

Kurzfassung

Hier steht der deutsche Text der Kurzfassung. Eine evtl. gekürzte Version inklusive einiger Stichwörter muss auch im TUG-Online eingegeben werden.

Abstract

This is the English version of the abstract. It is also required to submit a short English abstract including some keywords to the TUG-Online system.

Danksagung

Diese Diplomarbeit wurde im (Studien)Jahr am Institut für Technische Informatik an der Technischen Universität Graz durchgeführt.

Danksagung an alle am Institut bzw. bei Firmen, die geholfen haben....

Danksagung an Freunde und Freundinnen für das Verständnis, ebenso den Eltern und allen sonstigen Sponsoren....

Graz, im Monat Jahr

Name des Diplomanden

Inhaltsverzeichnis

1	Einleitung	7
1.1	Motivation	7
1.2	Zielsetzung	8
1.3	Gliederung	9
2	Überschrift Kapitel zwei	10
2.1	Überschrift Abschnitt	10
2.2	Überschrift nächster Abschnitt	10
2.2.1	Überschrift Unterabschnitt	10
2.3	Überschrift NochEinAbschnitt	10
3	Überschrift Kapitel drei	11
3.1	Überschrift Abschnitt 3.1	11
4	Überschrift NochEinKapitel	12
5	Schlußbemerkung und Ausblick	13
A	z.B. Begriffsbestimmung	14
A.1	z.B. Definitionen	14
A.2	z.B. Verwendete Symbole	14
	Literaturverzeichnis	15

Abbildungsverzeichnis

1.1	Abstraktionsebenen eines physikalischen Systems.	8
-----	--	---

Tabellenverzeichnis

4.1	Laufzeit des Algorithmus.	12
-----	-----------------------------------	----

Kapitel 1

Einleitung

Dieses Dokument soll als Vorlage zum Verfassen von Diplomarbeiten am Institut für Technische Informatik der TU Graz mit Hilfe von L^AT_EX dienen. Es sind hier die wichtigsten Abschnitte der Diplomarbeit zusammengefaßt. Weiters werden die wichtigsten Befehle von L^AT_EX an einigen Beispielen kurz vorgestellt. Dieses Vorlage erhebt keinen Anspruch auf Vollständigkeit.

Die Einleitung enthält u.A. die Aufgabenstellung, die Vorgehensweise zur Lösung und die Gliederung der Arbeit. Es folgt nun ein **Beispieltext**.

1.1 Motivation

Qualitative Simulation ist ein Teilbereich des *qualitative reasonings* (qualitatives Schließen). Dabei wird versucht, Schlußfolgerungen über das Verhalten von Systemen auf höheren Abstraktionsebenen zu ziehen. Hauptanwendungsgebiete für die qualitative Simulation sind Bereiche der künstlichen Intelligenz. Die Abbildung 1.1 zeigt die verschiedenen Abstraktionsebenen eines Systems. Die erste Ebene stellt das physikalische System dar. Durch die Beobachtung des Systems kann das Verhalten bestimmt werden. In der zweiten Ebene wird das physikalische System mit Differentialgleichungen (*DGLen*) modelliert. Durch eine analytischen Berechnung der DGLen wird eine Beschreibung des Verhaltens mit reellwertigen Funktionen erzielt. In der kontinuierlichen Simulation werden häufig numerische Verfahren zur Berechnung dieser Funktionen eingesetzt. Eine weitere Abstraktion der DGLen in qualitative DGLen (*qualitative differential equations* - *QDEs*) wird in der dritten Ebene vollzogen. Die qualitative Simulation bestimmt aus diesem Modell eine qualitative Beschreibung des Systemverhaltens.

Im Gegensatz zur kontinuierlichen Simulation, welche für numerische Eingabegrößen die resultierenden Ausgabewerte berechnet, wird bei der qualitativen Simulation das Modell abstrakter betrachtet. Diese abstrakte Betrachtungsweise entspricht auch eher dem menschlichen Verständnis. Gründe für den Einsatz der qualitativen Simulation sind unter anderem:

- Das quantitative Wissen über ein System ist unvollständig oder nicht gegeben.
- Das Modell ist zu komplex und daher zu zeitaufwendig für die kontinuierliche Simulation.

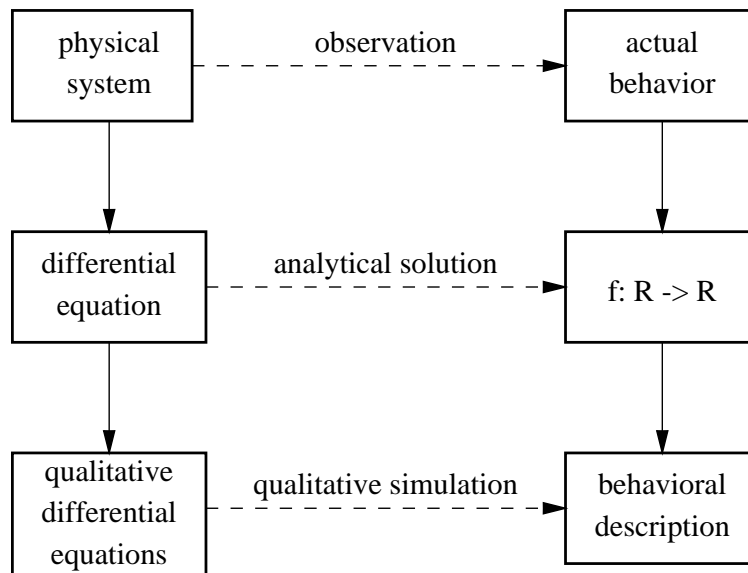


Abbildung 1.1: Die verschiedenen Abstraktionsebenen eines physikalischen Systems.

- Bestimmte Anwendungen benötigen gar nicht die detaillierten Ergebnisse der kontinuierlichen Simulation. Qualitative Aussagen sind ausreichend.

Aufgrund dieser Eigenschaften wird die qualitative Simulation immer öfter auch in technischen Anwendungen, wie zum Beispiel in der Prozeß- und Designüberwachung oder in der Fehlerdiagnose, eingesetzt.

Der bekannteste Algorithmus für die qualitative Simulation ist QSIM [Kui94].. Am Institut für Technische Informatik wurde ein Forschungsprojekt gestartet, um die Laufzeit von QSIM zu reduzieren. Im Rahmen dieses Projektes werden Coprozessoren für rechenzeitintensive Teilfunktionen entwickelt und Parallelisierungsmöglichkeiten untersucht.

1.2 Zielsetzung

In dieser Arbeit wird die parallele Implementierung von rechenzeitintensiven Teilaufgaben von QSIM aufgezeigt und bewertet. Die Ziele der parallelen Implementierung werden in den folgenden Absätzen aufgelistet.

Verarbeitungszeit: Für QSIM Anwendungen werden schnelle Simulatoren benötigt. Ein Schwerpunkt des Forschungsprojektes und dieser Arbeit liegt daher darin, die Verarbeitungszeit gegenüber bestehenden Implementierungen von QSIM zu verbessern.

Skalierbarkeit: Ein weiterer wichtiger Aspekt einer parallelen Implementierung ist ein skalierbares Design. Für eine höhere Prozessoranzahl soll auch die Leistung des Systems steigen.

Speedup: Das Verhältnis der Laufzeiten des Einprozessorsystems zum Mehrprozessorsystem wird untersucht und bewertet. Die Beschreibung dieses Verhältnisses erfolgt mit Hilfe des Speedups.

Die Implementierung erfolgt auf einer Multi-DSP Architektur. Als Prozessor wird der digitale Signalprozessor TMS320C40 von Texas Instruments verwendet. Ein Transtech Multi-C40 Board bildet die Plattform für die Implementierung. Dieses System ist als PC-Einsteckkarte ausgeführt. Die Programmierung der Prozessoren erfolgt mit dem verteilten Echtzeit-Betriebssystem Virtuoso.

1.3 Gliederung

In **Kapitel 2** wird QSIM allgemein erläutert. Die Ansätze für eine parallele Implementierung werden dabei mitbehandelt. Es zeigt sich, daß gewisse rechenzeitintensive Aufgaben für eine parallele Implementation geeignet sind. Die parallelisierbaren Aufgaben werden dann charakterisiert und auf ihre spezifischen Eigenschaften untersucht.

In **Kapitel 3** wird das Scheduling-Problem nach grundlegenden Kriterien kategorisiert. Mit dieser Einteilung und der Charakterisierung aus dem Kapitel 2 werden Schedulingverfahren beschrieben. Es werden Heuristiken für verschiedene Schedulingverfahren vorgestellt und die Abweichung zur Optimallösung beschrieben. Die Heuristiken werden auch auf ihre Komplexität untersucht.

Kapitel 4 beschreibt die Implementierung und die experimentellen Ergebnisse. Im ersten Abschnitt wird die Multi-DSP Architektur und das verwendete Betriebssystem beschrieben. Der zweite Abschnitt beschäftigt sich mit der konkreten parallelen Implementierung der QSIM Kernfunktionen.

Eine Diskussion der parallelen Implementierung von QSIM und einen Ausblick auf weitere Entwicklungen an der parallelen Implementierung von QSIM werden in **Kapitel 5** angeführt und bilden den Abschluß dieser Arbeit.

Kapitel 2

Überschrift Kapitel zwei

Neue Kapitel beginnen automatisch immer auf einer neuen Seite.

Die Schriftgrößen für die Überschriften der einzelnen Unterabschnitte erfolgen automatisch abgestuft. Ab der 3. Unterebene erfolgt keine Numerierung mehr.

2.1 Überschrift Abschnitt

2.2 Überschrift nächster Abschnitt

2.2.1 Überschrift Unterabschnitt

Überschrift Unter-Unterabschnitt

2.3 Überschrift NochEinAbschnitt

Kapitel 3

Überschrift Kapitel drei

Die Kapitel sollten jedenfalls die Aufgabenstellung, den Stand der Technik, die Vorgehensweise zur Lösung des Problems und die Lösung enthalten. Programmlistings sollten nicht Teil der Diplomarbeit sein. Evtl. können elegante Lösungen bzw. wesentliche Elemente dargestellt werden. Hinweise auf den Anhang und (gekürzte) Auszüge dort sind aber zulässig.

3.1 Überschrift Abschnitt 3.1

- Es sollte eine sinnvolle Aufteilung in Text-Files (Kapitel, Abschnitte, Unterabschnitte) erfolgen.
 - Ziehen Sie sinnvolle Label-Bezeichner mit den Überschriften mit.
 - Sortieren Sie die Bilder in Unterverzeichnisse, gleich mit Label und Caption versehen.
1. Listen kann man auch numerieren.
 2. Auch bei Querverweisen kann man sich auf \LaTeX verlassen. Hier wird auf Kapitel 2 verwiesen.
 3. Das Referenzieren funktioniert ebenfalls sehr einfach. Setzen Sie die Referenzen immer innerhalb eines Satzes. Beispiel dazu sind:
 - Wie in [Voa97] gezeigt, ... oder
 - Referenzieren von Quellen ist sehr einfach [Kui94].

Die Quellenangaben werden am besten in einer eigenen Datei erfasst (hier `diplom.bib`) und mit dem Programm `bibtex` in eine eigene \LaTeX -Datei (hier `diplom.bbl`) umgewandelt.

Kapitel 4

Überschrift NochEinKapitel

Tabellen können einfach erzeugt werden. Ein Beispiel dazu ist in Tabelle 4.1 angegeben.

<i>observ. variables</i>	<i>subspaces</i>	<i>sampling rate [Hz]</i>	<i>t_{real} [s]</i>	<i>t_{mon} [s]</i>
T1, T2, T3	1	0,1	90,0	3,56
		1	90,0	1,26
		10	89,6	2,20
T1, T2, T3	32	0,1	90,0	110,84
		1	88,0	32,10
		10	87,7	26,60
T2	1	0,1	150,0	4,58
		1	146,0	1,55
		10	145,1	3,33
T2	32	0,1	150,0	139,20
		1	136,0	29,00
		10	134,3	40,70

Tabelle 4.1: Laufzeit des Algorithmus. Hier kann zusätzlicher Text angegeben werden, der nicht im Tabellenverzeichnis aufscheint.

Ähnliches gilt auch für mathematische Formeln. Sie können entweder direkt im Text, wie z.B. $a^2 + b^2 = c^2$, oder als eigenes Element mit oder ohne Nummerierung angegeben werden.

$$\mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1}, \mathbf{u}_{t-1}, \mathbf{p}_{t-1}) \quad (4.1)$$

Kapitel 5

Schlußbemerkung und Ausblick

Anhang A

z.B. Begriffsbestimmung

A.1 z.B. Definitionen

A.2 z.B. Verwendete Symbole

Literaturverzeichnis

- [GPW93] Robert Ginthör, Marco Platzner, and Reinhold Weiss. Interprocessor Communication in Distributed Transputer Systems. Technical Report 355, IIG-Report-Series, February 1993.
- [Kui94] Benjamin Kuipers. *Qualitative Reasoning: Modeling and Simulation with Incomplete Knowledge*. Artificial Intelligence. MIT Press, 1994.
- [Man97] Christian Mandl. *Real-Time Searching Coprocessor Architectures*. PhD thesis, Graz University of Technology, 1997.
- [Sei88] Peter Ewald Seifter. Interaktives Grafiksystem zur Erstellung und Simulation hydraulischer Schaltungen. Master's thesis, Institute for Technical Informatics, Graz University of Technology, 1988.
- [Voa97] Jeffrey Voas. A defensive approach to certifying cots software. *RSTR-002-97-002.01, Reliable Software Technologies Corporation*, August 1997.
- [WB92] Reinhold Weiss and Eugen Brenner. Ein verteiltes Echtzeit-Expertensystem auf Transputerbasis. In *Comett Seminar: Industrielle Anwendungen paralleler Computersysteme*, pages 3–9, Graz, Austria, 1992.
- [Wei92] Reinhold Weiss. *Verteilte Echtzeitsysteme*. Automatisierungstechnisches Kolloquium der FESTO Akademie, Stuttgart, Germany, March 1992.