

Lehrstuhl Datenstrukturen und
Softwarezuverlässigkeit

Diplomarbeit



Brandenburgische Technische
Universität Cottbus

Invariantenbasierte Strukturierung von Petri-Netzen

Katja Winder

kwinder@informatik.tu-cottbus.de

31. Mai 2006

Betreuer / Gutachter : Prof. Dr.-Ing. Monika Heiner
Prof. Dr. rer. nat. Ina Koch

Eidesstattliche Erklärung

Die vorliegende Diplomarbeit wurde von mir selbständig angefertigt. Die verwendeten Hilfsmittel und Quellen sind im Literaturverzeichnis vollständig aufgeführt.

Eingetragene Warenzeichen und Copyrights werden anerkannt, auch wenn sie nicht explizit gekennzeichnet sind.

Cottbus, den 22. Juli 2006

Katja Winder

Danksagung

An dieser Stelle möchte ich mich bei allen Personen für die vielseitige Unterstützung während meiner Diplomarbeitsphase bedanken. Einen besonderen Dank verdienen die folgenden Personen:

Frau Prof. Heiner für unsere fruchtbaren Diskussionen, für Denkanstöße und Hinweise und für die engagierte Betreuung meiner Diplomarbeit.

Frau Prof. Krebs, meine Mentorin, die mir immer mit Rat und Tat zur Seite stand.

Oliver S. für seine seelische und moralische Unterstützung.

Inhaltsverzeichnis

Inhaltsverzeichnis	vii
Abbildungsverzeichnis	ix
Tabellenverzeichnis	xi
1 Einleitung	1
1.1 Motivation	2
1.2 Ziel der Diplomarbeit	3
1.3 Aufbau der Diplomarbeit	3
2 Grundlagen Petri-Netze	5
2.1 Definitionen	5
2.2 Eigenschaften	7
2.2.1 Strukturelle Eigenschaften	8
2.2.2 Netzinvarianten	9
2.3 Konventionen	13
3 Ansätze zur Auswertung von Invarianten	15
3.1 Auswertung mit Data Mining	16
3.2 Abhängige Transitions Mengen	17
3.2.1 Definitionen	19
3.2.2 Berechnung der DTS	21
3.2.3 Eigenschaften der abhängigen Transitions Mengen	23
3.2.4 Anwendungen der abhängigen Transitions Mengen	26
3.3 Zusammenfassung der vorgestellten Verfahren	30
4 Hierarchisierung und Reduktion des Petri-Netzes	33
4.1 Hierarchische Strukturierung	34
4.2 Trägerbasierte Reduktion	36
4.3 Häufigkeitsbasierte Reduktionsregel	38
4.4 Einschränkungen der Reduktionsregeln	50
5 Fallstudie	53

5.1	DTS in biochemischen Netzwerken	53
5.1.1	Grundlagen	53
5.1.2	Biologische Interpretation der T-Invarianten	54
5.1.3	Biologische Bedeutung der DTS	56
5.2	Fallstudie: Glycolyse	56
6	Zusammenfassung und Ausblick	63
6.1	Zusammenfassung	63
6.2	Ausblick	64
A	Implementierung	67
A.1	Darstellung von Invarianten und anderen Knotenvektoren mit Snoopy	67
A.2	DTS-Berechnung und DTS-Anzeige mit PInA und Snoopy	69
A.3	Machbarkeitsstudie zu Hierarchisierung und Reduktion	70
	Literaturverzeichnis	71

Abbildungsverzeichnis

2.1	Schalten einer Transition	6
2.2	Einfaches Beispiel für einen Konflikt	9
2.3	Beispielnetz mit Inzidenzmatrix	9
2.4	Beispiel für minimale und einfache T-Invarianten	11
3.1	Beispielnetz für abstract DTS	20
3.2	Beispielnetze für Transitionstypen innerhalb einer DTS	25
3.3	Invariantenbeschreibung mit finitem Automat	29
4.1	Beispiel für Hierarchisierung mit abstract DTS	35
4.2	Beispiel für trägerbasierte Reduktion	37
4.3	Beispiel für zwei Kantenbewertungsvarianten	40
4.4	Beispiel für häufigkeitsbasierte Reduktion	43
4.5	Beispiel für ein Petri-Netz, das nach der häufigkeitsbasierten Reduktion be- schränkt ist	49
4.6	Beispiel für Verlust der Realisierbarkeit bei der Reduktion	50
4.7	Beispiel für Reduktion zusammenhängender Teilnetze	51
5.1	Glycolyse-Netzwerk	58
5.2	Reduziertes Glycolyse-Netzwerk	60

Tabellenverzeichnis

2.1	Minimale und einfache T-Invarianten	11
3.1	Eine Menge von T-Invarianten	17
3.2	Beispiele für Assoziationsregeln zu Tabelle 3.1	18
3.3	Abstract DTS des Netzes aus Abbildung 3.1	19
3.4	Strong DTS des Netzes aus Abbildung 3.1	21
4.1	Vergleich der T-Invarianten vor und nach der Reduktion	43
5.1	Minimale T-Invarianten des Glycolyse-Netzwerkes	59
5.2	Maximale strong DTS des Glycolyse-Netzwerkes	59
5.3	Reduzierte strong DTS des Glycolyse-Netzwerkes	59
5.4	Einfache T-Invarianten des reduzierten Glycolyse-Netzwerkes	61

Abkürzungsverzeichnis

abstract DTS	abstrahiert abhängige Transitionsmenge (engl. abstractly dependent transition set)
ADP	Adenosine diphosphate
ATP	Adenosine triphosphate
BDD	binäres Entscheidungsdiagramm (engl. binary decision diagram)
DEA	deterministischer endlicher Automat
DTS	abhängige Transitionsmenge (engl. dependent transition set)
INA	Integrated Net Analyser
MCS	minimale Schnittmenge (engl. minimal cut set)
P-Invariante	Platzinvariante
PInA	Petrinet Invariant Analyser
strong DTS	stark abhängige Transitionsmenge (engl. strongly dependent transition set)
T-Invariante	Transitionsinvariante

Aufgabe

Bei der qualitativen Analyse von Petri-Netzen können Invarianten besonders hilfreich sein, doch die Menge der Invarianten kann exponentiell mit der Menge der Knoten im Petri-Netz wachsen. Große Mengen an Invarianten können aber kaum mehr manuell analysiert werden, so dass eine automatisierte Unterstützung des Analyseprozesses hilfreich ist.

Ziel dieser Arbeit ist ein geeignetes Verfahren, um Invariantenmengen zu strukturieren und zu charakterisieren. Das Verfahren soll dem Anwender die Möglichkeit bieten, Petri-Netze mit Hilfe der zugehörigen Invarianten zu validieren und zentrale Abläufe im Petri-Netz zu extrahieren. Besondere Bedeutung soll dabei eine visualisierte Charakterisierung haben.

Es können sowohl exakte als auch abstrahierende Verfahren verwendet werden. Eine prototypische Umsetzung der Verfahren soll implementiert werden. Die Diplomarbeit ist so anzufertigen, dass sie ohne zusätzliche Literatur verständlich ist.

Kapitel 1

Einleitung

Der Mensch versucht seit jeher, seine Umgebung zu verstehen und die komplexen Prozesse in seiner Umwelt zu seinem Wohl anzuwenden und weiterzuentwickeln. Die menschliche Umgebung besteht aus einer Fülle von Systemen, die miteinander agieren und sich gegenseitig beeinflussen. Doch schon die einzelnen Systeme selbst sind häufig so komplex, dass sie nicht durch einfaches Beobachten verstanden werden können.

Deshalb erfolgt die Beschreibung von komplexen Systemen zumeist durch Modelle, die die Eigenschaften und das Verhalten des Systems vereinfacht, aber möglichst adäquat darstellen. Am Anfang des Weges vom realen System zum Modell steht ein grundlegendes Verständnis des Systems. Schrittweise wird dieses Verständnis in ein Modell umgesetzt. Die Eigenschaften des Modells werden im Bezug zu den erwarteten Eigenschaften des realen Systems verifiziert. Das Modell sollte helfen, Erklärungen für am realen System beobachtbare Vorgänge zu finden. Am Ende des Modellierungsprozesses steht ein Modell, dessen Verhalten als Vorhersage für das Verhalten des realen Systems dienen kann.

Petri-Netze als mathematisches Modell eignen sich zur Analyse und Simulation von dynamischen Systemen mit nebenläufigen Prozessen und nichtdeterministischem Verhalten. Ein wesentlicher Vorteil von Petri-Netzen als Modell ist die breite theoretische Grundlage, auf der die formale Modellierung und Analyse aufbauen können. Es gibt außerdem zahlreiche Simulations- und Analysewerkzeuge, die den Modellierungsprozess unterstützen.

Die Anwendung von Petri-Netzen zum Modellieren von realen Systemen ist vielfältig, Petri-Netz-Modelle werden im administrativen Bereich und in der Forschung eingesetzt. Zum Beispiel werden Fertigungslinien mit Petri-Netzen modelliert und simuliert. Ein Ziel ist es dabei, die Fehlerfreiheit des Systems nachzuweisen. Weiterhin eignen sich Petri-Netze dazu, Verfahren und Abfolgen zu optimieren und damit effizienter zu produzieren. Biochemische Netzwerke, als Petri-Netz dargestellt, werden zur Erforschung von Prozessen in Lebewesen eingesetzt. Das Verständnis der Abläufe in biochemischen Netzwerken spielt bei der Entwicklung von Therapien und Medikamenten für den Menschen eine große Rolle. Gut verstandene Prozesse können leichter unterbunden oder aufrechterhalten werden, je nachdem, ob es sich um die Fortpflanzung eines schädlichen Bakterienstammes oder die Produktion von lebenswichtigen Enzymen in den Zellen handelt.

1.1 Motivation

Bei der Validation von Petri-Netz-Modellen werden unter anderem sogenannte Invarianten benutzt. Eine Invariante ist im Allgemeinen eine Eigenschaft eines Systems, die sich während der Arbeit des Systems nicht ändert. Invarianten von Petri-Netzen sind in technischer Hinsicht die Lösungen bestimmter homogener Gleichungssysteme, die das Petri-Netz beschreiben. Ein wesentlicher Vorteil der Verwendung von Invarianten zur Validation ist deren Unabhängigkeit vom Zustandsraum des Petri-Netzes. Ist dieser unendlich, können viele andere Techniken nicht mehr eingesetzt werden, etwa Techniken, die auf dem sogenannten Erreichbarkeitsgraphen basieren.

Invarianten können in den verschiedenen Anwendungen der Petri-Netze unterschiedliche Bedeutungen haben. In Modellen für Fertigungslinien stellen sogenannte T-Invarianten zum Beispiel Zyklen im Fertigungssystem dar. Die Herstellung eines Produktes muss jeweils einem solchen Zyklus entsprechen, damit das Fertigungssystem an seinen Anfangszustand zurückkehrt. In [Fre00] wird beschrieben, wie T-Invarianten dafür benutzt werden, die Reihenfolge für die Fertigung von Produkten entsprechend der Auftragslage flexibel anzupassen.

Ein Anwendungsgebiet, in dem zur Zeit verstärkt Petri-Netze eingesetzt werden, ist die Systembiologie. Hier werden Methoden und Techniken gesucht, die das Verständnis biochemischer Prozesse fördern. Dabei werden Aufbau, Funktionalität und Fähigkeiten eines biologischen Systems untersucht. Ziel ist es, die Struktur des Systems aufzudecken, seine Komponenten und ihr Zusammenwirken zu untersuchen.

Dabei eignen sich Petri-Netze aufgrund ihrer mathematischen Eigenschaften sehr gut dazu, biochemische Reaktionen und den Zusammenhang zwischen den Reaktionen darzustellen. In [KJH05] wird am Beispiel der Zuckerzerlegung in der Kartoffelknolle ein Ansatz beschrieben, die Petri-Netz-Theorie zur Validation von Netzwerkmodellen zu verwenden. T-Invarianten beschreiben gerade das minimale Verhalten des biochemischen Systemmodells und tragen zu einem besseren Verständnis der einzelnen Abläufe bei. Sie ermöglichen damit die Entwicklung gezielter Maßnahmen am realen System.

Ein weiteres Anwendungsgebiet, in dem invariantenbasierte Techniken zum Einsatz kommen können, ist das Testen von Software. Die zum Finden geeigneter Testfälle benutzten Ursache-Wirkungs-Graphen [Lig02] können mit Petri-Netzen modelliert werden. Die T-Invarianten der modellierten Netze liefern dabei gerade die Testfälle, die für eine umfassende Ursache-Wirkungs-Analyse durchgeführt werden müssen [Hei04].

Leider kann schon bei Petri-Netzen mit wenigen Komponenten die Anzahl der Invarianten sehr groß werden. Damit können die einzelnen Invarianten ab einer bestimmten Anzahl nicht mehr manuell analysiert werden. Deshalb sind automatisierbare Methoden zur strukturellen Charakterisierung einer Menge von Invarianten notwendig, die während des Modellierungsprozesses und bei der Auswertung des Petri-Netzes des biochemischen Netzwerkes eingesetzt werden können.

1.2 Ziel der Diplomarbeit

Die Methoden und Werkzeuge, die im Rahmen dieser Diplomarbeit entwickelt werden, unterstützen den Modellierungsprozess und den Einsatz von Petri-Netz-Modellen. Die strukturelle Charakterisierung der Invarianten erfolgt automatisierbar und wird theoretisch fundiert dargestellt.

Die strukturelle Charakterisierung der Invarianten soll Vergleiche zwischen den Invarianten ermöglichen und eine komprimierte strukturierte Darstellung der Menge der Invarianten und des zugrunde liegenden Petri-Netzes erzeugen. Dafür sollen maximale Teil-T-Invarianten effizient identifiziert und in die Analysen einbezogen werden.

Mit den beschriebenen Methoden können Erkenntnisse über das Petri-Netz-Modell und damit das zugrunde liegende reale System gewonnen werden. Diese Erkenntnisse sind nicht nur für den Modellierungsprozess relevant, sondern auch für die folgende Phase, die Anwendung des fertigen Modells in der Erforschung des modellierten Systems.

1.3 Aufbau der Diplomarbeit

Nach der Einleitung in das Thema dieser Arbeit werden im zweiten Kapitel Petri-Netze und ihre grundlegenden Eigenschaften eingeführt. In Kapitel 3 wird auf die verschiedenen Möglichkeiten eingegangen, die Invarianten eines Petri-Netzes strukturell zu charakterisieren und auszuwerten. Auf eine der vorgestellten Möglichkeiten, Informationen über die Invarianten beziehungsweise das Petri-Netz zu verwerten, wird in Kapitel 4 genauer eingegangen. Dieses Kapitel behandelt ausführlich verschiedene neue Reduktionsmöglichkeiten und welchen Einfluss diese auf das Petri-Netz haben. Anschließend wird anhand einer Fallstudie erläutert, wie die vorgestellten Methoden eingesetzt werden können. Die Zusammenfassung am Schluss der Arbeit bietet einerseits einen Überblick über die gewonnenen Erkenntnisse, andererseits wird auf die noch offen gebliebenen Fragen und zukünftige Arbeiten eingegangen. Den im Rahmen der Diplomarbeit durchgeführten Implementierungen ist der Anhang A gewidmet.

Kapitel 2

Grundlagen Petri-Netze

Petri-Netze wurden 1962 als mathematisches Modell nebenläufiger Systeme von Carl Adam Petri definiert [Pet62]. Sie bieten die Möglichkeit, Systemeigenschaften wie Konkurrenz, Nebenläufigkeit, synchrone Abläufe und kausale Abhängigkeiten zu modellieren.

In diesem Kapitel werden zunächst Petri-Netze und die Begriffe, die für die Bestimmung und Auswertung der Invarianten von Petri-Netzen benötigt werden, eingeführt. Weiterhin werden wichtige Eigenschaften von Petri-Netzen und deren Invarianten definiert, die im weiteren Verlauf Gegenstand der Betrachtung sind. Für weiterführende Einblicke in die Petri-Netz-Theorie sind die Werke [Sta90], [Bau96] und [Rei91] zu empfehlen.

2.1 Definitionen

Ein **Petri-Netz** ist ein gerichteter, endlicher, bipartiter Graph ohne isolierte Knoten. Kanten verlaufen nur zwischen den beiden Knotenklassen, die als **Plätze** und **Transitionen** bezeichnet werden. Plätze werden in dieser Arbeit mit runden Knoten dargestellt, Transitionen mit rechteckigen oder quadratischen Knoten. Die Kanten des Petri-Netzes sind mit natürlichen Zahlen gewichtet. In Darstellungen werden Kanten mit dem Gewicht eins ohne Kantengewicht dargestellt, Kanten mit dem Gewicht null werden weggelassen.

Die Plätze des Petri-Netzes können Bedingungen oder Zustände darstellen, sie sind der passive Teil des Petri-Netzes. Der aktive Teil sind die Transitionen als atomare Einheiten der Zustandsveränderungen. Sie verändern die mit ihnen verbundenen Plätze und damit den Zustand des Petri-Netzes. Ein Platz, von dem eine Kante zu einer Transition führt, wird als **Vorplatz** dieser Transition bezeichnet. Ist die Richtung der Kante umgekehrt, wird die Bezeichnung **Nachplatz** verwendet.

Definition 2.1.1 (Petri-Netz) Ein Petri-Netz ist ein Tupel $N = (P, T, F, m_0)$ mit

- $P = \{p_1, p_2, \dots, p_m\}$ als Menge der Plätze, mit $m \in \mathbb{N} \setminus \{0\}$

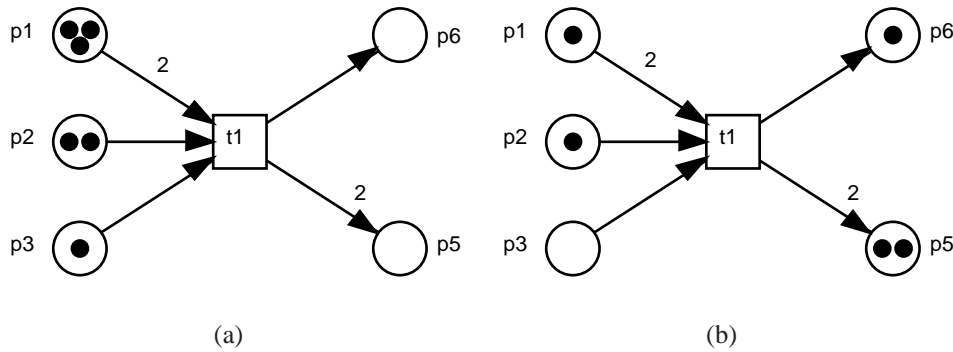


Abbildung 2.1: Schalten einer Transition

- $T = \{t_1, t_2, \dots, t_n\}$ als Menge der Transitionen, mit $n \in \mathbb{N} \setminus \{0\}$
- $P \cap T = \emptyset$
- $F : (T \times P) \cup (P \times T) \rightarrow \mathbb{N}_0$ als Kanten mit Gewichtung
- m_0 als Anfangsmarkierung

Die Menge aller Vorplätze einer Transition t wird mit $\bullet t = \{p \in P \mid F(p, t) \neq 0\}$, die Menge aller Nachplätze einer Transition mit $t \bullet = \{p \in P \mid F(t, p) \neq 0\}$ bezeichnet. Analog dazu ist die Menge aller Vor- und Nachtransitionen eines Platzes definiert.

Zum veränderlichen Teil eines Petri-Netzes gehören die **Marken**. Marken sind nicht unterscheidbare Elemente, die sich ausschließlich auf Plätzen befinden können. Zusammen mit der Anzahl der jeweils vorhandenen Marken stellen alle Plätze den Systemzustand des Netzes dar. Eine **Markierung** des Petri-Netzes N ist eine Funktion $m : P \rightarrow \mathbb{N}$, die den Systemzustand beschreibt und jedem Platz eine Menge von Marken zuordnet. Markierungen können als Vektoren der Länge $|P|$ dargestellt werden.

Eine Änderung des Systemzustandes wird durch das **Schalten** einer Transition herbeigeführt. Eine Transition kann schalten, (auch 'ist **aktiviert**' oder 'hat **Konzession**'), wenn ihre Vorplätze entsprechend der Kantengewichtungen mit Marken belegt sind. Das Schalten einer Transition entfernt Marken von allen Vorplätzen und legt Marken auf allen Nachplätzen ab. Die Anzahl der entfernten und abgelegten Marken hängt vom Kantengewicht ab.

Definition 2.1.2 (Schalten einer Transition) Eine Transition t ist aktiviert beziehungsweise hat Konzession unter einer Markierung m , wenn $F(p, t) \leq m(p) \forall p \in \bullet t$. Eine aktivierte Transition t kann schalten, indem sie die Markierung m in eine (unmittelbare) Folgemarkierung m' mit

$$m'(p) = m(p) - F(p, t) + F(t, p) \quad \forall p \in P$$

überführt (Schreibweise $m \xrightarrow{t} m'$).

Abbildung 2.1 zeigt ein Beispiel für das Schalten einer Transition. t_1 kann bei der Markierung des Netzes aus 2.1(a) schalten. Dabei werden entsprechend der Kantengewichtung Marken von den

Vorplätzen gelöscht und neue Marken auf den Nachplätzen erzeugt. Abbildung 2.1(b) zeigt das Netz von Abbildung 2.1(a), nachdem Transition t_1 geschaltet hat.

Transitionen ohne Vorplätze sind immer aktiviert und können immer schalten. Sie können dem Petri-Netz jederzeit Marken zuführen. Aktivierte Transitionen ohne Nachplätze können Marken ihrer Vorplätze konsumieren, ohne eine Marke neu zu erzeugen. Transitionen ohne Vor- oder Nachplätze nennt man **Randtransitionen**. Plätze, die keine Vor- oder Nachtransitionen besitzen, werden **Randplätze** genannt. Ein Petri-Netz wird **transitionsberandet** genannt, wenn es Randtransitionen, aber keine Randplätze aufweist. Analog dazu existieren in einem platzberandeten Petri-Netz Randplätze, aber keine Randtransitionen.

Markierungen eines Petri-Netzes N können durch das Schalten von Transitionen ineinander überführt werden. Seien m und m' Markierungen, dann wird eine Folge von Transitionen $\tau = t_1, t_2, \dots, t_k$ mit $m = m_0 \xrightarrow{t_1} m_1 \xrightarrow{t_2} m_2 \cdots m_{k-1} \xrightarrow{t_k} m_k = m'$ **Schaltfolge** genannt (Schreibweise $m \xrightarrow{\tau} m'$). Eine Markierung m' heißt **erreichbar** unter einer Markierung m , wenn es eine Schaltfolge τ gibt mit $m \xrightarrow{\tau} m'$.

Viele Analyseverfahren benutzen als Grundlage den **Erreichbarkeitsgraphen** eines Petri-Netzes N . Die Knoten des Erreichbarkeitsgraphen symbolisieren gerade die Menge aller von einer Anfangsmarkierung erreichbaren Markierungen $R_N(m_0)$. Die Kanten sind mit der Transition markiert, deren Schalten die entsprechenden Markierungen ineinander überführt.

Definition 2.1.3 (Erreichbarkeitsgraph) *Es sei $N = (P, T, F, m_0)$ ein Petri-Netz. Als Erreichbarkeitsgraph von N bezeichnet man den Graphen $EG(N) := [R_N(m_0), K_N]$, der die in N erreichbaren Markierungen als Knoten und die Menge K_N von mit Transitionen beschrifteten Kanten hat. Dabei ist*

$$\begin{aligned} R_N(m_0) &= \left\{ m' \mid \exists \text{ Schaltfolge } \tau : m_0 \xrightarrow{\tau} m' \right\} \\ K_N &= \left\{ [m, t, m'] \mid m, m' \in R_N(m_0) \wedge t \in T \wedge m \xrightarrow{t} m' \right\}. \end{aligned}$$

Das Tupel $[m, t, m']$ beschreibt dabei eine Kante von Knoten m nach m' mit der Transition t als Kantenbewertung.

2.2 Eigenschaften

Die Verhaltenseigenschaften des Petri-Netzes hängen im Allgemeinen von der Anfangsmarkierung des Petri-Netzes ab. Die im Folgenden genannten Eigenschaften können für die Analyse des modellierten Systems relevant sein.

Eine der wichtigsten Eigenschaften ist die **Lebendigkeit**. Eine Transition t ist **lebendig**, wenn kein Zustand erreichbar ist, von dem aus t nie wieder schalten kann. Ein Petri-Netz ist lebendig, wenn alle seine Transitionen lebendig sind. Dabei spielt allerdings immer die jeweilige Markierung eine Rolle.

Definition 2.2.1 (Lebendigkeit) *Sei $N = (P, T, F, m_0)$ ein Petri-Netz.*

1. Eine Transition t heißt *tot* bei der Markierung m von P , wenn von m aus keine Markierung erreichbar ist, bei der t aktiviert ist.
2. Eine Transition t von N heißt *lebendig* bei der Markierung m von P , wenn von m aus keine Markierung erreicht werden kann, bei der t tot ist.
3. Eine Markierung m von P wird *lebendig* in N genannt, wenn alle Transitionen $t \in T$ lebendig bei m in N sind.
4. Das Petri-Netz N heißt *lebendig*, wenn seine Anfangsmarkierung m_0 lebendig in N ist.

Im Petri-Netz-Modell mit einer Anfangsmarkierung kann die Frage interessant sein, wieviele Marken sich in allen erreichbaren Markierungen maximal auf einem Platz befinden können. Ist diese Anzahl bei allen Plätzen beschränkt, ist die Menge der erreichbaren Zustände endlich. Bei einem System mit einer endlichen Zustandsmenge ist es damit zum Beispiel zumindest theoretisch möglich, alle Zustände nach bestimmten Kriterien zu prüfen und das Netz auf diesem Weg zu validieren.

Definition 2.2.2 (Beschränktheit) Es sei $N = (P, T, F, m_0)$ ein Petri-Netz, m eine Markierung von P und $p \in P$.

1. p wird *beschränkt* bei m genannt, wenn ein $k \in \mathbb{N}$ existiert, so dass für jede von m in N erreichbare Markierung m' gilt: $m'(p) \leq k$
2. Das Petri-Netz N heißt *beschränkt* bei m , wenn alle seine Plätze beschränkt bei m sind.

2.2.1 Strukturelle Eigenschaften

Neben den Eigenschaften von Petri-Netzen, die von Markierungen abhängig sind, gibt es auch Eigenschaften, die weder von einer Anfangsmarkierung, noch von anderen Markierungen abhängen. Diese Eigenschaften werden *strukturelle Eigenschaften* genannt.

Eine wichtige strukturelle Eigenschaft, die ein Petri-Netz enthalten kann, ist ein **Konflikt**. Ein Konflikt ist eine nichtnebenläufige gleichzeitige Aktiviertheit von zwei Transitionen. Man unterscheidet verschiedene Arten von Konflikten, hier wird vorrangig der sogenannte **statische Konflikt** betrachtet. Ein statischer beziehungsweise struktureller Konflikt liegt vor, wenn es zwei Transitionen t_1 und t_2 mit $\bullet t_1 \cap \bullet t_2 \neq \emptyset$ gibt (Schreibweise: $t_1 \# t_2$). In Abbildung 2.2 stehen zum Beispiel die Transitionen t_1 und t_2 miteinander in Konflikt, da der Platz p_1 ein Vorplatz beider Transitionen ist.

Liegt ein Konflikt vor, kann eine Transition einer anderen Transition die Konzession entziehen. Mit Konflikten ist ein Modellieren nichtdeterministischer Abläufe möglich, da eventuell jede der beteiligten Transitionen schalten kann. Welche Transition schaltet, wenn beide Transitionen Konzession haben, ist zufällig.

Neben der **Konfliktrelation** können zwei Transitionen auch in einer **Abhängigkeitsrelation** oder in einer **Unabhängigkeitsrelation** stehen. Zwei Transitionen t_1 und t_2 stehen in einer Abhängigkeitsrelation (auch kausale Relation genannt, Schreibweise: $t_1 < t_2$), wenn es einen nichtleeren

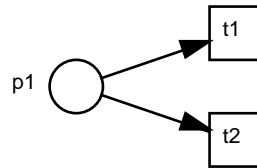


Abbildung 2.2: Einfaches Beispiel für einen Konflikt

Weg von t_1 nach t_2 gibt. Die beiden Transitionen stehen in einer Unabhängigkeitsrelation (Schreibweise $t_1 || t_2$), wenn weder $t_1 < t_2$ noch $t_2 < t_1$ noch $t_1 \# t_2$ gilt.

Eine **Lesekante** oder **Testkante** ist ein Paar von Kanten $\{(t, p), (p, t)\}$ für das gilt: $F(t, p) = F(p, t)$. Das Schalten der Transition t ändert damit die Anzahl der Marken auf dem Platz, mit dem sie durch eine Lesekante verbunden ist, nicht. Für ihr Schalten ist trotzdem eine genügend große Anzahl an Marken auf dem Platz erforderlich.

2.2.2 Netzinvarianten

Petri-Netze kann man vereinfacht auch mit einer sogenannten **Inzidenzmatrix** darstellen. Die Definition des Petri-Netzes liefert eine feste Ordnung über den Transitionen und Plätzen, die für die Inzidenzmatrixdarstellung benötigt wird. Diese statische Darstellung spiegelt allerdings nur Teile der Struktur des Petri-Netzes wieder. Nicht enthalten in der Darstellung als Inzidenzmatrix sind zum Beispiel die Markierungen des Petri-Netzes.

Definition 2.2.3 (Inzidenzmatrix) Die Matrix C zum Petri-Netz $N = (P, T, F, m_0)$ mit

- $C = (c_{i,j})_{1 \leq i \leq |P|, 1 \leq j \leq |T|}$ mit $c_{i,j} = \Delta t_j(p_i)$
- $\Delta t_j(p_i) := F(t_j, p_i) - F(p_i, t_j)$

wird *Inzidenzmatrix* von N genannt.

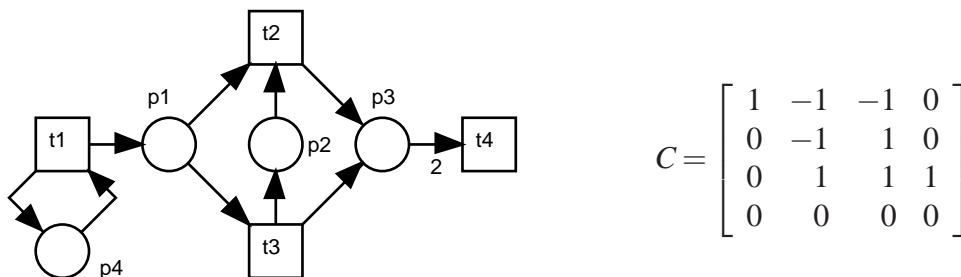


Abbildung 2.3: Beispielnetz mit Inzidenzmatrix

In Abbildung 2.3 werden ein Petri-Netz und seine Inzidenzmatrix dargestellt. Eine wesentliche Beobachtung wird an diesem Beispiel deutlich: Die beiden Kanten zwischen t_1 und p_4 erscheinen nicht in der Inzidenzmatrix, da sich die Auswirkungen des Schaltens der Transition t_1 auf den Platz p_4 gerade aufheben. Dieser Sachverhalt wird im späteren Verlauf dieser Arbeit noch Bedeutung erlangen.

Ergänzend zur Inzidenzmatrix werden nun noch sogenannte Platz- und Transitionsvektoren eingeführt. Jeder positive ganzzahlige Vektor von $|T|$ Elementen wird Transitionsvektor, jeder positive ganzzahlige Vektor von $|P|$ Elementen wird Platzvektor genannt, falls ihre Elemente durch T beziehungsweise P geordnet sind. Einen Transitionsvektor nennt man auch Parikhvektor. Insbesondere hat jede Schaltfolge τ einen zugehörigen Parikhvektor v mit

$$v[t] = \#(t, \tau) \forall t \in T.$$

Die Trägermenge $\text{supp}(v)$ eines Transitionsvektors (eines Platzvektors) v , häufig auch kurz Träger genannt, ist die Menge der Transitionen (Plätze) t mit $v[t] \neq 0$, d.h. deren zugehörige Komponente im Vektor nicht null ist.

Die Inzidenzmatrix ermöglicht es, Invarianten des Petri-Netzes zu bestimmen. Invarianten sind Systemeigenschaften, die während der Arbeit des Systems unabhängig vom konkreten Ablauf erhalten bleiben. Hier werden Lösungen bestimmter homogener Gleichungssysteme als Invarianten bezeichnet. Dabei unterscheidet man zwischen zwei Typen von Invarianten, die Platzinvarianten und die Transitionsinvarianten.

Definition 2.2.4 (Platzinvariante) Sei N ein Petri-Netz und C seine Inzidenzmatrix. Eine positive Platzinvariante (kurz P-Invariante) ist ein nichttrivialer ganzzahliger Platzvektor y des Gleichungssystems

$$y \cdot C = 0, y \geq 0, y \in \mathbb{N}_0^{|P|}.$$

Bei einer P-Invariante y ist die mit y gewichtete Anzahl der Marken auf den Plätzen in $\text{supp}(y)$ invariant. Da die gewichtete Summe der Marken konstant ist, sind die Plätze der Trägermenge einer P-Invariante immer beschränkt. Eine P-Invariante des Petri-Netzes von Abbildung 2.3 ist $(0, 0, 0, 1)$.

Definition 2.2.5 (Transitionsinvariante) Sei $N = (P, T, F, m_0)$ ein Petri-Netz und C seine Inzidenzmatrix. Eine positive Transitionsinvariante (kurz T-Invariante) ist ein nichttrivialer ganzzahliger Transitionsvektor x des Gleichungssystems

$$C \cdot x = 0, x \geq 0, x \in \mathbb{N}_0^{|T|}.$$

Eine Transitionsinvariante i heißt realisierbar in N unter m_0 , wenn es eine Schaltfolge τ mit dem zugehörigen Parikhvektor i und eine in N erreichbare Markierung m gibt mit $m \xrightarrow{\tau} m$.

Wenn eine Schaltfolge abläuft, deren Parikhvektor einer T-Invariante entspricht, dann ist der Zustand des Petri-Netzes, also dessen Markierung, invariant. Jeder Kreis im Erreichbarkeitsgraphen stellt somit eine realisierbare T-Invariante dar. Es kann allerdings verschiedene Reihenfolgen geben, mit der die Transitionen der Schaltfolge zu der T-Invariante schalten. Deshalb umfasst eine realisierbare T-Invariante eine Menge von Kreisen im Erreichbarkeitsgraphen. Eine T-Invariante des Netzes aus Abbildung 2.3 ist $(2, 1, 1, 1)$.

Zwischen P- und T-Invarianten besteht außerdem ein dualer Zusammenhang. Als **duales Petri-Netz** wird das Netz N^d bezeichnet, das entsteht, wenn Transitionen und Plätze aus N die Rollen

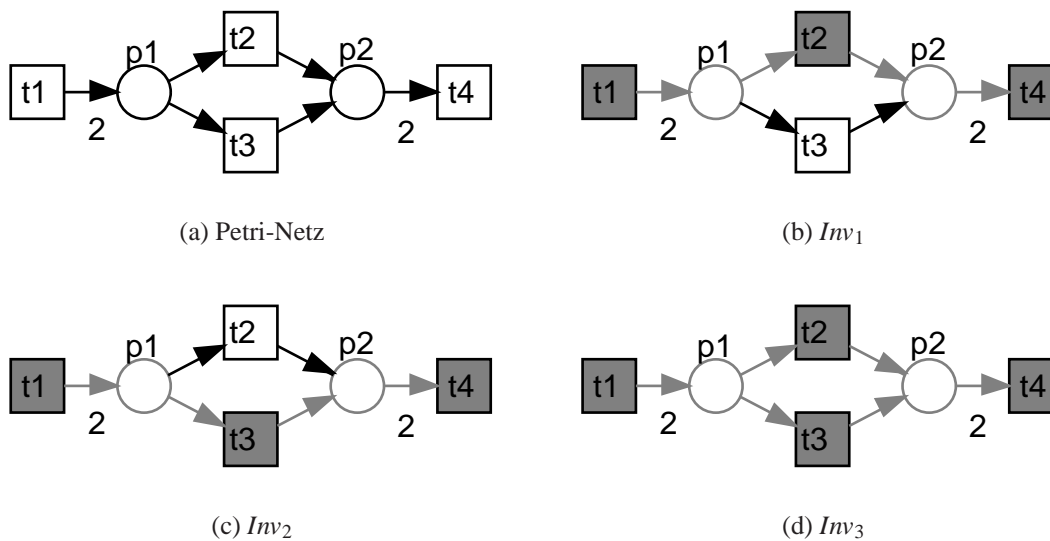


Abbildung 2.4: Beispiel für minimale und einfache T-Invarianten

Transition	Inv_1	Inv_2	Inv_3
t1	1	1	1
t2	2	0	1
t3	0	2	1
t4	1	1	1

Tabelle 2.1: Minimale und einfache T-Invarianten

tauschen und die Kanten gerade umgekehrt verlaufen [Mül04, S. 52]. Dem entspricht als Matrixoperation das Transponieren der Inzidenzmatrix C des Petri-Netzes N . Damit ist eine T-Invariante des Petri-Netzes N gerade die Transponierte einer P-Invariante von N^d .

Im Verlauf dieser Arbeit sind besonders die minimalen T-Invarianten beziehungsweise P-Invarianten interessant. Eine P-Invariante (T-Invariante) v heißt **minimal**, gdw. es keine andere P-Invariante (T-Invariante) w gibt mit $\text{supp}(w) \subset \text{supp}(v)$ und wenn der größte gemeinsame Teiler aller Komponenten von v eins ist.

In Abbildung 2.4 zeigt ein Petri-Netz 2.4(a), dessen minimale T-Invarianten Inv_1 und Inv_2 aus Tabelle 2.1 in 2.4(b) und 2.4(c) hervorgehoben sind.

Eine P-Invariante (T-Invariante) x heißt **einfach**, wenn sie sich nicht als Summe zweier (nicht notwendig verschiedener) P-Invarianten (T-Invarianten) darstellen lässt, d. h. $\nexists v, w : x = v + w$. Jede minimale P-Invariante (T-Invariante) ist somit auch eine einfache P-Invariante (T-Invariante).

Das Petri-Netz aus Abbildung 2.4(a) hat neben den beiden minimalen T-Invarianten noch eine weitere einfache T-Invariante, Inv_3 aus Tabelle 2.1. In 2.4(d) sind alle an dieser T-Invariante beteiligten Transitionen hervorgehoben.

Die Menge aller T- bzw. P-Invarianten des Petri-Netzes N ist abgeschlossen gegenüber der Bildung von Linearkombinationen der T- bzw. P-Invarianten mit ganzzahligen Koeffizienten und der Division durch den größten gemeinsamen Teiler. Die minimalen T- bzw. P-Invarianten bilden ein \mathbb{Q} -Erzeugendensystem, mit dem alle T- bzw. P-Invarianten dargestellt werden können:

$$kx = \sum_{i=1}^n a_i \cdot x_i \quad \begin{array}{l} a_i \in \mathbb{N}, x_i \text{ ist minimale P- bzw. T-Invariante} \\ n = \text{Anzahl der minimalen P- bzw. T-Invarianten} \\ k = \text{größter gemeinsamer Teiler} \end{array}$$

Die ganzzahlige Division ist nicht notwendig, wenn man die einfachen T- bzw. P-Invarianten zur Verfügung hat. Diese können nicht mehr ganzzahlig zerlegt werden, so dass bei der Zerlegung eine kleinere Invariante entsteht.

Die Menge aller T- bzw. P-Invarianten des Petri-Netzes N ist abgeschlossen gegenüber der Bildung von Linearkombinationen der einfachen T- bzw. P-Invarianten. Für das \mathbb{N} -Erzeugendensystem der einfachen T- bzw. P-Invarianten gibt es eine geschlossene Darstellung (aus [Run04, S. 26]):

$$x = \sum_{i=1}^n a_i \cdot x_i \quad \begin{array}{l} a_i \in \mathbb{N}, x_i \text{ ist einfache P- bzw. T-Invariante} \\ n = \text{Anzahl der einfachen P- bzw. T-Invarianten} \end{array}$$

Ein Verfahren für die Berechnung der minimalen T- und P-Invarianten ist in [Sta90] angegeben. Es kann im Bezug zur Knotenanzahl des Netzes exponentiell viele T- oder P-Invarianten geben, damit ist der Aufwand zur Berechnung im schlimmsten Fall ebenfalls exponentiell.

Induzierte Teilnetze

Die Trägermenge $\text{supp}(v)$ zu einem Transitionsvektor v und damit auch zu jeder T-Invariante induziert ein Teilnetz. Dieses Teilnetz besteht aus den Transitionen der Trägermenge, deren Vor- und Nachplätzen und allen Kanten zwischen diesen Knoten. Als Beispiel dienen die Abbildungen 2.4(b) bis 2.4(d). Zu der jeweiligen T-Invariante ist jeweils das gesamte induzierte Teilnetz grau markiert.

Das **induzierte Teilnetz** einer minimalen T-Invariante i ist zusammenhängend, aber nicht zwingend stark zusammenhängend. Wäre es nicht zusammenhängend, gäbe es mindestens zwei getrennte Teilnetze N' und N'' . Der Teil der T-Invariante, der das Schalten im Teilnetz N' komplett enthält, reproduziert unabhängig von den anderen Teilen der T-Invariante die Markierung in diesem Teilnetz. Dieser Teil ist damit eine neue T-Invariante, er hat aber eine Trägermenge, die Teilmenge der Trägermenge der T-Invariante i ist. Damit wäre i nicht minimal gewesen.

Diese Aussagen sind direkt auf P-Invarianten übertragbar, da diese im dualen Petri-Netz T-Invarianten darstellen.

2.3 Konventionen

Da die graphische Darstellung eines Petri-Netzes schon bei geringer Knotenanzahl durch viele Kantenüberschneidungen unübersichtlich werden kann, wurden sogenannte **logische Knoten** als Repräsentanten eines Knotens eingeführt. Alle zu einem Knoten k des Netzes gehörenden logischen Knoten stehen symbolisch für diesen Knoten, sie besitzen den gleichen Namen und werden meist andersfarbig dargestellt. Das bedeutet, dass jede Kante zu einem der logischen Knoten für eine Kante zu k im Petri-Netz steht. Alle logischen Repräsentanten zu einem Platz p haben jederzeit die gleiche Markierung.

Befindet sich eine Testkante unter den Kanten, das heißt es existieren zwei Kanten $((p, t), k)$ und $((t, p), k) \in F$ zwischen den beiden Knoten p und t , so wird diese Kante als Linie mit zwei Pfeilenden dargestellt.

Für die graphische Darstellung ist eine hierarchische Gliederung des Petri-Netzes sinnvoll. Sie ermöglicht durch Verfeinerung oder Vergrößerung auf verschiedenen Hierarchieebenen eine visuelle Abstraktion. Dazu wurden **hierarchische Knoten** eingeführt. Sie ersetzen ein entweder transitions- oder platzberandetes Teilnetz, das sich als Subnetz in der nächsten Hierarchiestufe befindet. Hierarchische Knoten ändern das Petri-Netz nicht semantisch, sondern nur optisch. Sie werden in dieser Arbeit als doppelt umrandete Quadrate dargestellt.

Neben der Hierarchisierung besteht auch die Möglichkeit, Petri-Netze nicht nur optisch sondern auch semantisch zu verfeinern oder zu vergrößern [Bau96]. Bei dieser lokalen Abstraktion wird ein transitions- oder platzberandetes Teilnetz durch eine Transition oder einen Platz ersetzt.

Kapitel 3

Ansätze zur Auswertung von Invarianten

Die Invarianten eines Petri-Netzes bieten die Möglichkeit, das Petri-Netz und damit das modellierte System besser zu verstehen und Fehler und Inkonsistenzen des Modells zu finden und zu beheben. Dabei spielt die endliche Menge der minimalen Invarianten eine besondere Rolle, da ihre Berechnung unabhängig vom Erreichbarkeitsgraphen erfolgt und nicht von der Markierung des Petri-Netzes abhängt. Die Berechnung des vollständigen Erreichbarkeitsgraphen ist bei unbeschränkten Netzen nicht möglich, da der Zustandsraum unendlich groß ist. Die Berechnung der Menge der minimalen Invarianten ist dagegen immer endlich, auch wenn der Aufwand dafür exponentiell sein kann. Deshalb wird die Analyse der minimalen Invarianten als strukturelles Instrument häufig zur Verifikation von Petri-Netz-Modellen eingesetzt. Auch die einfachen Invarianten eignen sich für die Analyse. Allerdings wird zur Zeit noch nach effizienten Verfahren gesucht, um die Menge der einfachen Invarianten zu bestimmen [DHJ02].

Minimale und einfache Invarianten sind je nach Anwendung verschieden interpretierbar. Ein ausführliches Beispiel für ihre Bedeutung bei biochemischen Netzwerken wird in Kapitel 5 vorgestellt.

Es ist auch möglich, beliebige nach bestimmten Kriterien ausgewählte Mengen von Invarianten zu bilden. Ein mögliches Kriterium könnte die Schaltanzahl einer bestimmten Transition sein. Es ist darauf zu achten, dass diese Mengen eine endliche, mit vertretbarem Aufwand berechenbare und damit analysierbare Basis besitzen. Die Anzahl der so ausgewählten Invarianten, beziehungsweise deren Basis, kann ebenfalls sehr groß sein.

Die Auswertung großer Mengen von Invarianten kann nicht mehr manuell erfolgen. Durch den möglichen exponentiellen Anstieg der Anzahl der minimalen Invarianten kann es aber schon bei relativ kleinen Netzen zu einer großen Anzahl von Invarianten kommen. Deren Gemeinsamkeiten und Unterschiede, ihre Struktur und eventuelle Abhängigkeiten innerhalb der Komponenten der Invarianten sind interessant für das Verständnis des Petri-Netzes. In diesem Kapitel wird beschrieben, welche Strategien und Verfahren sich als nützlich erweisen können, solche Mengen von Invarianten zu beschreiben und auszuwerten.

Aufgrund der Tatsache, dass sich Transitionen und Plätze dual zueinander verhalten, werden im nächsten Abschnitt und im nächsten Kapitel ausschließlich Transitionsinvarianten analysiert. Die Ergebnisse sind auf Platzinvarianten übertragbar. Im Folgenden werden Transitionsinvarianten kurz T-Invarianten genannt.

3.1 Auswertung mit Data Mining

Eine Menge von Invarianten ist, technisch gesehen, eine Menge von Vektoren. Bei der Auswertung einer Menge von Invarianten kommen deshalb prinzipiell alle Methoden in Betracht, die mit (Daten)-Feldern beliebiger Art umgehen. Fasst man eine Transition als Attribut, eine Schaltzahl als Wert dieses Attributes und eine T-Invariante als Datensatz über diesen Attributen auf, ist die Anwendung von Methoden aus dem Bereich Data Mining naheliegend.

Es gibt zahlreiche Methoden und Verfahren, die aus einer Menge von Datensätzen Informationen gewinnen. Um die Eignung einer Methode zu prüfen, ist zunächst der Nutzen, der vom Ergebnis der Anwendung der Methode gezogen werden kann, von Bedeutung. Deshalb werden im Folgenden einige der in [IH00] und [Web03] genannten Ergebnisarten geprüft. Folgende Ergebnisarten kommen dabei in Frage:

1. Entscheidungstabellen und Entscheidungsbäume
2. Klassifikationsregeln und Assoziationsregeln
3. Cluster

Entscheidungsbäume und Entscheidungstabellen sowie Klassifikationsregeln eignen sich nur bedingt für die Auswertung von Invariantenmengen. Verfahren, die Ergebnisse dieser Art erzeugen, nehmen eine Menge klassifizierter Beispiele entgegen und verarbeiten diese. Mit Hilfe der genannten Ergebnisarten ist es dann möglich, neue Beispiele zu klassifizieren oder Vorhersagen für bestimmte Attribute zu treffen.

Da bei Invarianten im Allgemeinen keine vorher bekannte Klassifikation existiert, sondern eher eine solche Klassifikation gefunden werden soll, sind diese Ergebnisarten weniger interessant. Anhand der Invarianten sollen auch keine Vorhersagen für neue Invarianten getroffen werden, denn die zu analysierende Invariantenmenge ist bereits vollständig.

Auch Assoziationsregeln dienen der Vorhersage, allerdings können damit auch Attribute oder Attributmengen anhand anderer Attributmengen vorhergesagt werden. Die Strukturen und Gemeinsamkeiten, die durch das Finden von relevanten Assoziationsregeln gefunden werden, sind deshalb durchaus interessant.

Eine Assoziationsregel besteht aus einer Menge von Bedingungen, die an die Werte einer Teilmenge der Attribute gestellt werden, und einer Folgerung, die mit einer gewissen Wahrscheinlichkeit eintritt. Die Folgerung kann dabei ein Wert eines weiteren Attributes oder ebenfalls eine Menge von Attributen mit bestimmten Werten sein.

Transition	Inv_1	Inv_2
t0	2	3
t1	0	1
t2	1	2
t3	0	1
t4	1	1
t5	1	1
t6	1	0

Tabelle 3.1: Eine Menge von T-Invarianten

Bei der Ermittlung von relevanten Assoziationsregeln, also Regeln, die für die meisten oder alle Invarianten zutreffen, oder Regeln, die möglichst viele Attribute erfassen, werden Attributmengen gebildet. Diese Attributmengen sind dahingehend interessant, dass sie eventuell häufig vorkommende Teil-Invarianten beschreiben. Die Assoziationsregeln können auf direkte oder indirekte Abhängigkeiten zwischen Attributen hinweisen. Ein Teil dieser Assoziationsregeln hat sich innerhalb der Invariantenauswertung als nützlich für die weitere Betrachtung erwiesen. Darauf wird im folgenden Abschnitt genauer eingegangen.

Die letzte genannte Art der Data-Mining-Ergebnisse ist die Clusterung. Eine Clusterung ist eine Zerlegung der Datensätze in disjunkte Mengen, sogenannte Cluster. Mit Hilfe einer Clusterung lassen sich relevante Teilmengen finden und clusterbasierte Aussagen treffen. Einem Cluster werden dabei zum Beispiel alle Datensätze zugeordnet, die eine große Ähnlichkeit aufweisen. Es gibt verschiedene Verfahren, die die Ähnlichkeit zweier Invarianten bestimmen. Es gibt weiterhin eine Reihe von Verfahren, die auf Basis dieser Ähnlichkeiten eine Clusterung vornehmen. Da diese Clusterungs-Verfahren und geeignete Ähnlichkeitsmaße bereits in einer parallel zu dieser Diplomarbeit laufenden Masterarbeit untersucht wurden [GB06], wird hier nicht weiter darauf eingegangen.

3.2 Abhängige Transitionsmengen

Im Kontext der Analyse von T-Invarianten besteht eine Assoziationsregel $s \Rightarrow u$ aus einer Prämisse s und einer Konklusion u der Regel, die jeweils aus einer Multimenge von Transitionen bestehen. Basis ist eine Menge von T-Invarianten I . Es ist auch möglich, eine solche Regel trägerbasiert zu formulieren, also als Menge. In diesem Fall wird hinter den Elementen nicht deren Häufigkeit festgehalten.

Einige Regeln zu den T-Invarianten aus Tabelle 3.1 werden in Tabelle 3.2 angegeben.

Zu einer Assoziationsregel $s \Rightarrow u$ lässt sich die Häufigkeit des Auftretens (Support) und die Stärke der Korrelation (Confidence) bestimmen [Ada01]. Zu beachten ist, dass der Begriff „Support“ in diesem Zusammenhang nichts mit dem Begriff „Träger“ der T-Invarianten zu tun hat. Der Support (Formel 3.1) gibt an, in wievielen Invarianten die Regel im Vergleich zu allen Invarianten gilt.

Nr.	Regel	Support	Confidence
1	$\{t4(1)\} \Rightarrow \{t5(1)\}$	1	1
2	$\{t0(2)\} \Rightarrow \{t2(1)\}$	0.5	0.5
3	$\{t0\} \Rightarrow \{t2\}$	1	1
4	$\{t1\} \Rightarrow \{t3\}$	0.5	1
5	$\{t5\} \Rightarrow \{t6\}$	0.5	0.5

Tabelle 3.2: Beispiele für Assoziationsregeln zu Tabelle 3.1

Die Confidence (Formel 3.2) gibt den Anteil der betrachteten Invarianten an, in denen $s \Rightarrow u$ bei Erfüllung der Prämisse s gilt.

$$\text{support}(s \Rightarrow u) = \frac{|\{i \in I \mid (s \cup u) \dot{\subseteq} i\}|}{|I|} \quad (3.1)$$

$$\text{confidence}(s \Rightarrow u) = \frac{\text{support}(s \Rightarrow u)}{\text{support}(s)} = \frac{|\{i \in I \mid (s \cup u) \dot{\subseteq} i\}|}{|\{i \in I \mid s \dot{\subseteq} i\}|} \quad (3.2)$$

Eine Regel $s \Rightarrow u$ gilt für eine T-Invariante, wenn in ihr sowohl die in s als auch die in u aufgezählten Transitionen mit genau der in s und u angegebenen Häufigkeit schalten (Schreibweise $\dot{\subseteq}$). Diese Definition ist verschieden von der Standarddefinition aus [Ada01], da hier mit Multimengen und nicht mit Attributen und diskretisierten Attributwerten gearbeitet wird.

Für die in Tabelle 3.2 angegebenen Regeln sind in den letzten beiden Spalten der Support und die Confidence angegeben.

Bei einer Menge von T-Invarianten weisen Assoziationsregeln $s \Rightarrow u$, die mit einer Confidence von 1 auftreten und bei denen nur eine Transition mit eventuell unterschiedlichen Häufigkeiten in der Prämisse s auftritt, auf abhängige Transitionsmengen hin. Abhängige Transitionen kommen entweder gemeinsam oder gar nicht in den T-Invarianten vor. Beispielsweise folgt aus der ersten Regel aus Tabelle 3.2, dass in allen T-Invarianten $t4$ immer zusammen mit $t5$ auftritt. Aus den Regeln 2 und 3 folgt, dass die Transitionen $t0$ und $t2$ zwar immer gemeinsam vorkommen, aber nicht immer mit der gleichen Häufigkeit.

Außerdem können einseitige Abhängigkeiten vorhanden sein, das heißt, dass eine Transition immer in Verbindung mit einer zweiten auftritt, aber diese zweite Transition auch ohne die erste in einer T-Invariante vorkommen kann. Diese letztere Form der Abhängigkeit weist auf das Vorhandensein von Alternativen im Netz hin. Im Rahmen dieser Arbeit sind aber vor allem die gegenseitig voneinander abhängigen Transitionen interessant.

Die Idee gegenseitig voneinander abhängiger Transitionsmengen im Kontext von Signaltransduktionswegen wurde unter dem Begriff T-Gruppe bereits in [Sac05] und in [BNSM04] als „Coupled Reaction Set“ eingeführt. Dieser Ansatz wird im Folgenden aufgegriffen und erweitert, da die genauere Untersuchung abhängiger Transitionsmengen aussichtsreich im Bezug zur Auswertung der T-Invarianten erscheint.

Es ist anzunehmen, dass auch andere Assoziationsregeln, etwa solche mit einer besonders hohen Abdeckung und eventuell geringerer Genauigkeit relevante Ergebnisse liefern können. Allerdings sprengt ihre eingehende Untersuchung den Rahmen dieser Arbeit. Assoziationsregeln werden im weiteren Verlauf der Arbeit keine besondere Rolle mehr spielen, obwohl sie einen indirekten Weg darstellen, abhängige Transitionsmengen zu berechnen.

3.2.1 Definitionen

Eine abhängige Transitionsmenge ist eine Menge von Transitionen, die immer gemeinsam in einer Menge von T-Invarianten vorkommen. Dabei kann man von den konkreten Häufigkeiten abstrahieren und nur den Support der T-Invariante beachten. Dies kann unter Umständen nützlich sein, wenn nur an das reine Vorkommen der Transition interessant ist und nicht unterschieden wird, wie oft eine Transition schalten muss.

Definition 3.2.1 (abstract DTS) Sei I eine Menge von T-Invarianten eines Petri-Netzes $N = (P, T, F, m_0)$. Eine Menge von Transitionen $A \subseteq T$ wird abstrahiert abhängige Transitionsmenge bezüglich I (abstractly dependent transition set, kurz abstract DTS) genannt, wenn

$$\forall i \in I: A \cap \text{supp}(i) = A \vee A \cap \text{supp}(i) = \emptyset.$$

Eine abstract DTS A wird maximal genannt, wenn man keine weitere Transition t aus T zu A hinzufügen kann, so dass A zusammen mit t eine abstract DTS darstellt. Eine abstract DTS heißt echt, wenn ihre Transitionen in mindestens einer T-Invariante aus I vorkommen.

Zu beachten ist die Tatsache, dass die Menge von T-Invarianten, auf deren Basis die abstract DTS berechnet werden, nicht unbedingt die Menge der minimalen oder einfachen T-Invarianten sein muss. Es ist zum Beispiel sinnvoll, bei den minimalen T-Invarianten triviale Invarianten auszuschließen. Der Träger von trivialen T-Invarianten enthält nur zwei Elemente. Da solche T-Invarianten im Normalfall uninteressant sind, sollten sie nicht in die Berechnung einbezogen werden, denn sie können das Ergebnis der Berechnung beeinflussen.

Im Gegensatz zu [Sac05] müssen abstract DTS laut Definition 3.2.1 nicht maximal sein. Für viele Anwendungen mögen zwar eher maximale DTS interessant sein, aber wie später noch erläutert wird, ist auch die Betrachtung von nicht maximalen DTS interessant. Alle maximalen abstract DTS bilden auf der Menge der Transitionen eine Zerlegung in disjunkte Teilmengen.

Das Petri-Netz aus Abbildung 3.1 hat zwei T-Invarianten, aufgelistet in der nebenstehenden Tabelle. Es hat drei maximale abstract DTS, aufgelistet in Tabelle 3.3.

$$\begin{aligned} \text{DTS1} &= \{t_0, t_2, t_4, t_5\} \\ \text{DTS2} &= \{t_1, t_3\} \\ \text{DTS3} &= \{t_6\} \end{aligned}$$

Tabelle 3.3: Abstract DTS des Netzes aus Abbildung 3.1

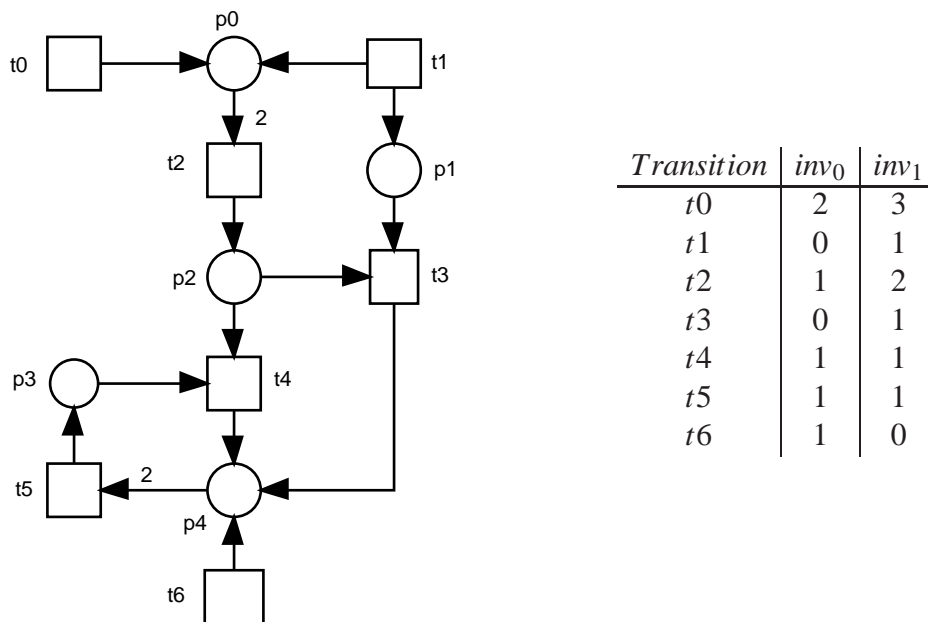


Abbildung 3.1: Beispielnetz für abstract DTS

Unter den abstract DTS auf Basis einer T-Invariantenmenge I gibt es unter Umständen eine besondere DTS, die einzige nicht echte abstract DTS. Sie umfasst alle Transitionen, die in keiner der T-Invarianten aus I vorkommen. Die Erkenntnis, dass diese Transitionen in keiner T-Invariante vorkommen, ist durchaus interessant, beispielsweise wenn I die Menge aller nichttrivialen T-Invarianten darstellt. Im biochemischen Kontext etwa kann es bedeuten, dass für die einer Transition entsprechenden Reaktion die Ausgangsstoffe fehlen, beziehungsweise nicht in ausreichendem Maße produziert werden können. Da die Transitionen einer DTS sonst nicht die im Folgenden verwendeten Gemeinsamkeiten aufweisen, wird diese DTS häufig von den beschriebenen Methoden ausgeschlossen.

Durch das Abstrahieren von den konkreten Häufigkeiten der Transitionen in den T-Invarianten kann es vorkommen, dass Transitionen vom Standpunkt der Häufigkeiten her unabhängig voneinander agieren können. In einer T-Invariante kann zum Beispiel Transition t_1 zweimal schalten und Transition t_2 dreimal, in einer anderen T-Invariante gerade umgekehrt.

In dem Petri-Netz aus Abbildung 3.1 trifft dieser Sachverhalt auf die Transitionen t_0 und t_2 zu. Sie treten zwar in allen minimalen T-Invarianten gemeinsam auf, deshalb befinden sie sich in einer abstract DTS. Ihre Häufigkeiten in den beiden T-Invarianten sind jedoch unabhängig voneinander, das heißt diese beiden Transitionen schalten nicht immer im gleichen Verhältnis.

Man kann deshalb von einer Menge von abhängigen Transitionen zusätzlich verlangen, dass auch die Häufigkeiten in den T-Invarianten eine Abhängigkeit widerspiegeln. Das ermöglicht Aussagen der Art „Wenn Transition t_1 in einer T-Invariante dreimal schaltet, schaltet Transition t_2 immer sechsmal“. Sollen Aussagen dieser Art getroffen werden, genügen die Anforderungen der abstract DTS an eine Transitionsmenge nicht.

DTS1	=	{t0}
DTS2	=	{t2}
DTS3	=	{t4, t5}
DTS4	=	{t1, t3}
DTS5	=	{t6}

Tabelle 3.4: Strong DTS des Netzes aus Abbildung 3.1

Definition 3.2.2 (strong DTS) Sei I eine Menge von T -Invarianten eines Petri-Netzes $N = (P, T, F, m_0)$. Eine stark abhängige Transitionsmenge bezüglich I (strongly dependent transition set, kurz strong DTS) ist eine Multimenge S aus Transitionen mit dem zugehörigen Transitionsvektor $v \in \mathbb{N}_0^T$, für den gilt:

$$\forall i \in I \exists k \in \mathbb{N} \forall t \in S : i[t] = k \cdot v[t]$$

Eine strong DTS S nennt man maximal, wenn man keine in S noch nicht vorhandene Transition t aus T in beliebiger Anzahl zu S hinzufügen kann, so dass S zusammen mit t eine strong DTS darstellt. Eine strong DTS S heißt echt, wenn die Transitionen aus S in mindestens einer T -Invariante von N vorkommen.

Diese zweite Definition der DTS stellt schärfere Anforderungen an die Transitionen beziehungsweise deren Häufigkeiten in den T -Invarianten. Der Träger einer strong DTS ist immer auch eine abstract DTS, aber nicht für jede abstract DTS A kann ein Transitionsvektor v gefunden werden, so dass A den Träger einer strong DTS darstellt.

Die durch die maximalen abstract DTS gebildete Zerlegung der Transitionsmenge kann durch die maximalen strong DTS verfeinert werden.

Berechnet man zu den beiden minimalen T -Invarianten aus Abbildung 3.1 die maximalen strong DTS, erhält man die Multimengen aus Tabelle 3.4. Hier liegt eine Verfeinerung der Zerlegung der Transitionsmenge durch die strong DTS gegenüber der Zerlegung durch die abstract DTS vor.

Für die Berechnung aller einzelnen Häufigkeiten der Transitionen aus S in einer Invariante genügt die Kenntnis des Reduktionsvektors v sowie der Zahl k . Dank der Informationen einer strong DTS sind „Wenn, ... dann“-Aussagen verbindlich möglich. Das ermöglicht eine Zusammenfassung der beteiligten Transitionen ohne Informationsverlust bei der Menge der Invarianten. Dieser Ansatz wird im weiteren Verlauf verwendet.

3.2.2 Berechnung der DTS

Für die Bestimmung der maximalen abstract DTS eines Petri-Netzes $N = (P, T, F, m_0)$ mit der T -Invariantenmenge I , gegeben als Menge von Parikh-Vektoren, bietet sich Algorithmus 3.1 an. Er partitioniert die Menge der Transitionen in maximale disjunkte nichtleere Teilmengen. Jede dieser Teilmengen erfüllt die Anforderungen an eine abstract DTS.

Algorithmus 3.1 Algorithmus zur Berechnung der abstract DTS

Require: I := Menge der T-Invarianten

```

1:  $R := T$ ;
2:  $D$  := Container für abstract DTSs;
3: for all  $t$  in  $R$  do
4:    $R := R \setminus \{t\}$ ;
5:    $d$  := neue abstract DTS;
6:    $d := d \cup t$ ;
7:    $D := D \cup d$ ;
8:   for all  $s$  in  $R$  do
9:     for all  $i$  in  $I$  do
10:      if ( $i[t] = 0$  und  $i[s] = 0$ ) oder ( $i[t]! = 0$  und  $i[s]! = 0$ ) then
11:        continue 9;
12:      else
13:        continue 8;
14:      end if
15:    end for
16:     $d := d \cup s$ ;
17:     $R := R \setminus \{s\}$ ;
18:  end for
19: end for

```

Der Aufwand für den Algorithmus 3.1 zum Berechnen der DTS ist in $O(|T|^2 \cdot |I|)$. Diese Abschätzung ergibt sich aus den drei ineinander geschachtelten for-Schleifen in Zeile 3, 8 und 9. Allerdings muss dazu die Datenstruktur für die Invarianten so gewählt sein, dass die Häufigkeit einer Transition in einer Invariante mit einem Aufwand in $O(1)$ bestimmt werden kann.

Wurde die Menge der abstract DTS D von N bestimmt, so bilden diese die Grundlage für die Bestimmung der strong DTS mit Algorithmus 3.2. Jede abstract DTS wird darin gegebenenfalls noch einmal in disjunkte Teilmengen partitioniert. Darin werden jeweils die Transitionen einer abstract DTS zusammengefasst, deren Häufigkeiten immer das gleiche Verhältnis besitzen. Zusammen mit dem jeweiligen Transitionsvektor bildet jede der so erzeugten Teilmengen die Menge der Transitionen einer strong DTS. Der Algorithmus 3.2 kann mit geringfügigen Anpassungen auch zur direkten Berechnung der strong DTS benutzt werden. Der zeitliche Aufwand ist aufgrund der vierten for-Schleife in $O(|T|^3 \cdot |I|)$, liegt aber praktisch darunter, da durch die Einschränkung auf abstract DTS, jeweils in Zeile 3, die zu durchsuchende Knotenmenge in den Zeilen 4 und 8 eingeschränkt werden kann.

Der Transitionsvektor v einer strong DTS $S = (T_S, v)$ wird anhand einer Invariante i bestimmt, in der die Transitionen aus T_S mindestens einmal vorkommen. Dazu wird der größte gemeinsame Teiler ggT der Häufigkeiten bestimmt, mit denen Transitionen aus T_S in i schalten. Darauf folgend werden diese Häufigkeiten durch ggT geteilt und das Ergebnis in v festgehalten. Formaler:

$$v[t] = \begin{cases} 0 & \text{falls } t \notin T_S \\ \frac{i[t]}{ggT} & \text{falls } t \in T_S \end{cases}.$$

Die Bestimmung der DTS ist auch für andere Mengen von T-Invarianten möglich, die von der Menge der minimalen T-Invarianten verschieden sind. Für Reduktionen, die nicht auf den DTS aller minimalen T-Invarianten beruhen, gelten jedoch im Allgemeinen nicht die im Kapitel 4 nachgewiesenen Eigenschaften der Reduktion. In solchen Fällen muss das Ergebnis der Reduktion immer im Zusammenhang mit der zugrunde liegenden T-Invariantenmenge betrachtet werden.

3.2.3 Eigenschaften der abhängigen Transitionsmengen

Nach dem Finden abhängiger Teilmengen der Transitionen sind zunächst verschiedene statische Eigenschaften dieser Mengen interessant. Die Beziehungen der Transitionen untereinander können relevant sein. Das Gesamtverhalten der Transitionen nach außen, das heißt im Bezug zum restlichen Petri-Netz, kann wichtige Erkenntnisse über das Petri-Netz liefern. Die im Folgenden getroffenen Aussagen gelten immer für beide Arten der DTS, es sei denn, die DTS-Art wird explizit erwähnt.

Eine erste Aussage für die echten DTS lässt sich leicht feststellen: Transitionen einer echten DTS befinden sich immer in einem zusammenhängenden Teilnetz. Da die DTS auf den T-Invarianten basierend berechnet werden, und zu jeder T-Invariante ein zusammenhängendes Teilnetz gehört, in dem die Transitionen der DTS entweder alle oder gar nicht enthalten sind, gehören folglich alle Transitionen einer DTS zu einem zusammenhängenden Teilnetz. Die Transitionen einer DTS müssen jedoch nicht direkt über einen Platz miteinander verbunden sein, sie können im gesamten Teilnetz verteilt auftreten. Betrachtet man die von der DTS induzierten Teile des Netzes, müssen diese damit nicht notwendig zusammenhängen.

Interessant ist die Frage, ob bestimmte Netzstrukturen in den Teilnetzen auftreten können, die durch die echte DTS induziert werden. Transitionen, die in einer kausalen Relation stehen, können natürlich in einer echten DTS vorkommen. Zwei Transitionen einer echten DTS können aber auch in einer Konfliktrelation oder in einer Unabhängigkeitsrelation stehen. Abbildung 3.2 zeigt Beispielnetze für diese Fälle. Die beiden Transitionen t_2 und t_3 stehen miteinander in Konflikt, aber die Rückkopplung durch Platz p_1 bewirkt, dass sie sich trotzdem in einer DTS befinden. Transition t_6 und t_7 stehen in einer Unabhängigkeitsrelation, befinden sich aber durch den gemeinsamen Vorgänger t_5 in einer DTS.

Daraus kann man schließen, dass je nach Aufbau des zugrunde liegenden Petri-Netzes auch komplexe Strukturen in den von echten DTS induzierten Teilnetzen vorkommen können.

Es folgt noch eine später nützliche Aussage:

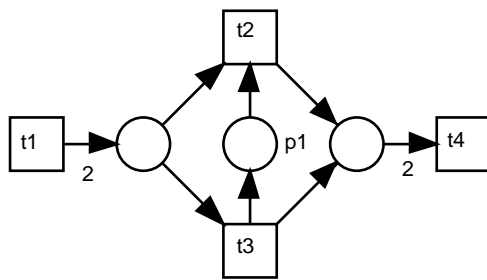
Lemma 3.2.3 *Sei $N = (P, T, F, m_0)$ ein Petri-Netz, S eine echte strong DTS auf Basis aller minimalen T-Invarianten, v der Transitionsvektor von S und sei $p \in P$ ein Platz, dessen Vor- und*

Algorithmus 3.2 Algorithmus zur Berechnung der strong DTS

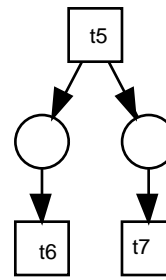
Require: I := Menge der T-Invarianten ;

Require: D := Menge der abstract DTS;

```
1:  $S$  := Container für strong DTSs;
2: for all  $d$  in  $D$  do
3:    $R := d$ ;
4:   for all  $t$  in  $d$  do
5:      $R := R \setminus \{t\}$  ;
6:      $g :=$  neue strong DTS;
7:      $g := g \cup \{t\}$ ;
8:     for all  $s$  in  $R$  do
9:        $relation := 0$ ;
10:      for all  $i$  in  $I$  do
11:        if  $i[t] = 0$  then
12:          continue 10;
13:        end if
14:        if  $relation = 0$  then
15:           $relation := i[t]/i[s]$ ;
16:        else
17:          if  $relation = i[t]/i[s]$  then
18:            continue 10;
19:          else
20:            continue 8;
21:          end if
22:        end if
23:      end for
24:       $g := g \cup \{s\}$  ;
25:       $R := R \setminus \{s\}$  ;
26:    end for
27:     $S := S \cup \{g\}$ ;
28:  end for
29: end for
```



(a) Konflikt in einer DTS



(b) Unabhängige Transitionen in einer DTS

Abbildung 3.2: Beispielnetze für Transitionstypen innerhalb einer DTS

Nachtransitionen ausschließlich in S sind. Dann ist die mit v gewichtete Summe der Kantenbewertungen der Kanten zwischen Transitionen aus S und p null.

$$(\forall a \in \bullet p \cup p\bullet : a \in S) \Rightarrow \sum_{a \in S} v[a](F(a, p) - F(p, a)) = 0.$$

Beweis: Die strong DTS S wurde auf der Basis aller T-Invarianten berechnet. Daraus kann man schließen, dass entweder alle Transitionen aus S in einer T-Invariante vorkommen oder keine einzige. Sei τ eine beliebige Schaltfolge zu einer beliebigen minimalen T-Invariante i . Nachdem τ geschaltet hat, muss die Markierung wieder mit der Markierung des Petri-Netzes vor dem Schalten von τ übereinstimmen, denn anderenfalls wäre I keine T-Invariante. Da p nur Vor- und Nachtransitionen in S hat, verändert sich die Markenanzahl auf p während des Schaltens von τ nur durch Transitionen aus S , und befindet sich am Ende des Schaltvorgangs wieder beim Ausgangswert.

$$\sum_{a \in S} \#(a, \tau)(F(a, p) - F(p, a)) = 0.$$

Die Schalthäufigkeiten von Transitionen aus S in τ sind laut Definition 3.2.2 gerade durch ein k -Vielfaches von v beschrieben:

$$\sum_{a \in S} k \cdot v[a] \cdot (F(a, p) - F(p, a)) = 0.$$

□

Plätze, die nur Transitionen der selben DTS d als Vor- und Nachtransition besitzen, werden **innere Plätze** der DTS d genannt. Alle anderen mit den Transitionen der DTS verbundenen Plätze heißen **Randplätze** von d .

Da die Transitionen einer echten DTS immer gemeinsam in T-Invarianten vorkommen, kann man diese zusammenfassen. Es gibt verschiedene Möglichkeiten, die Transitionen zusammenzufassen und damit das Verständnis des Petri-Netzes zu erleichtern.

Dabei muss beachtet werden, um welche Art der DTS es sich handelt: eine abstract DTS oder eine strong DTS. Nicht zuletzt ist auch das Ziel relevant, das die Zusammenfassung der Transitionen erreichen soll. Ein mögliches Ziel ist das Gewinnen eines Überblicks über die Menge der T-Invarianten, also eine Vereinfachung oder kompaktere Darstellung der T-Invarianten. Zielstellung kann es aber auch sein, mit Hilfe der T-Invarianten einen Überblick über das zugrunde liegende Petri-Netz zu erhalten.

3.2.4 Anwendungen der abhängigen Transitionsmengen

In den folgenden Abschnitten werden verschiedene Varianten vorgestellt, die Informationen über die DTS sinnvoll einzusetzen.

Teilmengen zur Darstellung der T-Invarianten

Spiele die Häufigkeiten bei der Charakterisierung der T-Invarianten eine untergeordnete Rolle, bietet sich eine vereinfachte Darstellung der T-Invarianten an, die nur den Support einbezieht. Unter dieser Betrachtungsweise bilden alle minimalen oder alle einfachen T-Invarianten voneinander verschiedene Teilmengen der Transitionsmenge. Diese Teilmengen können durch Boolesche Formeln beschrieben und durch sogenannte binäre Entscheidungsdiagramme (binary decision diagrams, kurz BDD) dargestellt werden [Bry86].

BDD wurden zuerst in [Bry86] eingeführt und werden seitdem häufig praktisch eingesetzt, um Teilmengen einer endlichen Grundmenge effizient zu repräsentieren. Verschiedene Operationen, die für den Vergleich und die Analyse von T-Invarianten nützlich sein können, lassen sich auf einer BDD-basierten Datenstruktur effizient implementieren. Dazu gehören Operationen wie Schnitt und Vereinigung von Teilmengen.

Für eine effiziente Umsetzung solcher Operationen muss eine günstige Reihenfolge der Variablen des BDD festgelegt werden. Von der Reihenfolge der Variablen hängt die Größe der Datenstruktur ab, die das BDD repräsentiert. Eine günstige Reihenfolge der Variablen kann bisher nur durch Heuristiken effizient bestimmt werden, denn das Finden der optimalen Reihenfolge ist ein NP-vollständiges Problem (siehe [BW96]). Die abstract DTS können eine Grundlage der Heuristik bilden, denn abhängige Variablen sollten möglichst hintereinander angeordnet werden, damit die Knotenanzahl des BDDs möglichst gering ist.

Die Transitionen einer abstract DTS können zusammengefasst als eine Variable im Booleschen Ausdruck betrachtet werden, ohne dass es zu einem Informationsverlust kommt. Der Träger jeder T-Invariante lässt sich anhand der DTS-Informationen und anhand des BDDs eindeutig rekonstruieren.

Automat zur Darstellung der T-Invarianten

In diesem Abschnitt bilden deterministische endliche Automaten die Grundlage für eine Darstellung der T-Invarianten. Ziel ist es, in möglichst kompakter Form eine endliche Menge von T-Invarianten zu beschreiben. Dafür wird zunächst in kurzer Form der deterministische endliche Automat formal eingeführt. Eine ausführliche Beschreibung des Modells befindet sich in [HU79].

Definition 3.2.4 *Ein deterministischer endlicher Automat ist ein Tupel $A = (Q, \Sigma, \delta, q_0, F)$ mit*

1. Q als endliche Menge von Zuständen
2. Σ als endliche Menge von Eingabesymbolen
3. $\delta : Q \times \Sigma \rightarrow Q$ als Übergangsfunktion
4. $q_0 \in Q$ als Startzustand
5. $F \subseteq Q$ als Menge der Endzustände.

Ein deterministischer endlicher Automat, kurz DEA, verarbeitet Zeichenreihen, indem er anhand des gelesenen Zeichens in einen neuen Zustand wechselt. Welche Zustandsübergänge möglich sind, beschreibt die Übergangsfunktion δ . Eine Zeichenreihe aus Eingabesymbolen wird von einem DEA M akzeptiert, wenn sich dieser nach der Verarbeitung aller Zeichen des Wortes in einem Endzustand befindet. Die Funktion δ^* ist induktiv über die Länge der Worte definiert:

1. $\forall q \in Q : ((q, \varepsilon), q) \in \delta^*$
2. $\forall w \in \Sigma^*, a \in \Sigma$ und $p, q, r \in Q$ gilt: ist $((p, w), q) \in \delta^*$ und $((q, a), r) \in \delta$, so ist auch $((p, wa), r) \in \delta^*$.

Mit Hilfe von δ^* ist die von M akzeptierte Sprache L_M folgendermaßen definiert:

$$L_M = \{w \mid \exists p \in F : ((q_0, w), p) \in \delta^*\}.$$

Für jede endliche Sprache kann man einen eindeutigen minimalen DEA generieren, der genau diese Sprache erkennt. Der Knuth-Morris-Pratt-Algorithmus entwickelt einen DEA für einzelne Worte. Er wird ausführlich in [Sed83] beschrieben. Da endliche Sprachen unter Vereinigung abgeschlossen sind, ist ein Automat für die Sprache aus den Automaten für die einzelnen Worte konstruierbar.

Fasst man die Transitionen als Menge von Eingabesymbolen und Schaltfolgen als Worte auf, definiert eine endliche Menge von Schaltfolgen eine Sprache. Würde nun jeder T-Invariante eine Schaltfolge zugeordnet werden, könnte man für die entsprechende Menge von Schaltfolgen einen Automaten generieren und hätte eine vollständige Beschreibung der T-Invarianten mit Hilfe eines DEA. Jeder Weg vom Startzustand zu einem Endzustand entspräche dann einer T-Invariante.

Leider gibt es zu jeder T-Invariante, die mehr als eine Transition enthält, mehrere Schaltfolgen. Zudem hängt die Anzahl der Zustände im Automaten sehr stark von den gewählten Schaltfolgen ab. Deshalb wird für die Darstellung von T-Invarianten mit DEA eine beliebige aber feste Reihenfolge der Transitionen gewählt. Wird die Reihenfolge der Transitionen festgelegt, kann man die

Paare (t, h) als Eingabesymbole, die T-Invarianten als Worte aus diesen Symbolen und die Menge der T-Invarianten als Sprache auffassen. Ein Wort zu einer T-Invariante besteht dann aus einer Aufzählung von Paaren (t, h) , wobei t eine Transition und h deren Häufigkeit in der T-Invariante darstellt. Dabei muss die Reihenfolge der Transitionen eingehalten werden. Jeder T-Invariante wird so eine eindeutige Schaltfolge und damit ein eindeutiges Wort zugeordnet.

Die festgelegte Reihenfolge hat Auswirkungen auf die Größe des Automaten. Sind die Transitionen günstig angeordnet, hat der Automat wenige Verzweigungen und Zustände. Die Kenntnis der strong DTS kann dafür benutzt werden, die Anzahl der Verzweigungen und Zustände zu verringern. Transitionen einer DTS sollten möglichst hintereinander angeordnet werden, denn da sie immer zusammen schalten, gibt es an diesen Stellen möglicherweise weniger Verzweigungen im Automaten als bei einer beliebigen Anordnung.

Analog zur Zusammenfassung von Transitionen einer abstract DTS im BDD kann mit Hilfe der strong DTS die Anzahl der Zustände im DEA nochmals verkleinert werden. Das geschieht, indem eine Kette von Zuständen, deren Zustandsübergänge gerade der strong DTS entsprechen, zu einem Zustandsübergang zusammengefasst wird. Aus den Informationen der strong DTS kann jederzeit wieder auf die konkrete Häufigkeit der Transition in den T-Invarianten geschlossen werden.

Da die DEA-basierte Darstellung der T-Invarianten die konkreten Häufigkeiten in die Eingabesymbole einbezieht, eignen sich für das Festlegen der Reihenfolge der Transitionen nur die strong DTS. Verzweigungen können bei letzteren innerhalb der zu der strong DTS gehörenden Zustände immer noch entstehen, da die Häufigkeiten der Transitionen innerhalb der strong DTS voneinander verschieden sein können.

Abbildung 3.3 zeigt einen Automaten, der die Menge der minimalen Invarianten des Petri-Netzes aus der Fallstudie 5.2 in Kapitel 5 zeigt. Es gibt vier minimale Invarianten und fünf strong DTS. Statt den Kanten des Automaten Paare aus Transitionen und Häufigkeiten als Beschriftung zuzuweisen, werden die Kanten mit den Namen der DTS und einer Häufigkeit beschriftet. Ist die Häufigkeit eins, kann sie analog zur Notation bei den Petri-Netzen weggelassen werden.

Die Häufigkeit beschreibt, wie oft die Transitionen der DTS in ihrer Gesamtheit schalten. Mit dem Transitionsvektor der DTS kann dann auf die Schalthäufigkeit einer Transition in der DTS geschlossen werden, indem die Häufigkeit an der Kante mit der Häufigkeit einer Transition im Transitionsvektor multipliziert wird. Das Ergebnis ist ein Automat mit vier Zuständen. Vom Startzustand aus, der durch einen Pfeil ohne Quellknoten gekennzeichnet ist, gibt es vier verschieden beschriftete Wege zu dem akzeptierenden Endzustand. Dieser ist durch eine doppelte Umrahmung hervorgehoben. Der Automat in Abbildung 3.3 erkennt die endliche Sprache $\{(DTS1, 1)(DTS2, 1)(DTS4, 1), (DTS1, 1)(DTS2, 1)(DTS5, 1), (DTS1, 1)(DTS3, 1)(DTS4, 1), (DTS1, 1)(DTS3, 1)(DTS5, 1)\}$. Diese Worte beschreiben komprimiert die T-Invarianten des Petri-Netzes.

Für die Bestimmung einer günstigen Reihenfolge der DTS können ähnliche Algorithmen beziehungsweise Heuristiken verwendet werden, die auch zu Bestimmung der Reihenfolge der Variablen in BDD eingesetzt werden.

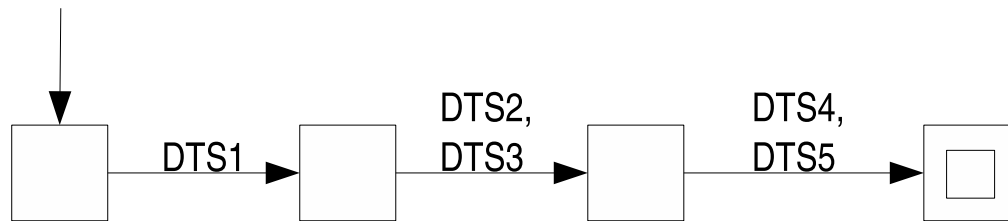


Abbildung 3.3: Invariantenbeschreibung mit finitem Automaten

Hierarchisierung und Reduktion des Petri-Netzes

Neben der Darstellung der T-Invarianten mit Hilfe von DEA und BDD kann ein weiteres Ziel die Verbesserung des Petri-Netz-Modells durch Transformationen anhand der T-Invarianten sein.

Bei der Modellierung eines Systems mit Petri-Netzen kommt es darauf an, anschaulich und analytisch aussagekräftig das System darzustellen. Der Detaillierungsgrad, mit dem einzelne Teile des Systems herausgearbeitet werden, wirkt sich auf diese Eigenschaften stark aus.

Vergrößerungen und Verfeinerungen von Netzteilen sind geeignete Transformationen, Systemteile mit dem jeweils gewünschten Detaillierungsgrad zu modellieren. Inhaltlich zusammenhängende transitions- beziehungsweise platzberandete Teilnetze werden durch hierarchische Transitionen beziehungsweise Plätze ersetzt. Das ersetzte Teilnetz bleibt vollständig auf der nächsten Hierarchieebene bestehen. Die Eigenschaften des Petri-Netzes bleiben vollständig erhalten, da sich das Netzverhalten bei dieser Transformation nicht ändert. Einzig die Darstellung des Systems ändert sich.

Eine gute Hierarchisierung eines Petri-Netzes zeichnet sich durch einen sinnvollen Detaillierungsgrad auf jeder Hierarchieebene und eine sinnvolle Auswahl der Teilnetze aus, die sich auf der nächsten Hierarchieebene befinden. Dabei sollte jede Hierarchieebene strukturell abgeschlossen und verständlich sein, ohne Kenntnis des genauen Inhaltes der hierarchischen Knoten.

Die durch die DTS induzierten Teilnetze haben die Eigenschaft, dass basierend auf den T-Invarianten ein enger Zusammenhang zwischen den in der DTS enthaltenen Transitionen besteht. Dies ist eine wichtige Voraussetzung für eine sinnvolle Hierarchisierung.

Eine weitere Technik zur Vereinfachung von Petri-Netzen ist die schrittweise Reduktion mit Hilfe von Reduktionsregeln. Dabei wird in jedem Reduktionsschritt ein Teilnetz des Petri-Netzes durch ein anderes Teilnetz ersetzt. Zumeist verringert sich dabei die Anzahl der Knoten und die Komplexität des Petri-Netzes. Bei der Reduktion sollen sich die wesentlichen Eigenschaften des Petri-Netzes nicht verändern. Wesentliche Eigenschaften können die Invarianten, die Lebendigkeit oder die Beschränktheit des Petri-Netzes sein.

Es gibt bereits eine Reihe von Reduktionsregeln, die zur Reduktion von Petri-Netzen verwendet werden. Einige sind in [Sta90] aufgeführt. Diese Regeln basieren auf der lokalen Suche von bestimmten Teilnetzen, die dann durch einfachere Teilnetze ersetzt werden können. Sie sind unabhängig von den T-Invarianten des Petri-Netzes und erhalten meist die Lebendigkeit und die Beschränktheit des Petri-Netzes, aber nicht die Invarianten. Werden beispielsweise mit der Reduk-

tionsregel 3 aus [Sta90, Kapitel Reduktion] parallele Transitionen entfernt, kann sich die Anzahl der T-Invarianten im Petri-Netz um den Faktor zwei verringern.

Wenn die Menge der minimalen T-Invarianten des Petri-Netzes zur Verfügung steht und bei der Reduktion des Petri-Netzes die T-Invarianten des Petri-Netzes erhalten bleiben sollen, können auf Basis der DTS weitere Reduktionsregeln formuliert werden. Diese neuen Reduktionsregeln sollten auch Teilnetze erfassen, für die keine der bekannten lokalen Reduktionsregeln anwendbar ist. Die auf den DTS basierende Reduktion eines Petri-Netzes ist die Fortführung des Gedankens der Hierarchisierung, denn das reduzierte Teilnetz wird nicht auf die nächste Hierarchieebene verschoben und durch eine hierarchische Transition ersetzt, sondern direkt durch eine Transition ersetzt.

3.3 Zusammenfassung der vorgestellten Verfahren

Die Standardverfahren aus dem Bereich Data-Mining erzielen, abgesehen von der Clusterung, scheinbar nicht direkt verwertbare Ergebnisse. Eine Teilmenge der Assoziationsregeln, die zu einer Menge von Invarianten ermittelt werden, ermöglicht weiterführende Schlüsse über die Invarianten von Petri-Netzen. Speziell die daraus ableitbaren abhängigen Transitions Mengen erscheinen aussichtsreich für eine weitere Verwendung. Die Clusterung von T-Invarianten ist trotz offensichtlicher Eignung aus oben genannten Gründen nicht Gegenstand dieser Arbeit.

Die Kenntnis der voneinander abhängigen Transitions Mengen, also der DTS eines Petri-Netzes, kann einerseits dazu benutzt werden, die T-Invarianten komprimiert zu verwalten und darzustellen. Dabei erscheint eine Beschreibung der T-Invarianten mit Hilfe von DEA als sinnvoll. Die Darstellung mit Hilfe von DEA hat den Vorteil, dass sie graphisch leicht zu veranschaulichen ist. Außerdem kann ein DEA mit Standardverfahren, zum Beispiel in [HU79] beschrieben, in eine rechtslineare Grammatik beziehungsweise in einen regulären Ausdruck umgewandelt werden.

Ein Nachteil der Methode mit DEA ist das Festlegen einer konkreten Reihenfolge der Transitionen für die Darstellung. Es suggeriert vor allem bei Visualisierungen die Tatsache, dass die Schalthihenfolge der Transitionen in einer T-Invariante feststeht. Eine T-Invariante kann man aber als eine Multimenge interpretieren, bei der die Reihenfolge der Elemente keine Rolle spielt. Wenn für alle T-Invarianten eine einzige Reihenfolge der Transitionen beim Schalten festgelegt wird, geht dieser Charakter der T-Invariante verloren.

Die Reihenfolge spielt dagegen bei der Darstellung von T-Invarianten auf Basis von Teilmengen zunächst keine Rolle. Eine effiziente Art der Repräsentation dieser Teilmengen, die BDD, eignen sich zwar gut dazu, intern die Teilmengen zu verwalten und bestimmte Operationen wie Schnitt und Vereinigung darauf auszuführen. Da jedoch auch eine effiziente Verwaltung von T-Invarianten nur durch weitere Auswertungen neue Erkenntnisse bringen kann, und BDD nicht dazu geeignet sind, die Informationen über die T-Invarianten für einen menschlichen Betrachter geeignet darzustellen, wird die BDD-basierte Darstellung in dieser Arbeit nicht weiter verfolgt. Leider gehen außerdem bei der Repräsentation der T-Invarianten durch Teilmengen die quantitativen Informationen über die T-Invarianten verloren.

Neben der Verwendung der DTS in Methoden, welche T-Invariantenmengen darstellen, können die DTS auch dazu benutzt werden, das Petri-Netz zu vereinfachen. In Frage kommen dafür verschiedene Transformationen, die eine Hierarchisierung oder eine Reduktion des Petri-Netzes zur Folge haben. Dabei sind abstrahierende Verfahren auf Basis der abstract DTS sowie die Genauigkeit erhaltende Verfahren auf Basis der strong DTS möglich.

Eine automatisierbare Erkennung von geeigneten Teilnetzen für eine Hierarchisierung oder eine Reduktion anhand der rein statischen Eigenschaften des Petri-Netzes, zu denen die T-Invarianten gehören, könnte den praktischen Modellierungsprozess verbessern. Deshalb wird im Folgenden genauer auf diese Verwendungsmöglichkeiten der DTS eingegangen und im folgenden Kapitel genauer vorgestellt.

Kapitel 4

Hierarchisierung und Reduktion des Petri-Netzes

Eine hierarchische Gliederung eines Petri-Netzes kann durch Abstraktion an geeigneten Stellen dazu beitragen, dem menschlichen Betrachter das Verständnis zu erleichtern. In diesem Kapitel wird eine automatische Hierarchisierung anhand der DTS, die im vorhergehenden Kapitel ausführlich beschrieben wurden, vorgestellt.

Werden in einem Petri-Netz Transitionen oder Plätze mit Hilfe von Reduktionsregeln zu neuen Transitionen und Plätzen zusammengefasst, kann das entstehende Petri-Netz übersichtlicher werden. Durch Reduktionen kann sich der Aufwand für die Berechnung des Erreichbarkeitsgraphen verringern, und die Informationen, die das Petri-Netz enthält, können auf Wesentliches reduziert werden.

Bei Reduktionen sollen sich möglichst wenige wesentliche Eigenschaften des Netzes ändern. Neben der Hierarchisierung werden im Folgenden verschiedene neue Reduktionsregeln vorgestellt.

Man unterscheidet im Allgemeinen bei den Reduktionsregeln für Petri-Netze zwischen lokalen und globalen Reduktionsregeln. Lokale Reduktionsregeln setzen im Gegensatz zu globalen Reduktionsregeln keine Kenntnis des Erreichbarkeitsgraphen voraus [Sta90]. Für die in dieser Arbeit eingeführten Reduktionsregeln muss die Menge der minimalen T-Invarianten bekannt sein. Die minimalen T-Invarianten werden zwar nicht mit Hilfe des Erreichbarkeitsgraphen berechnet, setzen aber die Kenntnis des gesamten Petri-Netzes voraus. Außerdem ist der Aufwand zur Berechnung der minimalen T-Invarianten eines Petri-Netzes im schlimmsten Fall exponentiell. Deshalb sind die hier vorgestellten Reduktionsregeln ebenfalls den globalen Reduktionsregeln zuzuordnen, wenn auch der Erreichbarkeitsgraph zur Bestimmung der zu reduzierenden Knoten nicht benötigt wird.

Da die Reduktion einer nicht echten DTS nach bisherigen Erkenntnissen nicht sinnvoll erscheint, wird im Folgenden immer von echten DTS gesprochen.

4.1 Hierarchische Strukturierung

Mit einer Hierarchisierung eines Petri-Netzes soll die Übersichtlichkeit erhöht werden, indem an geeigneten Stellen hierarchische Knoten Teilnetze symbolisieren. Das hierarchisch gegliederte Petri-Netz hat dabei das gleiche Verhalten, denn durch die Hierarchisierung ändert sich nur die optische Darstellung der Struktur, nicht das Petri-Netz selbst.

In der Regel wird die Hierarchisierung mit menschlichem Sachverstand während des Modellierungsprozesses durchgeführt. Inhaltlich eng zusammenhängende transitionsberandete oder platzberandete Netze werden hierarchischen Transitionen oder Plätzen zugeordnet und durch diese symbolisch dargestellt.

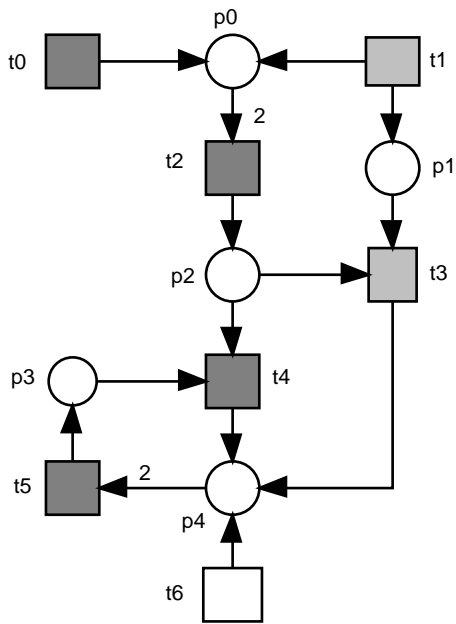
Stehen zu einem Petri-Netz-Modell die T-Invarianten zur Verfügung, können mit den in Kapitel 3.2.2 vorgestellten Algorithmen die maximalen DTS des Modells berechnet werden. Jede Teilmenge einer abstract (strong) DTS ist ebenfalls eine abstract (strong) DTS. Die Transitionen jeder DTS induzieren platzberandete Teilnetze, die sich allerdings an Randplätzen überlappen können. Reduziert wird das induzierte Teilnetz einer DTS ohne die Randplätze. Die reduzierten Teilnetze sind damit transitionsberandet.

Abbildung 4.1(a) zeigt noch einmal das Beispielnetz aus Abbildung 3.1, mit verschiedenen markierten Transitionen der drei maximalen abstract DTS. In Abbildung 4.1(b), 4.1(d) und 4.1(c) ist das hierarchisierte Netz mit den Teilnetzen abgebildet. Es entsteht durch die oben beschriebene Ersetzung der Teilnetze, die durch die mehrelementigen maximalen abstract DTS DTS_1 und DTS_2 induziert werden. Der entstehende hierarchische Knoten wird jeweils mit dem Namen der abstract DTS benannt.

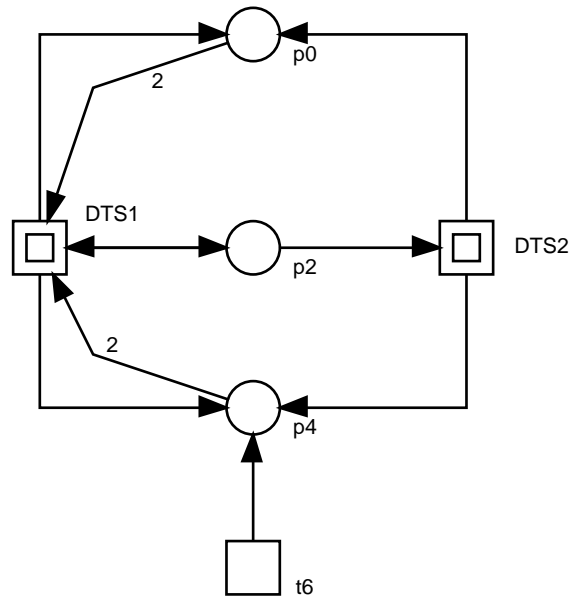
Da zwischen den Transitionen einer DTS bezüglich der T-Invarianten eine enge Verbindung besteht, bilden diese induzierten Teilnetze eine geeignete Grundlage für eine Hierarchisierung des Petri-Netzes. Die Transitionen einer DTS können im Bezug zu den T-Invarianten als funktionale Einheit betrachtet werden. Alle Transitionen, die zu einer DTS gehören, sowie alle inneren Plätze der DTS, können als Teilnetz von einer hierarchischen Transition symbolisiert werden.

Für die hierarchische Strukturierung eines Petri-Netzes mit durch DTS induzierten Teilnetzen können sowohl strong DTS als auch abstract DTS benutzt werden. Denkbar ist auch eine mehrstufige Hierarchisierung. Es kann zum Beispiel sinnvoll sein, mehrere zusammenhängende DTS durch eine einzige hierarchische Transition darzustellen, die dann selbst für jede DTS eine hierarchische Transition enthält. Denkbar ist auch, für die erste Hierarchiestufe die zu den abstract DTS gehörenden Teilnetze durch hierarchische Transitionen zu ersetzen. In einem zweiten Schritt können die strong DTS dann eine weitere Hierarchiestufe bilden.

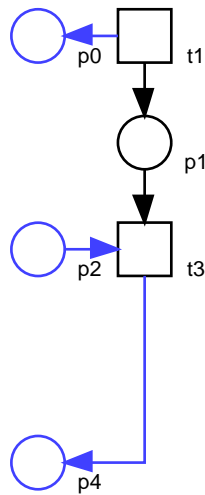
Neben der Möglichkeit, Netze zu hierarchisieren, kann auch die bestehende Hierarchisierung überprüft werden, beziehungsweise können funktionell zusammenhängende, aber über verschiedene Hierarchieebenen verteilte Transitionen identifiziert werden.



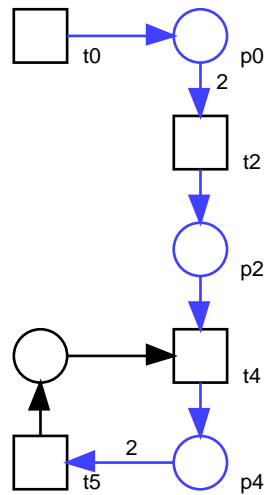
(a) Originalnetz



(b) Hierarchisiertes Hauptnetz



(c) Hierarchisches Netz für DTS2



(d) Hierarchisches Netz für DTS1

Abbildung 4.1: Beispiel für Hierarchisierung mit abstract DTS

4.2 Trägerbasierte Reduktion

Statt die Teilnetze durch hierarchische Transitionen zu ersetzen, können Teilnetze auch zu Transitionen reduziert werden. Eine Reduktion des Netzes ist im Gegensatz zur Hierarchisierung eine strukturelle Veränderung, die die Eigenschaften des Netzes beeinflussen kann.

Die erste und einfachste Reduktion ist die trägerbasierte Reduktion. Sie dient dazu, anhand der Informationen der T-Invarianten das Petri-Netz qualitativ in disjunkte Teile zu zerlegen. Qualitativ bedeutet hier, dass die konkreten Häufigkeiten nicht in die Reduktion einbezogen werden. Quantitative, die Häufigkeiten betreffende Aspekte spielen keine Rolle bei dieser Reduktion, denn die Transitionen einer abstract DTS werden trotz möglicher Unterschiede bei den Häufigkeiten in den T-Invarianten zu einer Einheit zusammengefasst. Da die Transitionen einer abstract DTS in den T-Invarianten immer gemeinsam vorkommen, bilden die entstehenden Einheiten eine Möglichkeit, ein quantitatives Petri-Netz zumindest qualitativ zu vereinfachen und damit die durch T-Invarianten beschriebenen Abläufe besser zu erkennen.

Reduktionsregel 4.2.1 (Trägerbasierte Reduktionsregel) *Es sei $N = (P, T, F, m_0)$ ein Petri-Netz und I eine Teilmenge der T-Invarianten von N . Durch die trägerbasierte Reduktion entsteht das Petri-Netz N' mit der leeren Markierung und der Kantenbewertung F' folgendermaßen:*

Voraussetzung: Seien alle Transitionen aus A in einer echten abstract DTS, die auf Basis von I berechnet wurde.

Anwendung: Entferne alle Transitionen aus A aus dem Petri-Netz. Entferne ebenfalls alle Plätze, deren Vor- und Nachtransitionen alle in A enthalten sind. Füge eine neue Transition r in das Netz ein. Die Markierung von N' ist die leere Markierung. Die Kantenbewertung wird wie folgt festgelegt:

$$F'(p, t) = \begin{cases} 1 & \text{falls } t \notin A \cup \{r\} \wedge F(p, t) > 0 \text{ oder } t = r \wedge \exists a \in A : a \in p \bullet \\ 0 & \text{sonst} \end{cases}$$

$$F'(t, p) = \begin{cases} 1 & \text{falls } t \notin A \cup \{r\} \wedge F(t, p) > 0 \text{ oder } t = r \wedge \exists a \in A : a \in \bullet p \\ 0 & \text{sonst} \end{cases}$$

Die Kantenbewertungen des gesamten ursprünglichen Netzes werden bei der Berechnung der Kantenbewertung F' für das reduzierte Netz N' neu bestimmt. Es sind die beiden Kantenbewertungen null und eins möglich. Dies geschieht, weil die Informationen über die Schalthäufigkeiten bei der Reduktion nicht in allen T-Invarianten gleich sind. Damit kann auch die neue Kantenbewertung im Allgemeinen nicht mehr eindeutig festgelegt werden. Beispielsweise sind die Schalthäufigkeiten der reduzierten Transitionen in den T-Invarianten des Petri-Netzes aus Abbildung 4.1 bei der Reduktion nicht eindeutig auf die neue Transition umrechenbar.

Die Analyse der beiden T-Invarianten dieses Netzes ergibt laut Tabelle 3.3, dass sich in jeweils einer abstract DTS die Transitionen $\{t1, t3\} = DTS1$ und $\{t0, t2, t4, t5\} = DTS2$ sowie $\{t6\} = DTS3$ befinden. Bei der Zusammenfassung der beiden Transitionen $t1$ und $t3$ ist leicht

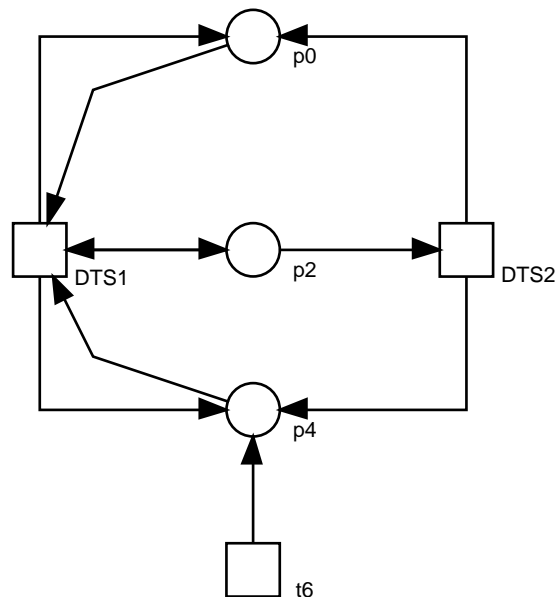


Abbildung 4.2: Beispiel für trägerbasierte Reduktion

zu bestimmen, welche Auswirkungen ein gemeinsames Schalten dieser Transitionen auf die Markenzahl der Vor- und Nachplätze hat. Hier wird in beiden T-Invarianten inv_0 und inv_1 des Netzes aus Abbildung 3.1 nur eine Marke von p_2 entfernt und jeweils eine Marke auf p_0 und p_4 abgelegt. Anders ist es beim Zusammenfassen der Transitionen von $DTS2$. Es ist nicht klar, ob die neue Transition insgesamt die Markenzahl von p_4 um eins erhöht, wie in inv_0 , oder die Markenzahl insgesamt nicht verändert, wie in inv_1 . Deshalb wird bei der neuen Kantenbewertung immer dann die Kante zu oder von der neuen Transition r mit eins bewertet, wenn der entsprechende Platz Vor- oder Nachplatz einer der zusammengefassten Transitionen war.

Die Bewertungen der Kanten zu der neuen Transition r lassen sich also höchstens noch qualitativ festlegen, nicht mehr quantitativ. Da so Teile des Netzes nur noch qualitativ spezifiziert sind, wird auch der verbleibende Teil der Kantenbewertungen nur noch qualitativ, also mit null oder eins, angegeben.

Das Petri-Netz aus Abbildung 4.1, bei dem die beiden mehrelementigen abstract DTS zu jeweils einer Transition reduziert wurden, wird in Abbildung 4.2 gezeigt.

Neben der nicht eindeutig festzulegenden Kantenbewertung ist auch eine exakte Behandlung von nichtleeren Markierungen im Allgemeinen nicht möglich. Die naheliegende Idee, alle Markierungen auf im neuen Netz ebenfalls vorhandenen Plätzen zu erhalten, führt nicht zu einem äquivalenten Verhalten des neuen Netzes. Da die neuen Kantenbewertungen nur null oder eins betragen können, werden durch die Reduktion möglicherweise Transitionen schaltfähig, die vorher nicht schaltfähig waren. Das Verhalten des neuen Netzes wäre komplett verschieden von dem Verhalten des alten Netzes, es suggeriert aber durch die weiterbestehende Markierung ein ähnliches Verhalten.

Ein Beispiel liefert das Originalnetz in Abbildung 4.1(a). Liegt eine Marke auf Platz p_4 und eine auf p_2 , wenn t_4 und t_5 zu einer neuen Transition r reduziert werden, so hat r Konzession, obwohl im ursprünglichen Netz weder t_4 noch t_5 Konzession hatten.

Im trägerbasiert reduzierten Netz werden damit grundsätzlich alle Kantenbewertungen größer als eins zu eins, und alle Markierungen werden entfernt. Das Ergebnis eines so reduzierten Netzes ist eine qualitative Zusammenfassung der Informationen aus dem ursprünglichen Petri-Netz und seinen T-Invarianten.

Ein wesentlicher Nachteil der trägerbasierten Reduktion wird schnell deutlich: das Petri-Netz ist damit zwar in qualitativ zusammenhängende Einheiten einteilbar und die Übersichtlichkeit des Netzes verbessert sich. Quantitative Aussagen sind jedoch im Allgemeinen nach der Reduktion nicht mehr möglich. Zudem kann sich das Netzverhalten ändern. Die Reduktion dient somit im Wesentlichen dazu, inhaltlich zusammenhängende Netzteile zusammenzufassen und von den konkreten Schalthäufigkeiten abstrahieren zu können.

Durch eine Einschränkung der Reduktionsmöglichkeiten kann erreicht werden, dass das grundlegende Verhalten des Netzes durch die Reduktion weniger stark beeinflusst wird. Hier wird die Menge der T-Invarianten des Petri-Netzes als wichtige Komponente des grundlegenden Verhaltens angesehen. Deshalb erfolgt die Reduktion im nächsten Abschnitt so, dass die Menge der T-Invarianten des neuen Netzes äquivalent zur Menge der T-Invarianten des alten Petri-Netzes bleibt.

4.3 Häufigkeitsbasierte Reduktionsregel

Sollen die Transitionen mit Hilfe der minimalen T-Invarianten so zusammengefasst werden, dass die quantitativen Informationen des Petri-Netzes erhalten bleiben, müssen die Schalthäufigkeiten der Transitionen in den T-Invarianten in die Überlegungen einbezogen werden. Nur so können Widersprüche beim Ermitteln der Kantenbewertung zu den neuen Knoten vermieden werden. Die Schalthäufigkeiten sind bei den abstract DTS im Allgemeinen nicht auf bestimmte Werte eingeschränkt. Im Gegensatz dazu stellen die strong DTS höhere Anforderungen an die Schalthäufigkeiten und bilden Transitions Mengen, die sich unter Beachtung der quantitativen Informationen zusammenfassen lassen.

Vorüberlegungen

Die Reduktion eines Petri-Netzes $N = (P, T, F, m_0)$ zu einem neuen Netz $N' = (P', T', F', m'_0)$ wird zunächst ähnlich der trägerbasierten Reduktion durchgeführt. Die reduzierten Transitionen S sind die Transitionen, die in einer echten strong DTS S enthalten sind. Zu jeder strong DTS kann es mehrere Transitionsvektoren geben, aber nur einen minimalen Transitionsvektor v . Dieser minimale Vektor hat als kleinsten gemeinsamen Teiler aller seiner Komponenten eins und bildet die Grundlage der Reduktion. Er wird deshalb **Reduktionsvektor** genannt.

Eine neue Transition r ersetzt die reduzierten Transitionen aus S . Die damit neu entstehenden Kanten erhalten eine Bewertung, die auf der Bewertung der Kanten zwischen Transitionen aus S und den Plätzen in N basiert. Bei der Berechnung der Kantenbewertung einer neu hinzukommenden Kante zwischen einem Platz p und der neuen Transition r können folgende Fälle auftreten:

1. p ist nicht gleichzeitig Nachplatz und Vorplatz von Transitionen aus S .
2. p ist sowohl Nachplatz als auch Vorplatz von Transitionen aus S . Hier sind folgende Fälle möglich:
 - (a) p ist ein 'Kontaktplatz' von S : $\bullet p \cup p \bullet \not\subseteq S$
 - (b) p ist ein 'innerer' Platz von S : alle Vor- und Nachtransitionen von p sind in S , also $\bullet p \cup p \bullet \subseteq S$.

Im ersten Fall ergibt sich die Kantenbewertung aus der Summe der Kantenbewertungen zwischen p und den Transitionen aus S :

$$F'(p, r) = \sum_{t \in S} v[t] \cdot F(p, t), \quad F'(r, p) = \sum_{t \in T_S} v[t] \cdot F(t, p). \quad (4.1)$$

Ein Kontaktplatz von S ist ein Randplatz der von T_S induzierten Teilnetze. Schalten die Transitionen aus T_S so oft wie in v angegeben, entnehmen sie im Fall 2a vom Platz p Marken, legen aber auch Marken auf p ab. Die Schaltreihenfolge der Transitionen ist nicht eindeutig festgelegt. Es ist also nicht klar, ob zuerst Marken abgelegt und dann entnommen werden oder umgekehrt.

Es ergeben sich zwei Möglichkeiten, die beiden Kanten f_1 und f_2 zwischen einem Kontaktplatz p und der reduzierten Transition r zu bewerten.

Die erste Variante der Kantenbewertung bei der Reduktion bewertet beide Kanten unabhängig voneinander mit der Summe der durch v gewichteten Kantenbewertungen. In diesem Fall ist die Berechnungsformel für F' wie im ersten Fall Formel 4.1. Dies hat den Vorteil, dass erkennbar bleibt, dass Plätze Marken tragen müssen, damit die Transitionen in S schalten können. Das Verhalten des Petri-Netzes wird durch die Reduktion bezüglich der T-Invarianten dahingehend verändert, dass jetzt alle Kontaktplätze genügend Marken tragen müssen, damit die neue Transition schalten kann. Im ursprünglichen Netz müssen im Allgemeinen nicht so viele Marken auf den Kontaktplätzen vorhanden sein, da dort die Transitionen nacheinander schalten und damit zum Teil selbst die benötigten Marken produzieren.

Bei der zweiten Möglichkeit wird die Differenz $d(p, r)$ aus insgesamt abgelegten und entnommenen Marken, gewichtet mit den Schalthäufigkeiten aus v , gebildet:

$$d(p, r) = \sum_{t \in S} v[t] \cdot F(t, p) - \sum_{t \in T_S} v[t] \cdot F(p, t). \quad (4.2)$$

Ist diese Differenz $d(p, r)$ negativ, das heißt die Transitionen entnehmen mehr Marken als sie ablegen, wird die Kante (p, r) mit dem Betrag der Differenz bewertet, die Kante (r, p) mit null. Legen die Transitionen aus S mehr Marken ab als sie entnehmen, wird der Betrag der Differenz

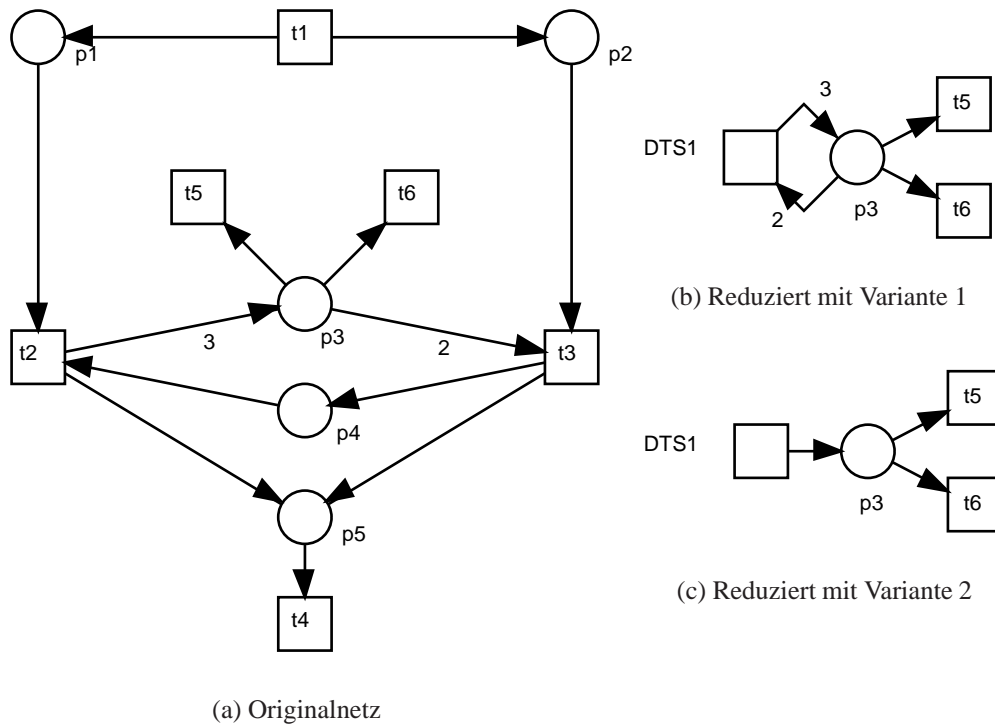


Abbildung 4.3: Beispiel für zwei Kantenbewertungsvarianten

als Kantenbewertung für die Kante (p, r) benutzt und null für die Rückkante (r, p) . In die Kantenbewertung geht damit nur die Gesamtwirkung der Transitionen aus S auf den Platz p innerhalb der T-Invarianten ein. Bei dieser Variante der Kantenbewertung ist am entstandenen Petri-Netz N' aber nicht mehr erkennbar, dass mindestens eine der reduzierten Transitionen im ursprünglichen Netz nur dann schalten kann, wenn mindestens eine Marke auf p liegt.

Beide Varianten der Kantenbewertung ändern das Verhalten des Netzes. Da die Adjazenzmatrix des entstehenden Petri-Netzes in beiden Fällen die gleichen Werte enthält, hat die Entscheidung, welche der beiden Varianten benutzt wird, keinen Einfluss auf die T-Invarianten des neuen Netzes. Die zweite Variante hat im Gegensatz zur ersten Variante den Vorteil, dass für das Schalten der Transitionen einer T-Invariante keine neuen Anforderungen ohne Entsprechung im ursprünglichen Netz hinzukommen, sondern höchstens Anforderungen entfallen. Da mit der Reduktion eine Vereinfachung des Petri-Netzes erreicht werden soll, wird in dieser Arbeit die zweite Variante benutzt.

Abbildung 4.3 zeigt beide Varianten der Kantenbewertung bei der Reduktion an einem Beispiel. Das abgebildete Petri-Netz 4.3(a) hat zwei T-Invarianten und eine mehrelementige DTS $DTS1 = \{t1, t2, t3, t4\}$. Die Transitionen von $DTS1$ werden reduziert, der Platz $p3$ ist der einzige Kontaktpunkt. Der Vektor $(1, 1, 1, 2, 0, 0)$ ist der Reduktionsvektor v bei der Ordnung $t1 < t2 < t3 < t4 < t5 < t6$ über T , mit $DTS1$ wird die neue Transition bezeichnet. Die Transition $t2$ legt in allen minimalen T-Invarianten drei Marken auf $p3$ ab, $t3$ entfernt in allen minimalen T-Invarianten wieder zwei

davon. Damit entsteht bei Variante eins der Kantenbewertung das Netz 4.3(b), bei der Nutzung von Variante 2 entsteht das Netz 4.3(c).

Eine ähnliche Entscheidung ist bei der Bestimmung von Kantenbewertungen im reduzierten Petri-Netz N' zu treffen, wenn der Fall 2b auftritt. Bei inneren Plätzen heben sich nach Lemma 3.2.3 die Wirkungen der Transitionen aus S auf, wenn diese mit den Schalthäufigkeiten aus ν schalten. Ein solcher Platz kann nicht Randplatz von N sein und kann deshalb nicht unter Fall 1 behandelt werden. Sonst könnten die Transitionen aus S entweder nur Token entnehmen oder nur Token ablegen. Da p nur mit Transitionen aus S verbunden ist, dürfte keine der Transitionen aus T_S in einer T-Invariante von N vorkommen. Dies wäre ein Widerspruch zu der Festlegung, dass nur echte strong DTS reduziert werden, deren Transitionen demnach in mindestens einer T-Invariante vorkommen.

Wie im Fall 2a ist nicht klar, ob zuerst Marken auf dem entsprechenden Platz abgelegt werden und dann wieder entnommen werden oder ob dies umgekehrt geschieht. Bei inneren Plätzen kann man deshalb entweder eine Testkante einfügen, oder beide Kanten mit null bewerten. In dieser Arbeit wird analog zu der Behandlung von Kontaktplätzen die zweite Variante benutzt. Da innere Plätze nur mit Transitionen aus S verbunden sind und das entstehende Petri-Netz keine isolierten Knoten enthalten darf, werden bei dieser Variante die inneren Plätze gelöscht.

Reduktion

Nach diesen Vorüberlegungen kann eine Reduktionsregel formuliert werden, mit der unter Beachtung der Häufigkeiten der Transitionen in den T-Invarianten ein Petri-Netz reduziert wird.

Reduktionsregel 4.3.1 (häufigkeitsbasierte Reduktionsregel) *Es sei $N = (P, T, F, m_0)$ ein Petri-Netz und I die Menge der minimalen T-Invarianten von N . Durch die häufigkeitsbasierte Reduktion entsteht das Petri-Netz N' mit der Kantenbewertung F' folgendermaßen:*

Voraussetzung: Sei S eine echte strong DTS auf Basis von I mit dem Reduktionsvektor v

Anwendung: Entferne alle Transitionen aus S aus dem Petri-Netz. Entferne ebenfalls alle Plätze, deren Vor- und Nachtransitionen alle in S enthalten sind. Füge eine neue Transition r in das Netz ein. Die Kantenbewertung wird wie folgt festgelegt:

$$F'(a, b) = \begin{cases} F(a, b) & \text{falls } a, b \notin S \cup \{r\} \\ d(a, b) & \text{falls } a = r \wedge d(a, b) > 0 \\ -d(b, a) & \text{falls } b = r \wedge d(b, a) < 0 \\ 0 & \text{sonst.} \end{cases}$$

Dabei ist d die Differenz aus auf p abgelegten und von p entnommenen Marken der Transitionen aus S :

$$d(a, b) = \sum_{t \in S} v[t] \cdot (F(t, b) - F(b, t)).$$

Die häufigkeitsbasierte Reduktion wird in Abbildung 4.4 am Beispielnetz 4.4(a) durchgeführt. Hier wurden die strong DTS markiert. Reduziert werden die beiden zweielementigen maximalen strong DTS $DTS1 = \{t1(1), t3(1)\}$ und $DTS2 = \{t4(1), t5(1)\}$. Zu beachten ist, dass zum reduzierten Petri-Netz weiterhin genau zwei minimale T-Invarianten gehören, die den T-Invarianten im Originalnetz entsprechen. Als Nachweis sind in Tabelle 4.1 noch einmal die T-Invarianten des Originalnetzes und die T-Invarianten des reduzierten Netzes aufgelistet.

Auch bei der häufigkeitsbasierten Reduktion stellt sich die Frage, wie Markierungen des Netzes behandelt werden können. Die bei der trägerbasierten Variante benutzte Strategie, alle Markierungen zu entfernen, ist auch hier anwendbar. Damit die Funktionalität des entstehenden Petri-Netzes jedoch der Funktionalität des ursprünglichen Netzes ähnlich ist, werden Markierungen von Plätzen, die auch im reduzierten Netz enthalten sind, beibehalten. Gemeinsam mit den bei der Reduktion aus dem Netz entfernten Plätzen wird auch deren Markierung entfernt. Die Markierung wird nicht auf andere Plätze übertragen. Damit gehen Teilinformationen über den Zustand des Petri-Netzes bei der Reduktion verloren. Da der Zustand aber keinen Einfluss auf T-Invarianten hat, und bei einer kompletten Entfernung aller Markierungen wesentlich mehr Zustandsinformationen verloren gehen können, wird diese Strategie für die häufigkeitsbasierte Reduktion benutzt.

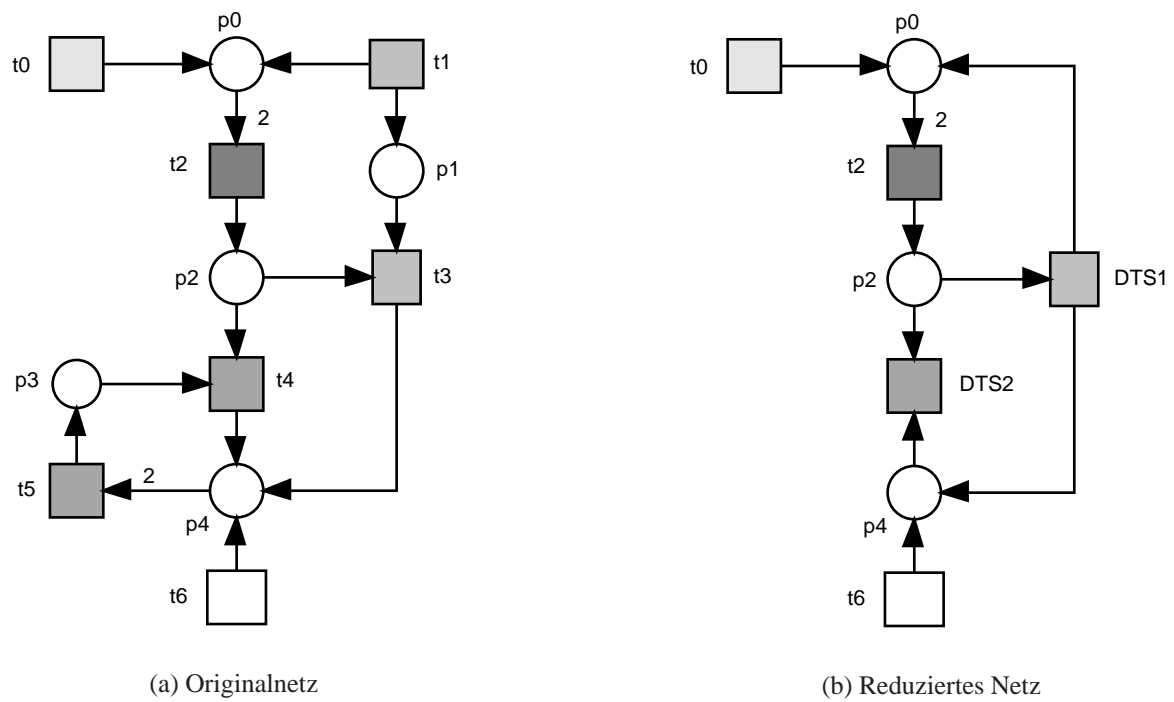


Abbildung 4.4: Beispiel für häufigkeitsbasierte Reduktion

Transition	inv_1	inv_2
t0	2	3
t1	0	1
t2	1	2
t3	0	1
t4	1	1
t5	1	1
t6	1	0

(a) T-Invarianten des Originalnetzes

Transition	inv_1	inv_2
t0	2	3
DTS1	0	1
t2	1	2
DTS2	1	1
t6	1	0

(b) T-Invarianten des reduzierten Netzes

Tabelle 4.1: Vergleich der T-Invarianten vor und nach der Reduktion

Eigenschaften der Reduktion

Ziel der häufigkeitsbasierten Reduktion bei einem Petri-Netz ist es, wesentliche Eigenschaften des Petri-Netzes zu erhalten. Wichtig ist in dieser Arbeit die Erhaltung der T-Invarianten. Der Nachweis, dass die häufigkeitsbasierte Reduktion die T-Invarianten des Petri-Netzes erhält, wird im nächsten Abschnitt geführt.

In [Sta90] wurden die Auswirkungen der dort vorgestellten Reduktionen auf Lebendigkeit und Beschränktheit des Petri-Netzes untersucht. Um die Ergebnisse dieser Arbeit besser mit den schon in [Sta90] vorgestellten Reduktionen zu vergleichen, wird in den folgenden Abschnitten die Auswirkung der hier vorgestellten Reduktionen auf diese Eigenschaften betrachtet. Aufgrund der Wichtigkeit der Realisierbarkeit einer T-Invariante für viele Anwendungen wird außerdem untersucht, wie sich die Reduktion auf die Realisierbarkeit einer T-Invariante auswirkt.

Invarianten

Ein wesentliches Ziel der häufigkeitsbasierten Reduktion ist die Erhaltung der T-Invarianten während der Reduktion. Das heißt, die T-Invarianten im Ergebnisnetz sollen äquivalent zu den T-Invarianten des ursprünglichen Netzes sein. Um zu zeigen, dass dieses Ziel erreicht wurde, muss zunächst definiert werden, was Äquivalenz in diesem Fall bedeutet.

Definition 4.3.2 Ein Petri-Netz $N = (P, T, F, m_0)$ mit Inzidenzmatrix C und den minimalen T-Invarianten I heißt T-Invarianten-äquivalent zu einem Petri-Netz $N' = (P', T', F', m'_0)$ mit Inzidenzmatrix C' und minimalen T-Invarianten I' (Schreibweise $N \trianglerighteq N'$), falls gilt:

1. es existiert eine bijektive Funktion $f : I \rightarrow I'$
2. es existiert eine totale und surjektive Funktion $h : T \rightarrow T'$ mit:

$$\forall i \in I \forall t \in T : i[t] \neq 0 \Leftrightarrow f(i)[h(t)] \neq 0.$$

Gilt $N \trianglerighteq N'$ für zwei Petri-Netze N und N' , so garantiert der erste Teil dieser Definition die naheliegende Forderung, dass die Anzahl der minimalen T-Invarianten bei beiden Netzen gleich ist. Es wird weiterhin gefordert, dass es eine Abbildung h gibt, die Transitionen von N eindeutig auf die Transitionen von N' abbildet.

Diese Abbildung h wurde eingeführt, damit nicht beliebige Netze mit gleicher Anzahl der minimalen T-Invarianten sofort T-Invarianten-äquivalent sind. Durch h wird eine gewisse Ähnlichkeit der Struktur der Petri-Netze und deren T-Invariantenmengen gefordert. Ist eine Transition t in einer T-Invariante i enthalten, so muss das Bild $t' = h(t)$, also das 'Äquivalent' von t in N' , auch in $f(i)$ enthalten sein. Da h total ist, hat jede Transition von N ein solches Äquivalent. Die Surjektivität von h garantiert, dass jede Transition in T' mindestens ein Urbild hat. Allerdings ist h im Allgemeinen nicht injektiv, es können mehrere Transitionen aus T auf eine Transition aus T' abgebildet werden.

Definition 4.3.3 (Reduzierbare Transitionsmenge) Gegeben sei ein Petri-Netz $N = (P, T, F, m_0)$ mit Inzidenzmatrix C . Sei I die Menge der Transitionsvektoren der minimalen T -Invarianten von C . Über P und T sei eine beliebige aber feste Ordnung definiert, mit Hilfe derer die Inzidenzmatrix sowie alle Transitions- und Platzvektoren indiziert werden können.

Eine *reduzierbare Transitionsmenge* G ist eine Multimenge mit Transitionen aus T und einem zugehörigen Reduktionsvektor v , für den gilt:

1. $v[t] = 0$, falls $t \notin G$
2. $v[t] \neq 0$, falls $t \in G$
3. $\forall i \in I \exists k \in \mathbb{N} \forall t \in G: (i - k \cdot v)[t] = 0$.

Die Transitionen aus einer reduzierbaren Transitionsmenge G mit dem Reduktionsvektor v eines Petri-Netzes N sind entweder gar nicht in einer T -Invariante i von N enthalten oder in einer durch ein Vielfaches von v gewichteten Anzahl.

Fasst man die Transitionen aus G zu einer neuen Transition r zusammen, sind die Auswirkungen auf die Inzidenzmatrix zu untersuchen.

Definition 4.3.4 (Matrixreduktion) Sei $N = (P, T, F, m_0)$ ein Petri-Netz, G eine reduzierbare Transitionsmenge von N , v ihr Reduktionsvektor und r eine neue Transition, die die Transitionen aus G in N ersetzt.

Aus der Inzidenzmatrix C von N entsteht die neue Matrix C' , indem eine neue Spalte \vec{r} für r eingefügt wird und alle Spalten für Transitionen aus G gelöscht werden. Die einzelnen Einträge ergeben sich aus den Kanten zwischen Transitionen aus G und den Plätzen von N und deren Kantenbewertungen:

$$\vec{r}[p] = \sum_{t \in G} F(t, p) \cdot v[t] - \sum_{t \in G} F(p, t) \cdot v[t]. \quad (4.3)$$

Lemma 4.3.5 Sei $N = (P, T, F, m_0)$ ein Petri-Netz mit der T -Invariantenmenge I und der Inzidenzmatrix C . Sei G eine reduzierbare Transitionsmenge mit dem Reduktionsvektor v , r die neue Transition für G und C' die aus der Inzidenzmatrix C des Petri-Netzes durch die Matrixreduktion 4.3.4 entstandene Inzidenzmatrix.

Dann sind N und jedes Petri-Netz $N' = (P, T', F', m'_0)$ mit

$$\begin{aligned} T' &= T \setminus G \cup \{r\} \\ F' &= F \setminus \{(u, v) \mid u \in G \vee v \in G\} \cup \{(p, r) \mid p \in P\} \cup \{(r, p) \mid p \in P\} \end{aligned}$$

und einer Kantenbewertung F' , so dass C' der Inzidenzmatrix von N' entspricht, T -Invarianten-äquivalent.

Beweis

Sei I' die Menge der minimalen T-Invarianten von N' . Zu zeigen ist die T-Invariantenäquivalenz $N \succeq N'$ vermöge f und h . Dazu werden die zwei Funktionen $f : I \rightarrow I'$ und $h : T \rightarrow T'$ folgendermaßen definiert:

$$f(i) = j \text{ mit } j[t] = \begin{cases} i[t] & \text{falls } t \in T \setminus G \\ k & \text{falls } t = r \text{ und } \forall g \in G \text{ gilt } (i - k \cdot v)[g] = 0 \end{cases} \quad (4.4)$$

$$h(t) = \begin{cases} t & \text{falls } t \in T \cap T' \\ r & \text{falls } t \in G \end{cases} . \quad (4.5)$$

Ist f bijektiv, so ist die Bedingung 1 der Definition 4.3.2 erfüllt. Aus der Definition von h folgt, dass h total und surjektiv ist. Damit ist noch zu zeigen:

1. $i \in I \Rightarrow f(i) \in I'$
2. f ist bijektiv:
 - (a) $\forall i, j \in I, i \neq j : f(i) \neq f(j)$
 - (b) $\forall j \in I' \exists i \in I : f(i) = j$
3. $\forall i \in I \forall t \in T : i[t] = 0 \Leftrightarrow (f(i))[h(t)] = 0$.

Zu 1: Sei $i \in I$. Da G eine reduzierbare Transitionsmenge ist, muss es für jedes i ein $k \in \mathbb{N}$ mit $(i - k \cdot v)[t] = 0$ geben.

$$\begin{aligned} C \cdot i = 0 & \Rightarrow \\ \forall p \in P : 0 & = \sum_{t \in T} C[p][t] \cdot i[t] \\ & = \sum_{t \in T \setminus G} C[p][t] \cdot i[t] + \sum_{t \in G} C[p][t] \cdot i[t] \\ & = \sum_{t \in T \setminus G} C[p][t] \cdot i[t] + \sum_{t \in T \setminus G} F(t, p) \cdot i[t] - \sum_{t \in T \setminus G} F(p, t) \cdot i[t] \\ & = \sum_{t \in T \setminus G} C[p][t] \cdot i[t] + \sum_{t \in T \setminus G} F(t, p) \cdot k \cdot v[t] - \sum_{t \in T \setminus G} F(p, t) \cdot k \cdot v[t] \\ & \stackrel{(4.3)}{=} \sum_{t \in T \setminus G} C'[p][t] \cdot i[t] + C'[p][r] \cdot k \\ & \Rightarrow C' \cdot f(i) = 0 \end{aligned}$$

Zu 2a: Zwei voneinander verschiedene minimale T-Invarianten i und j unterscheiden sich an mindestens einer Stelle im entsprechenden Transitionsvektor. Sei t eine Transition, an deren Stelle sich die beiden T-Invarianten unterscheiden, also $i[t] \neq j[t]$. Ist $t \in T \setminus G$, so ist nach Definition 4.4 $f(i)[t] = i[t] \neq j[t] = f(j)[t]$. Andernfalls ist $t \in G$. Da es für jedes $x \in I$ ein $k_x \in \mathbb{N}$ mit $(x - k_x \cdot v)[t] = 0$ und $i[t] \neq j[t]$ gibt, muss $k_i \neq k_j$ sein. Nach Definition 4.4 ist damit $f(i)[r] = k_i \neq k_j = f(j)[r]$ und somit gilt auch $f(i) \neq f(j)$.

Zu 2b: Sei j in I' und $k = j[r]$. Sei i ein Transitionsvektor von N mit

$$i[t] = \begin{cases} j[t] & \text{falls } t \in T \setminus G \\ k \cdot v[t] & \text{falls } t \in G \end{cases} \quad (4.6)$$

Dann ist nach Definition 4.4 $f(i) = j$. Der Transitionsvektor i ist eine T-Invariante von N :

$$\begin{aligned} \forall p \in P: \quad (C \cdot i)[p] &= \sum_{t \in T} C[p][t] \cdot i[t] \\ &= \sum_{t \in T \setminus G} C[p][t] \cdot i[t] + \sum_{t \in G} C[p][t] \cdot i[t] \\ &\stackrel{(4.6)}{=} \sum_{t \in T \setminus G} C[p][t] \cdot i[t] + k \cdot \left(\sum_{t \in G} C[p][t] \cdot v[t] \right) \\ &= \sum_{t \in T \setminus G} C[p][t] \cdot i[t] + k \cdot \left(\sum_{t \in G} (F(t, p) - F(p, t)) \cdot v[t] \right) \\ &\stackrel{(4.3)}{=} \sum_{t \in T \setminus G} C'[p][t] \cdot j[t] + k \cdot C'[p][r] \\ &= (C' \cdot j)[p] = 0 \\ &\Rightarrow \forall p \in P: \quad (C \cdot i)[p] = 0 \\ &\Rightarrow C \cdot i = 0 \end{aligned}$$

Zu 3: Nach Definition 4.4 ist für jede T-Invariante i und jede Transition t die zweite Bedingung aus 4.3.2 erfüllt. \square

Werden die echten strong DTS auf Basis aller minimalen T-Invarianten bestimmt, entsprechen sie gerade den reduzierbaren Transitions Mengen. Jeder Transitionsvektor v aus Definition 3.2.2 entspricht einem Reduktionsvektor.

Die häufigkeitsbasierte Reduktionsregel führt gerade eine Matrixreduktion durch, wie sie in 4.3.4 beschrieben wurde. Bei der Reduktion mit Regel 4.3.1 werden die Kantenbewertungen analog zu Definition 4.3.4 eingeführt. Der einzige Unterschied zwischen Matrixreduktion und häufigkeitsbasierter Reduktionsregel ist die Entfernung von Plätzen bei der Reduktion.

Entfernt werden dabei nur innere Plätze. Der Effekt der reduzierten Transitionen auf innere Plätze ist laut Lemma 3.2.3 null. Daraus folgt, dass die Einträge der Spalte zu der neuen Transition r in Zeilen zu inneren Plätzen während der Matrixreduktion auf null gesetzt werden. Alle anderen Einträge sind bereits null, denn nur die reduzierten Transitionen waren mit inneren Plätzen verbunden. Die Spalten für die reduzierten Transitionen wurden aber entfernt. Damit ist die zu einem inneren Platz gehörende Zeile der Matrix null. Es gibt, wie schon in den Vorüberlegungen zur häufigkeitsbasierten Reduktionsregel erläutert, zwei Interpretationsmöglichkeiten für einen Null-eintrag in der Matrix. Entweder die Differenz aus abgelegten und entnommenen Marken ist null und es entsteht eine Testkante oder beide Kanten sind mit null bewertet. Da in der häufigkeitsbasierten Reduktion immer der letzte Fall angenommen wird, kann der Platz entfernt werden. Auf die Menge der T-Invarianten hat das Entfernen eines solchen Platzes keinen Einfluß.

Damit kann jedes Petri-Netz, zu dem mehrelementige echte strong-DTS existieren, in ein Petri-Netz mit weniger Transitionen überführt werden. Das reduzierte Petri-Netz N' hat eine äquivalente Menge von T-Invarianten wie das ursprüngliche Petri-Netz N . Ist zu jeder neuen Transition r in N' der Reduktionsvektor v bekannt, kann jede T-Invariante aus N mit der Definition 4.6 eindeutig in eine T-Invariante des Petri-Netzes N' überführt werden. Den umgekehrten Weg leistet gerade die Funktion f (4.4).

Die Anzahl der Plätze kann sich bei der Reduktion des Petri-Netzes ebenfalls verringern. Das Verfahren kann auf mehrere reduzierbare Transitionsmengen angewendet werden. Die Reihenfolge der Anwendung spielt dabei keine Rolle.

Beschränktheit

Ein Petri-Netz N ist beschränkt, wenn alle seine Plätze beschränkt sind. Um die Auswirkungen der Reduktion auf ein beschränktes Petri-Netz zu untersuchen, wird der Erreichbarkeitsgraph $EG(N)$ benutzt. Der Erreichbarkeitsgraph von N' ist gegenüber dem Erreichbarkeitsgraphen von N höchstens kleiner, da einige Zustände nicht mehr erreichbar sind. Die Transitionen aus der Menge der reduzierten Transitionen A können in N unabhängig voneinander schalten. In N' schaltet anstelle der einzelnen Transitionen aus A nur noch die neue Transition r . Alle Zustandsübergänge, die das Schalten einer Transition aus A verursacht, entfallen. Dafür kommen Kanten mit der Transition r als Beschriftung hinzu. Genauer wirkt sich diese Veränderung folgendermaßen aus:

- Alle Kanten, die mit Transitionen aus A beschriftet sind, entfallen.
- Für eine Menge von Pfaden τ in $EG(N)$, dessen Transitionsvektor dem Reduktionsvektor v entspricht und deren Anfangs- und Endzustand jeweils gleich sind, wird eine neue Kante vom Anfangszustand des Pfades zum Endzustand mit der Bewertung r eingefügt.

Wird ein unbeschränktes Petri-Netz reduziert, kann das entstehende Netz beschränkt sein, denn durch die Reduktion können alle unbeschränkten Plätze entfernt werden. Ein Beispiel ist das Petri-Netz von Abbildung 4.5(a). Es besitzt zwei T-Invarianten, $\{t1, t2\}$ und $\{t3, t4\}$. Werden die beiden Transitionen $t3$ und $t4$ häufigkeitsbasiert zur Transition $DTS1$ reduziert, wird der einzige unbeschränkte Platz $p3$ des Netzes gelöscht, so dass das Ergebnisnetz 4.5(b) beschränkt ist.

Damit kann die Beschränktheit eines Petri-Netzes N durch die häufigkeitsbasierte Reduktion zum Netz N' nicht verlorengehen, aber ein unbeschränktes Netz kann durch die Reduktion beschränkt werden.

Realisierbarkeit

Die Realisierbarkeit einer T-Invariante hängt von der aktuellen Markierung des Netzes ab. Bei der häufigkeitsbasierten Reduktion entsteht ein Petri-Netz mit äquivalenten T-Invarianten und auch Markierungen bleiben erhalten. Damit stellt sich die Frage, ob auch die Realisierbarkeit der T-Invarianten erhalten bleibt.

Am Beispiel 4.6 ist erkennbar, dass die Realisierbarkeit der T-Invarianten bei der häufigkeitsbasierten Reduktion im Allgemeinen nicht erhalten bleibt. In allen minimalen T-Invarianten dieses Petri-Netzes kommen die Transitionen t_1 , t_2 und t_3 gemeinsam vor. Da auch die Häufigkeiten immer im gleichen Verhältnis stehen, bilden diese Transitionen die strong DTS $DTS1$. Wird $DTS1$ reduziert, entsteht das Petri-Netz in Abbildung 4.6(b).

Die T-Invariante $inv_1 = \{t_1, t_2, t_3, t_4(2), t_6\}$ ist im ursprünglichen Petri-Netz 4.6(a) unter der leeren Markierung realisierbar, im reduzierten Petri-Netz 4.6(b) ist die zu inv_1 äquivalente T-Invariante $inv'_1 = \{DTS1, t_4(2), t_6\}$ unter der leeren Markierung nicht realisierbar.

Lebendigkeit

Zusammen mit der Realisierbarkeit der T-Invarianten kann auch die Lebendigkeit des Petri-Netzes mit der häufigkeitsbasierten Reduktion verloren gehen. Das Petri-Netz aus Abbildung 4.6(a) ist lebendig, das Reduktionsergebnis nicht mehr.

Folgerungen

Bei der Reduktion eines Petri-Netzes mit Hilfe der strong DTS können redundante Informationen bezüglich der DTS aus dem Netz entfernt werden. Bleiben die Invarianten des Netzes bei der Reduktion erhalten, ist das Resultat ein Netz, das in seiner Kernfunktionalität das gleiche Verhalten wie das ursprüngliche Netz aufweist. Alle Transitionsmengen, die nur in ihrer Gesamtheit in Invarianten vorkommen, sind zu einer neuen Transition zusammengefasst. Daher ist je nach Grad der Reduktion der Anteil der Knoten, die einen echten Beitrag zur Vielseitigkeit des beschriebenen Systems beitragen, im Vergleich zur Gesamtanzahl der Knoten im Netz höher. Damit erhöht sich auch die Informationsdichte des Netzes. Grundlegende Abläufe, die durch die Invarianten beschrieben werden, sind leichter zu identifizieren.

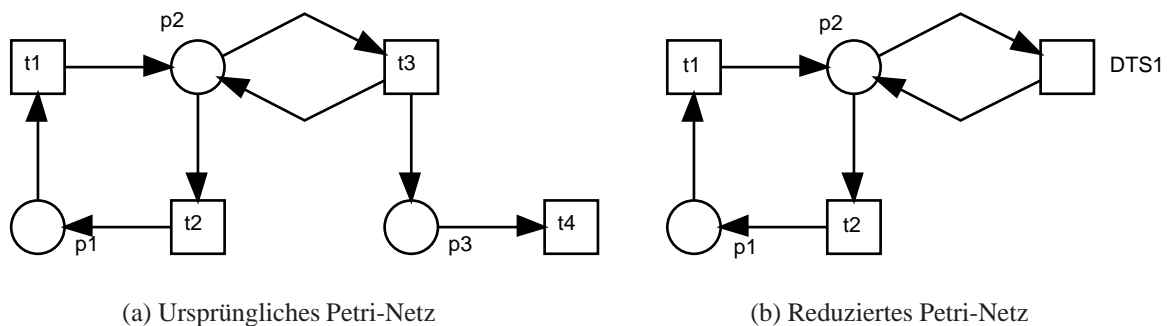


Abbildung 4.5: Beispiel für ein Petri-Netz, das nach der häufigkeitsbasierten Reduktion beschränkt ist

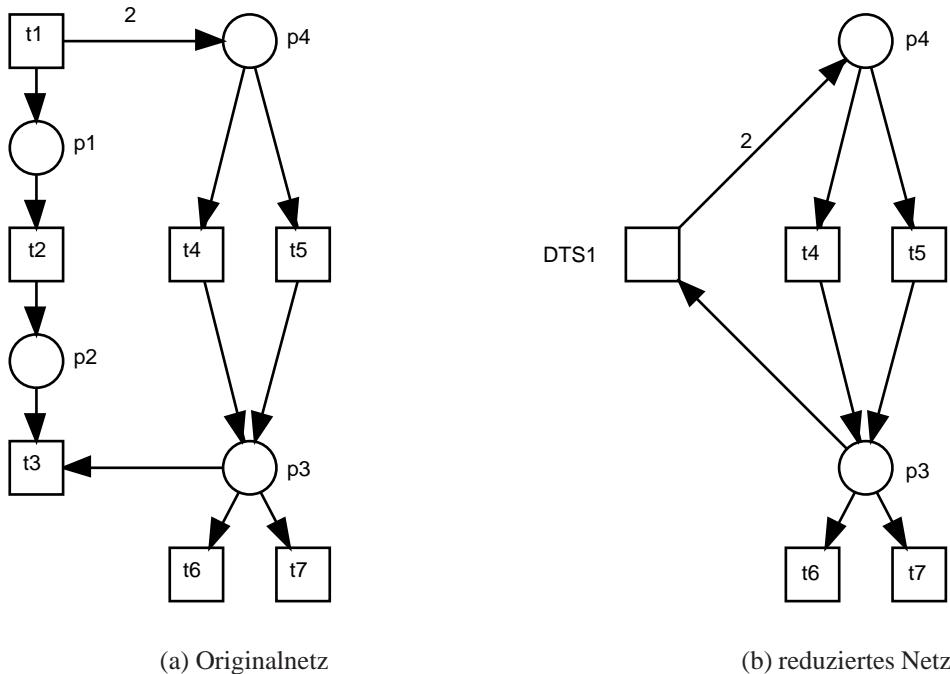


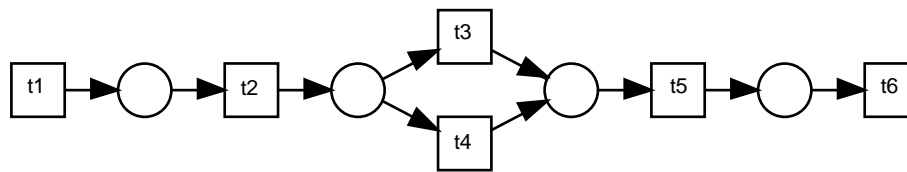
Abbildung 4.6: Beispiel für Verlust der Realisierbarkeit bei der Reduktion

Die vorgestellte häufigkeitsbasierte Reduktionsregel erfasst zudem größere Teilnetze als die Regeln in [Sta90]. Die Transitionen t_2 , t_3 und t_4 des Petri-Netzes aus Abbildung 4.3 können mit Hilfe der Regeln aus [Sta90] nicht zu einer Transition reduziert werden.

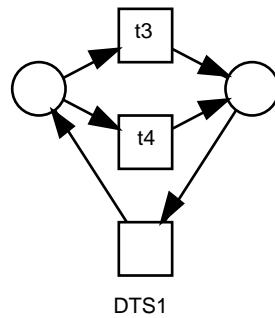
4.4 Einschränkungen der Reduktionsregeln

Laut den bisherigen Ausführungen spricht nichts dagegen, jeweils nur maximale DTS zu reduzieren. Abgesehen von der Anforderung, dass sich alle reduzierten Transitionen in einer abstract oder strong DTS befinden müssen, sind aber noch andere Einschränkungen bei der Auswahl möglich. Es kann zum Beispiel sinnvoll sein, nur solche Transitionen zusammenzufassen, die ein zusammenhängendes Teilnetz induzieren. Im Normalfall ist davon auszugehen, dass die Übersichtlichkeit des Petri-Netzes unter einer Zusammenfassung von Transitionen leidet, die nicht zusammenhängende Teilnetze induzieren. Ein Beispiel zeigt Abbildung 4.7(a). Hier könnten aufgrund der Zuordnung von t_1, t_2, t_3 und t_4 zu einer strong DTS alle diese Transitionen zu einer Transition $DTS1$ zusammengefasst werden.

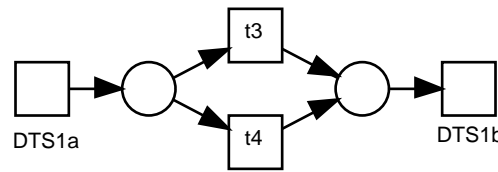
Das Ergebnis der Anwendung der Reduktionsregeln wäre dann das Petri-Netz aus Abbildung 4.7(b). Es ähnelt nur noch wenig dem ursprünglichen Netz, vor allem wurde nun ein Kreis im Petri-Netz geschlossen, der vorher nicht vorhanden war. Eine Reduktion wie in Abbildung 4.7(c),



(a) Originalnetz



(b) alle reduziert



(c) nur zusammenhängende reduziert

Abbildung 4.7: Beispiel für Reduktion zusammenhängender Teilnetze

bei der nur Transitionen reduziert werden, deren induziertes Teilnetz zusammenhängend ist, kann dies verhindern.

Eine weitere sinnvolle Einschränkung ist die ausschließliche Anwendung der Reduktion auf Transitionen, die nicht Randtransitionen sind. Randtransitionen spielen häufig eine wichtige Rolle, da sie Marken erzeugen aus dem Netz abführen können. Auf Grund dieser wichtigen Rolle sollten in manchen Situationen Randtransitionen nicht in einer zu reduzierenden Transitionsmenge enthalten sein.

Als Beispiel kann das Petri-Netz aus Abbildung 4.6 dienen. Hier wurden die beiden einzigen Randtransitionen reduziert. Dadurch wurde erstens wie in Beispiel 4.7(c) ein Kreis geschlossen. Zweitens ist am reduzierten Netz nicht mehr ersichtlich, dass im ursprünglichen Netz Marken aus der Umgebung in das Netz gespeist und wieder entfernt werden können.

Kapitel 5

Fallstudie

Petri-Netze werden zur Modellierung verschiedenster Systeme eingesetzt. Bei der Validation der Modelle werden häufig T-Invarianten benutzt. Diese können sehr zahlreich sein, so dass Methoden erforderlich sind, sie systematisch und möglichst automatisch darzustellen und zentrale Informationen zu extrahieren.

Nachdem in den letzten beiden Kapiteln verschiedene Möglichkeiten der T-Invariantenauswertung diskutiert wurden, wird in diesem Kapitel anhand einer konkreten Fallstudie gezeigt, welche Ergebnisse die Methoden liefern und wie diese verwendet werden können. Dazu wird auf ein bereits vorhandenes Petri-Netz-Modell zurückgegriffen. Ziel ist es, die Methoden, ihre Einsatzfähigkeit und ihren praktischen Nutzen anhand des Beispiels darzustellen und die Ergebnisse zu validieren.

5.1 DTS in biochemischen Netzwerken

Biochemische Netzwerke werden zunehmend mit Petri-Netzen modelliert. Eine wichtige Rolle beim Erstellen und Auswerten solcher Modelle spielen die T-Invarianten. Am Beispiel eines biochemischen Netzwerkes von geringerem Umfang sollen die in dieser Arbeit vorgestellten Methoden exemplarisch auf die T-Invarianten des Netzwerkes angewendet werden.

5.1.1 Grundlagen

Die chemischen Reaktionen der Stoffwechselprozesse in Lebewesen müssen ständig unter nahezu konstanter Temperatur ablaufen. Das bedeutet, dass die meisten Reaktionen von Enzymen katalysiert und reguliert werden. Die hier betrachteten metabolischen Netzwerke bestehen aus vielen einzelnen atomaren chemischen Reaktionen, die zusammen ein komplexes Netzwerk bilden. Eine atomare Reaktion ist dabei bestimmt durch ihre Eingangsstoffe, ihre Ausgangsstoffe und deren stochiometrisches Verhältnis. Sie wandelt ihre Eingangsstoffe im angegebenen Verhältnis in die Ausgangsstoffe um.

In diesem Netzwerk finden metabolische Stoffwechselprozesse statt, beschrieben durch die beteiligten atomaren Reaktionen, die eine Menge von Eingangsstoffen über Zwischenprodukte in eine Menge von Ausgangsstoffen umwandeln. Solch ein Stoffwechselprozess bildet ein zusammenhängendes, eventuell verzweigtes (Teil-)Netzwerk.

In den Zellen, in denen die Stoffwechselprozesse ablaufen, werden Stoffe, die essentiell für den Lebenserhalt sind, als primäre Stoffe bezeichnet. Bestimmte Gruppen von Molekülen, die sekundären Stoffe, sind nicht essentiell für den Lebenserhalt und liegen meist in einer sehr hohen Konzentration vor. Sie werden auch als allgegenwärtige Moleküle (engl. ubiquitous molecules) bezeichnet. Außerdem unterscheidet man begrifflich die sogenannten Zwischenprodukte und die Ein- und Ausgangsstoffe des Stoffwechselprozesses. Letztere werden dem Prozess von außen zu- oder abgeführt, alle anderen Stoffe gelten als Zwischenprodukte.

Metabolische Netzwerke werden für eine qualitative Analyse im sogenannten Fließgewicht (eng. steady state) modelliert. Es wird davon ausgegangen, dass die Konzentrationen der Zwischenprodukte im Fließgewicht konstant sind, nur die Konzentrationen der Ein- und Ausgangsstoffe ändern sich (siehe [HK04] und [PPW⁺03]). Bei der Modellierung mit einfachen Petri-Netzen werden Reaktionsraten nicht betrachtet und Enzyme nicht mitmodelliert. Es wird angenommen, dass die für eine Reaktion benötigten Enzyme in ausreichender Menge vorhanden und aktiv sind.

Chemische Reaktionen lassen sich direkt mit Petri-Netzen modellieren. Jeder Platz entspricht einem Stoffwechselprodukt. Die Anzahl der Marken spiegelt die aktuelle Konzentration des entsprechenden Stoffes wider. Die chemischen Reaktionen werden mit Transitionen dargestellt. Jede Transition entspricht einer Reaktionsrichtung, das heißt, reversible Reaktionen werden mit zwei Transitionen, irreversible Reaktionen mit einer Transition modelliert.

Auch die Unterscheidung zwischen Ein-, Ausgangsstoffen und Zwischenprodukten kann mit Hilfe von Petri-Netz-Mitteln direkt umgesetzt werden.

Auf die genaue Beschreibung der Methodik zum Entwickeln eines metabolischen Netzwerkes wird in [Run04] und in [KJH05] eingegangen und ist nicht Gegenstand dieser Arbeit. Wohl aber ist die Auswertung der T-Invarianten solcher Petri-Netze Teil der Methodik. Deshalb wird im folgenden Abschnitt die Bedeutung der T-Invarianten für Petri-Netz-Modelle biochemischer Netzwerke genauer erläutert.

5.1.2 Biologische Interpretation der T-Invarianten

Ist ein metabolisches Netzwerk durch ein Petri-Netz modelliert, bestimmt zum Beispiel das Tool INA (siehe [Sta03]) die minimalen T-Invarianten des Netzwerkes. Im biochemischen Kontext werden diese meist als „Elementarmoden“ bezeichnet, obwohl deren Definition nicht exakt mit der Definition von T-Invarianten übereinstimmt. Sie beschreiben minimale Reaktionspfade im biochemischen Netzwerk, das heißt, sie beschreiben eine minimale Menge von Reaktionen beziehungsweise den zugehörigen Enzymen, die im Fließgewicht arbeiten können. Schalten alle Transitionen einer T-Invariante hintereinander, stellen sie die Markierung des Petri-Netzes wieder her. Das ent-

spricht damit gerade dem Beibehalten des Fließgewichtes. Jedes Verhalten des Systems kann als Linearkombination der minimalen T-Invarianten beschrieben werden [KJH05].

Mit Hilfe der T-Invarianten kann das Petri-Netz gegenüber dem modellierten System validiert werden [HK04]. Jede T-Invariante sollte ein biologisch sinnvolles Verhalten widerspiegeln. Weiterhin sollte sich auch jedes biologisch relevante Verhalten im Petri-Netz und damit in einer T-Invariante wiederfinden. Außerdem wird mit Hilfe der Invariantenüberdeckung geprüft, ob das Modell widerspruchsfrei ist. Eine Transition, die in keiner minimalen T-Invariante enthalten ist, weist auf einen schlecht gewählten Ausschnitt des Gesamtsystems oder einen Fehler im Modell hin. Eine solche Transition kann aber auch ein Indiz für eine Krankheit des modellierten Systems sein [Run04].

Die Anzahl der minimalen Invarianten in Petri-Netzen ist schlimmstenfalls exponentiell im Bezug zur Anzahl der Knoten im Netzwerk. Praktische Studien haben jedoch gezeigt, dass die Anzahl der minimalen Invarianten in Modellen biochemischer Netzwerke häufig weit unter dieser oberen Grenze liegt. Einige Gründe dafür werden in [KS02] aufgeführt. Deshalb erweist sich die Auswertung der Invarianten als effektive Möglichkeit, Petri-Netz-Modelle für biochemische Systeme qualitativ und quantitativ zu untersuchen.

Trotzdem nimmt mit der Komplexität des modellierten Systems auch die Komplexität des Modells zu, so dass die Anzahl der berechneten Invarianten leicht über ein manuell analysierbares Maß hinausgeht. Verfahren, um automatisch bestimmte Informationen aus der Menge der minimalen T-Invarianten eines biochemischen Netzwerkes zu gewinnen, sind in der Literatur kaum zu finden.

Ein Beispiel ist die Bestimmung von minimalen Schnittmengen (engl. minimal cut set, MCS) der T-Invarianten in [KG04]. Eine minimale Schnittmenge ist eine minimale Menge von chemischen Reaktionen eines Netzwerkes, deren Inaktivität bestimmte Funktionen des Netzwerkes definitiv unterbindet. Definitiv bedeutet hier, dass die Funktion nicht eventuell durch andere, die inaktiven Reaktionen ausgleichende Reaktionen gewährleistet werden kann.

Es existieren außerdem Ansätze, mit Hilfe von Clustering-Verfahren die Menge der minimalen T-Invarianten biologisch sinnvoll einzuteilen.

Ansätze zu der in dieser Arbeit definierten DTS und der vorgestellten automatisierbaren Bestimmung der DTS aus den T-Invarianten wurden schon in [Sac05] genannt. Dort spielen jedoch die quantitativen Informationen keine Rolle. In [BNSM04] wurden zwar die quantitativen Informationen in die Betrachtungen einbezogen, jedoch werden die DTS im Wesentlichen dazu benutzt, eine Dekompositionierung durchzuführen, um das biochemische Netzwerk flexibler zu gestalten.

Neu ist die ausführliche Untersuchung der DTS und die Nutzung der quantitativen Informationen für eine Hierarchisierung oder eine Reduktion des Petri-Netzes. Die mögliche biologische Bedeutung der DTS wird im folgenden Abschnitt untersucht.

5.1.3 Biologische Bedeutung der DTS

Zunächst sind abstract und strong DTS im biologischen Sinn Mengen von chemischen Reaktionen, die in allen das Fließgewicht erhaltenden metabolischen Prozessen eines biochemischen Netzwerkes gemeinsam vorkommen. Sie können als kleinste funktionale Einheiten interpretiert werden, die eine biologische Bedeutung besitzen [Sac05].

Die erste Schlussfolgerung aus dieser Tatsache betrifft Methoden, die vor allem das Unterbinden bestimmter Prozesse im biochemischen Netzwerk zum Ziel haben. Fehlt ein Enzym, das eine der chemischen Reaktionen einer DTS katalysiert, so können auch die anderen Reaktionen der DTS nicht mehr unter dem Fließgewicht stattfinden.

Dies ist ein Mehrwert an Information gegenüber den minimalen T-Invarianten allein. Es ist bekannt, dass das Unterdrücken einer chemischen Reaktion alle minimalen Prozesse des Netzwerkes im Fließgewicht unterbindet, in denen diese chemische Reaktion enthalten ist. Wäre dies nicht der Fall, wäre der Prozess nicht minimal. Ist die unterbundene Reaktion t in einer mehrelementigen DTS enthalten, so kann statt der Reaktion t auch jede andere Reaktion der DTS unterbunden werden und das gleiche Ziel wird erreicht. Hier wird ein Zusammenhang zu den minimalen Schnittmengen aus [KG04] deutlich. Ist eine der chemischen Reaktionen t einer MCS in einer mehrelementigen DTS, so kann sie in der MCS durch jede andere Reaktion der DTS ersetzt werden.

Da die Reaktionen einer DTS immer gemeinsam in allen T-Invarianten vorkommen, bietet sich eine Zusammenfassung der DTS an. In dieser Arbeit wurden verschiedene Methoden vorgestellt, das Petri-Netz zu reduzieren. Die DTS bilden die Grundlage dieser Reduktionen.

Die Supportbasierte Reduktion auf Basis der abstract DTS bietet die Möglichkeit, das Modell des biochemischen Netzwerkes rein qualitativ in disjunkte Teilnetze zu zerlegen und von konkreten Häufigkeiten abstrahierend einen Überblick über in engem Zusammenhang stehende chemische Reaktionen zu gewinnen.

Ergänzend dazu kann die Verwendung der häufigkeitsbasierten Reduktion mit den strong DTS das Netz unter Beibehaltung der quantitativen Informationen vereinfachen. Dadurch können zentrale Prozesse im biochemischen Netzwerk leichter erkannt und nachvollzogen werden.

Doch nicht nur das Petri-Netz-Modell des biochemischen Netzwerkes, sondern auch die T-Invariantenmenge selbst kann mit Hilfe der DTS vereinfacht beziehungsweise komprimiert dargestellt werden. Das kann zum Beispiel bei der Berechnung der MCS auf Basis der minimalen T-Invarianten hilfreich sein und die Rechenzeit verkürzen.

5.2 Fallstudie: Glycolyse

Der Glycolyse-Weg (kurz: Glycolyse) ist einer der wichtigsten Stoffwechselprozesse der meisten Organismen. Dieser biochemische Abbauweg beschreibt die Umwandlung von Glucose in Pyruvat unter Energiefreisetzung. Die Energie wird unter anderem in Form von ADP (Adenosine

diphosphate) und ATP (Adenosine triphosphate) gespeichert. Die betrachteten Reaktionen finden im Zytoplasma der Zellen statt. Abbildung 5.1 zeigt das hier untersuchte Modell des Glycolyse-Wegs, das auf [Run03] basiert. Alle beteiligten Stoffe und Reaktionen sowie eine ausführliche Beschreibung sind in [Run03] und [Run04] zu finden. In dieser Arbeit wird nur kurz auf die biologische Bedeutung des Modells eingegangen, da die Struktur, vor allem im Bezug auf die DTS, rein technisch untersucht wird.

Primärer Eingangsstoff des Modells ist die Glucose, die primären Ausgangsstoffe sind Pyruvat und Lactat. Alle beteiligten Reaktionen des modellierten Glycolyse-Wegs sind als irreversibel modelliert, das heißt mit nur einer Transition. Dies ist zulässig, da einige Reaktionen des Glycolyse-Wegs irreversibel sind und damit die Reaktionsrichtung vorgegeben ist. Es gibt im Petri-Netz-Modell der Glycolyse zwei Konflikte: die Marken auf dem Platz *BPS* können sowohl von Transition *r7* als auch von *r7a* entnommen werden, das gleiche gilt für den Platz *Pyr* und die Transitionen *r11* und *r_Pyr*.

Konflikte sind in biochemischen Netzwerken bezüglich der T-Invarianten interessante Strukturen: hier kann eine Marke meist verschiedene Wege durchlaufen. Erst durch Konflikte und die entsprechende Kombinatorik werden T-Invarianten relevant für die Analyse. Ist für jede Marke jeweils nur ein einziger Weg durch das Netzwerk möglich, sind auch die dazugehörigen T-Invarianten verhältnismäßig uninteressant für die Auswertung.

Das Petri-Netz-Modell der Glycolyse ist transitionsberandet. Die vier T-Invarianten des Modells sind in Tabelle 5.1 festgehalten.

Der Algorithmus 3.1 liefert fünf abstract DTS. Diese werden in Algorithmus 3.2 nicht noch einmal aufgespaltet, da die Häufigkeiten, mit denen die Transitionen in den T-Invarianten auftreten, schon den Anforderungen der strong DTS genügen.

Tabelle 5.2 listet die maximalen strong DTS des Glycolyse-Petri-Netzes auf. Da der Reduktionsvektor einer DTS bei realen Netzen nur sehr wenige nicht-null-Einträge enthält, wird im Folgenden in Klammern hinter jeder Transition einer DTS ihr Eintrag im Reduktionsvektor festgehalten. Alle anderen Einträge im Reduktionsvektor sind null.

In der strong DTS *DTS1* befinden sich zwei Transitionsmengen, die jeweils ein zusammenhängendes Teilnetz induzieren. Alle anderen strong DTS des untersuchten Netzwerkes induzieren zusammenhängende Teilnetze.

Da die Transitionsmengen der abstract DTS mit den Transitionsmengen der strong DTS übereinstimmen, kann das Glycolyse-Petri-Netz sofort mit der invariantenerhaltenden Reduktionsregel in eine komprimierte Form überführt werden. Dabei entsteht ein übersichtlicheres Petri-Netz, das bezüglich der T-Invarianten das gleiche Verhalten aufweist wie das ursprüngliche Petri-Netz des Glycolyse-Wegs. Abbildung 5.2 zeigt das komprimierte Petri-Netz.

Tabelle 5.3 listet die Transitionsmengen auf, die reduziert wurden. Bei der Reduktion des Glycolyse-Petri-Netzes wurden nicht alle maximalen DTS reduziert. Die Transitionsmenge von *DTS1* wurde in zwei Transitionsmengen aufgeteilt, die jeweils ein zusammenhängendes Teilnetz induzieren. Der Grund dafür ist die Kantenrichtung der Kanten zwischen den Transitionen und den

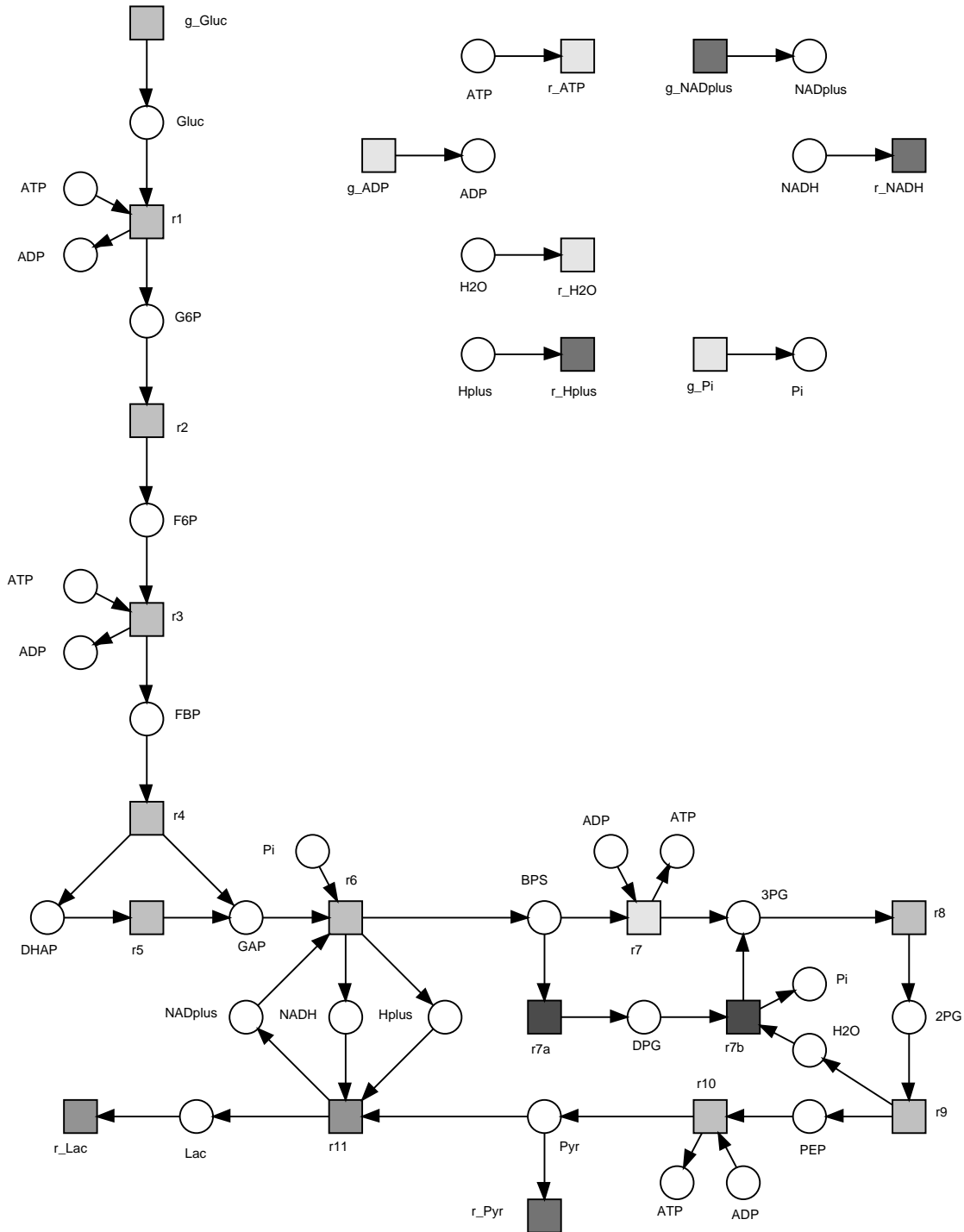


Abbildung 5.1: Glycolyse-Netzwerk

Transition	inv_1	inv_2	inv_3	inv_4
r7b	2	0	2	0
r7a	2	0	2	0
r2	1	1	1	1
r1	1	1	1	1
r3	1	1	1	1
r4	1	1	1	1
r6	2	2	2	2
r7	0	2	0	2
r8	2	2	2	2
r9	2	2	2	2
r10	2	2	2	2
g_Gluc	1	1	1	1
r_Lac	2	2	0	0
r_Pyr	0	0	2	2
g_ADP	0	2	0	2
r_ATP	0	2	0	2
r_H2O	0	2	0	2
r_Hplus	0	0	2	2
g_NADplus	0	0	2	2
g_Pi	0	2	0	2
r_NADH	0	0	2	2
r5	1	1	1	1
r11	2	2	0	0

Tabelle 5.1: Minimale T-Invarianten des Glycolyse-Netzwerkes

DTS1	=	{r1(1), r2(1), r3(1), r4(1), r5(1), r6(2), r8(2), r9(2), r10(2), g_Gluc(1)}
DTS2	=	{r7a(1), r7b(1)}
DTS3	=	{r7(1), g_ADP(1), r_ATP(1), r_H2O, g_Pi}
DTS4	=	{r_Lac(1), r11(1)}
DTS5	=	{r_Pyr(1), r_Hplus(1), g_NADplus(1), r_NADH(1)}

Tabelle 5.2: Maximale strong DTS des Glycolyse-Netzwerkes

DTS1a	=	{r1(1), r2(1), r3(1), r4(1), r5(1), g_Gluc(1), r6(2)}
DTS1b	=	{r8(1), r9(1), r10(1)}
DTS2	=	{r7a(1), r7b(1)}
DTS3	=	{r7(1), g_ADP(1), r_ATP(1), r_H2O, g_Pi}
DTS4	=	{r_Lac(1), r11(1)}
DTS5	=	{r_Pyr(1), r_Hplus(1), g_NADplus(1), r_NADH(1)}

Tabelle 5.3: Reduzierte strong DTS des Glycolyse-Netzwerkes

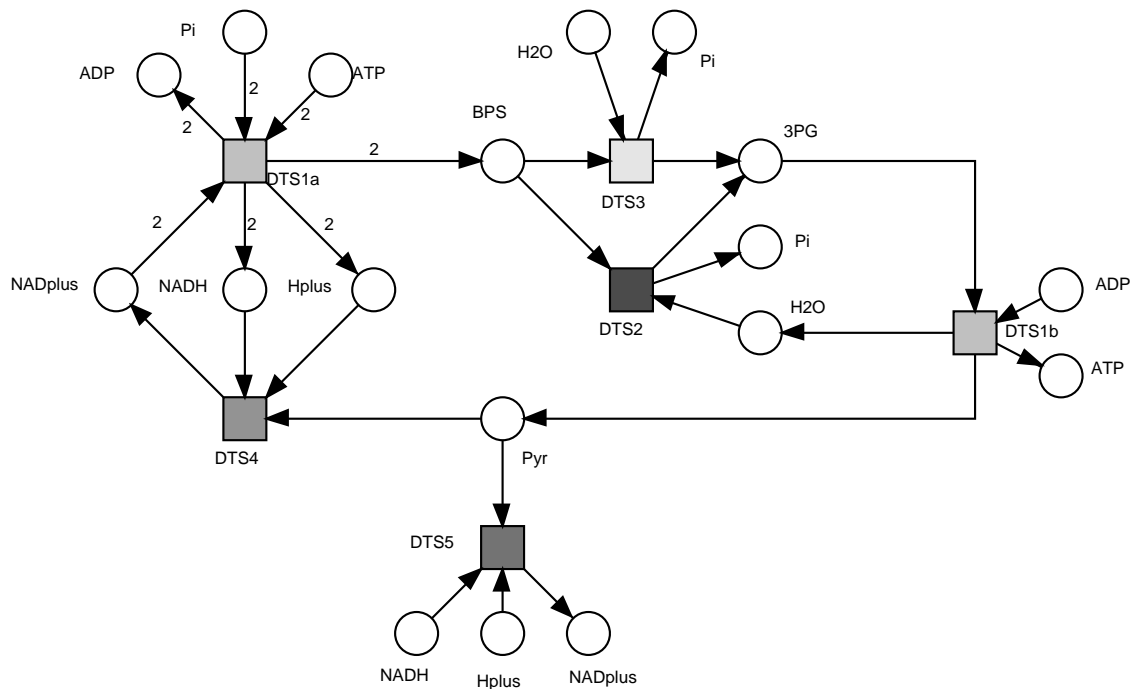


Abbildung 5.2: Reduziertes Glycolyse-Netzwerk

primären Stoffen. Diese Kanten definieren eine Halbordnung über den Knoten des Petri-Netzes. Wird die Transitionsmenge von $DTS1$ bei der Reduktion nicht aufgeteilt, entsteht im reduzierten Netz ein Kreis, der nur Kanten zwischen primären Stoffen und den Transitionen enthält. Die Reaktionsabfolge im Netz ist dann nicht mehr eindeutig zu erkennen, weil die entsprechenden Kanten keine Halbordnung mehr definieren. Trotz der Annahme, dass sich das gesamte Modell im Fließgewicht befindet, in dem alle Reaktionen gleichzeitig ablaufen, kann diese Halbordnung von Bedeutung sein. Deshalb wurde darauf verzichtet, $DTS1$ im Ganzen zu reduzieren.

Durch die Teilung von $DTS1$ kann der Reduktionsvektor für die Transitionen aus $DTS1b$ durchgehend auf eins gesetzt werden, da diese drei Transitionen in jeder T-Invariante zweimal schalten müssen. Der nur zu $DTS1b$ gehörende Transitionsvektor kann somit ganzzahlig durch zwei geteilt werden.

Die Transitionen, die die primären Ein- und Ausgangsstoffe und einige allgegenwärtige Moleküle des Glycolyse-Wegs zu- und abführen, wurden nicht von der Reduktion ausgenommen. Eine Reduktion, die alle oder ausgewählte Randtransitionen des Netzes nicht reduziert, ist jedoch ebenfalls möglich.

Im Netzwerk des Glycolyse-Wegs sind folgende Plätze Vor- und Nachplatz von Transitionen, die ausschließlich zu einer einzigen DTS gehören:

DPG, GAP, DHAP, FBP, F6P, G6P, Gluc, Lac, PEP, 2PG

Diese Plätze werden deshalb bei der mehrfachen Anwendung der Reduktionsregel 4.3.1 entfernt.

Transition	inv_1	inv_2	inv_3	inv_4	inv_5	inv_6	inv_7	inv_8	inv_9
DTS1a	1	1	1	1	1	1	1	1	1
DTS1b	2	2	2	2	2	2	2	2	2
DTS2	2	0	2	0	2	0	1	1	1
DTS3	0	2	0	2	0	2	1	1	1
DTS4	2	2	0	0	1	1	2	0	1
DTS5	0	0	2	2	1	1	0	2	1

Tabelle 5.4: Einfache T-Invarianten des reduzierten Glycolyse-Netzwerkes

Die minimalen T-Invarianten des komprimierten Petri-Netzes sind in Tabelle 5.4 aufgelistet. Aus diesen T-Invarianten lassen sich mit Hilfe der Reduktionsvektoren alle T-Invarianten des ursprünglichen Netzes ermitteln. In Tabelle 5.4 sind ebenfalls die einfachen T-Invarianten des reduzierten Petri-Netzes enthalten. Diese T-Invarianten ergeben sich folgendermaßen durch Kombination der minimalen T-Invarianten und ganzzahlige Teilung (siehe auch [Run04]):

$$\begin{aligned}
 inv_5 &= (inv_1 + inv_3)/2 \\
 inv_6 &= (inv_2 + inv_4)/2 \\
 inv_7 &= (inv_1 + inv_2)/2 \\
 inv_8 &= (inv_3 + inv_4)/2 \\
 inv_9 &= (inv_1 + inv_2 + inv_3 + inv_4)/4.
 \end{aligned}$$

Anhand des reduzierten Netzes können sehr viel leichter alle minimalen beziehungsweise alle einfachen T-Invarianten des Glycolyse-Modells nachvollzogen werden, als dies im ursprünglichen Netz der Fall ist. Finden unter Erhaltung des Fließgewichts Reaktionen des Glycolyse-Wegs statt, so muss dafür im reduzierten Modell die Transition *DTS1a* einmal schalten und legt zwei Marken auf dem Platz *BPS* ab. Da der Reduktionsvektor für *DTS1a* minimal gewählt ist, ist auch klar, dass die in *DTS1a* reduzierten Transitionen im Fließgewicht ebenfalls immer zwei Token produzieren, obwohl von Transition *r6* bei ihrem Schalten nur eine Marke auf *BPS* abgelegt wird. Im nicht reduzierten Petri-Netz ist erst auf den zweiten Blick klar, dass dies geschieht. Hier legt die Transition *r4* zwei Marken ab, so dass über *r5* eine weitere Marke auf Platz *GAP* gelangt.

Die zwei Marken auf Platz *BPS* können nun jeweils getrennt von Transition *DTS2* oder *DTS3* verbraucht werden. Die beiden primären Produkte dieser Transitionen, zwei Marken auf Platz *3PG*, werden beide von Transition *DTS1b* in zwei Marken auf *Pyr* umgewandelt. Diese beiden Marken können im reduzierten Modell jeweils von den Transitionen *DTS5* und *DTS4* konsumiert werden. Damit ergeben sich insgesamt neun bezüglich ganzzahliger Teilung minimale verschiedene Möglichkeiten für die zwei Marken, das Netz zu durchlaufen. Diese werden gerade durch die einfachen T-Invarianten beschrieben. Die neun Möglichkeiten existieren natürlich auch im ursprünglichen Netz, allerdings ist dort die Entstehung der neun einfachen T-Invarianten nicht so offensichtlich nachvollziehbar, da nicht sofort klar ist, dass im Fließgewicht zwei Marken den unteren Teil des Netzes durchlaufen müssen.

Kapitel 6

Zusammenfassung und Ausblick

Dieser Abschnitt fasst die Ergebnisse der Arbeit zusammenfassend und gibt einen Ausblick auf mögliche Weiterentwicklungen oder Verbesserungen der Konzepte.

6.1 Zusammenfassung

Invarianten spielen für die Analyse von Petri-Netz-Modellen in unterschiedlichen Anwendungen, insbesondere im Bereich Systembiologie, eine wichtige Rolle. Sie werden für das Finden relevanter Teilnetze und zum Überprüfen verschiedener Petri-Netz-Eigenschaften eingesetzt. Invarianten können unabhängig vom Erreichbarkeitsgraphen berechnet werden, so dass eine Analyse der Invarianten als rein statisches Verfahren auch auf nicht beschränkte Netze angewendet werden kann.

Es existieren bereits Verfahren, um eine Basis der Invarianten zu berechnen, die sogenannten minimalen Invarianten. Da die Anzahl der minimalen Invarianten und ähnlicher berechenbarer endlicher Teilmengen der Invarianten eines Netzes exponentiell im Vergleich zur Knotenzahl im Netz sein kann, ist eine manuelle Analyse schon bei relativ kleinen Netzen nicht mehr möglich.

In dieser Arbeit werden verschiedene automatisierbare Verfahren beschrieben, Mengen von T-Invarianten zu analysieren. Die Ergebnisse sind auf P-Invarianten übertragbar. Zunächst werden Standardverfahren und Ergebnisse aus dem Bereich Data-Mining untersucht. Für Teile dieser Ergebnisse, sogenannte abhängige Transitions Mengen, bietet sich eine weitere Verwendung an.

Es werden zwei Arten der abhängigen Transitions Mengen vorgestellt: die abstractly dependent transition sets (abstract DTS) und die strongly dependent transition sets (strong DTS). Abhängige Transitions Mengen kommen in bestimmten T-Invarianten des Petri-Netzes immer gemeinsam vor. Abstract DTS basieren auf dem Träger der Invarianten, das heißt die konkreten Häufigkeiten werden in die Berechnungen nicht einbezogen. Im Unterschied dazu werden für die Berechnung der strong DTS die Häufigkeiten der Transitionen in den T-Invarianten benutzt. Maximale abstract DTS bilden eine Zerlegung der Transitions Menge eines Petri-Netzes, die Zerlegung durch die strong DTS verfeinert diese Zerlegung im Allgemeinen nochmals.

Abhängige Transitions Mengen können für eine komprimierte Visualisierung von Invarianten Mengen mit finiten Automaten benutzt werden. Dank der abhängigen Transitions Mengen ist auch eine effiziente Datenrepräsentation von T-Invarianten mit binären Entscheidungsdiagrammen möglich. Algorithmen für die Berechnung der abhängigen Transitions Mengen werden vorgestellt.

Weiterhin bieten die auf den T-Invarianten basierten abhängigen Transitions Mengen die Möglichkeit, das zugrunde liegende Petri-Netz zu hierarchisieren und zu reduzieren.

Dazu werden anhand der abhängigen Transitions Mengen inhaltlich eng zusammenhängende Netzteile identifiziert. Sie können einer Hierarchisierung dienen, bei der die ermittelten Netzteile auf eine Hierarchieebene tiefer verschoben werden.

Im Vergleich zur Hierarchisierung, die die Struktur und damit die Semantik des Netzes nur optisch verändert, geht die Reduktion noch einen Schritt weiter. Mit zwei in dieser Arbeit vorgestellten Reduktionsregeln kann das Petri-Netz in eine reduzierte Form überführt werden. Die trägerbasierte Reduktionsregel reduziert Teilnetze, die durch die abstract DTS definiert werden. Die häufigkeitsbasierte Reduktionsregel benutzt dafür die durch strong DTS definierten Teilnetze.

Die Verwendung der trägerbasierten Reduktion kann dabei die Übersichtlichkeit des Petri-Netzes erheblich verbessern. Sie erhält aber die meisten Eigenschaften des Petri-Netzes nicht, da die Reduktion starke strukturelle Veränderungen nach sich zieht.

Werden für die häufigkeitsbasierte Reduktion die über allen minimalen T-Invarianten berechneten strong DTS benutzt, bleiben während der Reduktion die T-Invarianten des Petri-Netzes nachweislich erhalten. Damit kann zum Beispiel eine Vereinfachung eines metabolischen Netzwerks unter Beibehaltung wichtiger Abläufe erreicht werden. Andere Eigenschaften, zum Beispiel die Lebendigkeit oder die Unbeschränktheit, können bei der Reduktion verloren gehen.

Es hat sich gezeigt, dass die abhängigen Transitions Mengen in biochemischen Netzwerken eine biologische Bedeutung besitzen können. Abhängige Transitions Mengen sind biologisch interpretierbare funktionale Einheiten biologischer Netzwerke. Mit dieser Arbeit wurde die Grundlage geschaffen, die abhängigen Transitions Mengen auch strukturell zusammenzufassen.

6.2 Ausblick

Die in dieser Arbeit vorgestellten Techniken und Konzepte bilden die Grundlage für eine automatisierte Auswertung von Invarianten eines Petri-Netzes. Sie sind sowohl bei biochemischen Netzwerken als auch bei technischen Netzwerken anwendbar. Diese Techniken können noch weiterentwickelt werden.

Bei der eingeführten häufigkeitsorientierten Reduktion bleiben zum Beispiel die Beschränktheit und die Lebendigkeit des Netzes nicht invariant. Zudem kann die Realisierbarkeit einer Invariante durch die Reduktion verändert werden. Denkbar ist eine strukturelle Einschränkung der Reduktion auf kleinere Teilnetze, die diese Eigenschaften erhält. Hilfreich könnte die strukturelle Einschränkung auf Teilnetze analog zur Modularisierung nach [Zai05] sein.

In biochemischen Netzwerken gibt es Primär- und Sekundärsubstanzen. Eine Ergänzung der Reduktionsregeln um eine Unterscheidung dieser Substanzen erscheint sinnvoll. Eventuell können bei der Reduktion entstehende Lesekanten zu Sekundärsubstanzen zur Verbesserung Übersichtlichkeit entfernt werden. Dagegen sollten Lesekanten zu Primärsubstanzen erhalten werden. Für eine solche Unterscheidung ist eine vorhergehende manuelle Angabe der Primär- und Sekundärsubstanzen notwendig.

Bisher wurde die Bestimmung der abhängigen Knotenmengen nur über allen minimalen Invarianten durchgeführt. Für die Bestimmung der abhängigen Knotenmengen kann eine gezielte Auswahl der Invarianten mit Hilfe einer allgemeingültigen Anfragesprache in Verallgemeinerung zur Anfragesprache in [KSH04] die Relevanz der Ergebnisse verbessern. Außerdem lässt sich dadurch auch die Zeitkomplexität der Berechnungen erheblich verringern. Mit Hilfe der Anfragesprache muss es beispielsweise möglich sein, triviale Invarianten auszuschließen. Zur Bearbeitung und Verwaltung großer Invariantenmengen ist außerdem die Entwicklung von geeigneten Datenstrukturen und Algorithmen notwendig.

Für die Berechnung der abhängigen Knotenmengen ist bisher die Bestimmung einer Menge von Invarianten erforderlich. Es gibt Hinweise auf eine rein strukturelle Erkennbarkeit zumindest eines Teils solcher Mengen. Dieser direkte Weg vermeidet den im schlimmsten Fall exponentiellen Aufwand bei der Berechnung der Invarianten. Ein Ansatz mit Hilfe der linearen Programmierung für die Berechnung sogenannter „Coupled Reaction Sets“ wird in [BNSM04] verfolgt. Es ist zu prüfen, ob dieser Ansatz auf die Petri-Netz-Theorie und die abhängigen Transitions-mengen übertragbar ist.

Sowohl die in [KG04] benutzten minimalen Schnittmengen als auch die Reaktionsmengen in [LB05] und [BNSM04] scheinen in einem engen Zusammenhang mit den abhängigen Knoten-mengen zu stehen. Dieser ist genauer zu untersuchen.

Neben der bereits vorhandenen Anzeige der abhängigen Transitions-mengen ist eine Implementierung zu entwickeln, mit der die Reduktion von Teilnetzen praktisch am Petri-Netz durchgeführt werden kann. Strukturell ist die Auswirkung der Reduktionen auf das Petri-Netz klar. Aufgrund der meist vorhandenen Hierarchisierung von Petri-Netzen ist ein geeignetes Konzept zu entwickeln, wie die Reduktion an einem hierarchisierten Netz erfolgen kann.

Sowohl die Bestimmung der abhängigen Transitions-mengen als auch die Reduktion und die Hierarchisierung wurden für einfache Petri-Netz-Typen entwickelt. Eine Übertragung auf höhere Petri-Netz-Typen, zum Beispiel gefärbte oder zeitbewertete Petri-Netze, erscheint aussichtsreich, da sowohl für gefärbte als auch für zeitbewertete Petri-Netze Invarianten berechnet werden können (siehe [NV86] und [Zai04]).

Anhang A

Implementierung

Zur Unterstützung dieser Arbeit wurden verschiedene Implementierungen durchgeführt. Sie beziehen sich ausschließlich auf die Arbeit mit Invarianten von Petri-Netzen.

A.1 Darstellung von Invarianten und anderen Knotenvektoren mit Snoopy

Beim manuellen Analysieren der Invarianten von Petri-Netzen kann mit Visualisierungen schnell ein Überblick über die an der Invariante beteiligten Transitionen und deren Zusammenarbeit gewonnen werden. Zum Beispiel sind in Abbildung 2.4 im Kapitel 2 jeweils die beteiligten Transitionen und deren Umgebung im Petri-Netz hervorgehoben, so dass sofort die relevanten Netzteile identifiziert werden können. Im praktischen Teil dieser Diplomarbeit entstand deshalb eine Implementierung, mit der Invarianten in Petri-Netzen visuell hervorgehoben werden können.

Eine Invariante eines Petri-Netzes ist als Multimenge von Knoten des Petri-Netzes interpretierbar, also als Menge von Knoten und ihrer Häufigkeit. Das induzierte Teilnetz kann jeweils über die inzidenten Kanten zu den betroffenen Knoten ermittelt werden. Dies ist jedoch nicht nur eine Eigenschaft von Invarianten, sondern von allen Transitions- oder Platzvektoren.

Deshalb wurde eine allgemein einsetzbare Technik angestrebt, um Knotenvektoren in Netzwerken zu visualisieren. Dazu wurde das Werkzeug „Snoopy“, vorgestellt in [Fie04], um eine Implementierung ergänzt, mit der extern generierte Daten zu Mengen von Knotenvektoren eingelesen, bearbeitet und graphisch angezeigt werden können.

Färbung

Um eine Menge von Knotenvektoren zu einem Petri-Netz in Snoopy temporär einzufärben, wird die entsprechende Datei geladen. In dem sich öffnenden Auswahldialog kann der Nutzer einen

oder mehrere Knotenvektoren aus einer Liste auswählen. Sobald ein Knotenvektor markiert ist, werden alle enthaltenen Knoten flächig eingefärbt. Die Umgebung der Knoten wird ebenfalls gefärbt. Bei benachbarten Knoten wird nur die Umrahmung eingefärbt. Damit ist es möglich, die Knoten und deren Umgebung voneinander zu unterscheiden. Der Name und ein Kommentar zu jedem Knotenvektor kann im Auswahldialog editiert und abgespeichert werden.

Werden mehrere Knotenvektoren ausgewählt, können entweder alle in mindestens einem Knotenvektor enthaltenen Knoten, oder alle in allen Knotenvektoren enthaltenen Knoten eingefärbt werden. Dies entspricht den Mengenoperationen „union“ und „intersection“ auf den Trägermengen der Vektoren. Der Nutzer kann zwischen beiden Varianten wählen.

Neben der einheitlichen Färbung von Knotenvektoren in einer auswählbaren Farbe kann auch eine Färbung mit einem Farbverlauf vorgenommen werden. Die Farbintensität richtet sich nach der Häufigkeit eines Knotens in den ausgewählten Knotenvektoren. Bei der Auswahl eines einzelnen Knotenvektors wird die maximale und die minimale, von null verschiedene Häufigkeit der Knoten in allen Knotenvektoren bestimmt. Knoten mit der maximalen Häufigkeit werden am intensivsten gefärbt, Knoten mit der minimalen Häufigkeit werden am wenigsten intensiv gefärbt. Alle Knoten mit dazwischenliegenden Werten bekommen eine entsprechende Farbintensität, die anhand der Position zwischen Minimum und Maximum bestimmt wird.

Etwas anders wird die Farbverlaufsfärbung berechnet, wenn mehrere Knotenvektoren ausgewählt werden. Hier werden die Häufigkeiten für jeden vorkommenden Knoten aufsummiert. Die maximale und minimale Summe der eingefärbten Knoten bestimmen den Bereich, nach dem die Farbintensität der Färbung bestimmt wird.

Dateiformat

Da bisher häufig das Werkzeug „INA“ [Sta03] benutzt wird, um zu einem Petri-Netz alle minimalen Invarianten zu berechnen, wurde das Dateiformat zum Datenaustausch kompatibel zu den von INA benutzten Formaten gewählt.

Ein Knotenvektordokument ist nach folgender Grammatik, festgehalten in der erweiterten Backus-Naur-Form, aufgebaut:

```

S           = text1 '=' DATA
DATA       = VECTORS '@' VECTORTEXTS | VECTORS
VECTORS    = VECTORS VECTOR | VECTOR
VECTORTEXTS = VECTORTEXTS VECTORTEXT | ε
VECTOR     = num '|' NODEINFOS
NODEINFOS  = NODEINFOS ',' '|' NODEINFO | NODEINFO
VECTORTEXT = num '|' text2 '|' text2 '|'
NODEINFO   = num ':' ident ':' num

```

Das Terminalsymbol *num* steht dabei für eine natürliche Zahl und *ident* steht für einen Bezeichner eines Knotens. Das Terminalsymbol *text1* (*text2*) kann dabei zu einem beliebigen Text ohne das

Zeichen '='('|'), das als Trennzeichen dient, abgeleitet werden. Damit automatisch erkannt werden kann, um welche Knotenart es sich bei den Knotenvektoren handelt, muss der aus *text* 1 abgeleitete Text ein Schlüsselwort enthalten, das mit dem jeweiligen Namen der Knotenklasse in Snoopy übereinstimmt.

Listing A.1 zeigt ein Dokument mit den minimalen Invarianten des Petri-Netzes aus Abbildung 5.2. Da es sich um T-Invarianten handelt, ist im ersten Textteil das Schlüsselwort „transition“ enthalten.

```
semipositive transition invariants =

1 |      1.DTS2           :  2,
  |      7.DTS1a         :  1,
  |     10.DTS1b         :  2,
  |     23.DTS4          :  2
2 |      1.DTS2           :  2,
  |      7.DTS1a         :  1,
  |     10.DTS1b         :  2,
  |     14.DTS5          :  2
3 |      7.DTS1a         :  1,
  |      8.DTS3          :  2,
  |     10.DTS1b         :  2,
  |     23.DTS4          :  2
4 |      7.DTS1a         :  1,
  |      8.DTS3          :  2,
  |     10.DTS1b         :  2,
  |     14.DTS5          :  2

@

1 | Erste Invariante | Dies ist die erste Invariante |
2 | Zweite Invariante | Dies ist die zweite Invariante |
```

Listing A.1: Invariantendokument zu Abbildung 5.2

A.2 DTS-Berechnung und DTS-Anzeige mit PInA und Snoopy

Für die in dieser Arbeit eingeführten abstract und strong DTS wurden in Kapitel 3 Algorithmen vorgestellt, die auf Basis einer Menge von Invarianten die DTS berechnen. Diese Algorithmen wurden in Java implementiert und in das bereits existierende Tool „PInA“ eingebunden. Da eine strong DTS eine Multimenge von Transitionen und damit einen Knotenvektor darstellt, bietet sich ein Export der Ergebnisse in das gleiche Format an, das auch für Invarianten benutzt wird. Dies gilt auch für abstract DTS. Allerdings ist zu beachten, dass als Häufigkeit der Transitionen einer abstract DTS im Dokument immer eins angegeben wird.

Da für das Speichern der DTS das selbe Format benutzt wird wie für die Invarianten, können die Transitionen, die zu einer DTS gehören, im Petri-Netz mit Hilfe von Snoopy visuell hervorgehoben, benannt und mit Kommentaren versehen werden. Dies erleichtert die Analyse der abhängigen Transitions Mengen in Petri-Netzen.

A.3 Machbarkeitsstudie zu Hierarchisierung und Reduktion

Neben der visuellen Hervorhebung von DTS ist auch eine automatisierte Hierarchisierung denkbar. Eine Implementierung in Snoopy bietet sich an, da Hierarchisierungskonzepte und das Einlesen und Bearbeiten der DTS und der Invarianten bereits realisiert sind.

Das Ersetzen eines Teilnetzes durch ein anderes Teilnetz kann schrittweise manuell von der Benutzeroberfläche von Snoopy aus durchgeführt werden. Deshalb sollte das direkte Ansteuern der dabei benutzten Funktionen ebenfalls möglich sein.

Sei d die zu hierarchisierende DTS, T_d ihre beteiligten Transitionen. Beim Erzeugen eines Subnetzes für d werden die Transitionen aus T_d sowie alle inneren Plätze von d in das neue Subnetz verschoben. Kanten von nicht verschobenen Plätzen zu Transitionen aus T_d führen nun zu der neuen hierarchischen Transition h . Für jeden mit h verbundenen Platz wird ein Repräsentant in das Subnetz eingefügt. Statt mit den Plätzen im Hauptnetz werden die Transitionen nun mit den entsprechenden Repräsentanten verbunden. Damit auf diese Weise hierarchisiert werden kann, müssen sich alle zu hierarchisierenden Knoten im gleichen hierarchischen Netz befinden.

Die Koordinaten des neuen Knotens im Netzwerk kann zum Beispiel durch den Schwerpunkt aller hierarchisierten Transitionen bestimmt werden. Dabei verändern sich die Koordinaten aller anderen, nicht in die nächste Hierarchieebene verschobenen Netzknoten nicht. Stehen, wie in Snoopy, automatisch generierbare Layouts zur Verfügung, kann das Layout nach dem Hierarchisieren auch auf diese Art bestimmt werden. Dabei können sich die Koordinaten der nicht von der Hierarchisierung betroffenen Netzbestandteile ändern.

Die Auswahl der zu hierarchisierenden DTS kann über das Fenster, mit dem derzeit die DTS angezeigt werden, erfolgen. Für einen besseren Bedienkomfort sollten die nicht trivialen DTS automatisch ausgewählt werden können. Dazu muss ein Trivialitätskriterium festgelegt und eine Trivialitätsprüfung implementiert werden. Beispielsweise können einelementige DTS als trivial bezeichnet werden.

Befinden sich nicht alle zu hierarchisierenden Knoten im gleichen Teilnetz, sollte nach bisherigen Erkenntnissen nicht hierarchisiert werden. Die Bedeutung der bestehenden Hierarchisierung kann dabei zerstört werden. Es ist außerdem nicht klar, an welcher Stelle der entstehende hierarchische Knoten eingefügt werden soll. Möglich ist eine vorhergehene Dehierarchisierung. Das dehierarchisierte Netz kann dann mit den oben beschriebenen Methoden anhand der von den DTS vorgegebenen Struktur neu hierarchisiert werden.

Literaturverzeichnis

- [Ada01] ADAMO, J.: *Data Mining for Association Rules and Sequential Patterns*. New York : Springer, 2001
- [Bau96] BAUMGARTEN, Bernd: *Petri-Netze*. Berlin, Heidelberg : Spektrum Akademischer Verlag, 1996
- [BNSM04] BURGARD, Anthony P. ; NIKOLAEV, Evgeni V. ; SCHILLING, Christophe H. ; MARRANAS, Costas D.: Flux Coupling Analysis of Genome-Scale Metabolic Network Reconstructions. In: *Genome Res.* 14 (2004), February, Nr. 2, S. 301–312
- [Bry86] BRYANT, Randal E.: Graph-Based Algorithms for Boolean Function Manipulation. In: *IEEE Transactions on Computers* 35 (1986), Nr. 8, S. 677–691
- [BW96] BOLLIG, Beate ; WEGENER, Ingo: Improving the Variable Ordering of OBDDs Is NP-Complete. In: *IEEE Trans. Comput.* 45 (1996), Nr. 9, S. 993–1002
- [DHJ02] DURAND, Arnaud ; HERMANN, Miki ; JUBAN, Laurent: STEPP - Search Tool for Exploration of Petri net Paths: A new tool for Petri net-based path analysis in biochemical networks. In: *Theor. Comput. Sci.* 270 (2002), Nr. 1-2, S. 625–642
- [Fie04] FIEBER, Markus: *Entwurf und Implementierung eines generischen, adaptiven Werkzeugs zur Arbeit mit Graphen*, Brandenburgische Technische Universität Cottbus, Diplomarbeit, 2004
- [Fre00] FREY, Georg: Modellierung flexibler Fertigungslinien und Bestimmung gültiger Produktionsfolgen mit Hilfe einer ereignisdiskreten Zustandsbeschreibung. In: *at - Automatisierungstechnik* 48 (2000), S. 16–25
- [GB06] GRAFAHREND-BELAU, Eva: *Klassifizierung von T-Invarianten in biochemischen Petrinetzen auf der Grundlage verschiedener Clusteranalyseverfahren*, TFH Berlin, Diplomarbeit, 2006
- [Hei04] HEINER, Monika: *Evaluation of cause effect graphs by Petri nets*. 2004
- [HK04] HEINER, Monika ; KOCH, Ina: Petri Net Based Model Validation in Systems Biology. In: *ICATPN*, 2004, S. 216–237
- [HU79] HOPCROFT, JE ; ULLMAN, JD: *Introduction to Automata Theory, Languages, and Computation*. Reading, Massachusetts : Addison-Wesley, 1979

- [IH00] IAN H. WITTEN, Eibe F.: *Data Mining, Practical Machine Learning Tools and Techniques with Java Implementations*. USA : Morgan Kaufmann, 2000
- [KG04] KLAMT, S. ; GILLES, ED.: Minimal cut sets in biochemical reaction networks. In: *Bioinformatics* 20 (2004), S. 226–234
- [KJH05] KOCH, Ina ; JUNKER, Björn H. ; HEINER, Monika: Application of Petri net theory for modelling and validation of the sucrose breakdown pathway in the potato tuber. In: *Bioinformatics* 21 (2005), Nr. 7, S. 1219–1226
- [KS02] KLAMT, S. ; STELLING, J.: Combinatorial complexity of pathway analysis in metabolic networks. In: *Molecular Biology Reports* 29 (2002), S. 233–236
- [KSH04] KOCH, Ina ; SCHÜLER, Markus ; HEINER, Monika: STEPP - Search Tool for Exploration of Petri net Paths: A new tool for Petri net-based path analysis in biochemical networks. In: *In Silico Biology* 5 (2004)
- [LB05] LARHLIMI, Abdelhalim ; BOCKMAYR, Alexander: Minimal Metabolic Behaviors and the Reversible Metabolic Space. In: *Matheon Preprint* 299 (2005)
- [Lig02] LIGGESMEYER, Peter: *Software-Qualität: Testen, Analysieren und Verifizieren von Software*. Heidelberg Berlin : Spektrum Akademischer Verlag, 2002
- [Mül04] MÜLLER, Jörg R.: *Dualität und Analyse von Formalen Modellen - Prädikat/Transitions-Netze und ihr Bezug zur Linearen Algebra*. Koblenz : Verlag Dietmar Fölbach, 2004
- [NV86] NARAHARI, Y. ; VISWANADHAM, N.: On the invariants of coloured Petri nets. In: *Advances in Petri Nets 1985, covers the 6th European Workshop on Applications and Theory in Petri Nets-selected papers*. London, UK : Springer-Verlag, 1986, S. 330–345
- [Pet62] PETRI, Carl A.: *Kommunikation mit Automaten*, Rhein.-Westf. Institut für Instr. Mathematik an der Universität Bonn, Schrift Nr. 2, Diss., 1962
- [PPW⁺03] PAPIN, J. A. ; PRICE, N. D. ; WIBACK, S. J. ; FELL, D. A. ; PALSSON, B. O.: Metabolic pathways in the post-genome era. In: *Trends Biochem Sci* 28 (2003), May, Nr. 5, S. 250–258
- [Rei91] REISIG, Wolfgang: *Petrinetze, Eine Einführung*. Berlin, Heidelberg, New York, Tokyo : Springer Verlag, 1991
- [Run03] RUNGE, Thomas: *Modellierung und Analyse von Stoffwechselprozessen mit Petri Netzen, dargestellt am Beispiel der Glykolyse*, Brandenburgische Technische Universität Cottbus, Studienarbeit, 2003
- [Run04] RUNGE, Thomas: *Methodik zur Modellierung und Validierung von biochemischen Netzwerken mit gefärbten Petri Netzen, dargestellt am Beispiel der Glykolyse*, Brandenburgische Technische Universität Cottbus, Diplomarbeit, 2004

- [Sac05] SACKMANN, Andrea: *Modellierung und Simulation von Signaltransduktionswegen in *Saxxharomyces cerevisiae* auf der Basis der Petrinetz-Theorie*, Ernst-Moritz-Arndt-Universität Greifswald, Diplomarbeit, 2005
- [Sed83] SEDGEWICK, Robert: *Algorithms*. Addison-Wesley, 1983
- [Sta90] STARKE, Peter H.: *Analyse von Petri-Netz-Modellen*. Stuttgart : Teubner Verlag, 1990
- [Sta03] STARKE, Peter. H.: Integrated Net Analyzer INA. (2003). <http://www.informatik.hu-berlin.de/lehrstuehle/automaten/ina/>
- [Web03] WEBB, G. I.: Association Rules. In: YE, Dr. N. (Hrsg.): *The Handbook of Data Mining, Chapter 2*. Lawrence Erlbaum Associates, 2003, S. 25 – 39
- [Zai04] ZAITSEV, D. A.: Invariants of Timed Petri Nets. In: *Cybernetics and Sys. Anal.* 40 (2004), Nr. 2, S. 226–237
- [Zai05] ZAITSEV, Dmitry A.: Functional Petri Nets. (2005)