



Research Report Series

Lehrstuhl für Rechnertechnik und  
Rechnerorganisation (LRR-TUM)  
Technische Universität München

<http://www.bode.in.tum.de/>

Editor: Prof. Dr. A. Bode

Vol. 32

---

---

**Automatic Performance Analysis for  
Memory Hierarchies and Threaded  
Applications on SMP Systems**

---

---

**Edmond Kereku**

**SHAKER**

**V E R L A G**

Aachen 2006

**Bibliographic information published by Die Deutsche Bibliothek**

Die Deutsche Bibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data is available in the internet at <http://dnb.ddb.de>.

Zugl.: München, Techn. Univ., Diss., 2006

Copyright Shaker Verlag 2006

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission of the publishers.

Printed in Germany.

ISBN-10: 3-8322-5408-0

ISBN-13: 978-3-8322-5408-7

ISSN 1432-0169

Shaker Verlag GmbH • P.O. BOX 101818 • D-52018 Aachen

Phone: 0049/2407/9596-0 • Telefax: 0049/2407/9596-9

Internet: [www.shaker.de](http://www.shaker.de) • e-mail: [info@shaker.de](mailto:info@shaker.de)

---

---

# Abstract

---

---

SMP systems rank among the most commonly used parallel computer architectures. As powerful single machines or as nodes of clusters and supercomputers, SMPs provide the computing power needed by a wide range of HPC applications. Besides concurrently using the multiple resources offered by SMPs, a primary requirement for such applications is the efficient use of resources. Thus, software developers spend a considerable effort on finding the performance bottlenecks and on optimizing their programs.

One of the common sources of inefficiency in parallel programs is the poor utilization of the memory architectures. Due to the high complexity of parallel computers, finding such problems in the application is a difficult task even for experienced developers. This thesis addresses this problem by introducing an approach for the automatic detection of memory hierarchy-related performance bottlenecks of parallel programs running on SMP systems.

The approach includes new concepts for online monitoring of multithreaded applications. A highly configurable monitoring architecture is developed which detects cache-related performance problems for selected code regions and data structures of the program by using the performance counters available in most microprocessors. A well defined interface, named MRI, is used by the performance tools to configure the monitoring architecture and to retrieve the collected performance data in the form of profiles, traces, and memory access histograms.

The main research topic is the automatic analysis of the memory access behavior of multithreaded applications based on the performance data delivered by the monitoring architecture. The collected performance data is captured in an object-oriented data model and the performance bottlenecks are defined in terms of performance properties using the APART Specification Language (ASL).

Several strategies are specified which perform an online search for performance properties on the code regions and the data structures of the program. The strategies are built using so called strategy bricks. A strategy brick specifies an algorithm which may refine the search either on code regions and data structures or on the performance problem itself. The automatic analysis concepts are implemented in an automatic performance tool called AMEBA.

---

---

# Kurzfassung

---

---

Von großer Bedeutung für die effiziente Nutzung heutiger Parallelrechner ist neben der Parallelisierung der Anwendungen die Optimierung des Speicherzugriffsverhaltens. Im Rahmen dieser Doktorarbeit wurden neue Methoden zur automatischen Analyse der Speicherhierarchienutzung entwickelt. Der Focus liegt hierbei auf der Optimierung von OpenMP-Anwendungen auf SMP-Systemen, die als einzelne Parallelrechner, aber auch als Komponenten hochparalleler Systeme eingesetzt werden.

Die Analyseumgebung basiert auf einer Hardware zur adressbezogenen Erfassung von Ereignissen in der Speicherhierarchie, z.B., Cache-Fehler und -Treffer für die einzelnen Cache-Stufen. Diese Hardware-Zähler, die u.a. auf Itanium-Prozessoren verfügbar sind, werden durch ein weitgehend konfigurierbares Monitorsystem kontrolliert. Die Konfiguration erfolgt über das Monitor Request Interface (MRI).

Die eigentliche Analyse des Zugriffsverhaltens erfolgt automatisch anhand einer vordefinierten Menge von Leistungseigenschaften, die in der APART Spezifikationsprache (ASL) vorgegeben werden. Die Analysekomponente verwendet anwendungsspezifische Suchstrategien um Leistungsengpässe automatisch zu erkennen.