

Eine Modellwelt zur Integration von Warteschlangen- und Petri-Netz-Modellen

Falko Bause, Heinz Beilner
Informatik IV, Universität Dortmund
Postfach 50 05 00, 4600 Dortmund 50

Zusammenfassung:

Rechensysteme werden sowohl bzgl. funktionaler Aspekte (z.B. Betrachtung von Deadlocks) als auch hinsichtlich quantitativer Aspekte (z.B. Betrachtung des Durchsatzes) untersucht. Der Einsatz von Petri-Netzen bzw. Warteschlangennetzen bietet sich hierbei an, da für beide Modellwelten eine fundierte Theorie existiert. Um zu vermeiden, daß zwei Modelle für ein System erstellt werden müssen, existieren bereits einige Modellwelten zur Verbindung der Betrachtung funktionaler und quantitativer Gesichtspunkte eines Systems. Besonders stark ist dabei in den letzten Jahren die Linie der Time(d)-Petri-Netze hervorgetreten. Nachteilig wirkt sich bei ihnen allerdings aus, daß die Beschreibungsmittel stark an der Petri-Netz-Philosophie angelehnt sind und häufig für Warteschlangenmodellierer unpassend sind. Diese Arbeit stellt eine Modellwelt vor, welche Warteschlangennetz- und Petri-Netz-Beschreibungen vereint und so den Beschreibungsaufwand reduziert bzw. die Beschreibung vereinfacht. Ferner wird untersucht, inwieweit funktionale Ergebnisse Aussagen über das quantitative Modell zulassen.

1 Motivation und Einführung

Die bekanntesten Modelle zur Untersuchung von Rechensystemen sind Petri-Netze und Warteschlangennetze. Petri-Netze dienen dabei der Untersuchung funktionaler Aspekte (im Sinne "globaler" Funktionalität und Korrektheit), wogegen Warteschlangennetze zur Leistungsbewertung des modellierten Systems eingesetzt werden. In der Regel sollte ein System sowohl unter funktionalen Gesichtspunkten betrachtet als auch in seiner Leistungsfähigkeit bewertet werden.

Zur Leistungsbewertung (quantitativen Analyse) eines Systems eignen sich Petri-Netze nicht, da sie den Begriff der Zeit nicht beinhalten; andererseits sind Warteschlangenmodelle nur bedingt zur funktionalen Analyse einsetzbar, da Mechanismen wie Synchronisation von Prozessen, wechselseitiger Ausschluß und ähnliche Probleme zumindest standardmäßig nicht beschrieben werden können. Ist man an der Analyse eines Systems unter sowohl funktionalen als auch quantitativen Gesichtspunkten interessiert, so ist es häufig nötig, mehrere Modellbeschreibungen eines Systems zu erstellen.

Um diesen Mehraufwand zu vermeiden, sind bereits zahlreiche Modelle zur Verbindung der funktionalen und der quantitativen Analyse entwickelt worden. Schon Ende der 60er, Anfang der 70er Jahre wurden hierzu Modelle vorgestellt (GMC/GMB ([RAES77], [ESTR78]), E-Netze ([NUTT72a,b]), Pro-Netze ([NOE75])). Eine wesentliche Aufgabe dieser Modelle ist die Unterstützung des Benutzers bei der Modellierung und Bewertung von Rechensystemen. Oft erweitern diese Modelle die Petri-Netz-Struktur, um eine einfache Handhabung der rechnergestützten Modellierungswerkzeuge zu ermöglichen. Dies hat zur Folge, daß die Modelle

fast ausschließlich simulativ bearbeitet werden können und eine funktionale Analyse nur für eingeschränkte Netzklassen praktikabel ist.

Ab Anfang der 70er Jahre entstanden sogenannte Time(d)-Petri-Netz-Modelle, die weitgehend auf eine Erweiterung der Petri-Netz-Struktur verzichten und so eine funktionale Analyse zulassen. Ferner wird in vielen Fällen die Zeit so in die Beschreibung integriert (Markov-Modelle), daß algebraisch-numerische Methoden, teils auch algebraisch-analytische Methoden zur Leistungsanalyse eingesetzt werden können ([MOLL81], [AMBC84], [AMBB85], WTPN85]). Die Beschreibung der Time(d)-Petri-Netze ist allerdings (verständlicherweise) stark an der Beschreibung von Petri-Netzen angelehnt, was für Warteschlangenmodellierer häufig ungewohnt oder sogar unpassend ist.

In diesem Beitrag wird eine Modellwelt vorgestellt, welche versucht, Petri-Netze und Warteschlangennetze zu vereinen. Die Struktur eines Modells wird dabei durch ein Petri-Netz beschrieben und der Begriff der "Zeit" mit den Stellen des Netzes in Verbindung gebracht. Durch Einführung zeitbehafteter Vorgänge in einem Petri-Netz kann, außer für bestimmte Fälle (z.B. SPNs [MOLL81]), die Erreichbarkeitsmenge nicht mehr oder nur noch teilweise zur quantitativen Analyse genutzt werden, da der Zustandsraum des "zeitbehafteten" Modells i.a. andere Eigenschaften besitzt als der des "zeitlosen" Modells. Diese Probleme treten auch in der hier vorgestellten Modellwelt auf; sie sind allerdings auch bei bereits bekannten Modellwelten, z.B. der GSPN-Modellwelt ([AMBC84]), zu finden. Es wird untersucht, unter welchen Bedingungen sich gewisse funktionale Aussagen, wie z.B. Lebendigkeit, auf den quantitativen Fall übertragen lassen und ob ein einem funktional "korrekten" Modell zugeordneter Markov-Prozeß eine stationäre Zustandsverteilung besitzen kann. Damit wird das Ziel verfolgt, Aussagen der funktionalen Analyse als "Voranalyse" zur quantitativen Analyse zu nutzen.

2 Grundlegende Begriffe und Aussagen aus der Petri-Netz-Theorie

Ein Petri-Netz (auch Stellen/Transitions-Netz genannt) ist ein 5-Tupel $PN = (S, T, W, W^+, M_0)$ mit

- a) S ist eine endliche, nichtleere Menge von Stellen.
- b) T ist eine endliche, nichtleere Menge von Transitionen.
- c) $S \cap T = \emptyset$
- d) W, W^+ sind Funktionen aus $[S \times T \rightarrow \mathbf{N}_0]$.
 W^- heißt Rückwärtsinzidenzfunktion und W^+ Vorwärtsinzidenzfunktion.
- e) M_0 ist eine Funktion aus $[S \rightarrow \mathbf{N}_0]$, die Anfangsmarkierung von PN.

Wie üblich werden wir zur Bezeichnung der Ein- und Ausgabeelemente die Punktnotation verwenden, so ist z.B. $\bullet s = \{t \in T \mid W^+(s, t) > 0\}$ die Menge der Eingabetransitionen zu $s \in S$; analog ist $s \bullet$ die Menge der Ausgabetransitionen zu s , und $\bullet t$ bzw. $t \bullet$ die Menge der Ein- bzw. Ausgabestellen zu $t \in T$. Eine Transition t heißt aktiviert in einer Markierung M genau dann, wenn $M(s) \geq W^-(s, t), \forall s \in \bullet t$. Eine aktivierte Transition $t \in T$ kann feuern; die Markierung M geht dabei über in die (Folge-)Markierung M' mit $M'(s) = M(s) + W^+(s, t) - W^-(s, t), \forall s \in S$ (Schreibweise: $M \rightarrow M'$ oder $M \xrightarrow{-t} M'$). Sei \rightarrow^* der reflexive, transitive Abschluß der Relation \rightarrow . Dann bezeichnet $E(PN) = \{M \mid M_0 \rightarrow^* M\}$ die Erreichbarkeitsmenge von PN. Das Petri-Netz PN heißt beschränkt, falls $E(PN)$ eine endliche Menge ist. PN heißt lebendig genau dann, wenn $\forall t \in T, \forall M \in E(PN): \exists M' \in E(PN): M \rightarrow^* M': t$ ist aktiviert in M' .

Für allgemeine Petri-Netze sind Eigenschaften wie Beschränktheit und Lebendigkeit nur mit großem Aufwand nachprüfbar. Schränkt man die Netzstruktur dagegen ein, so existieren leicht nachprüfbare Bedingungen, insbesondere um die Lebendigkeit des Petri-Netzes zu sichern.

Definition (Netzklassen):

Sei $PN = (S, T, W, W^+, M_0)$ ein Petri-Netz mit $W(s, t) \leq 1$ und $W^+(s, t) \leq 1, \forall s \in S, t \in T$. PN heißt

- a) marked graph-Netz $\forall s \in S: |\bullet s| = |s\bullet| = 1,$
- b) state machine-Netz $\forall t \in T: |\bullet t| = |t\bullet| = 1.$
- c) free choice-Netz $:\Leftrightarrow \forall s \in S: |s\bullet| > 1 \implies \bullet(s\bullet) = \{s\}$. D.h. besitzen zwei Transitionen die gleiche Eingabestelle, so ist dies ihre einzige Eingabestelle.
- d) simple-Netz $:\Leftrightarrow \forall t \in T: |\{s \in \bullet t \mid |s\bullet| > 1\}| \leq 1$. D.h. jede Transition besitzt höchstens eine Eingabestelle, welche auch Eingabestelle einer anderen Transition ist.

Zwischen o. g. Netzklassen besteht folgende echte Teilmengenbeziehung:

$$\text{state machine-Netze} \subset \text{free choice-Netze} \subset \text{simple-Netze} \\ \cup \\ \text{marked graph-Netze}$$

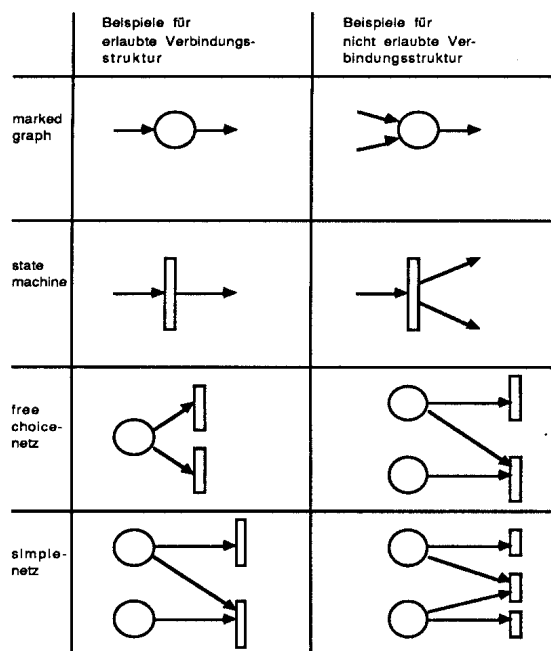


Abb. 1: Netzklassen

Um einfache Bedingungen zur Sicherung der Lebendigkeit obiger Netzklassen angeben zu können, spielen die Begriffe deadlock und trap eine wichtige Rolle. Ein Stellenmenge $S' \subseteq S$ heißt deadlock, falls $\bullet S' \subseteq S'\bullet$ bzw. trap, falls $S'\bullet \subseteq \bullet S'$, wo $\bullet S' := \cup_{s \in S'} \bullet s$ und $S'\bullet := \cup_{s \in S'} s\bullet$. Wenn ein deadlock unter einer Markierung M nicht markiert (leer; $M(s) = 0, \forall s \in S'$) ist, so ist er auch unter allen Folgemarkierungen von M nicht markiert. Ist ein trap unter einer Markierung M markiert, so ist er auch unter allen Folgemarkierungen von M markiert.

Satz (Lebendigkeitscharakterisierung für free choice-Netze):

Sei PN ein free choice-Netz. Dann gilt:

PN ist lebendig \iff Jeder deadlock besitzt einen unter M_0 markierten trap.

Beweis: z.B. [REIS82]•

Satz (hinreichende Bedingung für die Lebendigkeit in simple-Netzen):

Sei PN ein simple-Netz. Dann gilt:

Jeder deadlock besitzt einen unter M_0 markierten trap \implies PN ist lebendig.

Beweis: z.B. [COMM72]•

Obige Bedingungen lassen sich mit geringem Aufwand nachprüfen, da sie sich nur auf die Struktur des Netzes und die Anfangsmarkierung stützen. Ebenso wie die Lebendigkeit läßt sich häufig auch die Beschränktheit eines Petri-Netzes mit einfachen Mitteln nachprüfen. So ist ein Petri-Netz beispielsweise beschränkt, wenn alle Stellen durch S-Invarianten überdeckt sind ([REIS82]).

3 Die WSPN-Modellwelt

Zur Bewertung der Leistung von Rechensystemen ("quantitative Analyse") werden verbreitet Warteschlangenmodelle eingesetzt. Für die Analyse solcher Modelle existiert ein großes Spektrum an Techniken, welche die Ermittlung von Leistungsgrößen gestatten. So ist neben dem Einsatz stochastischer, ereignisorientierter Simulation insbesondere die Abbildung der Modelle auf Markov-Prozesse üblich, welche einerseits einer direkten numerischen Analyse zugänglich sind, andererseits (bei geeigneter Einschränkung der Problemklasse und der zu ermittelnden Leistungscharakteristika) den Einsatz hocheffizienter Analyse-Algorithmen erlauben, wie etwa im Falle der sog. separablen Netze. Die Darstellung jener Eigenschaften eines Rechensystems, die für seine globale Korrektheit eine zentrale Rolle spielen (Synchronisation von Prozessen, Populationsbeschränkungen samt Blockaden u.a.m.) ist in Warteschlangenmodellen zwar möglich, geschieht aber üblicherweise mit Beschreibungsmitteln, die einer formalen, modellgestützten Beurteilung der Korrektheit nicht oder nur sehr schwer zugänglich sind. Es sei in diesem Kontext daran erinnert, daß zur leistungsorientierten Bewertung solcher Modelle im Prinzip nur "allgemeine" Techniken wie Simulation und numerische Markov-Analyse einsetzbar sind.

Zur Beurteilung der globalen Korrektheit von Rechensystemen ("funktionale Analyse") werden verbreitet Modelle des Petri-Netz-Typs eingesetzt. Für die Analyse solcher Modelle existieren sehr effiziente Algorithmen zur Feststellung gewisser Systemeigenschaften (Invariantenanalyse, Algorithmen für spezielle Netzklassen). Die Darstellung (der Länge) konkreter Zeitintervalle, die für die Leistung eines Rechensystems eine zentrale Rolle spielen, ist in reinen Petri-Netzen nicht vorgesehen. Zur Einbringung von Zeitaspekten in Petri-Netze bieten sich unmittelbar zwei alternative Möglichkeiten an, zum einen die Quantifizierung der Verweilzeit von Marken in Stellen, zum anderen die Quantifizierung der Schaltdauer von Transitionen, wobei die gegenwärtige Tendenz (mit nennenswertem Erfolg) fast ausschließlich der letztgenannten Alternative gilt. Es sei in diesem Zusammenhang daran erinnert, daß das Verhalten des Modells (etwa in Form der möglichen Zustands-Trajektorien) in jedem Fall durch das Einbringen des Zeitaspekts potentiell verändert wird. Die im folgenden geschilderte "Warteschlangen-Petri-Netz-Modellwelt" (WSPN-Modellwelt) stellt einen weiteren Versuch dar, eine Spezifikation von Rechensystem-Modellen zu ermöglichen, die wesentliche Vorteile unter folgende Gesichtspunkten liefert:

- Sie vereint die Darstellungsannehmlichkeiten und -Gewohnheiten der Warteschlangennetze und der Petri-Netze.
- Sie ist zugänglich für die Analyse-Techniken der Warteschlangennetze (quantitative Analyse) und der Petri-Netze (funktionale Analyse).

Die Beschreibung der WSPN-Modellwelt geschieht hier verbal. Eine formale Charakterisierung findet sich in [BAUS86].

Die Struktur eines WSPN-Modells ist durch ein Stellen/Transitions-Netz gegeben. Die Stellen des Netzes dürfen durch Stationsbeschreibungen erweitert werden in der Form (s. Abb. 2), daß eine Strukturierung der Stelle in zwei Komponenten entsteht, wobei die "vordere" Komponente den Charakter einer Warteschlangen-"Station" annimmt, die "hintere" Komponente den einer "normalen" Petri-Netz-Stelle. Dynamisch gesehen, gelangen Marken ("Kunden"), die auf solch eine erweiterte Stelle ("zeitbehaftete Stelle") gefeuert werden, in einen (Stations-)Warteraum, aus dem heraus sie gemäß einer (Stations-)Bediendiziplin (z.B. FCFS, PS, ...) unmittelbar oder verzögert einer Bedieneinrichtung zugeführt werden; bei Bedienung werden Marken zeitlos im Abgabebereich der zeitbehafteten Stelle abgelegt, wo sie für Ausgabetransitionen dieser erweiterten Stelle verfügbar sind. Die Bedienzeit (der zeitliche Bedienbedarf) für Marken bezüglich des Stationsteils erweiterter Stellen wird durch kontinuierliche, auf $[0, \infty[$ konzentrierte Verteilungen beschrieben angenommen.

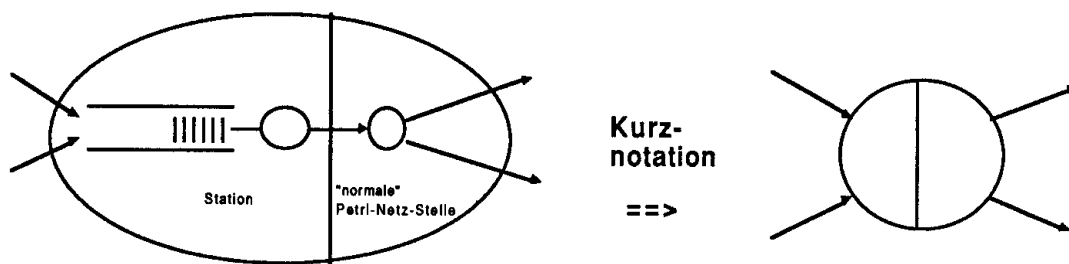


Abb. 2: Integration einer Station in eine Stelle des Petri-Netzes: zeitbehaftete Stelle

Alle Transitionen eines WSPN-Modells feuern im aktivierten Zustand zeitlos und unmittelbar nach Aktivierung. Sind mehrere Transitionen gleichzeitig aktiviert (Konfliktfall), so bestimmt eine diesbezügliche diskrete Verteilung die feuernde Transition. Der Anfangszustand eines WSPNs wird über die Anfangsmarkierung des zugrundeliegenden Petri-Netzes definiert. Marken, die zeitbehafteten Stellen zugeordnet sind, befinden sich dabei in der Stationskomponente, und die Station befindet sich in einem beliebigen zulässigen Stationszustand.

Die WSPN-Modellwelt gestattet damit beispielsweise die Beschreibung reiner Petri-Netze, wenn nämlich die Stellen nicht durch Stationsbeschreibungen erweitert werden, und die Beschreibung üblicher Warteschlangennetze, wenn als Netzstruktur ein state machine-Netz zugrundegelegt und in jede Stelle eine Stationsbeschreibung integriert wird. Beide Modellwelten werden also als Spezialfälle durch die WSPN-Modellwelt erfaßt. Der wesentliche Vorteil der WSPN-Modellwelt liegt darin, daß einerseits Mechanismen, wie Synchronisation, Populationsbeschränkungen etc. in der gewohnten Petri-Netz-Manier dargestellt werden, andererseits Scheduling-Strategien in der Form beschrieben werden, wie sie aus der Warteschlangentheorie bekannt sind. Die Beschreibung von Scheduling-Strategien mittels Time(d)-Petri-Netz-Modellen ist dagegen häufig recht umständlich. Durch die WSPN-Modellbeschreibung wird, analog zu Generalized Stochastic Petri Nets (GSPN)-Modellen ([AMBC84]), ein stochastischer Prozeß (SP) beschrieben. Der Zustandsraum des WSPN-Modells setzt sich zusammen aus den jeweiligen Stationszuständen und den Markierungen der Petri-Netz-Stellen bzw. der Petri-Netz-Komponenten zeitbehafteter Stellen.

Der SP verbringt nun eine positive Zeit in Zuständen, in denen keine Transition aktiviert ist, wohingegen er Zustände, in denen Transitionen aktiviert sind, in Nullzeit verläßt. Zustände des ersten Typs werden üblicherweise 'tangible' und die des letzteren 'vanishing' genannt (vgl. [AMBC84]). Werden die Bedienzeitverteilungen der Stationen durch Phasenverteilungen beschrieben, so kann der SP einen Markov-Prozeß beschreiben. Dieser Markov-Prozeß läßt sich mit üblichen Methoden analysieren, z.B. mit numerischen Techniken bei endlichem Zustandsraum oder auch mittels Simulation. Diese Analyse gilt quantitativen Eigenschaften des WSPN-Modells, wohingegen eine funktionale Analyse des WSPN-Modells auf Basis des zugrundeliegenden Petri-Netzes erfolgt.

4 Eigenschaften des WSPN-Modells

Durch Einführung von Stationen ändern sich gewisse Eigenschaften des zugrundeliegenden Petri-Netzes, da der Zustandsraum des WSPN-Modells i.a. nicht mehr die gleichen Eigenschaften aufweist wie die Erreichbarkeitsmenge des Petri-Netzes. Dies liegt hauptsächlich darin begründet, daß zum einen ein Zeitverbrauch an den Stellen des Netzes stattfindet, wodurch Marken für gewisse Zeiträume den Transitionen nicht zur Verfügung stehen und zum anderen aktivierte Transitionen in Nullzeit feuern müssen. So kann z.B. das zugrundeliegende Petri-Netz lebendig sein, wo diese Eigenschaft für das WSPN-Modell nicht erfüllt ist. Es treten ähnliche Probleme auf wie bei den Time(d)-Petri-Netzen. Eigenschaften werden an dieser Stelle informell beschrieben, und es werden auch nur "einfarbige" Netze behandelt. Aussagen über farbige WSP-Netze werden zum Abschluß des Kapitels angerissen. Eine formale Behandlung der angesprochenen Problemstellungen findet man in [BAUS86].

Anstelle des komplexen Zustandsraums des WSPN-Modells läßt sich ein reduzierter Zustandsraum betrachten, der dadurch entsteht, daß man bzgl. der Stationskomponente nur die Anzahl enthaltener Marken betrachtet und als gesichert annimmt, daß nach einer gewissen Zeitspanne eine (jede) Marke die Station verläßt. Viele Eigenschaften des WSPN-Modells lassen sich auf Basis dieses reduzierten Zustandsraumes analysieren, so beispielsweise die "Lebendigkeit" des WSP-Netzes und Bedingungen für einen möglicherweise eingebetteten Markov-Prozeß.

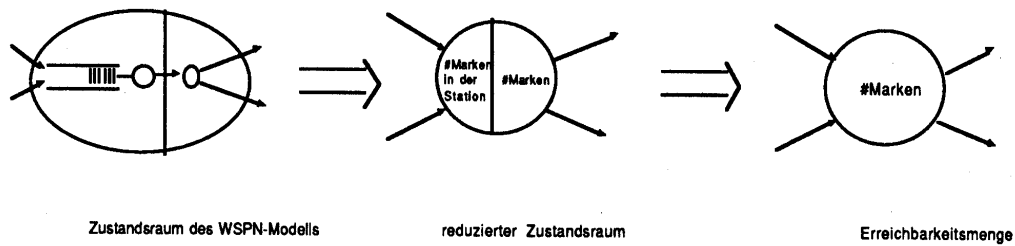


Abb. 3: Reduktion des Zustandsraumes

Faßt man noch weitergehend im reduzierten Zustandsraum die Anzahl der Marken in der Stationskomponente und die Marken auf der Petri-Netz-Komponente zusammen, so ist der resultierende Zustandsraum eine (i.allg. echte) Untermenge der Erreichbarkeitsmenge des zugrundeliegenden Petri-Netzes. Es ist damit offensichtlich, daß der Zustandsraum des durch das WSPN beschriebenen SP endlich ist, sofern das zugrundeliegende Petri-Netz beschränkt ist und Stationen bei endlicher Population auch nur endlich viele Zustände annehmen können. Die Umkehrung gilt allerdings nicht, wie Abb. 4 zeigt.

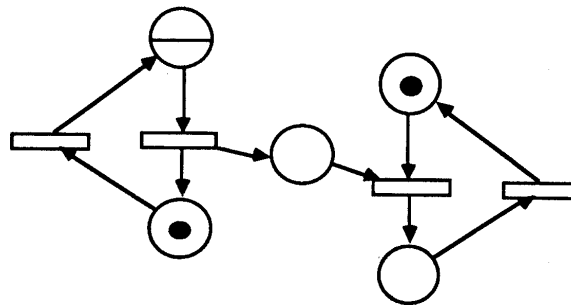


Abb. 4: WSPN dessen Zustandsraum endlich ist, obwohl das Petri-Netz nicht beschränkt ist

Obiges Netz stellt ein Produzenten-/Konsumentensystem dar, wobei der Produzent zur Produktion eine gewisse Zeitspanne benötigt, der Konsument allerdings in der Lage ist, Produktionsgüter sofort zu verbrauchen. Im folgenden werden wir nur noch beschränkte Netze betrachten.

4.1 Lebendigkeit

Die Lebendigkeit des WSP-Netzes zu sichern, ist in vielen Fällen wünschenswert, da Transitionen zur Modellierung der Übergänge zwischen Stationen einzusetzen sind. Es ist daher wichtig sicherzustellen, daß jeder im WSP-Netz definierte Übergang bei Betrachtung des SP möglich ist. Wir wollen im folgenden annehmen, daß von jedem Zustand des SP aus mit Sicherheit ein 'tangible' Zustand erreichbar ist.

Definition (P-lebend):

Ein WSP-Netz heißt P-lebend genau dann, wenn das zugrundeliegende Petri-Netz (also das Netz ohne Beachtung der Stationsbeschreibungen) lebendig ist.

Definition (WP-lebend):

Ein WSP-Netz heißt WP-lebend genau dann, wenn es zu jeder Transition und zu jedem erreichbarem Zustand des SP einen Folgezustand gibt, in dem die Transition aktiviert ist.

Es läßt sich nun danach fragen, ob P-Lebendigkeit, welche für spezielle Netzklassen effizient nachprüfbar ist (vgl. Kap. 2), hinreichend für die WP-Lebendigkeit eines Netzes ist. Dies ist nicht allgemein der Fall, wie Abb. 5 zeigt.

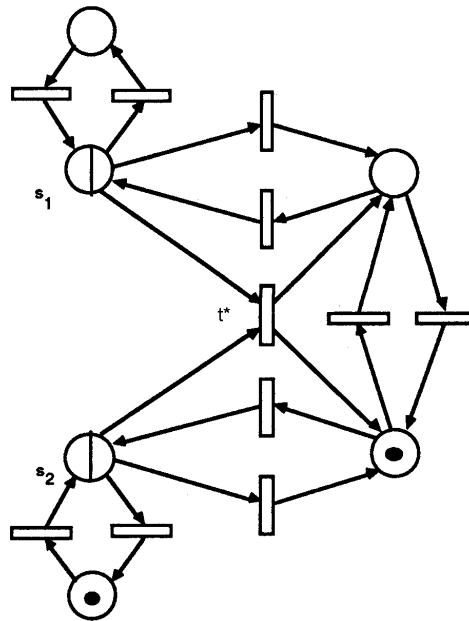


Abb. 5: WSP-Netz, welches P-lebend, aber nicht WP-lebend ist

Obiges Netz ist P-lebend, aber nicht WP-lebend, da eine der zu t^* in Konflikt stehenden Transitionen jeweils sofort schaltet, wenn im Abgabebereich von s_1 (bzw. s_2) eine Marke abgelegt wird. Eine gleichzeitige Anwesenheit von Marken in beiden Abgabebereichen (die Aktivierungsbedingung für t^*) tritt wegen der als kontinuierlich angenommenen Bedienzeitverteilungen nur mit Wahrscheinlichkeit 0 auf. Die Ursache der skizzierten Diskrepanz scheint in der Tatsache zu liegen, daß t^* zwei vorwärtsverzweigte Eingabestellen besitzt. Es liegt somit nahe, die Struktur auf Netze einzuschränken, die maximal eine vorwärtsverzweigte Eingabestelle besitzen, also simple-Netze.

Satz.

Falls das dem WSP-Netz zugrundeliegende Petri-Netz ein beschränktes simple-Netz ist, gilt:
WSP-Netz P-lebend \implies WSP-Netz WP-lebend.

Beweisskizze: Nimmt man an, daß das WSP-Netz nicht WP-lebend ist, so existiert eine Transition $t \in T$, welche ab einem bestimmten Zeitpunkt nicht mehr aktivierbar ist. Da das zugrundeliegende Petri-Netz ein simple-Netz ist, muß zumindest eine Stelle $s \in \bullet t$ existieren, welche nicht markiert und auch nicht mehr markierbar ist. Ist s nicht markierbar, sind auch alle $t \in \bullet s$ nicht mehr aktivierbar. Damit existiert eine Stellenmenge S_{leer} und eine Transitionenmenge T_{tot} mit $\bullet S_{\text{leer}} \subseteq T_{\text{tot}}$ und $S_{\text{leer}} \bullet = T_{\text{tot}}$ (wegen simple-Netz). Für die Stellenmenge S_{leer} gilt: $\bullet S_{\text{leer}} \subseteq$

$S_{\text{leer}} \bullet$. S_{leer} wäre ein nicht markierter deadlock, im Widerspruch zur P-Lebendigkeit des WSP-Netzes. •

Die Umkehrung obiger Aussage gilt für simple-Netze nicht, wie Abb. 6 zeigt. Das angegebene simple-Netz ist nicht P-lebend, wie das Feuere von t_3 und nachfolgend t_1 zeigt. Es ist dagegen WP-lebend, da t_2 eher schalten muß als t_1 .

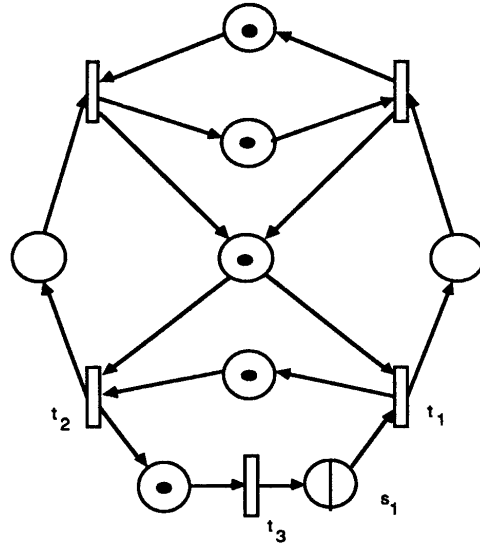


Abb. 6: simple-Netz, welches WP-lebend, aber nicht P-lebend ist

Schränkt man die Konfliktstruktur weiter ein, so ist folgende Aussage gültig.

Satz:

Falls das dem WSP-Netz zugrundeliegende Petri-Netz ein free choice-Netz ist, gilt:
WSP-Netz WP-lebend \implies WSP-Netz P-lebend.

Beweisskizze: Annahme: Das zugrundeliegende Petri-Netz sei nicht P-lebendig.

Dann gilt: \exists deadlock S_d , welcher unter dem Anfangszustand keinen markierten trap enthält (vgl. charakterisierende Lebendigkeitsbedingung für free choice-Netze).

Es ist $\bullet S_d \subseteq S_d \bullet$. Definiere $S' = S_d$. Die Stellenmenge S' wird nun sukzessive verkleinert.

Solange (S' kein trap) führe aus:

Wähle $t \in S' \bullet / \bullet S'$. Ein solches t existiert, falls S' kein trap ist.

Die Marken auf der Stellenmenge $\bullet t$ können bei Betrachtung des durch das WSP-Netzes beschriebenen SP mit positiver Wahrscheinlichkeit verfeuert werden. Die Marken auf diesen Stellen können somit als "verloren" für die Stellenmenge S_d gelten.

Setze $S' = S' / \{ \bullet t \}$.

end(solange).

Da S_d eine endliche Menge ist, bricht o.a. Verfahren ab und es gilt abschließend: S' ist ein trap (evtl. $S' = \emptyset$). Damit existiert bzgl. des SP eine Trajektorie, die alle Marken aus der Stellenmenge S_d entfernt, da nach Annahme S_d keinen markierten trap enthält. Es ist also bzgl. des SP ein Zustand erreichbar, in dem S_d ein leerer deadlock ist. Widerspruch zur WP-Lebendigkeit des WSP-Netzes. •

Obige Sätze zeigen, daß sich für gewisse Netzstrukturen die Eigenschaft des Petri-Netzes, lebendig zu sein, auf das "zeitbehaftete" Modell überträgt und damit eine einfache Überprüfbarkeit der WP-Lebendigkeit gewährleistet ist. Für free-choice-Netze gilt sogar die Charakterisierung "P-lebendig \iff WP-lebendig". Hier kann also durch Einführung der Zeit eine Lebendigkeit des Netzes nicht erzwungen werden.

4.2 Existenz der stationären Verteilung des zugeordneten Markov-Prozesses

Von großer Bedeutung bei der Analyse von WSP-Netzen zugeordneten Markov-Prozessen ist die Existenz einer stationären Verteilung. Dabei ist es wünschenswert, effizient nachprüfbar Kriterien zur Sicherstellung der Existenz einer stationären Verteilung an der Hand zu haben, um unsinnige Analyseversuche vermeiden zu können. An dieser Stelle wird eine hinreichende Bedingung vorgestellt, um eine notwendige Bedingung für die Existenz der stationären Zustandsverteilung zu sichern. Zuvor muß jedoch sichergestellt werden, daß dem WSPN ein Markov-Prozeß zugeordnet werden kann. Eine Vorbedingung hierfür ist offensichtlich die Möglichkeit, einen Markov-Zustandsraum aus ausschließlich "tangible" Zuständen konstruieren zu können. Diese ist allerdings nicht immer gegeben, wie nachfolgende Betrachtung zeigt.

Nullzeit-Falle

Die Definition eines WSP-Netzes läßt Aktionsfolgen in Nullzeit zu, da auch zeitunabhängige Stellen erlaubt sind. Transitionen, die nur solche Stellen als Eingabestellen besitzen, feuern daher sofort nach Belegung dieser Stellen. Es ist durchaus möglich, daß von einem "tangible" Zustand aus nur noch Aktionen in Nullzeit ausgeführt werden, d.h. nicht mit Sicherheit ein neuer "tangible" Zustand erreicht wird. Das Netz befindet sich dann in einer "Nullzeit-Falle", wie im Bsp. der Abb. 7.

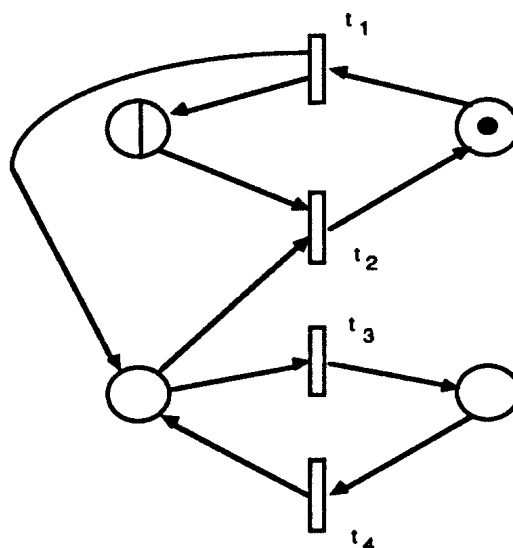


Abb. 7: WSP-Netz mit Nullzeitfalle

Nach Feuern von t_1 können t_3 und t_4 nacheinander beliebig oft in Nullzeit feuern. Ein Fortschritt des Netzverhaltens im Sinne seiner Zustandstrajektorie ist damit nicht mehr ohne weiteres beschrieben. Nullzeitfallen können unter folgender hinreichender Bedingung nicht auftreten.

Bedingung NZF:

Es sei $T' := \{t \in T \mid \bullet t \text{ enthält nur zeitunabhängige Stellen}\} \subseteq T$.

Es gelte: $\forall T_i \subseteq T', T_i \neq \emptyset: \bullet T_i \neq T_i \bullet$

Satz: Ist Bedingung NZF erfüllt, so ist für beschränkte WSP-Netze eine Nullzeit-Falle ausgeschlossen.

Beweisskizze: Man betrachte hierzu für ein $T_i \subseteq T'$ die Stellenmenge $S' := (\bullet T_i / T_i \bullet) \cup (T_i \bullet / \bullet T_i) \neq \emptyset$. Falls S' eine Eingabestelle s von T_i enthält (welche dann nicht Ausgabestelle von T_i ist) werden, nach (evtl. mehrfachen) Feuerungen der Transitionen aus $s \bullet$, die Transitionen aus $s \bullet \cap T_i$ nicht mehr aktiviert. Daher können diese Transitionen nicht beliebig oft in Nullzeit feuern. Falls S' eine Ausgabestelle s von T_i enthält (welche dann nicht Eingabestelle von T_i ist) so wäre das Petri-Netz bei beliebig häufigen Feuerungen der Transitionen aus T_i unbeschränkt. Das beliebig häufige Feuern von Transitionen in Nullzeit ist daher nur für eine echte Teilmenge $T_j \subset T_i$ möglich. Dies wird durch Bedingung NZF ebenfalls verhindert. •

Besitzt das WSP-Netz keine Nullzeitfallen, so kann ihm ein Markov-Prozeß zugeordnet werden, wenn die Dynamik aller Stationskomponenten geeignet gewählt ist (z.B. im Falle der Wahl "üblicher" Stationstypen der separablen Netzklasse). Dieser Prozeß besitzt bei endlichem Zustandsraum eine stationäre Verteilung, falls er irreduzibel ist, d.h. genau eine abgeschlossene Zustandsmenge besitzt.

mehrere abgeschlossene Zustands-Teilmengen

Ist eine Nullzeit-Falle ausgeschlossen, so ist gesichert, daß der dem WSPN zugeordnete Markov-Prozeß rekurrente Zustände besitzt. Es kann aber der Fall eintreten, daß mehrere abgeschlossene Zustandsmengen existieren. In Abb. 8 ist ein solcher Fall dargestellt. Die jeweiligen abgeschlossenen Zustandsmengen zeichnen sich dadurch aus, daß auf den Stellenmengen $\{s_2, s_3\}$ und $\{s_4, s_5\}$ jeweils 0, 1 oder 2 Marken vorhanden sein können.

Für simple-Netze ist es möglich, die Existenz genau einer abgeschlossenen Zustandsmenge zu sichern.

Bedingung REK-SIMPLE:

$T_{RS} := \{t \in T \mid \forall t' \neq t: \bullet t' \cap \bullet t = \emptyset\}$. $\forall t \in T/T_{RS}: \bullet t$ enthält eine zeitbehaftete Stelle.

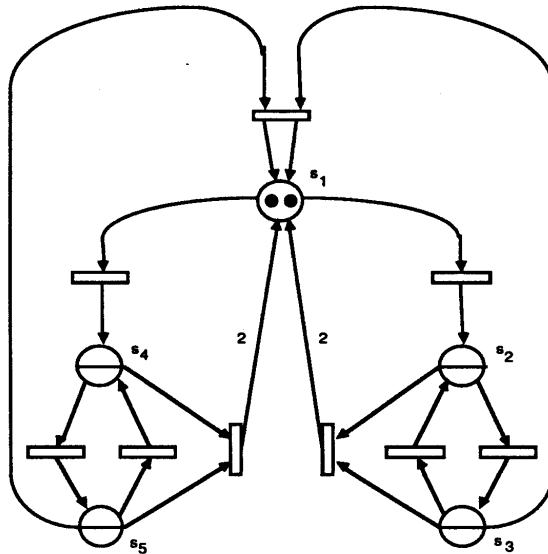


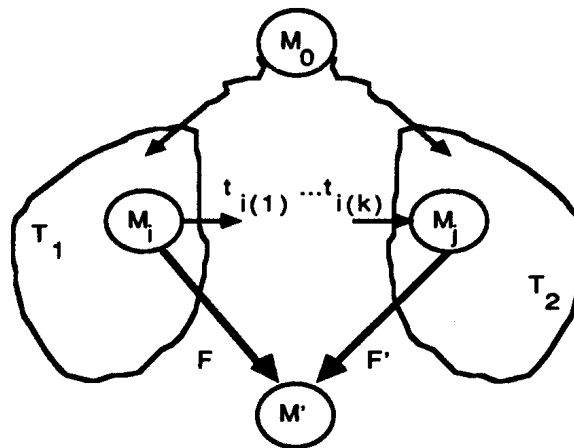
Abb. 8: WSPN mit mehreren abgeschlossenen Zustandsmengen

Satz:

Das dem WSP-Netz zugrundeliegende Petri-Netz sei ein beschränktes und lebendiges simple-Netz, dessen Erreichbarkeitsgraph genau eine abgeschlossene Zustandsmenge besitzt, und es gelten die Bedingungen REK-SIMPLE und NZF.

==> Der dem WSPN zugeordnete Markov-Prozeß besitzt genau eine abgeschlossene Zustandsmenge.

Beweisskizze: Die Beweisführung erfolgt über den reduzierten Zustandsraum (vgl. Abb. 3).
Annahme: Der reduzierte Zustandsraum besitzt mehrere abgeschlossene Zustandsmengen.



$$H \subseteq \text{TRS}, F' \equiv H, F \equiv t_{i(1)} \dots t_{i(k)} + H$$

Abb. 9

Betrachtet man die jeweiligen Markierungen (Zustände), so existieren zwei abgeschlossene Zustandsmengen T_1 und T_2 mit $T_1 \cap T_2 = \emptyset$ (vgl. Abb. 9). Seien $M_i \in T_1$ und $M_j \in T_2$ zwei beliebige Zustände in diesen Mengen. Da die Erreichbarkeitsmenge des zugrundeliegenden Petri-Netzes genau eine abgeschlossene Zustandsmenge besitzt, existiert eine Feuerungsfolge $t_{i(1)}$

... $t_{i(k)}$, die durch sukzessives Feuern M_i in M_j überführt (oder umgekehrt; OE gelte $M_i \rightarrow^* M_j$). Gemäß obiger Annahme ist diese Feuerungsfolge im WSPN nicht mehr realisierbar. Von M_i ausgehend kann jetzt eine Feuerungsfolge F konstruiert werden, die alle Transitionen aus $t_{i(1)} \dots t_{i(k)}$ enthält sowie weitere Transitionen aus $H \subseteq T_{RS}$, und welche M_i in M' überführt. Da die Transitionen aus H nicht in Konflikt zu anderen Transitionen stehen können, existiert ausgehend von M_j eine Feuerungsfolge F' bestehend aus Transitionen aus H , welche M_j in M' überführt. Damit existiert eine Markierung $M' \in T_1 \cap T_2$ - Widerspruch. \implies Behauptung. •

Vorbedingungen für die Existenz einer stationären Verteilung des angenommenen Markov-Prozesses lassen sich also über strukturelle Eigenschaften des zugrundeliegenden Petri-Netzes sichern. Man beachte, daß die oben geforderte Lebendigkeit des Netzes ebenfalls mit effizienten Mitteln nachprüfbar ist (vgl. Kap. 2).

4.3 Farbige WSP-Netze

Bisher wurden nur "einfarbige" Netze betrachtet, bei denen also nur eine Klasse von Marken (Kunden/Aufträge) existiert. Dies ist für die reale Anwendung von WSP-Netzen nur von eingeschränktem Interesse. Eine Erweiterung auf farbige Netze ist z.B. durch Einsatz farbiger Petri-Netze ([JENS81]) möglich.

Ein farbiges Petri-Netz ist ein 6-Tupel

$CPN = (S, T, C, W^-, W^+, M_0)$ mit

- a) S ist eine endliche, nichtleere Menge von Stellen.
- b) T ist eine endliche, nichtleere Menge von Transitionen.
- c) $S \cap T = \emptyset$.
- d) C ist eine Farbenfunktion. Sie ist definiert auf $S \cup T$ in endliche, nichtleere Mengen.
- e) W^-, W^+ sind (Inzidenz-)Funktionen definiert auf $S \times T$, so daß $W^-(s, t)$ bzw. $W^+(s, t) \in [C(t) \rightarrow [C(s) \rightarrow \mathbf{N}_0]]$, $\forall (s, t) \in S \times T$.
- f) M_0 ist eine Funktion definiert auf S , so daß $M_0(s) \in [C(s) \rightarrow \mathbf{N}_0]$, $\forall s \in S$.
 M_0 heißt Anfangsmarkierung des farbigen Petri-Netzes.

Ein farbiges Petri-Netz läßt sich eindeutig in ein Stellen/Transitions-Netz transformieren. Diese Transformation nennt man Entfaltung. Begriffe wie Aktivierung, Feuerung etc. lassen sich somit für farbige Petri-Netze über deren zugehörige entfalteten Stellen/Transitions-Netze definieren.

Definition (Entfaltung):

Sei $CPN = (S, T, C, W^-, W^+, M_0)$ ein farbiges Petri-Netz. Die Entfaltung läßt sich wie folgt durchführen:

1. Erzeuge für jede Stelle $s \in S$ und für jede Farbe $c \in C(s)$ eine Stelle (s, c) .
2. Erzeuge für jede Transition $t \in T$ und für jede Farbe $c' \in C(t)$ eine Transition (t, c') .

3. Definiere die Funktionen W^- , W^+ und M_0 des entfalteten Petri-Netzes durch $W^-((s, c), (t, c')) = W^-(s, t)(c')(c)$, $W^+((s, c), (t, c')) = W^+(s, t)(c')(c)$ und $M_0((s, c)) = M_0(s)(c)$.

$PN = (\bigcup_{s \in S} \bigcup_{c \in C(s)} (s, c), \bigcup_{t \in T} \bigcup_{c' \in C(t)} (t, c'), W^-, W^+, M_0)$
 ist dann das eindeutig bestimmte entfaltete Petri-Netz.

Farbige Petri-Netze sind damit, funktional betrachtet, nur kompaktere Beschreibungen für Stellen/Transitions-Netze. Alle angegebenen Sätze lassen sich somit auf entsprechende farbige Netze übertragen. Die Stationsbeschreibungen des WSP-Netzes können jetzt mehrere Kundenklassen behandeln, wobei die Stationsbeschreibungen den üblichen Beschreibungen aus der Warteschlangentheorie entsprechen. Allerdings treten dadurch zusätzliche Probleme auf, was zum Abschluß durch folgendes Beispiel illustriert werden soll.

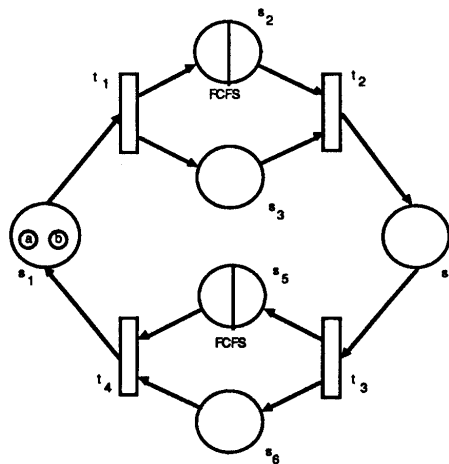


Abb. 10: Farbiges WSPN mit 2 abgeschlossenen Zustandsmengen

Das Verhalten des Netzes aus Abb. 10 ist wie folgt:

Die Stationen in den Stellen s_2 und s_5 sind beliebige FCFS-Stationen. Die Aktivierung und Feuerung der Transitionen ist folgendermaßen definiert: t_1 , t_3 sind jeweils aktiviert, wenn eine Marke des Typs a oder b auf der Eingabestelle liegt. Feuert die Transition, so wird eine Marke des selben Typs auf s_2 bzw. s_5 abgelegt (gelangt in die Station). Auf s_3 bzw. s_6 wird jeweils eine Marke des "Komplementärtyps", also $\{a, b\}/\text{Typfarbe}$, abgelegt. Die Transitionen t_2 , t_4 sind jeweils aktiviert, wenn eine Marke des Typs a oder b auf s_2 bzw. s_5 bedient wurde (sich also auf der "normalen" Petri-Netz-Stelle der Stelle befindet) und eine Marke der Komplementärfarbe auf s_3 bzw. s_6 vorhanden ist. Feuert die Transition, so werden die vorher genannten Marken entfernt und auf s_1 bzw. s_4 eine Marke des Typs a bzw. b (also des selben Typs wie zuvor auf s_2 bzw. s_5) abgelegt.

Aus der FCFS-Bedienstrategie der Stationen ergibt sich, daß eine einmal eingetretene Reihenfolge " a vor b " oder " b vor a " immer beibehalten wird. Der SP besitzt also zwei abgeschlossene Zustandsmengen.

Obiges Beispiel zeigt, daß Bedienstrategien, die alle Marken einer Farbe (Kunden einer Klasse) blockieren, also zwischenzeitlich von der Bedienung ausschließen, zusätzliche Probleme aufwerfen. Selbst für einfache Netzstrukturen (das Netz aus Abb. 10 ist ein marked-graph-Netz) sind wichtige Vorbedingungen für eine sinnvolle quantitative Analyse nicht mehr erfüllt.

5 Schlußwort

Die WSPN-Modellwelt ermöglicht die Integration von Warteschlangen- und Petri-Netzen in einer Modellbeschreibung und unterstützt so die Analyse des Modells unter sowohl funktionalen als auch quantitativen Gesichtspunkten. Die Modellbeschreibung ist dabei so gewählt, daß sie "Petri-Netz-Modellierern" und "Warteschlangen-Modellierern" entgegenkommt. Bestehende effiziente Analysetechniken aus der Petri-Netz-Theorie lassen sich dazu verwenden, Aussagen über wichtige quantitative Eigenschaften des Modells treffen zu können, sofern man sich auf gewisse Netzklassen beschränkt. Mit diesen Netzklassen lassen sich aber bereits wesentliche Merkmale eines Rechensystems beschreiben, die über sonst übliche Warteschlangen-Beschreibungen hinausgehen. Bei Einführung von mehreren Kundenklassen (farbige Netze) treten zusätzliche Probleme auf, die einer weiteren Untersuchung bedürfen.

6 Literatur

- [AMBC84] M. Ajmone Marsan, M.G. Balbo, G. Conte, A Class of Generalized Stochastic Petri Nets for the Performance Evaluation of Multiprocessor Systems, ACM Trans. on Computer Systems, Vol. 2, No. 2, Mai 1984, S. 93-122.
- [AMBB85] M. Ajmone Marsan, G. Balbo, A. Bobbio, G. Chiola, G. Conte, A. Cumani, On Petri Nets with Stochastic Timing, in [WTPN 85], S. 80-87.
- [BAUS86] F. Bause, Funktional und quantitativ analysierbare Rechensystemmodelle; Über Petrinetz- und Warteschlangen-Modellwelten, Diplomarbeit, Universität Dortmund, 1986.
- [COMM72] F. Commoner, Deadlocks in Petri Nets, Report CA-7206-231 1, Massachusetts Computer Associates, Wakefield, 1972.
- [ESTR78] G. Estrin, A Methodology for Design of Digital Systems - supported by SARA at the Age of One, Proc. of the AFIPS National Computer Conference, Vol. 47, 1978, S. 313-324.
- [JENS81] K. Jensen, Colored Petri Nets and the Invariant Method, Theoretical Computer Science 14, 1981, S. 317-336.
- [MOLL81] M.K. Molloy, On the Integration of Delay and Throughput Measures in Distributed Processing Models, Ph. D. Diss., University of California, 1981.
- [NOE75] J.D. Noe, Pro-nets: For Modeling Processes and Processors, Technical Report 75-07-15, Dept. of Computer Science, University of Washington, Seattle, 1975.
- [NUTT 72a] G.J. Nutt, The Formulation and Application of Evaluation Nets, Ph. D. Diss., Computer Science Group, University of Washington, Seattle, Juli 1972.
- [NUTT72b] G.J. Nutt, Evaluation Nets for Computer Systems Performance Analysis, Proc. of the AFIPS Fall Joint Computer Conference, 1972, S. 279-286.
- [PNPM87] Proceedings of the International Workshop on Petri Nets and Performance Models, Madison, Aug. 1987.
- [RAES77] R.R. Razouk, G. Estrin, The Graph Model of Behaviour Simulator, Proc. of the Symposium on Design Automation and Microprocessors, Palo Alto, Feb. 1977.
- [REIS82] W. Reisig, Petri-Netze. Eine Einführung, 1982.
- [WTPN85] Proceedings of the International Workshop on Timed Petri Nets, Turin, Juli 1985.