

Lecture Notes in Computer Science

Edited by G. Goos, J. Hartmanis and J. van Leeuwen

1656

Springer

Berlin

Heidelberg

New York

Barcelona

Hong Kong

London

Milan

Paris

Singapore

Tokyo

Siddhartha Chatterjee Jan F. Prins
Larry Carter Jeanne Ferrante Zhiyuan Li
David Sehr Pen-Chung Yew (Eds.)

Languages and Compilers for Parallel Computing

11th International Workshop, LCPC'98
Chapel Hill, NC, USA, August 7-9, 1998
Proceedings



Springer

Volume Editors

Siddhartha Chatterjee, Jan F. Prins
Department of Computer Science, The University of North Carolina
Chapel Hill, NC 27599-3175, USA
E-mail: {sc/prins}@cs.unc.edu

Larry Carter, Jeanne Ferrante
Department of Computer Science and Engineering
University of California at San Diego
9500 Gilman Drive, La Jolla, CA 92093-0114, USA
E-mail: {carter/ferrante}@cs.ucsd.edu

Zhiyuan Li
Department of Computer Science, Purdue University
1398 Computer Science Building, West Lafayette, IN 47907, USA
E-mail: li@cs.purdue.edu

David Sehr
Intel Corporation
2200 Mission College Boulevard, RN6-18, Santa Clara, CA 95052, USA
E-mail: dsehr@gomez.sc.intel.com

Pen-Chung Yew
Department of Computer Science and Engineering, University of Minnesota
Minneapolis, MN 55455, USA
E-mail: yew@cs.umn.edu

Cataloging-in-Publication data applied for

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Languages and compilers for parallel computing : 11th
international workshop ; proceedings / LCPC '98, Chapel Hill, NC,
USA, August 7 - 9, 1998. S. Chatterjee ... (ed.). - Berlin ; Heidelberg
; New York ; Barcelona ; Hong Kong ; London ; Milan ; Paris ;
Singapore ; Tokyo : Springer, 1999
(Lecture notes in computer science ; Vol. 1656)
ISBN 3-540-66426-2

CR Subject Classification (1998): D.1.3, D.3.4, F.1.2, B.2.1, C.2

ISSN 0302-9743

ISBN 3-540-66426-2 Springer-Verlag Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag. Violations are liable for prosecution under the German Copyright Law.

© Springer-Verlag Berlin Heidelberg 1999
Printed in Germany

Typesetting: Camera-ready by author
SPIN: 10704088 06/3142 - 5 4 3 2 1 0 Printed on acid-free paper

Steering Committee

Utpal Banerjee	<i>Intel Corporation</i>
David Gelernter	<i>Yale University</i>
Alex Nicolau	<i>University of California at Irvine</i>
David Padua	<i>University of Illinois at Urbana-Champaign</i>

Program Committee

Larry Carter	<i>University of California at San Diego</i>
Siddhartha Chatterjee	<i>University of North Carolina at Chapel Hill</i>
Jeanne Ferrante	<i>University of California at San Diego</i>
Zhiyuan Li	<i>Purdue University</i>
Jan Prins	<i>University of North Carolina at Chapel Hill</i>
David Sehr	<i>Intel Corporation</i>
Pen-Chung Yew	<i>University of Minnesota</i>

Organizing Committee

Linda Houseman	<i>University of North Carolina at Chapel Hill</i>
----------------	--

External Reviewers

George Almasi	Asheesh Khare
Ana Azevedo	Jaejin Lee
Brian Blount	Yuan Lin
Calin Cascaval	Yunheung Paek
Walfredo Cirne	Nick Savoio
Paolo D'Alberto	Martin Simons
Vijay Ganesh	Weiyu Tang
Xiaomei Ji	

LCPC'98 Steering and Program Committes for their time and energy in reviewing the submitted papers. Finally, and most importantly, we thank all the authors and participants of the workshop. It is their significant research work and their enthusiastic discussions throughout the workshop that made LCPC'98 a success.

May 1999

Siddhartha Chatterjee
Program Chair

Preface

The year 1998 marked the eleventh anniversary of the annual Workshop on Languages and Compilers for Parallel Computing (LCPC), an international forum for leading research groups to present their current research activities and latest results. The LCPC community is interested in a broad range of technologies, with a common goal of developing software systems that enable real applications. Among the topics of interest to the workshop are language features, communication code generation and optimization, communication libraries, distributed shared memory libraries, distributed object systems, resource management systems, integration of compiler and runtime systems, irregular and dynamic applications, performance evaluation, and debuggers. LCPC'98 was hosted by the University of North Carolina at Chapel Hill (UNC-CH) on 7 - 9 August 1998, at the William and Ida Friday Center on the UNC-CH campus. Fifty people from the United States, Europe, and Asia attended the workshop.

The program committee of LCPC'98, with the help of external reviewers, evaluated the submitted papers. Twenty-four papers were selected for formal presentation at the workshop. Each session was followed by an open panel discussion centered on the main topic of the particular session. Many attendees have come to regard the open panels as a very effective format for exchanging views and clarifying research issues. Using feedback provided both during and after the presentations, all of the authors were given an opportunity to improve their papers before submitting the final manuscript contained in this volume. This collection documents important research activities from the past year in the design and implementation of programming languages and environments for parallel computing.

The major themes of the workshop included both classical issues (Fortran, instruction scheduling, dependence analysis) as well as emerging areas (Java, memory hierarchy issues, network computing, irregular applications). These themes reflect several recent trends in computer architecture: aggressive hardware speculation, deeper memory hierarchies, multilevel parallelism, and “the network is the computer.” In this final editing of the workshop papers, we have grouped the papers into these categories.

In addition to the regular paper sessions, LCPC'98 featured an invited talk by Charles Leiserson, Professor of Computer Science at the MIT Laboratory for Computer Science, entitled “Algorithmic Multithreaded Programming in Cilk”. This talk was the first exposure to the Cilk system for many of the participants and resulted in many interesting discussions. We thank Prof. Leiserson for his special contribution to LCPC'98.

We are grateful to the Department of Computer Science at UNC-CH for its generous support of this workshop. We benefited especially from the efforts of Linda Houseman, who ably coordinated the logistical matters before, during, and after the workshop. Thanks also go out to our local team of volunteers: Brian Blount, Vibhor Jain, and Martin Simons. Special thanks are due to the

Table of Contents

Java

From Flop to MegaFlops: Java for Technical Computing	1
<i>J. E. Moreira, S. P. Midkiff and M. Gupta</i> (IBM T.J. Watson Research Center)	

Considerations in HPJava Language Design and Implementation	18
<i>Guansong Zhang, Bryan Carpenter, Geoffrey Fox, Xinying Li</i> <i>and Yuhong Wen</i> (Syracuse University)	

Locality

A Loop Transformation Algorithm Based on Explicit Data Layout Representation for Optimizing Locality	34
<i>M. Kandemir</i> (Northwestern University), <i>J. Ramanujam</i> (Louisiana State University), <i>A. Choudhary</i> (Northwestern University) <i>and</i> <i>P. Banerjee</i> (Northwestern University)	

An Integrated Framework for Compiler-Directed Cache Coherence and Data Prefetching	51
<i>Hock-Beng Lim</i> (University of Illinois) <i>and Pen-Chung Yew</i> (University of Minnesota)	

I/O Granularity Transformations	68
<i>Gagan Agrawal</i> (University of Delaware)	

Network Computing

<i>Stampede</i> : A Programming System for Emerging Scalable Interactive Multimedia Applications	83
<i>Rishiyur S. Nikhil</i> (Compaq), <i>Umakishore Ramachandran</i> (Georgia Tech), <i>James M. Rehg</i> (Compaq), <i>Robert H. Halstead, Jr.</i> (Curl Corporation), <i>Christopher F. Joerg</i> (Compaq) <i>and Leonidas Kontothanassis</i> (Compaq)	

Network-Aware Parallel Computing with Remos	100
<i>Bruce Lowekamp, Nancy Miller, Dean Sutherland, Thomas Gross,</i> <i>Peter Steenkiste and Jaspal Subhlok</i> (Carnegie Mellon University)	

Object-Oriented Implementation of Data-Parallelism on Global Networks	120
<i>Jan Borowiec</i> (GMD FIRST)	

Fortran

Optimized Execution of Fortran 90 Array Language on Symmetric Shared-Memory Multiprocessors	131
<i>Vivek Sarkar</i> (IBM T.J. Watson Research Center)	

Fortran RED — A Retargetable Environment for Automatic Data Layout	148
<i>Ulrich Kremer</i> (Rutgers University)	

Automatic Parallelization of C by Means of Language Transcription	166
<i>Richard L. Kennell and Rudolf Eigenmann</i> (Purdue University)	

Irregular Applications

Improving Compiler and Run-Time Support for Irregular Reductions Using Local Writes	181
<i>Hwansoo Han and Chau-Wen Tseng</i> (University of Maryland)	

Beyond Arrays — A Container-Centric Approach for Parallelization of Real-World Symbolic Applications	197
<i>Peng Wu and David Padua</i> (University of Illinois)	

SIPR: A New Framework for Generating Efficient Code for Sparse Matrix Computations	213
<i>William Pugh and Tatiana Shpeisman</i> (University of Maryland)	

HPF-2 Support for Dynamic Sparse Computations	230
<i>R. Asenjo</i> (University of Málaga), <i>O. Plata</i> (University of Málaga), <i>J. Touriño</i> (University of La Coruña), <i>R. Doallo</i> (University of La Coruña) and <i>E.L. Zapata</i> (University of Málaga)	

Instruction Scheduling

Integrated Instruction Scheduling and Register Allocation Techniques	247
<i>David A. Berson</i> (Intel Corporation), <i>Rajiv Gupta</i> (University of Pittsburgh) and <i>Mary Lou Soffa</i> (University of Pittsburgh)	

A Spill Code Placement Framework for Code Scheduling	263
<i>Dingchao Li, Yuji Iwahori, Tatsuya Hayashi and Naohiro Ishii</i> (Nagoya Institute of Technology)	

Copy Elimination for Parallelizing Compilers	275
<i>David J. Kolson, Alexandru Nicolau and Nikil Dutt</i> (University of California, Irvine)	

Potpourri

Compiling for SIMD Within a Register	290
<i>Randall J. Fisher and Henry G. Dietz</i> (Purdue University)	

Automatic Analysis of Loops to Exploit Operator Parallelism on Reconfigurable Systems	305
<i>Narasimhan Ramasubramanian, Ram Subramanian</i> and <i>Santosh Pande</i> (University of Cincinnati)	

Principles of Speculative Run-Time Parallelization	323
<i>Devang Patel and Lawrence Rauchwerger</i> (Texas A&M University)	

Dependence Analysis

The Advantages of Instance-Wise Reaching Definition Analyses in Array (S)SA	338
<i>Jean-François Collard</i> (University of Versailles)	
Dependency Analysis of Recursive Data Structures Using Automatic Groups	353
<i>D. K. Arvind and T. A. Lewis</i> (The University of Edinburgh)	
The I+ Test	367
<i>Weng-Long Chang and Chih-Ping Chu</i> (National Cheng Kung University)	
Author Index	383