

Modellgestützte Analyse und Optimierung

Übungsblatt 2

Ausgabe: 19. April, **Abgabe:** 26. April, 12 Uhr

Aufgabe 2.1 (4 Punkte)

Es gibt verschiedene Wege ein System zu untersuchen. Man kann mit dem realen System oder mit einem Modell des Systems experimentieren. Grundsätzlich gibt es zwei Modellarten: physikalische und mathematische Modelle. Mathematische Modelle kann man weiter in analytische und simulative Modelle unterteilen. Geben sie für die folgenden Systeme an, welche der Modellarten jeweils am effektivsten ist. Begründen sie dabei stichwortartig, wieso diese Modellart besser ist und die anderen schlechter geeignet sind:

- a) Ein kleiner Bereich einen bestehenden Fabrik
- b) Ein Kommunikationsnetzwerk für militärische Einsätze
- c) Ein Autobahnkreuz mit hoher Staugefahr
- d) Eine Pizzeria mit Lieferservice
- e) Eine Notfallstation eines Krankenhauses
- f) Shuttleservice für eine Großveranstaltung

Aufgabe 2.2 (8 Punkte)

An einem Supermarkt treffen „immer wieder“ Kunden ein. Jeder Kunde nimmt sich zunächst einen Einkaufskorb, sucht anschließend eine „gewisse Anzahl“ von Waren aus, was „eine gewisse Zeit“ dauert, wendet sich dann den Kassen zu (wo er unter Umständen auf Bedienung warten muss) und wird schließlich bedient, was wieder „eine gewisse Zeit“ in Anspruch nimmt. Genauere Spezifikation:

- Die Ankunftszeiten der verschiedenen Kunden sind festgelegt durch Zwischenankunftszeiten. Die Zwischenankunftszeiten sind durch Beobachtungen am realen Supermarkt bestimmt worden. Für den i -ten Kunden steht dieser Wert in der Variablen z_i .
- Die Zahl der Einkaufskörbe ist begrenzt auf M . Findet ein eintreffender Kunde keinen freien Korb, dann verlässt er den Supermarkt direkt wieder.
- Die Zahl der Waren die ein Kunde aussucht sei „schwankend“. Diese ist ebenfalls durch Beobachtungen am realen Supermarkt bestimmt worden und in der Variablen k_i gespeichert.

- Abhängig von der Anzahl k_i auszusuchender Waren benötigt ein Kunde die Zeit k_s für die Warenauswahl und die Zeit k_p für die Bedienung an der Kasse.
- Die Zahl der Kassen ist begrenzt auf N . Findet ein zahlungswilliger Kunde keine freie Kasse, dann wartet er bis eine Kasse frei wird.
- Fehlen noch Angaben? Treffen sie in diesem Fall „Ihre“ Annahmen und notieren diese tabellarisch.

Entwerfen Sie für das oben beschriebene System ein ereignisorientiertes Simulationsmodell wie in der Abbildung im Skript auf den Seiten 33 und 34. Machen Sie Gebrauch von Pseudo-Code (in Anlehnung an C/C++ oder Java) und umgangssprachlichen Beschreibungen wie Sie es für sinnvoll halten. Beschreiben Sie tabellarisch, welche Zustandsvariablen benötigt werden, welche Attribute zu einem Kunden gehören, welche Ereignisse auftreten können und wie die Ereignisroutinen aussehen. Zur Vereinfachung nutzen Sie bitte bei der Verwendung von Warteschlangen folgende Funktionen: Die Funktion `ws_einreihen(schlange, element)` fügt ein Element am Ende in eine Warteschlange ein, `ws_erster(schlange)` liest das erste Element einer Warteschlange aus, `ws_entfernen(schlange)` entfernt das erste Element einer Warteschlange, und `ws_leer(queue)` testet, ob eine Warteschlange leer ist.