehen. Das nicht-deterministische Schalten von Reaktionen und auch das bevorzugte Schalten, wenn genügend Marken die Vorbedingung von Transitionen erfüllen, erzeugt ein anderes Nutzungsverhalten von Reaktionen um die Biomasse zu bilden, als bei der Optimierung. Grundsätzlich kann mit dem Gerstesamen Petri-Netz Modell Biomasse gebildet werden, aber die Stabilität, also das stetige Erzeugen von Biomasse über viele Simulationsschritte, bedarf weiterer Anpassungen.

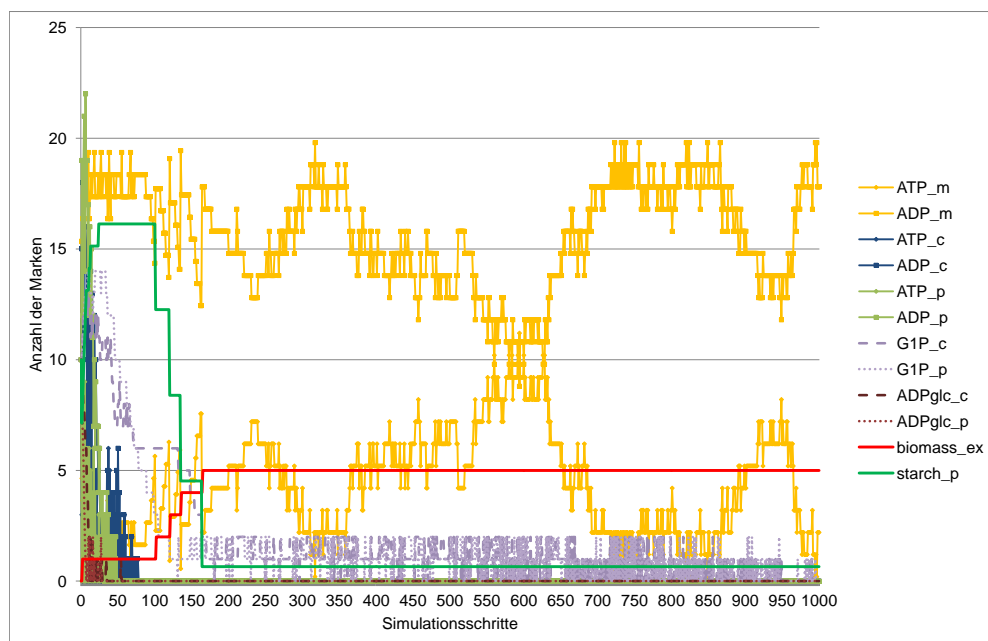


Abbildung 5.9: Markenverteilung im Gerstesamen Petri-Netz pro Simulationsschritt mit ausgeschaltetem Stärkeabbau. Es sind die Kurvenverläufe der Metabolite ATP und ADP der drei Kompartimente Zytosol (ATP_c, ADP_c), Plastid (ATP_p, ADP_p) und Mitochondrium (ATP_m, ADP_m), sowie die Stärke bildenden Metabolite G1P und ADPglc in den Kompartimenten Zytosol (G1P_c, ADPglc_c) und Plastid (G1P_p, ADPglc_p), die Biomasse (biomass_ex) und auch die Stärke (starch_p) dargestellt.

Das Petri-Netz Add-on konnte erfolgreich die Simulations- und Analysemethoden durchführen. Im ersten Anwendungsfall, der Glykolyse, wurden die P- und T-Invarianten, sowie der Erreichbarkeitsgraph zur Interpretation der Petri-Netz Eigenschaften herangezogen. Der zweite Anwendungsfall, der Saccharoseabbauweg in der Kartoffel, wurde mit P- und T-Invarianten analysiert und die Unzulänglichkeiten der T-Invariantenanalyse konnten durch die Simulation des Netzwerks kompensiert werden. Das Gerstesamen Modell, der dritte Anwendungsfall, wurde simuliert und somit validiert. Der Export der Simulationsergebnisse unterstützte die Interpretation der Ergebnisse. Auch die verschiedenen Möglichkeiten der Manipulation des Petri-Netzes, wie das Ausschließen von Reaktionen in der Simulation, konnten das Verständnis für die Funktionsweise des Modells verbessern.

6 Zusammenfassung und Ausblick

Die Anforderungen an diese Arbeit konnten mit dem Petri-Netz Add-on für VANTED erfolgreich umgesetzt werden. Das Petri-Netz Add-on für VANTED kann erweiterte Platz/Transitions-Netze editieren, den Markenfluss simulieren und mit Hilfe der Invariantenanalyse Platz- und Transitionsinvarianten berechnen, sowie den Erreichbarkeits- und Überdeckungsgraphen erstellen. Die Funktionalität des Editierens umfasst das Erstellen, Laden und Umwandeln bestehender biologischer Netzwerke in Platz/Transitions-Netze. Unterstützt wird der Im- und Export von Petri-Netzen durch das standardisierte Dateiformat PNML und allen bereits in VANTED verfügbaren Im- und Export Funktionen.

Die Petri-Netze können in verschiedenen Modi simuliert werden. Das Einzel- und Mehrschrittverfahren animiert die Markenverteilung im Petri-Netz. Visuell kann das Schalten der Transitionen zu jedem Simulationsschritt, durch farbliche Hervorhebung, nachvollzogen werden. Des Weiteren kann jede Markierung pro Simulationsschritt aufgezeichnet werden, sodass das Simulationsergebnis, zum Beispiel in Tabellenkalkulationsprogrammen aufbereitet und detailliert untersucht werden kann. Zukünftig könnten die aufgezeichneten Markierungen pro Simulationsschritt mit VANTED auf die Plätze gemappt werden. VANTED besitzt bereits eine Mappingfunktion, um Daten auf Netzwerkelemente, wie Knoten und Kanten zu mappen. Es wäre somit möglich für jeden Platz die Markenverteilung über die Simulationsschritte zu visualisieren. Der Nutzer kann durch diese Visualisierung direkt beurteilen, ob ein Platz zum Beispiel Marken ansammelt oder unzureichend Marken enthält. Die Verwendung zusätzlicher Tabellenkalkulationsprogramme, um das Simulationsergebnis detailliert untersuchen zu können, entfällt. Zusätzlich ist es auch denkbar die Schalthäufigkeit von Transitionen pro Simulationsschritt aufzuzeichnen und direkt auf die Transitionen zu mappen.

Die Analyse der Petri-Netze bietet die Möglichkeit, die berechneten Ergebnisse mit Hilfe der implementierten Interaktionstechnik, für den Nutzer visuell aufzubereiten. Die Invarianten können interaktiv im Petri-Netz betrachtet werden. Der Nutzer selektiert in der Tabelle des Dialoges eine Invariante, die somit interaktiv im Petri-Netz visualisiert wird. Diese Interaktionstechnik findet ebenso in der Erreichbarkeitsanalyse Verwendung. Hier können die Knoten (Zustände) oder Kanten

(Transitionen) des Erreichbarkeits- und Überdeckungsgraphen selektiert werden und die entsprechende Markierung des Zustandes oder die geschaltete Transition im Petri-Netz interaktiv visualisiert werden. Somit kann der Nutzer, beginnend mit der Anfangsmarkierung, alle möglichen Zustände des Petri-Netzes nachvollziehen und die sich daraus ergebenden Eigenschaften ablesen.

Anhand drei sich in ihrer Komplexität unterscheidender biologischer Netzwerke wird dargestellt, in welchem Umfang die Funktionalität des Petri-Netz Add-ons eingesetzt werden kann und wo die Grenzen der Analysen liegen. Die Analyse und Simulation kann für wenig komplexe Petri-Netze, wie der Glykolyse vollständig durchgeführt werden. Je Komplexer ein Petri-Netz wird, desto weniger Analysemethoden können in absehbarer Zeit Ergebnisse berechnen. Die Erreichbarkeitsanalyse ist Zeit und Ressourcen aufwändiger als die Invariantenanalyse und kann somit in komplexeren Petri-Netzen, wie dem Sacharoseabbauweg in der Kartoffel, auf Grund der Zustandsraum-Explosion, nicht mehr berechnet werden. Aber auch die Invariantenanalyse hat ihre Grenzen, sodass in sehr komplexen Petri-Netzen, wie dem Gerstesamen Modell, nur noch die Simulation Rückschlüsse auf Petri-Netz Eigenschaften zulässt.

Damit zukünftig der Nutzer Einfluss auf die Simulation nehmen kann, könnten den Transitionen im Petri-Netz Prioritäten hinzugefügt werden. Mit Hilfe von Prioritäten können Transitionen, die sich im Konflikt befinden, der Wichtigkeit nach geschaltet werden. Der Nutzer kann zum Beispiel reversiblen Reaktionen eine bevorzugte Richtungen vorgeben und somit den Markenfluss lenken.

Literaturverzeichnis

- [1] AMENGUAL, ALBERTO: *A computational model of attachment secure responses in the Strange Situation*. Technischer Bericht, International Computer Science Institute, 2009.
- [2] AZEVEDO, MÁRCIO: *JPetriNet*. <http://jpetrinet.sourceforge.net>, 2004. Abgerufen am 10.09.2011.
- [3] BALDAN, PAOLO, NICOLETTA COCCO, ANDREA MARIN und MARTA SIMEONI: *Petri nets for modelling metabolic pathways: a survey*. Natural Computing, 9:955–989, 2010.
- [4] BALZERT, HEIDE: *Lehrbuch der Objektmodellierung: Analyse und Entwurf*. Lehrbücher der Informatik. Spektrum Akademischer Verlag, Heidelberg-Berlin, 1999.
- [5] BERNARDINELLO, LUCA und FIORELLA DE CINDIO: *A survey of basic net models and modular net classes*. In: ROZENBERG, GRZEGORZ (Herausgeber): *Advances in Petri Nets: The DEMON Project*, Band 609 der Reihe *Lecture Notes in Computer Science*, Seiten 304–351, London, 1992. Springer-Verlag.
- [6] BONET, PERE, CATALINA LLADÓ, RAMON PUIGJANER und WILLIAM J. KNOTTENBELT: *PIPE v2.5: A Petri Net Tool for Performance Modelling*. In: *23rd Latin American Conference on Informatics (CLEI 2007)*, September 2007.
- [7] BRUSEY, JAMES: *Petri(LLD) Tutorial*, 2006.
- [8] BYG, JOAKIM, KENNETH YRKE JØRGENSEN und JIŘÍ SRBA: *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)*, Band 5799 der Reihe *Lecture Notes in Computer Science*, Seiten 84–89, Berlin-Heidelberg, 2009. Springer-Verlag.
- [9] CASPI, RON, TOMER ALTMAN, JOSEPH M. DALE, KATE DREHER, CAROL A. FULCHER, FRED GILHAM, PALLAVI KAIPA, ATHIKKATTUVALASU S. KAR-

- THIKEYAN, ANAMIKA KOTHARI, MARKUS KRUMMENACKER, MARIO LATENDRESSE, LUKAS A. MUELLER, SUZANNE PALEY, LIVIU POPESCU, ANURADHA PUJAR, ALEXANDER G. SHEARER, PEIFEN ZHANG und PETER D. KARP: *The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of pathway/genome databases*. Nucleic Acids Research, 38:D473–D479, 2010.
- [10] DEVILLERS, RAYMOND R.: *The semantics of capacities in P/T nets*. In: ROZENBERG, GRZEGORZ (Herausgeber): *European Workshop on Applications and Theory in Petri Nets*, Band 424 der Reihe *Lecture Notes in Computer Science*, Seiten 128–150, London, 1988. Springer-Verlag.
- [11] EICK, STEPHEN G. und GRAHAM J. WILLS: *High interaction graphics*. European Journal of Operational Research, 81(3):445–459, 1995.
- [12] ERHARD, FLORIAN, CAROLINE C. FRIEDEL und RALF ZIMMER: *FERN - a Java framework for stochastic simulation and evaluation of reaction networks*. BMC Bioinformatics, 9(1):356, 2008.
- [13] FRIEDRICHS, FRIEDRICH D.: *Referenznetze mit Anschriften in Scheme*. Diplomarbeit, Universität Hamburg, 2007.
- [14] FUNAHASHI, AKIRA, MINEO MOROHASHI, HIROAKI KITANO und NAOKI TANIMURA: *CellDesigner: a process diagram editor for gene-regulatory and biochemical networks*. Biosilico, 1(5):159–162, 2003.
- [15] GRAFAHREND-BELAU, EVA, FALK SCHREIBER, DIRK KOSCHÜTZKI und BJÖRN H. JUNKER: *Flux balance analysis of barley seeds: a computational approach to study systemic properties of central metabolism*. Plant Physiology, 149(1):585–598, 2009.
- [16] GURSKI, FRANK, IRENE ROTHE, JÖRG ROTHE und EGON WANKE: *Exakte Algorithmen für schwere Graphenprobleme*. Springer-Verlag, Berlin-Heidelberg, 2010.
- [17] HALDER, ABHISHEK: *A Study of Petri Nets: Modeling, Analysis and Simulation*. Dissertation, Indian Institute of Technology Kharagpur, Department of Aerospace Engineering, 2006.
- [18] HEITMANN, FRANK und DANIEL MOLDT: *Petri Nets World*. <http://www.informatik.uni-hamburg.de/TGI/PetriNets>, 2005. Abgerufen am 09.09.2011.

- [19] HILLAH, LOM MESSAN, EKKART KINDLER, FABRICE KORDON, LAURE PETRUCCI und NICOLAS TRÈVES: *A primer on the Petri Net Markup Language and ISO/IEC 15909-2*. Petri Net Newsletter (originally presented at the 10th International workshop on Practical Use of Colored Petri Nets and the CPN Tools - CPN'09), 76:9–28, October 2009.
- [20] HOFESTÄDT, RALF: *A petri net application to model metabolic processes*. Systems Analysis Modelling Simulation, 16:113–122, 1994.
- [21] JUNKER, BJÖRN H., CHRISTIAN KLUKAS und FALK SCHREIBER: *VANTED: A system for advanced data analysis and visualization in the context of biological networks*. BMC Bioinformatics, 7(1), 2006.
- [22] KELDER, THOMAS, MARTIJN P. VAN IERSEL, KRISTINA HANSPERS, MARTINA KUTMON, BRUCE R. CONKLIN, CHRIS T. EVELO und ALEXANDER R. PICO: *WikiPathways: building research communities on biological pathways*. Nucleic Acids Research, 2011.
- [23] KLINK, STEFAN, YU LI und ANDREAS OBERWEIS: *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, Seiten 14:1–14:8. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.
- [24] KLUKAS, CHRISTIAN: *Analyse und Visualisierung von Experimentdaten im Kontext biologischer Netzwerke*. Dissertation, Martin Luther Universität Halle-Wittenberg, Naturwissenschaftliche Fakultät III, 2009.
- [25] KOCH, INA, BJÖRN H. JUNKER und MONIKA HEINER: *Application of Petri net theory for modelling and validation of the sucrose breakdown pathway in the potato tuber*. Bioinformatics, 21(7):1219–1226, 2005.
- [26] KOCH, INA, WOLFGANG REISIG und FALK SCHREIBER (Herausgeber): *Modeling in Systems Biology, The Petri Net Approach*, Band 16 der Reihe *Computational Biology*. Springer-Verlag, London, 2011.
- [27] KUHL, FRED: *Tortuga*. <http://code.google.com/p/tortugades>, 2004. Abgerufen am 10.09.2011.

- [28] LE NOVÈRE, NICOLAS, MICHAEL HUCKA, HUAIYU MI, STUART MOODIE, FALK SCHREIBER, ANATOLY SOROKIN, EMEK DEMIR, KATJA WEGNER, MIRIT I. ALADJEM, SARALA M. WIMALARATNE, FRANK T. BERGMAN, RALPH GAUGES, PETER GHAZAL, HIDEYA KAWAJI, LU LI, YUKIKO MATSUOKA, ALICE VILLÉGER, SARAH E. BOYD, LAURENCE CALZONE, MELANIE COURTOT, UGUR DOGRUSOZ, TOM C. FREEMAN, AKIRA FUNAHASHI, SAMIK GHOSH, AKIYA JOURAKU, SOHYOUNG KIM, FEDOR KOLPAKOV, AUGUSTIN LUNA, SVEN SAHLE, ESTHER SCHMIDT, STEVEN WATTERSON, GUANMING WU, IGOR GORYANIN, DOUGLAS B. KELL, CHRIS SANDER, HERBERT SAURO, JACKY L. SNOEP, KURT KOHN und HIROAKI KITANO: *The Systems Biology Graphical Notation*. Nature Biotechnology, 27(8):735–741, 2009.
- [29] MARTIN, ALLEN R. und MATTHEW O. WARD: *High Dimensional Brushing for Interactive Exploration of Multivariate Data*. In: *Proceedings of the 6th conference on Visualization '95, VIS '95*, Seiten 271–278. IEEE Computer Society, 1995.
- [30] MARTINEZ, J. und MANUEL SILVA: *A Simple and Fast Algorithm to Obtain All Invariants of a Generalized Petri Net*. In: *Selected Papers from the First and the Second European Workshop on Application and Theory of Petri Nets*, Seiten 301–310, London, 1982. Springer-Verlag.
- [31] MURATA, TADAO: *Petri nets: Properties, Analysis and Applications*. Proceedings of the IEEE, 77(4):541–580, 1989.
- [32] NOWOSTAWSKI, MARIUSZ: *JFern Manual Version 4.0.0*, August 2009.
- [33] PADILHA, RICARDO S.: *JARP Petri Nets Analyzer*. <http://jarp.sourceforge.net>, 2001. Abgerufen am 10.09.2011.
- [34] PETRI, CARL ADAM: *Kommunikation mit Automaten*. Dissertation, Universität Hamburg, 1962.
- [35] REDDY, VENKATRAMANA N., MICHAEL L. MAVROVOUNIOTIS und MICHAEL N. LIEBMAN: *Petri Net Representations in Metabolic Pathways*. In: *Proceedings of the 1st International Conference on Intelligent Systems for Molecular Biology*, Band 1, Seiten 328–336. AAAI Press, 1993.
- [36] REISIG, WOLFGANG: *Petrinetze: Modellierungstechnik, Analysemethoden, Fallstudien*. Leitfäden der Informatik. Vieweg+Teubner Verlag, Wiesbaden, 2010.

- [37] RIESZ, MARTIN: *PNEditor*. <http://www.pneditor.org>, 2010. Abgerufen am 10.09.2011.
- [38] ROCH, STEPHAN und PETER H. STARKE: *INA-Integrated Net Analyzer*. <http://www2.informatik.hu-berlin.de/~starke/ina.html>, 1999. Abgerufen am 27.11.2011.
- [39] SCHREIBER, FALK, CHRISTIAN COLMSEE, TOBIAS CZAUDERNA, EVA GRAFAHREND-BELAU, ANJA HARTMANN, ASTRID JUNKER, BJÖRN H. JUNKER, MATTHIAS KLAPPERSTÜCK, UWE SCHOLZ und STEPHAN WEISE: *MetaCrop 2.0: managing and exploring information about crop plant metabolism*. Nucleic Acids Research, 2011.
- [40] SCHUSTER, STEFAN, THOMAS DANDEKAR und DAVID A. FELL: *Detection of elementary flux modes in biochemical networks: a promising tool for pathway analysis and metabolic engineering*. Trends in Biotechnology, 17(2):53–60, 1999.
- [41] SHANNON, PAUL, ANDREW MARKIEL, OWEN OZIER, NITIN S. BALIGA, JONATHAN T. WANG, DANIEL RAMAGE, NADA AMIN, BENNO SCHWIKOWSKI und TREY IDEKER: *Cytoscape: a software environment for integrated models of biomolecular interaction networks*. Genome Research, 13(11):2498–2504, 2003.
- [42] STARKE, PETER H.: *Analyse von Petri-Netz-Modellen*. Leitfäden und Monographien der Informatik. Teubner Verlag, Stuttgart, 1990.
- [43] WEBER, MICHAEL: *Allgemeine Konzepte zur software-technischen Unterstützung verschiedener Petrinetz-Typen*. Dissertation, Humboldt-Universität zu Berlin, Mathematisch-Naturwissenschaftliche Fakultät II, 2002.
- [44] WIMMEL, HARRO und LUTZ PRIESE: *Petri-Netze*. Springer-Verlag, Berlin-Heidelberg, 2. Auflage, 2008.
- [45] ZEVEDEI-OANCEA, IONELA und STEFAN SCHUSTER: *Topological analysis of metabolic networks based on Petri net theory*. In Silico Biology, 3:323–345, 2003.

Danksagung

Für die Vergabe meiner Masterarbeit bedanke ich mich ganz herzlich bei Herrn Prof. Dr. Burkhard Wrenger, Vizepräsident für Lehre und Internationalisierung an der Hochschule Ostwestfalen-Lippe.

Bei Herrn Prof. Dr. Falk Schreiber, Koordinator der Bioinformatik und Leiter der Arbeitsgruppe Pflanzenbioinformatik am Leibniz-Institut für Pflanzengenetik und Kulturpflanzenforschung (IPK) Gatersleben, bedanke ich mich für die Möglichkeit, die Masterarbeit im Umfeld des IPK durchführen zu dürfen.

Besonderer Dank gilt meinem Betreuer Hendrik Rohn, wissenschaftlicher Mitarbeiter der Arbeitsgruppe Pflanzenbioinformatik am IPK. Er hat mir auf der Suche nach Lösungswegen immer Rede und Antwort gestanden. Danke für die Unterstützung und vielseitigen Denkanstöße bei der Erstellung dieser Masterarbeit. Außerdem danke ich der gesamten Arbeitsgruppe Pflanzenbioinformatik, für die angenehme und konstruktive Arbeitsatmosphäre.

Vor allem möchte ich mich bei meiner Freundin für ihre enorme Geduld und ihr Interesse beim Entstehen dieser Arbeit bedanken.

Ohne die liebevolle Unterstützung während des Studiums durch meine Familie wäre diese Arbeit nicht entstanden. Sie haben mir eine erfolgreiche und sorgenfreie Studienzeit durch moralische und materielle Unterstützung geboten und standen mir bei Schwierigkeiten immer zur Seite. Ich danke euch von Herzen dafür!

Ehrenwörtliche Erklärung

Ich versichere, die von mir vorgelegte Masterarbeit selbständig verfasst zu haben. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder nicht veröffentlichten Arbeiten anderer entnommen sind, habe ich als entnommen kenntlich gemacht. Sämtliche Quellen und Hilfsmittel, die ich für die Masterarbeit benutzt habe, sind angegeben. Diese Masterarbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde, weder im In- noch im Ausland, vorgelegen.

Quedlinburg, den 2. Januar 2012

Kevin Pucknat

Hochschule Ostwestfalen-Lippe
University of Applied Sciences

HS Ostwestfalen-Lippe • An der Wilhelmshöhe 44 • 37671 Höxter

An den Vorsitzenden des Prüfungsausschusses
Master Environmental Sciences
 Herrn Prof. Dr. Winfried Türk

Prof. Dr. Burkhard Wrenger

An der Wilhelmshöhe 44
 37671 Höxter 9. Juni 2011
 Telefon: (05271) 687 0
 Durchwahl: (05271) 687 122
 Fax: (05271) 687 200
 Internet: <http://www.hs-owl.de>
 E-mail: burkhard.wrenger@hs-owl.de

Vergabe einer Masterarbeit an Herrn Kevin Pucknat

Hiermit vergebe ich eine Masterarbeit zum Thema „Erweiterung der Software VANTED zum Editieren und Simulieren biologischer Petri-Netze“.

Die Masterarbeit wird im Umfeld des Leibniz-Instituts für Pflanzengenetik und Kulturpflanzenforschung (IPK) in Gatersleben durchgeführt. Das IPK gehört zu den international bedeutsamen Zentren der Pflanzenforschung und bearbeitet Probleme der modernen Biologie vorrangig an Kulturpflanzen. Im Zentrum der grundlagen- und anwendungsorientierten Forschung steht die Erarbeitung neuer Erkenntnisse und Technologien mit dem Ziel einer umfassenden Nutzung pflanzengenetischer Ressourcen für eine optimierte Stoffproduktion und eine umweltverträgliche Landwirtschaft. Die Arbeitsgruppe Pflanzenbioinformatik der Abteilung Molekulare Genetik beschäftigt sich mit Modellierung, Analyse, Simulation und Visualisierung biologischer Netzwerke im Kontext pflanzenbiologischer Probleme. In der Arbeitsgruppe wurde in den vergangenen Jahren die Software VANTED für die Analyse biologischer Netzwerke entwickelt. Dieser soll im Rahmen der vergebenen Masterarbeit erweitert werden.

Ziel der Arbeit ist die Erstellung eines Add-ons für VANTED. Dabei sind folgende Punkte zu bearbeiten bzw. zu beachten:

1. Entwurf und Implementierung des Add-ons
2. Berücksichtigung von diskreten und kontinuierlichen Petri-Netzen
3. Möglichkeit zum Editieren und Simulieren (Einzelschritt, Mehrschritt) von Petri-Netzen
4. Implementierung von Analysefunktionen (P/T-Invarianten, Erreichbarkeit)
5. Import und Export von Petri-Netzen
6. Als Basis für die Entwicklung ist eine Opensource-Bibliothek vorzusehen
7. Bereitstellung von geeigneten Beispieldaten zur Dokumentation der Funktionalität

Der Arbeit ist ein elektronischer Datenträger mit folgendem Inhalt beizulegen:

- Die Arbeit in elektronischer Form
- Bei Online-Literatur die Originalquellen als Datei
- Quelltext- und Projektdateien
- Ein Abstract in deutscher und englischer Sprache

Herr Pucknat erklärt mit der Abgabe der Masterarbeit sein Einverständnis zur Veröffentlichung des Abstracts auf den Webseiten des Studiengangs bzw. des Fachgebietes.

Bu. Wrenger

Ausgabe: 05.09.2011

Abgabe: 05.01.2012

