Proceedings of ITG/GI-Fachtagung Messung, Modellierung und Bewertung von Rechen- und Kommunikationssystemen (MMB '97), Freiberg, September 1997, pp. 207-216.

Erwartungswerte von gesättigten Prioritätensystemen

Stefan Bodamer, Michael Schopp

Universität Stuttgart
Institut für Nachrichtenvermittlung und Datenverarbeitung
Prof. Dr.-Ing. Dr. h. c. P. J. Kühn
Pfaffenwaldring 47, 70569 Stuttgart
E-Mail: {bodamer|schopp}@ind.uni-stuttgart.de

Zusammenfassung

Zur Leistungsbewertung von Kommunikationssystemen ist häufig die Untersuchung von prioritätengesteuerten Bediensystemen, bestehend aus einer Bedieneinheit mit Markoff-Ankunftsprozeß und allgemeinem Bedienprozeß, erforderlich. Dabei durchlaufen Aufträge innerhalb des Systems i.a. mehrere Bearbeitungsphasen mit unterschiedlichen Prioritäten und Bedienzeit-Verteilungsfunktionen in sogenannten Meldungsketten. Man spricht in diesem Zusammenhang von M/G/1-Prioritätensystemen mit Rückkopplungen. Zusätzlich sind Gruppenankünfte, Verzweigungen und Aufspaltungen von Aufträgen sowie das Auftreten unterbrechender Prioritäten möglich. Aufbauend auf bereits bestehenden Lösungen wurde ein Algorithmus entwickelt, um unter Anwendung der Momentenmethode die mittleren Durchlaufzeiten dieser Aufträge durch die einzelnen Phasen sowie durch die ganze Kette exakt zu bestimmen. Dabei werden auch gesättigte Systeme mit einem Verkehrsangebot größer 1 berücksichtigt, in denen nur noch höherprioritäre Phasen in endlicher Zeit durchlaufen werden. Als Anwendungsbeispiel wird ein Stand-alone STP untersucht.

1 Einführung

Beim Entwurf und der Dimensionierung von Kommunikationsnetzen stellen Leistungsuntersuchungen einen unverzichtbaren Teil der Entwicklungsarbeit dar. Entscheidend für die Bewertung sind dabei die Reaktionszeiten der Systeme in den einzelnen Netzknoten auf von außen kommende Anforderungen. Als Beispiel sei hier die Verbindungsaufbauzeit in einem Signalisiernetz genannt [1]. Eine Anforderung wird im System in der Regel von mehreren Prozessen auf verschiedenen Protokollebenen bearbeitet, die miteinander durch den Austausch von Meldungen kommunizieren. Dabei sind die Prozesse im allgemeinen auf unterschiedliche Prozessoren verteilt. Im System wird somit durch jede Anforderung eine Meldungskette ausgelöst, die unter Umständen verschiedene Prozessoren durchläuft. Unter gewissen Voraussetzungen ist es möglich, ein komplexes System in einzelne Subsysteme zu zerlegen und die Analyse dieser Subsysteme getrennt durchzuführen, insbesondere dann, wenn lokale Interaktionen im Vergleich zu den Interaktionen zwischen den Subsystemen überwiegen. Ein Subsystem (ein Prozessor) kann dann als M/G/1-Prioritätensystem mit Rückkopplungen modelliert werden. Diese Technik der Dekomposition wird unter anderem in [1] und [20] angewandt.

Für die Analyse von M/G/1-Prioritätensystemen existieren verschiedene Ansätze. Eine Einführung in die Analyse von M/G/1-Systemen ohne Prioritäten ist in [13] gegeben. Takagi [19] sowie für allgemeine Unterbrechungssysteme Paterok und Ettl [17] zeigen, wie die Verteilungsfunktionen für die Antwortzeiten in M/G/1-Prioritätensystemen ohne Rückkopplung bestimmt werden können. Antwortzeit- und Warteschlangenlängenverteilungsfunktionen für Spezialfälle von Prioritätensystemen mit Rückkopplungen wurden unter anderem von Enns ([5], [6]) und Fontana ([7], [8], [9], [10]) untersucht.

Wenn nur die Erwartungswerte von Durchlaufzeiten und Warteschlangenlängen gesucht sind, erweist sich die Momentenmethode als günstig, da hier keine explizite Berechnung der Zustandswahrscheinlichkeiten erforderlich ist. Diese Methode beruht auf dem Gesetz von Little [14], dem PASTA-Theorem [21] sowie dem Ergebnis für die Vorwärts-Rekurrenzzeit aus der Erneuerungstheorie [13].

Die Momentenmethode wurde erstmals von Cobham auf rückkopplungsfreie Prioritätensysteme angewandt [3]. In [18] wurde eine Erweiterung auf Systeme mit Rückkopplungen vorgenommen. Paterok war es schließlich, der in [16] ein Verfahren vorstellte, mit dem auch Systeme mit Gruppenankünften, Verzweigungen und Aufspaltungen sowie verschiedenen Arten von unterbrechenden Prioritäten analysiert werden konnten.

2 Modellbeschreibung

Die hier untersuchten Modelle bestehen aus einer Bedieneinheit und einem Warteraum mit unbegrenzter Anzahl von Warteplätzen. Ein im System ankommender Auftrag durchläuft im allgemeinen mehrere Bearbeitungsphasen, die zusammengenommen eine Meldungskette bilden. Die Ankunft von Aufträgen erfolgt dabei in einem Poisson-Strom einzeln oder in einer Gruppe, wobei der Ankunftsprozeß einer jeden Meldungskette durch eine individuelle Ankunftsrate und Gruppengröße beschrieben wird. Innerhalb der Meldungsketten können auch Verzweigungen und Aufspaltungen auftreten. Bei einer Verzweigung („Branch“) stehen einem Auftrag mehrere Wege alternativ zur Auswahl, denen jeweils eine Verzweigungswahrscheinlichkeit zugeordnet ist. Im Gegensatz dazu entstehen bei einer Aufspaltung („Fork“) aus einem Auftrag mehrere Aufträge in gleichen oder unterschiedlichen Phasen. Auf diese Weise ergibt sich für jede Meldungskette im allgemeinen Fall eine Baumstruktur.

Beim Eintritt in eine Phase kommt der Auftrag zunächst in die Warteschlange des Systems. Sein Verhalten wird dabei von den Parametern der zugehörigen Phase bestimmt. Die Priorität der Phase steuert das Einordnen des Auftrags in die Warteschlange. Aufträge mit einem höheren Prioritätswert stehen in der Warteschlange vor solchen mit niedrigerer Priorität. Die Queueing-Strategie entscheidet, welche Position unter den Aufträgen mit gleicher Priorität der Auftrag bei seinem Eintritt in die Warteschlange einnimmt. Dabei werden die Strategien FIFO, LIFO und „Random“ unterschieden. Die Queueing-Strategie gilt jeweils für alle Phasen mit gleicher Priorität. Beim Eintritt in die Bedieneinheit wird dem Auftrag gemäß der Bedienzeit-Verteilungsfunktion, die der Phase zugeordnet ist, eine Bearbeitungszeit zugewiesen.

Im Falle von Unterbrechungssystemen geschieht die Festlegung, wann eine Unterbrechung stattfindet, mit Hilfe der globalen Unterbrechungsdistanz ([11], [15]). Ein im System eintreffender Auftrag kann einen momentan in der Bedieneinheit befindlichen Auftrag unterbrechen, wenn seine Priorität um mindestens den Wert der globalen Unterbrechungsdistanz größer ist als die Priorität des bedienten Auftrags. Die Unterbrechungsstrategie bestimmt das Verhalten des Auftrags, wenn er während seiner Bearbeitung unterbrochen wird. Bei der Strategie „resume“ bliebt die bereits geleistete Arbeit erhalten. Demgegenüber beginnt die Bearbeitung des Auftrags bei der Strategie „repeat“ nach jedem unterbrochenen Bedienversuch stets wieder von vorne, wobei zwischen „repeat with resampling“ (neues Auswürfeln der Bedienzeit nach jeder Unterbrechung) und „repeat without resampling“ (Beibehalten der einmal ausgewürfelten Bedienzeit) unterschieden wird.

Bild 1 zeigt beispielhaft das Modell eines Systems, in dem zwei unterschiedliche Arten von Anforderungen bearbeitet werden. Es handelt sich dabei um eine einzelne Bedieneinheit, zu der Anforderungen aus zwei (negativ-exponentiell verteilten) Ankunftsprozessen mit Ankunftsraten λ_I bzw. λ_{II} gelangen. Die Anforderungen des Ankunftsprozesses I werden

zuerst in einer Bedienphase mit dem Bedienprozeß G_1 bearbeitet und anschließend in einer Bedienphase mit dem Bedienprozeß G_2. Mit Wahrscheinlichkeit q_{20} ist ihre Bearbeitung danach beendet, während sie mit Wahrscheinlichkeit $q_{23} = 1 - q_{20}$ noch eine Bearbeitung in eine Bedienphase mit Bedienprozeß G_3 erfahren. Die Anforderungen des Ankunftsprozesses II werden zunächst mit G_4 bearbeitet, um sich dann zu vervielfachen. Jede Anforderung, die G_4 verläßt, erzeugt N_{45} Anforderungen, die dann mit G_5 bearbeitet werden, sowie N_{46} Anforderungen die mit G_6 bearbeitet werden. Vor jeder Bearbeitung ordnen sich die Anforderungen erneut in die globale Warteschlange ein. An welcher Position hängt dabei von der Priorität und der Queueing-Strategie ab, die der folgenden Bedienphase zugeordnet sind. Der Anschaulichkeit halber ist die globale Warteschlange in Form mehrerer Teilwarteschlangen dargestellt, die den einzelnen Bedienphasen zugeordnet sind und an denen die zu den Bedienphasen gehörigen Prioritäten notiert sind.

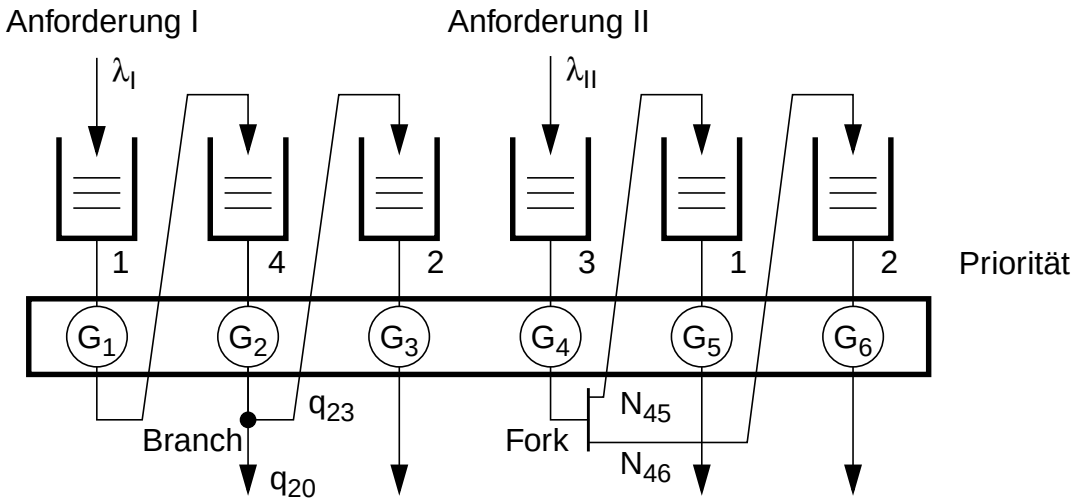

Bild 1: Beispiel eines Systems mit zwei Meldungsketten

3 Analyse

Bei der Analyse eines Systems mit Hilfe der Momentenmethode wird der Weg eines markierten Auftrags durch das System verfolgt. Dabei werden alle Teilverzögerungen, die dieser durch Aufträge verschiedener Klassen erfährt, aufsummiert mit dem Ziel, eine Bestimmungsgleichung für die mittlere Durchlaufzeit zu erhalten. Bei der Anwendung auf Prioritätensysteme mit Rückkopplungen erfolgt dieses Vorgehen schrittweise für jede Phase innerhalb einer Kette und für alle Ketten im System. Daraus erhält man für jede Phase eine Bestimmungsgleichung für die mittlere Durchlaufzeit. Alle Bestimmungsgleichungen zusammen ergeben ein lineares Gleichungssystem.

Ausgangspunkt für die Analyse einer Phase ist der Systemzustand, den der markierte Auftrag bei seiner Ankunft in der betrachteten Phase im Mittel vorfindet. Dieser mittlere Systemzustand wird einerseits bestimmt durch den Belegungszustand der Bedieneinheit und andererseits durch die mittlere Anzahl von Aufträgen, die sich in den von den einzelnen Phasen gebildeten Teilwarteschlangen befinden. Diese mittleren Warteschlangenlängen zu Ankunftszeitpunkten können durch Anwendung des PASTA-Theorems und des Gesetzes von Little als lineare Funktionen der jeweiligen Durchlaufzeiten ausgedrückt werden, falls der markierte Auftrag in einem Poisson-Strom in der Phase eintrifft. Dies ist jedoch i.a. nur bei Phasen mit externer Ankunft der Fall. Bei Phasen, in die Aufträge rückkoppeln, muß der mittlere Systemzustand zum Ankunftszeitpunkt des markierten Auftrags auf den mittleren Systemzustand bei Verlas-

sen der Vorgängerphase zurückgeführt werden, welcher wiederum aus dem Zustand bei Ankunft in der Vorgängerphase ermittelt wird.

Die Teilverzögerungen, die ein markierter Auftrag auf seinem Weg durch eine Phase erleidet, lassen sich grob auf drei Klassen von Aufträgen zurückführen:

- Aufträge, die sich bei Ankunft des markierten Auftrags in der Bedieneinheit befinden und deren Restbedienzeit er abwarten muß
- Aufträge, die sich bei Ankunft des markierten Auftrags bereits im Warteraum befinden und die vor ihm bedient werden
- Aufträge, die nach der Ankunft des markierten Auftrags in der Phase von außen im System eintreffen und ihn aufgrund ihrer Priorität oder Queueing-Strategie überholen

Jeder dieser Aufträge kann außerdem wieder durch Rückkopplung erzeugte Folgeaufträge haben, die möglicherweise ebenfalls eine Verzögerung des markierten Auftrags verursachen.

In [16] und [18] werden die Teilverzögerungen und damit die Bestimmungsgleichungen für die Durchlaufzeiten mit Hilfe komplexer mathematischer Ausdrücke unter Verwendung einer Vielzahl von Hilfsgrößen angegeben. In [2] haben wir einen effizienten Algorithmus entwikkelt, bei dem die Bestimmungsgleichungen direkt aus einer Nachbildung der Vorgänge innerhalb des Systems gewonnen werden. Dies geschieht unter Verwendung einer Warteschlange, in die sogenannte Queue Items (QIs) eingetragen werden. In einem Queue Item sind jeweils alle Aufträge zusammengefaßt, die zu derselben Kategorie (z.B. zu den „Überholern“) und zu derselben Phase gehören. Dazu werden innerhalb des QI im wesentlichen die mittlere Anzahl von Aufträgen, die er repräsentiert, sowie die Phase, zu der diese Aufträge gehören, gespeichert.

Der Algorithmus, der für jede Phase durchlaufen wird, besteht nun grob aus folgenden Schritten:

- Auffüllen der Warteschlange mit den entsprechenden QIs der Kategorien, die für die zu untersuchende Phase relevant sind, einschließlich eines markierten QI als Repräsentant für den markierten Auftrag
- Abarbeiten der QIs in der Warteschlange mit dem Ziel, die Verzögerung, welche die in den QIs enthaltenen Aufträge gegenüber dem markierten Auftrag verursachen, aufzuaddieren und damit zu einer Bestimmungsgleichung für die Durchlaufzeit für diese Phase zu gelangen

Die Abarbeitung der QIs beinhaltet außerdem die Realisierung von Rückkopplungen, indem nämlich die Folgeaufträge, die aus den Aufträgen eines abgearbeiteten QI entstehen, zu einem neuen QI zusammengefaßt werden, der wieder in die Warteschlange eingetragen wird. Der Eintrag eines QI erfolgt stets unter Berücksichtigung der Priorität und Queueing-Strategie der zugehörigen Phase.

Für Systeme mit Unterbrechung muß die Klasseneinteilung verfeinert und die Struktur der QIs erweitert werden. Außerdem muß in diesem Fall der eigentlichen Analyse die Berechnung von Hilfsgrößen wie Nichtunterbrechungswahrscheinlichkeiten und abzuwartende Restbedienzeiten vorausgehen. Die Formeln für die diese Größen wurden für die Unterbrechungsstrategien „resume“ und „repeat with resampling“ bereits in [16] teilweise unter Bezugnahme auf [4] und [18] abgeleitet, während wir für „repeat without resampling“ eine Herleitung in [2] angegeben haben.

Der Vorteil des beschriebenen Verfahrens besteht hauptsächlich in seiner einfachen Implementierbarkeit, die darin begründet liegt, daß die Struktur des Systems direkt nachgebildet wird. Dies erlaubt den Einsatz in einem Analysewerkzeug, bei dem das zu untersuchende System

durch den Anwender vorgegeben werden kann. Daneben ergibt sich vor allem für große Systeme ein Geschwindigkeitsvorteil bei der Berechnung der mittleren Durchlaufzeiten im Vergleich zu Implementierungen, die auf dem in [16] beschriebenen Ansatz beruhen.

4 Gesättigte Systeme

Ein gesättigtes System liegt vor, wenn das Gesamtangebot größer als 1 ist. Da die Auslastung der Bedieneinheit nie den Wert 1 übersteigen kann, hat dies zur Folge, daß im Durchschnitt mehr Aufträge das System betreten, als bearbeitet werden können. Dadurch nimmt die Zahl der Aufträge im System immer weiter zu, es stellt sich kein stationäres Gleichgewicht ein.

Da es sich bei dem vorliegenden Modell um ein Prioritätensystem handelt, werden bei der Verteilung der Gesamtauslastung $\rho = 1$ auf die einzelnen Phasen im gesättigten Fall Phasen mit hoher Priorität bevorzugt behandelt. Phasen mit niedrigerer Priorität bekommen gewissermaßen das, was die höherprioritären Phasen übriglassen, d.h. es können nicht so viele Aufträge in einer Zeiteinheit bearbeitet werden, wie in dieser Phase ankommen. Dies führt dazu, daß in einer solchen Phase die Ausgangsrate $\lambda_{out,\,i}$ um einen Faktor x_i geringer ist als die Ankunftsrate $\lambda_{in,\,i}$ in der Phase:

$$x_i = \frac{\lambda_{out,\,i}}{\lambda_{in,\,i}} \tag{1}$$

Bezüglich dieser im Rahmen der vorliegenden Arbeit als „Reduktionsfaktor“ bezeichneten Größe lassen sich für einen bestimmten Lastfall insgesamt drei Klassen von Phasen gemäß ihrer Priorität unterscheiden:

- Für Phasen, deren Priorität größer ist als die sogenannte „kritische“ Priorität, ist der Reduktionsfaktor 1, d.h. die Ausgangsrate stimmt mit der Ankunftsrate in der Phase überein, die Warteschlangenlänge und damit die mittlere Phasendurchlaufzeit bleiben also endlich.
- Bei Phasen, deren Prioritätswert der kritischen Priorität entspricht, nimmt der Reduktionsfaktor einen Wert zwischen 0 und 1 an, d.h. es werden durchaus noch Aufträge in dieser Phase bearbeitet, allerdings nicht so viele, wie angeboten werden. Die Erwartungswerte von Warteschlangenlänge und Phasendurchlaufzeit gehen gegen unendlich.
- In Phasen, deren Prioritätswert unterhalb der kritischen Priorität liegt, werden überhaupt keine Aufträge mehr bearbeitet, d.h. der Reduktionsfaktor ist 0. Auch hier wächst die Warteschlange bis ins Unendliche an, von einer Phasendurchlaufzeit kann überhaupt nicht mehr gesprochen werden.

In [2] zeigen wir, daß die Reduktionsfaktoren in allen Phasen mit kritischer Priorität den gleichen Wert besitzen, wenn diese Phasen alle mit der Queueing-Strategie FIFO arbeiten. Auch in bezug auf die Queueing-Strategie „Random“ gilt diese Aussage noch, während bei LIFO-Strategie die Reduktionsfaktoren im allgemeinen phasenabhängig sind.

Wenn alle Reduktionsfaktoren den gleichen Wert x haben, ergibt sich in einem System mit insgesamt n Phasen, davon k Phasen mit kritischer Priorität, allgemein für die Ausgangsrate einer Phase i

$$\lambda_{out,\,i} \sim x^{b_i} \text{ mit } 0 \le b_i \le k \le n \tag{2}$$

da innerhalb einer Kette wiederholt Phasen mit kritischer Priorität auftreten können. Dabei entspricht b_i der Anzahl der Vorgängerphasen mit kritischer Priorität in der gleichen Kette einschließlich Phase i selbst, wenn diese die kritische Priorität hat.

Der Auslastungsfaktor einer Phase errechnet sich mit Hilfe der mittleren Bedienzeit h_i zu

$$\rho_i = \lambda_{out,\,i} \cdot h_i = a_i \cdot x^{b_i} \text{ mit } 0 \le a_i \le 1,\ 0 \le b_i \le k \le n \tag{3}$$

wobei sich der Faktor a_i aus der Ankunftsrate in der Kette unter Berücksichtigung der Kettenstruktur ergibt.

Da die Summe der Auslastungsfaktoren aller n Phasen im System im gesättigten Fall 1 sein muß, erhält man allgemein als Bestimmungsgleichung für x eine nichtlineare Gleichung vom Typ

$$\sum_{i=1}^{n} a_i \cdot x^{b_i} = 1 \text{ mit } 0 \le a_i \le 1,\ 0 \le b_i \le k \le n \tag{4}$$

die für höhere Grade durch numerische Approximation gelöst werden kann. Mit Hilfe des Reduktionsfaktors lassen sich schließlich sämtliche Flußraten und Auslastungsfaktoren im System berechnen, deren Kenntnis Voraussetzung für die in Kapitel 3 geschilderte Analyse ist.

5 Ergebnisse

5.1 Beispiel mit Rückkopplung und mehreren Meldungsketten

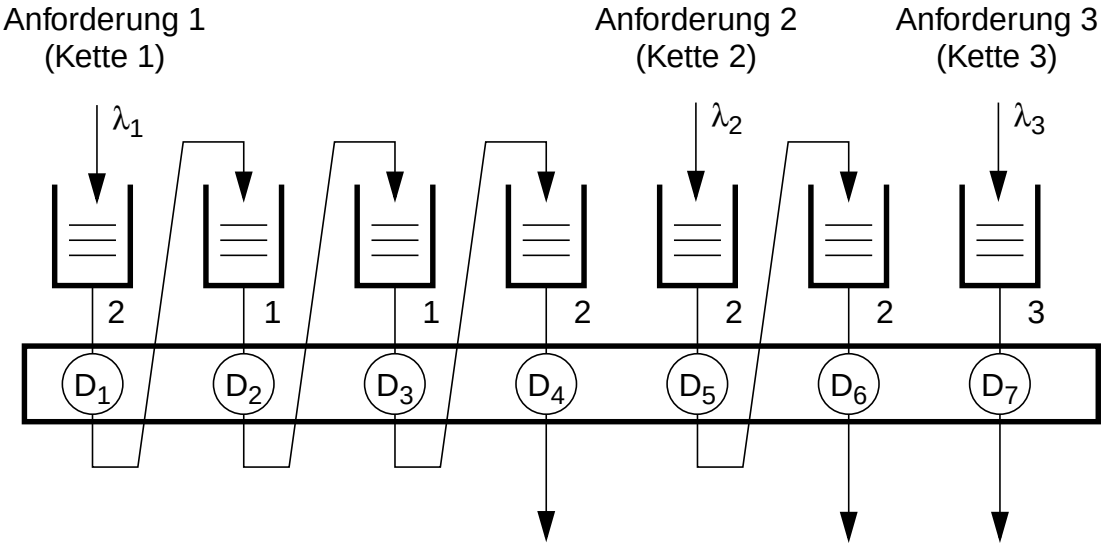

Bild 2: System mit drei Meldungsketten

Bei Systemen mit Rückkopplung ist vor allem der Fall interessant, bei dem mehrere Phasen mit der kritischen Priorität in einer Kette vorkommen. Für ein solches System mit insgesamt 7 Phasen in 3 Ketten (Bild 2) sind die Verläufe von Kettenauslastungsfaktoren und -durchlaufzeiten in Bild 3 und Bild 4 dargestellt. Als Werte für die mittleren Bearbeitungszeiten wurden

$h_1 = h_4 = h_5 = h_6 = 1$ und $h_2 = h_3 = 5$ gewählt. Das Verhältnis der Ankunftsraten ist festgelegt zu $\lambda_1 : \lambda_2 : \lambda_3 = 3 : 2 : 4$.

In diesem Fall wirkt sich die Nichtlinearität der Phasenauslastungen, die bedingt ist durch eine höhergradige Abhängigkeit vom Reduktionsfaktor, auch auf die Auslastungsfaktoren der Ketten aus. Darüber hinaus ergibt sich für die Kette 3, die nur aus einer hochprioritären Phase besteht, ein extremer Verlauf: Im ersten Sättigungsbereich ist die Durchlaufzeit durch diese Kette mit steigendem Gesamtangebot deutlich rückläufig. Dies ist vor allem darauf zurückzuführen, daß in diesem Bereich die Auslastung durch die Phasen 2 und 3, die einen hohen Restbedienzeitanteil verursachen, aufgrund der Sättigung zurückgeht.

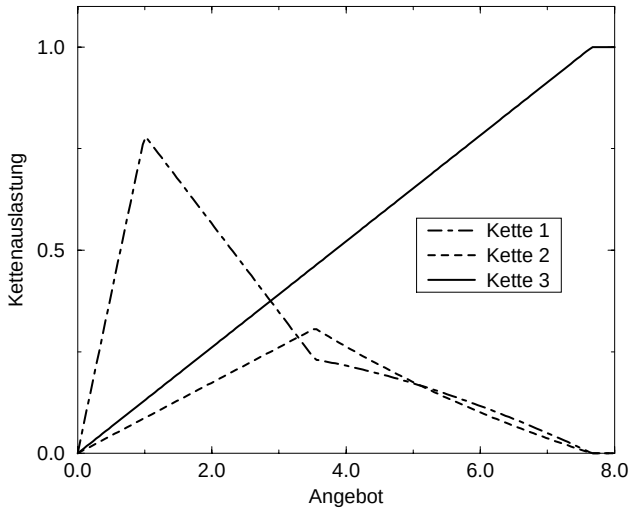

Bild 3: Kettenauslastungsfaktoren für ein System mit drei Ketten

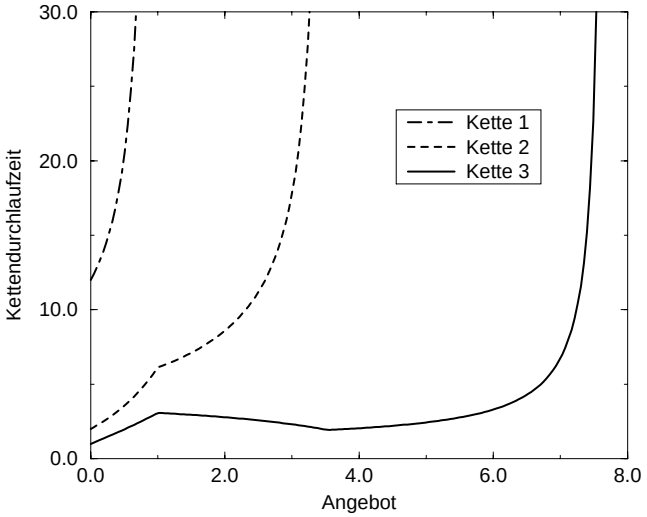

Bild 4: Kettendurchlaufzeiten für ein System mit drei Ketten

5.2 Anwendungsbeispiel: Stand-alone STP

Abschließend präsentieren wir ein mögliches Anwendungsbeispiel, wobei wir uns bei der Modellierung an die Methodik in [1] und [20] anlehnen.

In modernen Signalisiernetzen [12] spielen zunehmend sogenannte Stand-alone STPs (Signalling Transfer Points) eine wichtige Rolle. Solch ein STP zeichnet sich dadurch aus, daß er weder Ursprung noch Ziel von Signalisiermeldungen ist, sondern diese lediglich weitervermittelt. Neben der Vermittlung im Message Transfer Part (MTP) ist dabei auch eine Variante mit Global Title Translation (GTT) im Signalling Connection Control Part (SCCP) von Bedeutung. Ein Stand-alone STP wird durch mehrere unabhängige Signalisierverbindungen (Signalling Links) gespeist, die jeweils bereits MTP Level 1 und 2 realisieren. Die Überlagerung der verschiedenen Meldungsströme kann durch einen negativ-exponentiell verteilten Ankunftsprozeß angenähert werden. Gemäß der ITU-T Empfehlungen zum Signalisiersystem Nummer 7 [12], sind folgende funktionale Blöcke an der Bearbeitung von Signalisiermeldungen beteiligt: MTP Level 3: Message Discrimination (MDC), Message Distribution (MDT), Message Routing (MRT); SCCP: Routing Control Receiving (RCR), Routing Control Transmitting (RCT). Wir gehen davon aus, daß die Implementierung sich an dieser Aufteilung orientiert, daß die - im allgemeinen recht kurze - Bearbeitung einer Meldung innerhalb eines funktionalen Blocks nicht unterbrochen wird und daß der Stand-alone STP mit einem einzelnen Prozessor realisiert ist, dessen Laufzeitumgebung es erlaubt, den einzelnen funktionalen Blöcken unterschiedliche Prioritäten zuzuordnen. In Anlehnung an Q.706 und Q.716 [12], legen wir fest, daß die mittlere Level 3 STP Processor Handling Time T_{ph} den Wert *50 ms* nicht überschreiten darf und daß mit Global Title Translation die SCCP Relay Point Transfer Time (ohne MTP Level 2 Verzögerungen) im Mittel unter *100 ms* liegen muß. Somit ergeben sich für die Durchlaufzeiten der beiden möglichen Meldungsketten MDC-MRT und MDC-MDT-RCR-RCT-MRT maximale Mittelwerte von *50 ms* bzw. *100 ms*.

Wir stellen nun die Frage, welchen Einfluß die Priorisierung der verschiedenen funktionalen Blöcke auf die Leistungsfähigkeit des STP hat, wenn das Meldungsangebot hoch ist und 50% der Meldungen eine Global Title Translation benötigen. Dazu nehmen wir für zwei verschiedene Beispiele folgende Bearbeitungszeiten und Prioritäten an:

Funktionaler Block	Mittlere Bearbeitungszeit [ms]	Verteilung der Bearbeitungszeit	Prioritäten im Beispiel 1	Prioritäten im Beispiel 2
MDC	1	D	1	3
MDT	1	D	1	2
MRT	2	D	1	3
RCR	3	D	1	1
RCT	10	D	1	1

Tabelle 1: Bearbeitungszeiten und Prioritäten für die verschiedenen funktionalen Blöcke

In Beispiel 1 haben alle funktionalen Blöcke gleiche Priorität. In Beispiel 2 dagegen wird der MTP bevorzugt, wobei Message Distribution weniger hoch priorisiert ist als die anderen beiden Blöcke. Die Analyse führt zu folgenden Ergebnissen:

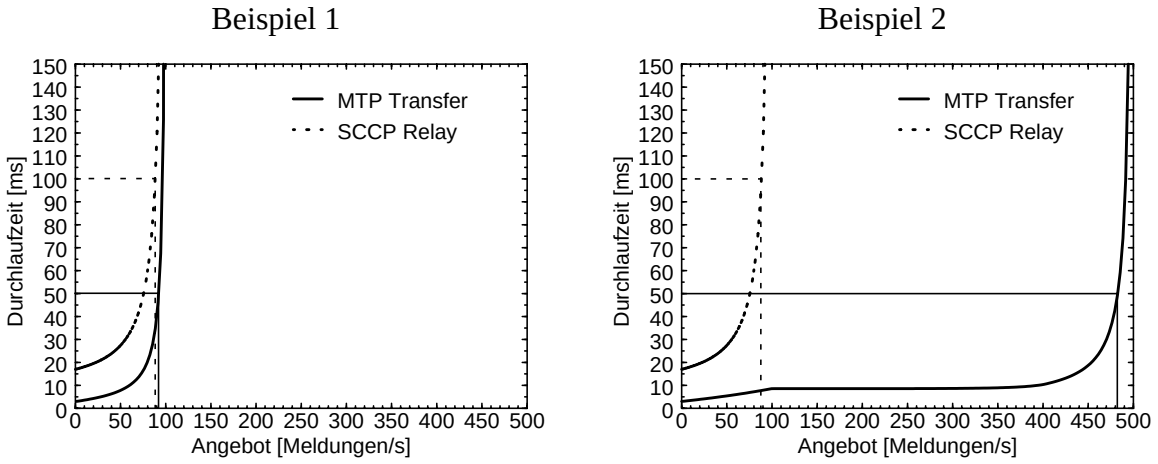

Bild 5: Durchlaufzeiten für MTP Transfer und SCCP Relay

Für SCCP Relay mit Global Title Translation ergibt sich in den beiden Beispielen kein Unterschied. Bei knapp 90 Meldungen (davon 45 mit GTT) pro Sekunde übersteigt der Mittelwert der Durchlaufzeit die Marke von *100 ms*. MTP Transfer (ohne GTT) dagegen profitiert stark von der Priorisierung: Während in Beispiel 1 bei einem Angebot von gut 90 Meldungen pro Sekunde die mittlere Durchlaufzeit den Wert *50 ms* erreicht, ist dies in Beispiel 2 erst bei einem Angebot von gut 480 Meldungen (davon 240 ohne GTT) pro Sekunde der Fall. Somit kann durch geschickte Priorisierung die Funktion des Stand-alone STP für reinen MTP Transfer auch bei hoher Last aufrechterhalten werden, ohne die Funktionsfähigkeit als SCCP Relay Point bei geringerer Last zu beeinträchtigen.

6 Zusammenfassung

Es wurde ein allgemeines Modell für Prioritätensysteme mit Markoff-Ankunftsprozeß und allgemeinem Bedienprozeß vorgestellt, in dem auch Rückkopplungen sowie Verzweigungen und Aufspaltungen im Meldungsfluß berücksichtigt werden. Dieses Modell eignet sich zur Untersuchung von Teilsystemen, in denen von außen kommende Anforderungen in unterschiedlichen Phasen bearbeitet werden, die zusammen eine Meldungskette bilden.

Mit Hilfe der Momentenmethode können die mittleren Durchlaufzeiten für einzelne Phasen und Ketten dieses Modells exakt bestimmt werden. Dazu wurde ein Algorithmus vorgestellt, der leicht an eine konkrete Konfiguration angepaßt werden kann und der eine effiziente Aufstellung der Bestimmungsgleichungen für die mittleren Durchlaufzeiten erlaubt. Er bietet sich somit für die Verwendung in einem Analysewerkzeug an.

Prioritätensysteme zeigen bei erhöhtem Angebot ein interessantes Verhalten. Für Phasen mit höherer Priorität ergeben sich endliche Durchlaufzeiten, während niederpriore Phasen ein instationäres Verhalten zeigen. Für Systeme mit Rückkopplungen können die Phasenauslastungen bei Sättigung im allgemeinen Fall mit Hilfe eines Reduktionsfaktors bestimmt werden, der sich durch Lösung einer nichtlinearen Gleichung ergibt.

Danksagung

Die Autoren danken Dr. Marcos Bafutto für seine grundlegenden Ideen und Ratschläge bei der Bearbeitung des Problems.

Literatur

[1] M. Bafutto, P. J. Kühn, G. Willmann, "Capacity and Performance Analysis of Signaling Networks in Multivendor Environments", *IEEE Journal on Selected Areas in Communication*, Vol. 12, No. 3, pp. 490-500, Apr. 1994

[2] S. Bodamer, *Mittlere Durchlaufzeiten in gesättigten M/G/1-Prioritätensystemen mit Rückkopplungen*, Diplomarbeit, Institut für Nachrichtenvermittlung und Datenverarbeitung, Universität Stuttgart, 1994

[3] A. Cobham, "Priority Assignments in Waiting Line Problems", *Operations Research*, Vol. 2, pp. 70-76, 1954

[4] R. W. Conway, W. L. Mexell, L. W. Miller, *Theory of Scheduling*, Addison-Wesley, Reading, Mass., 1967

[5] E. G. Enns, "Some Waiting Time Distributions for Queues with Multiple Feedback and Priorities", *Operations Research*, Vol. 17, pp. 519-525, 1969

[6] E. G. Enns, "A Simple Method of Calculating Queue Size Distributions of Priority Queueing Systems with Feedback", *Proc. of the 6th International Teletraffic Congress (ITC)*, München, Sep. 1970, paper 314

[7] B. Fontana, "Queue with Two Priorities and Feedback: Joint Queue-Length Distribution and Response Time Distributions for Specific Sequences", *Proc. of the 10th International Teletraffic Congress (ITC)*, Montreal, Juni 1983, paper 4.1-8

[8] B. Fontana, C. D. Bersoza, "Stationary Queue-Length Distributions in an M/G/1 Queue with Two Non-Preemptive Priorities and General Feedback", *Proc. of the Second International Symposium on Performance of Computer Communication Systems*, Zürich, 1984

[9] B. Fontana, C. D. Bersoza, "M/G/1 Queue with Two Non-Preemptive Priorities and Feedback: Response Time Distributions for any Particular Sequence", *Proc. of the 3rd International Seminar on Teletraffic Theory*, Moskau, Juni 1984, paper 16, pp. 113-116

[10] B. Fontana, C. D. Bersoza, "M/G/1 Queue with N-Priorities and Feedback: Joint Queue-Length Distributions and Response Time Distributions for any Particular Sequence", *Proc. of the 11th International Teletraffic Congress (ITC)*, Kyoto, 1985, paper 3.3A-4.1

[11] U. Herzog, *Verkehrsfluß in Datennetzen*, Habilitationsschrift, Institut für Nachrichtenvermittlung und Datenverarbeitung, Universität Stuttgart, 1973

[12] ITU-T, „Specifications of Signalling System No. 7", *ITU-T Recommendations Q.700 Series*. Genf: International Telecommunication Union, 1993.

[13] L. Kleinrock, *Queueing Systems Volume I: Theory*, John Wiley & Sons, New York et al., 1975

[14] J. D. C. Little, "A Proof of the Queueing Formula L=λW", *Operations Research*, Vol. 9, pp. 383-387, 1961

[15] M. Paterok, O. Fischer, "Feedback Queues with Preemption-Distance Priorities", *ACM SIGMETRICS Performance Evaluation Review*, Vol. 17, No. 1, pp. 136-145, May 1989

[16] M. Paterok, *Warteschlangensysteme mit Rückkopplungen und Prioritäten*, Arbeitsberichte des Instituts für Mathematische Maschinen und Datenverarbeitung, Vol. 23, No. 12, Friedrich Alexander Universität Erlangen Nürnberg, Oktober 1990

[17] M. Paterok, M. Ettl, „Sojourn Time and Waiting Time Distributions for M/GI/1 Queues with Preemption-Distance Priorities", *Operations Research*, Vol. 42, No. 6, pp. 1146-1161, 1994

[18] B. Simon, "Priority Queues with Feedback", *Journal of the ACM*, Vol. 31, pp. 134-149, 1984

[19] H. Takagi, *Queueing Analysis Volume I: Vacation and Priority Systems Part I*, North-Holland, Amsterdam et al., 1991

[20] G. Willmann, P. J. Kühn, "Performance Modeling of Signaling System No. 7", *IEEE Communications Magazine*, Vol. 28, No.7, pp. 44-56, 1990

[21] R. W. Wolff, "Poisson Arrivals see Time Averages", *Operations Research*, Vol. 20, pp. 223-231, 1982