

Analyse von Multiprozessorsystemen mit Prozeßblockierungen

Thomas Ruf, Andreas Blümle, Ian Fuat Akyildiz*

Stichworte: Multiprozessorsystem, Blockierung, Leistungsgrößen, Warteschlangennetz, Nachrichtenpuffer, Prozeßkommunikation, Mittelwertanalyse, Simulation

Zusammenfassung: In der vorliegenden Arbeit werden Warteschlangennetze vorgestellt, die es erlauben, die Leistungsgrößen von Multiprozessorsystemen zu ermitteln, wobei Blockierungen der im System befindlichen Prozesse durch Nachrichtenaustausch über endliche Puffer erfolgen können. Ein erstes solches Netz gibt zwar die zu Beginn der Arbeit gestellten Anforderungen exakt wieder, erlaubt aber keine Bestimmung der Leistungsgrößen auf analytischem Weg, da im Netz endliche Warteschlangenkapazitäten enthalten sind. Deshalb wird ein zweites Netz entwickelt, das einer Analyse zugänglich ist und dessen Leistungsgrößen mit den aus dem ersten Netz durch Simulation gewonnenen Vergleichswerten sehr gut übereinstimmen, wie eine graphische Gegenüberstellung an zwei Beispielen zeigt.

Analysis of multiprocessor systems with process locks

Key-words: multiprocessor system, process lock, performance measures, queuing network system, message buffer, process communication, mean value analysis, simulation

Abstract: In this paper, we present two queuing network systems which allow to evaluate the performance measures of multiprocessor systems where message exchange among processes is done using finite message buffers and may cause process locks. The first net presented exactly models the problem. However, analytic methods cannot be used to obtain performance measures due to the finite buffer capabilities of this net. Therefore we present a second, simplified queuing network system that can easily be analyzed. Performance measures are almost identical to the values received by simulation of the first network. This is illustrated by comparing those measures for two examples in form of graphs.

In dieser Arbeit werden Multiprozessorsysteme analysiert, in denen Prozesse über Puffer Nachrichten austauschen und aufgrund der Synchronisation Blockierungen entstehen können. Als Ergebnis der Analyse sind dabei vor allem die Auslastung, der Durchsatz, die mittlere Anzahl von und die mittlere Verweilzeit der Prozesse im Multiprozessor von Interesse sowie deren Blockierwahrscheinlichkeiten und Blockierzeiten aufgrund der Synchronisation.

Die zu untersuchenden Multiprozessorsysteme haben generell folgende Eigenschaften:

- a, Es gibt bis zu 49 identische Prozessoren.
- b, Die Prozessoren verfügen über eine einzige gemeinsame Warteschlange, die nach Disziplin FCFS bearbeitet wird.
- c, Die Bearbeitung der Prozesse findet ohne Verdrängung statt.
- d, Die Anzahl der Prozesse ist konstant (bis zu 49 Prozesse).
- e, Die Prozeßsynchronisation findet durch Nachrichtenaustausch statt, wobei jeder Prozeß mit jedem kommunizieren kann und Nachrichten dann gesendet bzw. verarbeitet werden können, wenn die entsprechenden Prozesse aktiv sind, d.h. auf Prozessoren laufen.
- f, Die Kapazität der Nachrichtenpuffer ist endlich; jedem Prozeß ist ein eigener Nachrichtenpuffer zugeordnet.
- g, Der Verarbeitung von Nachrichten durch einen Prozeß erfolgt nach Strategie FCFS, wobei die Verarbeitung selbst als zeitlos angenommen wird.
- h, Ein Prozeß kann aus drei Gründen blockiert werden:
 - i, Dem zur Ausführung anstehenden Prozess kann kein Prozessor zugewiesen werden, weil diese alle durch andere Prozesse belegt sind.
 - ii, Der Nachrichtenpuffer des Prozesses, an den eine Nachricht abgeschickt werden soll, ist voll belegt.
 - iii, Der Prozess versucht, eine Nachricht aus seinem eigenen Nachrichtenpuffer zu verarbeiten, obwohl dieser leer ist.
- k, Die Blockierung eines Prozesses wird dann aufgehoben, wenn entweder ein Prozessor frei wird (Fall i,) oder die Ausführungsbedingung der anstehenden kritischen Aktion (dies sind das Senden und Empfangen von Nachrichten, im folgenden mit SEND und RECEIVE bezeichnet) erfüllbar wird (Fälle ii, und iii.).

* Thomas Ruf, Andreas Blümle, Dr. Ian Fuat Akyildiz, Lehrstuhl für Betriebssysteme, IMMD IV, Universität Erlangen, D-8520 Erlangen

der Multiprozessor im System (im folgenden als Hardwareteil bezeichnet) folgendermaßen modellieren:

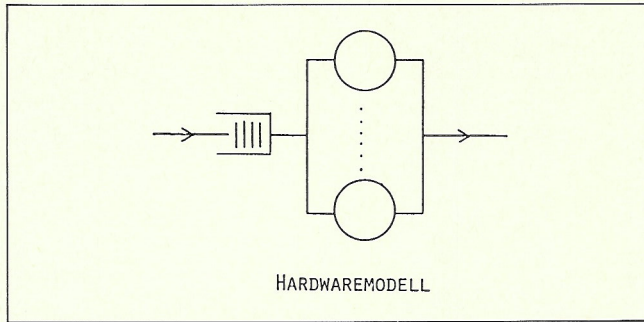


Bild 1 Hardwaremodell

Die in den Anforderungen e, und f, beschriebene Prozeßkommunikation legt die Modellierung durch das unten angegebene Kommunikationsnetz nahe:

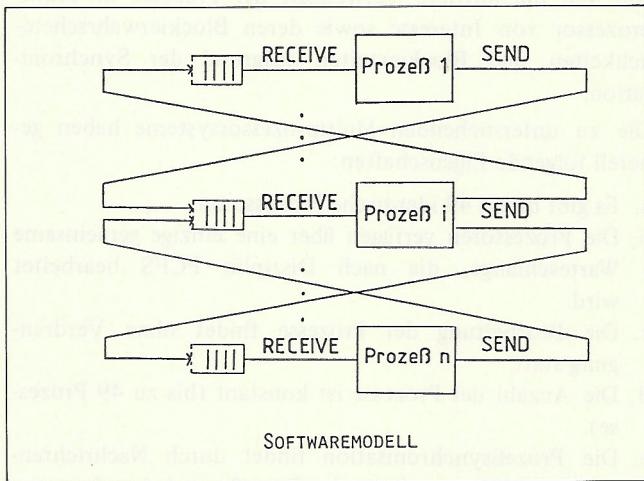


Bild 2 Softwaremodell

Bei der Entwicklung eines gemeinsamen Warteschlangennetzes, das Hard- und Software des Systems gleichermaßen berücksichtigt, sieht man sich der Schwierigkeit gegenübergestellt, daß im Hardwaremodell (Bild 1) Prozesse als Aufträge fungieren, während im Softwaremodell (Bild 2) Nachrichten als Aufträge im Modell zirkulieren.

Es ist unmittelbar einleuchtend, daß ein gemeinsames Modell für Hard- und Software des Systems auch über eine gemeinsame, einheitliche Art von Aufträgen verfügen muß. Deshalb wird jedem Prozeß ein Nachrichtenpuffer mit vorgelagerter Prozeßblockierwarteschlange zugeordnet. Eine kritische Aktion auszuführen bedeutet dann, daß der entsprechende Prozeß den ihm zugeteilten Prozessor verläßt und entweder in einem fremden Nachrichtenpuffer eine Nachricht ablegt (SEND) oder aber seinen eigenen Nachrichtenpuffer aufsucht und eine dort eingetragene Nachricht verarbeitet (RECEIVE). Falls er dabei nicht blockiert wird (siehe Anforderung h.), kehrt er anschließend in den ihm zugeteilten Prozessor zurück, wodurch wegen der angenommenen Zeitlosigkeit der Ausführung kritischer Aktionen (Anforderung g.) nach außen der Eindruck entsteht, daß der Prozess den Prozessor überhaupt nicht verlassen hat, was der gestellten Anforderung c, entspricht.

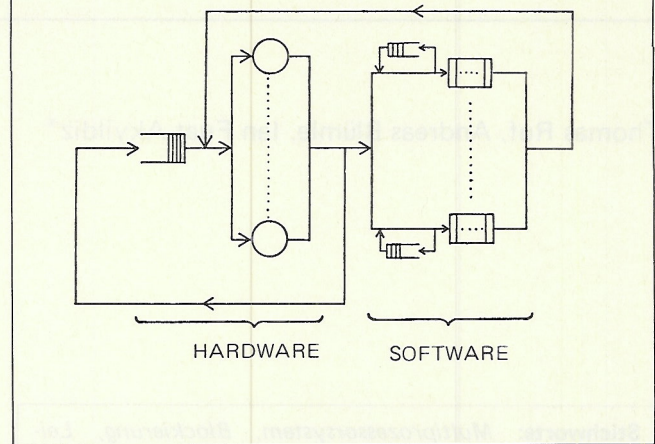


Bild 3 Gesamtmodell

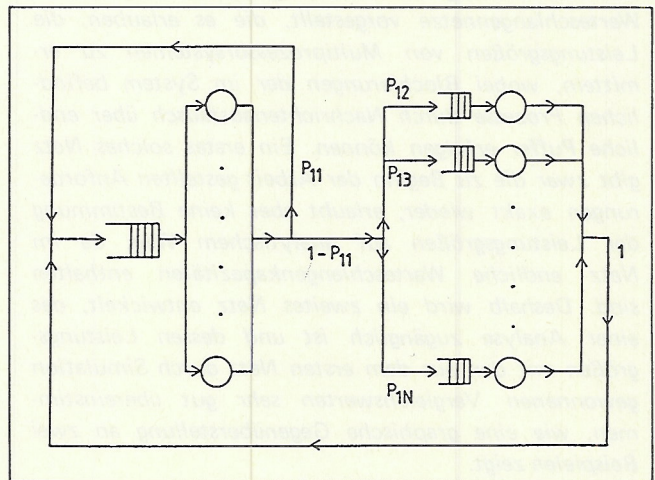


Bild 4 Vereinfachtes Modell

Die in Bild 3 zu erkennenden Prozeßblockierwarteschlangen vor den Nachrichtenpuffern nehmen alle Prozesse auf, deren Ausführungsbedingung der anstehenden kritischen Aktion nicht erfüllbar ist, wodurch gleichzeitig der dem zu blockierenden Prozeß bisher zugeteilte Prozessor freigegeben wird und neu belegt werden kann (Anforderung k.).

Da alle Prozesse im System zyklische Struktur aufweisen, kann ein Prozeßwechsel in der Hardware auch dann vorgenommen werden, wenn ein Prozeß sein Zyklusende erreicht hat. Die konstante Prozeßzahl im System schließlich (Anforderung d,) mußte deshalb vorausgesetzt werden, damit die Zuordnung der Prozesse zu den Nachrichtenpuffern im Softwareteil eindeutig ist.

Wie eben gezeigt, wird das in Bild 3 angegebene Gesamtmodell für Hard- und Software allen eingangs an die zu analysierenden Multiprozessorssysteme gestellten Anforderungen gerecht. Da laut Aufgabenstellung im wesentlichen nur die Mittelwerte der Leistungsgrößen an der Hardware zu bestimmen sind, würde es genügen, dieses Warteschlangennetz mit analytischen Methoden wie zum Beispiel der Mittelwertanalyse zu lösen [4]. Wegen der im Netz vorhandenen endlichen Warteschlangenkapazitäten ist dies

jedoch im allgemeinen Fall mit keiner der derzeit bekannten analytischen Methoden möglich. Deshalb wird im folgenden ein vereinfachtes Modell vorgestellt (Bild 4), dessen Leistungsgrößen analytisch leicht zu bestimmen sind und mit denen des in Bild 3 gezeigten Gesamtmodells nahezu übereinstimmen.

Wie man aus den Bildern 3 und 4 erkennen kann, ist die Modellierung des Hardwareteils in beiden Modellen dieselbe, während im Softwareteil statt der endlichen Nachrichtenpuffer und Prozeßblockierwarteschlangen im Gesamtmodell im vereinfachten Modell Typ-1-Stationen (M/M/1-FCFS) den Einfluß der Prozeßkommunikation auf das Gesamtsystem modellieren.

Das in Bild 4 gezeigte Warteschlangennetz ist sehr einfach mit analytischen Methoden zu lösen, wenn die Übergangswahrscheinlichkeiten zwischen und die Bedienzeiten in den Stationen bekannt sind.

Um den Zusammenhang zwischen Gesamtmodell und vereinfachtem Modell herzustellen, werden aus dem Gesamtmodell auf simulativem Weg die Blockierzeiten und Blockierwahrscheinlichkeiten der einzelnen Prozesse bestimmt [5]. Sodann werden die ermittelten Blockierzeiten als Bedienzeiten in den Typ-1-Stationen des vereinfachten Modells eingesetzt und aus den Blockierwahrscheinlichkeiten die Übergangswahrscheinlichkeiten P_{12} bis P_{1n} so bestimmt, daß die Summe der Einzelwahrscheinlichkeiten 1 ergibt. Diese sind noch mit der Wahrscheinlichkeit $1 - P_{11}$ zu multiplizieren, da ein Prozeß nur dann in eine der Verzögerungsstationen (= Typ-1-Stationen) gelangen soll, wenn eine softwareseitige Blockierung vorliegt, und nicht dann, wenn ein Prozeß sein Zyklusende erreicht hat und deshalb den ihm zugeteilten Prozessor freigibt (Übergang P_{11}). Als Bedienzeit der Bedieneinheiten in Station 1 (Multiprozessor) wird die mittlere Zyklusdauer aller im System befindlichen Prozesse eingesetzt.

Die Leistungsgrößen des vereinfachten Modells können nun mit bekannten analytischen Methoden wie z.B. der Mittelwertanalyse bestimmt werden. Zur Validierung dieser hybrid-simulativ gewonnenen Werte wurde das in Bild 3 gezeigte Gesamtmodell auch simuliert und die erhaltenen Leistungsgrößen miteinander verglichen. Die im Anhang angegebenen graphischen Gegenüberstellungen zeigen, daß die Abweichungen sehr gering sind (siehe hierzu auch [1]).

Bemerkungen

- 1) Bei hohen Auslastungen der Verzögerungsstationen hat es sich als sinnvoll erwiesen, diese abweichend von der Darstellung in Bild 4 statt durch Typ-1-Stationen durch Typ-3-Stationen (M/G/ ∞ -keine Warteschlange) zu modellieren [3]. In diesem Fall liegen die ermittelten Leistungsgrößen deutlich näher an den durch Simulation gewonnenen Vergleichswerten, während bei niedriger Auslastung der Verzögerungsstationen die Modellierung durch Typ-1-Stationen etwas bessere Ergebnisse liefert.

- 2) Inzwischen ist es gelungen, die Leistungsgrößen des Multiprozessorsystems unter Warteschlangendisziplin RR/PS am Multiprozessor auf rein analytischem Weg zu bestimmen [2]. Dazu wurde das Prozesskommunikationsmodell (Bild 2) analytisch untersucht und die Ergebnisse als Eingangsparameter für das vereinfachte Modell verwendet. Ein Vergleich sowohl mit den rein simulativ als auch mit den hybrid-simulativ gewonnenen Vergleichsgrößen hat gezeigt, daß die Ergebnisse sehr gut übereinstimmen.

Anhang

Die nachstehenden Bilder zeigen den Verlauf der Leistungsgrößen zweier Multiprozessorsysteme in Abhängigkeit von der Anzahl der Bedieneinheiten (Prozessoren) in der Hardware. Die durch Analyse des vereinfachten Modells (Bild 4) erhaltenen Größen sind dabei jeweils durch eine gestrichelte Linie verbunden, die durch Simulation des Gesamtmodells (Bild 3) gewonnenen Vergleichswerte durch eine durchgezogene Linie.

a, Zehn Prozesse:

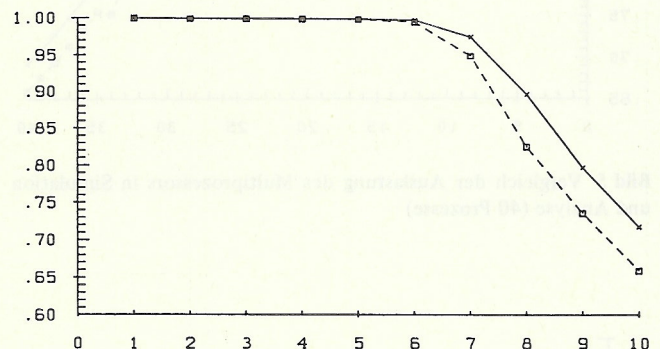


Bild 1 Vergleich der Auslastung des Multiprozessors in Simulation und Analyse (zehn Prozesse)

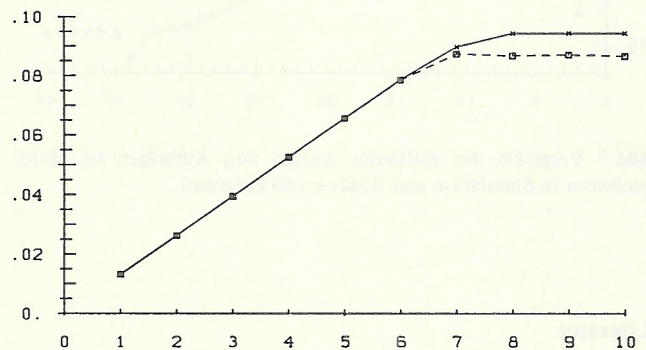


Bild 2 Vergleich des Durchsatzes des Multiprozessors in Simulation und Analyse (zehn Prozesse)

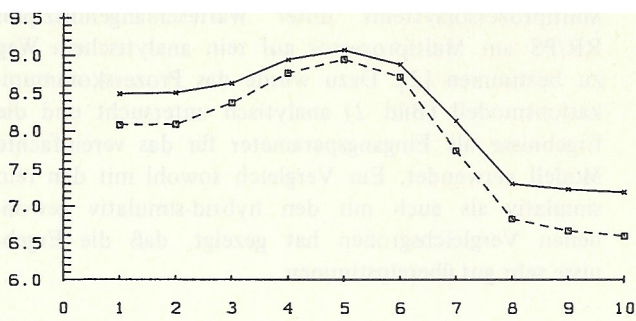


Bild 3 Vergleich der mittleren Anzahl von Aufträgen im Multiprozessor in Simulation und Analyse (zehn Prozesse)

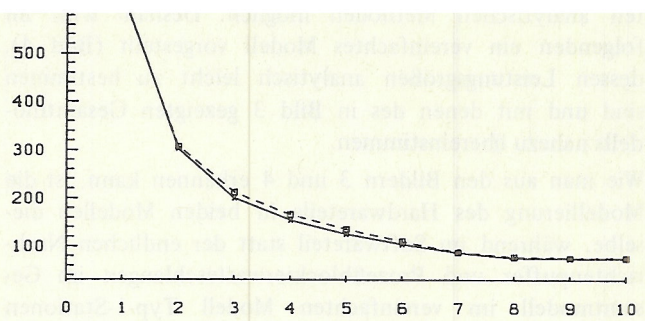


Bild 4 Vergleich der mittleren Verweilzeit im Multiprozessor in Simulation und Analyse (zehn Prozesse)

b, 40 Prozesse:

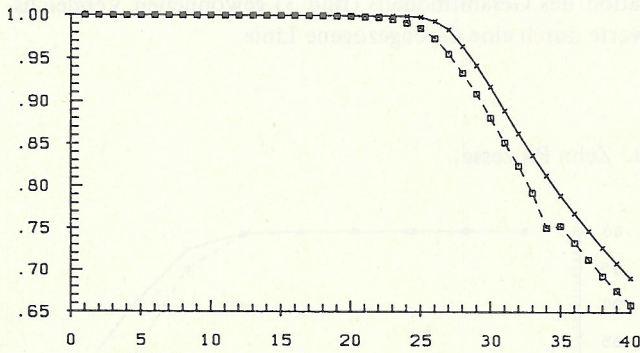


Bild 5 Vergleich der Auslastung des Multiprozessors in Simulation und Analyse (40 Prozesse)

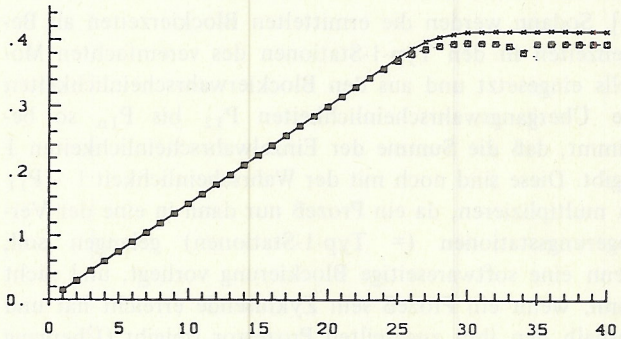


Bild 6 Vergleich des Durchsatzes des Multiprozessors in Simulation und Analyse (40 Prozesse)

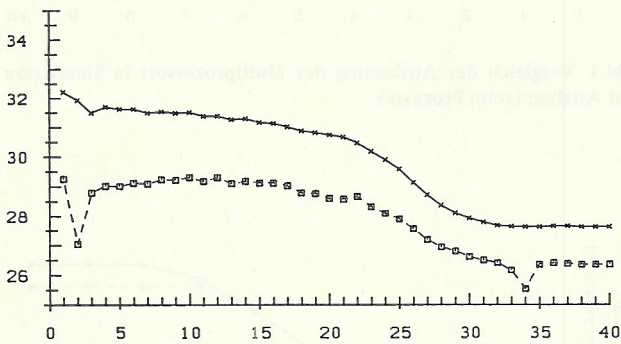


Bild 7 Vergleich der mittleren Anzahl von Aufträgen im Multiprozessor in Simulation und Analyse (40 Prozesse)

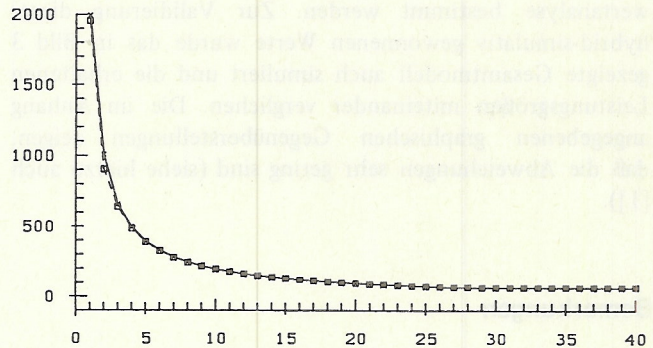


Bild 8 Vergleich der mittleren Verweilzeit im Multiprozessor in Simulation und Analyse (40 Prozesse)

Literatur

- [1] Akyildiz, I. F.; Blümle, A.; Ruf, T.: Hybrid-Simulation eines Multiprozessorsystems mit Prozeßsynchronisation. ASIM Simulationstechnik, Wien 1984
- [2] Akyildiz, I. F.: Leistungsanalyse von Multiprozessorsystemen mit Prozeßkommunikation. Arbeitsberichte des IMMD der FAU Erlangen-Nürnberg, Band 17, Nr. 8, Erlangen 1984
- [3] Blümle, A.; Ruf, T.: Kopplung von Software- und Hardwaremodellen zur Leistungsanalyse von Multiprozessorsystemen mit Prozeßsynchronisation. Studienarbeit am IMMD IV der FAU Erlangen-Nürnberg, Erlangen 1984
- [4] Bolch, G.; Akyildiz, I. F.: Analyse von Rechensystemen. Teubner Verlag, Stuttgart 1982
- [5] Schmidt, B.: GPSS-FORTRAN Version 3. RRZE-Dokumentation, Band 119, Erlangen o. J.