Lecture Notes in Computer Science 1239

David Sehr Utpal Banerjee David Gelernter
Alex Nicolau David Padua (Eds.)

Languages and Compilers for Parallel Computing

9th International Workshop, LCPC'96
San Jose, California, USA, August 8-10, 1996
Proceedings

Springer

Volume Editors

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

David Gelernter
Yale University, Department of Computer Science
51 Prospect Street, New Haven, CT 06520, USA
E-mail: gelernter@cs.yale.edu

Alex Nicolau
University of California, Department of Information and Computer Science
Irvine, CA 92717, USA
E-mail: nicolau@ics.uci.edu

David Padua
University of Illinois, Department of Computer Science
1308 West Main Street, Urbana, IL61801, USA
E-mail: padua@cs.uiuc.edu

Cataloging-in-Publication data applied for

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Languages and compilers for parallel computing : 9th international workshop ; proceedings / LCPC '96, San Jose, California, USA, August 8 - 10, 1996. David Sehr ... (ed.). - Berlin ; Heidelberg ; New York ; Barcelona ; Budapest ; Hong Kong ; London ; Milan ; Paris ; Tokyo : Springer, 1997
(Lecture notes in computer science ; Vol. 1239)
ISBN 3-540-63091-0

CR Subject Classification (1991): D.1.3, D.3.1, D.3.4,F.1.2, B.2.1

ISSN 0302-9743
ISBN 3-540-63091-0 Springer-Verlag Berlin Heidelberg New York

Typesetting: Camera-ready by author
SPIN 10548814 06/3142 – 5 4 3 2 1 0 Printed on acid-free paper

Foreword

My association with the Workshop on Languages and Compilers for Parallel Computing began in 1988, when the first one was hosted by Cornell University. That year the number of attendees was small enough to accomodate in a conference room, and the list read like a "who's who" of high-performance computing. We heard excellent presentations on a variety of topics, and the discussion was extremely lively. As a graduate student presenting his first paper, let alone a paper about Prolog, I was more than a little intimidated to address that group. But despite a few tough questions, I survived mostly intact and gained an insight into just how valuable interchanging ideas with such people can be. This workshop, because it brings together people from across the range of parallel computing, serves the purpose of generating new ideas and thoroughly reviewing existing ones. This workshop has traditionally drawn the best machine architecture, systems software, and algorithms people in the field to accomplish those tasks.

Moreover, having twice presented my work on Prolog, this workshop has always held a special place for me by its presentations tackling high performance computing through "non-traditional" approaches. In the past, papers have been presented running the gamut of non-imperative languages, including functional, logic, and constraint languages. Program transformation systems, visual programming aids, and other tools to aid parallel program development have also found a home in the workshop.

Of course the primary focus remains high-performance scientific and engineering computing, so Fortran retains a place of honor in the workshop. Because of this, many great papers on dependence analysis, loop transformations, instruction scheduling, and the like have been presented. As the technology for parallelizing Fortran has become more mature, though, the workshop has made room for explicit parallelism and data distribution directives, as well as venturing into C and more recently C++. In the current volume we find several papers addressing C and C++, and one even addressing extensions to those languages.

This year the ninth workshop was held from August 8 to August 10, 1996 at the Red Lion Hotel in San Jose, California. The attendance this year was approximately 100 people, representing fifteen U.S. states and ten other countries. Although the workshop has grown well beyond the size to fit in a conference room, it continues to have a single track and a relatively leisurely schedule. A very successful addition this year was the poster session, which afforded considerable interaction for poster session authors. The sponsor this year was my employer, Intel Corporation.

No effort such as this comes to fruition without the efforts of many. I would first like to thank the members of the committee, Utpal Banerjee, David Gelernter, Alex Nicolau, and David Padua, for having given me the chance to host the workshop and work with them. I would also like to thank the referees, especially those who gave reviews with very short notice. It is through the efforts of such conscientious reviewers that we are able to keep the workshop going. A very special mention goes to the two people who worked most closely with me on site arrangements, Karen Kmetz and Diane Proudfoot; they gave extremely gracious and helpful responses to even my most ridiculous requests. Also deserving many thanks are my colleagues Knud Kirkegaard and Peter Jensen, who helped with a variety of tasks. Dr. Robert Colwell, recently appointed an Intel Fellow, more than earned my thanks by consenting to give his terrific keynote speech. A last note of thanks goes to my superiors at Intel for allowing me the time to work on this task, and to Intel for sponsoring the event.

March 1997 David Sehr

Program Chair

List of Referees

Tarek Abdelrahman
Tito Autrey
Eduard Ayguadé
Milind Bhandarka
Dong-Yuan Chen
William Chen
Fabien Coelho
Béatrice Creusillet
Luiz De Rose
Rudi Eigenmann
Milind Girkar
John Grout
Junjie Frank Gu
Attila Gursoy
Mohammad Haghighat
Jay Hoeflinger
François Irigoin
Roy Ju
Knud Kirkegaard
Venkata Krishnan
Robert Kuhn
Jaejin Lee
Yong-Fong Lee
Calvin Lin
Bret Andrew Marsolf
Samuel Midkiff
Jose Moreira
Trung Nguyen
David Padua
Yunheung Paek
Vishesh Parikh
Paul Petersen
Bill Pottenger
Lawrence Rauschwerger
Martin Rinard
Hideki Saito
Dale Schouten
David Sehr
Ajay Sethi
Xiaowei Shen
Nicholas Stavrakos
Ernesto Su
Pedro Trancoso
Jenn-Yuan Tsai
Peng Tu
Pen-Chung Yew

Table of Contents

Automatic Data Distribution and Locality Enhancement

Cross-Loop Reuse Analysis and Its Application to Cache Optimizations 1
K. Cooper, K. Kennedy, N. McIntosh, *Rice University, Houston*

Locality Analysis for Distributed Shared-Memory Multiprocessors 20
V. Sarkar, *Massachusetts Institute of Technology, Cambridge*
G. R. Gao, *University of Delaware, Newark*
S. Han, *McGill University, Montreal*

Data Distribution and Loop Parallelization for Shared-Memory Multiprocessors 41
E. Ayguadé, J. Garcia, M. L. Grande, J. Labarta, *Polytechnic University of Catalunya, Barcelona*

Data Localization Using Loop Aligned Decomposition for Macro-Dataflow Processing 56
A. Yoshida, H. Kasahara, *Waseda University, Tokyo*

Program Analysis

Exploiting Monotone Convergence Functions in Parallel Programs 75
W. Pugh, E. Rosser, T. Shpeisman, *University of Maryland, College Park*

Exact versus Approximate Array Region Analyses 86
B. Creusillet, F. Irigoin, *École des mines de Paris, France*

Context-Sensitive Interprocedural Analysis in the Presence of Dynamic Aliasing 101
P. W. Sathyanathan, M. S. Lam, *Stanford University, California*

Initial Results for Glacial Variable Analysis 120
T. Autrey, M. Wolfe, *Oregon Graduate Institute, Portland*

Compiler Algorithms for Fine-Grain Parallelism

Compiler Algorithms on If-Conversion, Speculative Predicate Assignment and Predicated Code Optimizations 135
J. Fang, *Intel Corporation, Santa Clara*

Determining Asynchronous Pipeline Execution Times 154
V. Donaldson, J. Ferrante, *University of California, San Diego*

Compiler Techniques for Concurrent Multithreading with Hardware Speculation Support 175
J.-Y. Tsai, *University of Illinois, Urbana*
Z. Li, X. Wang, P.-C. Yew, B. Zhang, *University of Minnesota, Minneapolis*

Resource-Directed Loop Pipelining 192
S. Novack, A. Nicolau, *University of California, Irvine*

Instruction Scheduling and Register Allocation

Integrating Program Optimizations and Transformations with the Scheduling of Instruction Level Parallelism 207
D. A. Berson, P. Chang, *Intel Corporation, Santa Clara*
R. Gupta, M. L. Soffa, *University of Pittsburgh, Pennsylvania*

Bidirectional Scheduling: A New Global Code Scheduling Approach 222
P. Chang, D.-Y. Chen, Y.-F. Lee, Y. Wu, U. Banerjee, *Intel Corporation, Santa Clara*

Parametric Computation of Margins and of Minimum Cumulative Register Lifetime Dates 231
B. D. de Dinechin, *McGill University, Montreal*

Global Register Allocation Based on Graph Fusion 246
G.-Y. Lueh, A.-R. Adl-Tabatabai, *Carnegie-Mellon University, Pittsburgh*
T. Gross, *Carnegie-Mellon University, Pittsburgh & ETH, Zurich*

Parallelizing Compilers

Automatic Parallelization for Non-cache Coherent Multiprocessors 266
Y. Paek, D. A. Padua, *University of Illinois, Urbana*

Lock Coarsening: Eliminating Lock Overhead in Automatically Parallelized Object-Based Programs 285
P. Diniz, M. Rinard, *University of California, Santa Barbara*

Are Parallel Workstations the Right Target for Parallelizing Compilers? 300
R. Eigenmann, I. Park, M. J. Voss, *Purdue University, West Lafayette*

Optimal Reordering and Mapping of a Class of Nested-Loops for Parallel Execution 315
C.-C. Lam, P. Sadayappan, R. Wenger, *The Ohio State University, Columbus*

Communication Optimization

Communication-Minimal Tiling of Uniform Dependence Loops . 330
J. Xue, *University of New England, Armidale*

Communication-Minimal Partitioning of Parallel Loops and Data Arrays for Cache-Coherent Distributed-Memory Multiprocessors 350
R. Barua, D. Kranz, A. Agarwal, *Massachusetts Institute of Technology, Cambridge*

Resource-Based Communication Placement Analysis 369
K. Kennedy, A. Sethi, *Rice University, Houston*

Statement-Level Communication-Free Partitioning Techniques for Parallelizing Compilers 389
K.-P. Shih, J.-P. Sheu, *National Central University, Taiwan*
C.-H. Huang, *The Ohio State University, Columbus*

Generalized Overlap Regions for Communication Optimization in Data-Parallel Programs 404
A. Venkatachar, J. Ramanujam, A. Thirumalai, *Louisiana State University, Baton Rouge*

Compiling HPF

Optimizing the Representation of Local Iteration Sets and Access Sequences for Block-Cyclic Distributions 420
S. P. Midkiff, *IBM T. J. Watson Research Center, Yorktown Heights*

Interprocedural Array Redistribution Data-Flow Analysis 435
D. J. Palermo, *Hewlett-Packard Company, Richardson*
E. W. Hodges IV, *SAS Institute, Cary*
P. Banerjee, *Northwestern University, Evanston*

HPF on Fine-Grain Distributed Shared Memory: Early Experience 450
S. Chandra, J. R. Larus, *University of Wisconsin, Madison*

New Languages and New Compiler Techniques

Simple Qualitative Experiments with a Sparse Compiler 466
A. J. C. Bik, H. A. G. Wijshoff, *Leiden University, The Netherlands*

Factor-Join: A Unique Approach to Compiling Array Languages for Parallel Machines 481
B. L. Chamberlain, S.-E. Choi, E. C. Lewis, L. Snyder,
W. D. Weathersby, *University of Washington, Seattle*
C. Lin, *University of Texas, Austin*

Compilation of Constraint Systems to Procedural Parallel Programs 501
A. John, J. C. Browne, *University of Texas, Austin*

A Multithreaded Substrate and Compilation Model for the Implicitly Parallel Language pH 519
Arvind, A. Caro, J.-W. Maessen, *Massachusetts Institute of Technology, Cambridge*
S. Aditya, *Hewlett-Packard Laboratories*

Run Time Control of Parallelism

Threads for Interoperable Parallel Programming 534
L. V. Kalé, J. Yelon, T. Knauff, *University of Illinois, Urbana*

A Programming Environment for Dynamic Resource Allocation and Data Distribution 553
J. E. Moreira, V. K. Naik, R. B. Konuru, *IBM T. J. Watson Research Center, Yorktown Heights*

Dependence Driven Execution for Data Parallelism 570
S. Vajracharya, D. Grunwald, *University of Colorado, Boulder*

Poster Session Papers

σ-SSA and Its Construction Through Symbolic Interpretation . 585
H. Saito, C. D. Polychronopoulos, *University of Illinois, Urbana*

Compiler Support for Maintaining Cache Coherence Using Data Prefetching (Extended Abstract) 588
H.-B. Lim, *University of Illinois, Urbana*
L. Choi, *Intel Corporation, Santa Clara*
P.-C. Yew, *University of Minnesota, Minneapolis*

3D Visualization of Program Structure and Data Dependence for Parallelizing Compilers and Parallel Programming 591
M. Sasakura, *Okayama University, Okayama*
S. Kiwada, K. Joe, T. Nakanishi, *Nara Institute of Science and Technology, Nara*
K. Araki, *Kyushu University, Fukuoka*

Side Effect Analysis on User-Defined Reduction Functions with Dynamic Pointer-Linked Data Structures 594
Y.-S. Hwang, J. Saltz, *University of Maryland, College Park*

Estimating Minimum Execution Time of Perfect Loop Nests with Loop-Carried Dependences 597
T. Nakanishi, K. Joe, A. Fukuda, *Nara Institute of Science and Technology, Nara*
C. D. Polychronopoulos, *University of Illinois, Urbana*
K. Araki, *Kyushu University, Fukuoka*

Automatic Data and Computation Partitioning on Scalable Shared Memory Multiprocessors 600
S. Tandri, T. S. Abdelrahman, *The University of Toronto, Toronto*

The Loop Parallelizer LooPo – Announcement 603
M. Griebl, C. Lengauer, *Universität Passau, Passau*

A Generalized Forall Concept for Parallel Languages 605
P. F. G. Dechering, L. C. Breebaart, F. Kuijlman, C. van Reeuwijk, H. J. Sips, *Delft University, The Netherlands*

Memory Optimizations in the Intel Reference Compiler 608
K. Sridharan, P. Chang, U. Banerjee, R. Narayanaswamy, S. Rao, *Intel Corporation, Santa Clara*

Author Index 611