Lecture Notes in Computer Science 892
Edited by G. Goos, J. Hartmanis and J. van Leeuwen

Advisory Board: W. Brauer D. Gries J. Stoer

K. Pingali U. Banerjee D. Gelernter
A. Nicolau D. Padua (Eds.)

Languages and Compilers for Parallel Computing

7th International Workshop
Ithaca, NY, USA, August 8-10, 1994
Proceedings

Springer-Verlag
Berlin Heidelberg New York
London Paris Tokyo
Hong Kong Barcelona
Budapest

Series Editors

Gerhard Goos
Universität Karlsruhe
Vincenz-Priessnitz-Straße 3, D-76128 Karlsruhe, Germany

Juris Hartmanis
Department of Computer Science, Cornell University
4130 Upson Hall, Ithaka, NY 14853, USA

Jan van Leeuwen
Department of Computer Science, Utrecht University
Padualaan 14, 3584 CH Utrecht, The Netherlands

Volume Editors

Keshav Pingali
Department of Computer Science, Cornell University
Ithaca, NY 14853, USA

Utpal Banerjee
Intel Corporation
2200 Mission College Blvd. P.O. Box 58119, RN6-18
Santa Clara, CA 95052, USA

David Gelernter
Department of Computer Science, Yale University
51 Prospect St., New Haven, CT 06520, USA

Alex Nicolau
Department of Information & Computer Science, University of California
444 Computer Science Bldg., Irvine, CA 92717, USA

David Padua
Center for Supercomputing Research and Development
465 Computer and Systems Research Laboratory
1308 West Main St., Urbana, IL 61801, USA

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

ISBN 3-540-58868-X Springer-Verlag Berlin Heidelberg New York

CIP data applied for

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 1995
Printed in Germany

Typesetting: Camera-ready by author
SPIN: 10479285 45/3140-543210 - Printed on acid-free paper

Foreword

The papers in this volume are revised versions of papers presented at the Seventh Annual Workshop on Languages and Compilers for Parallel Computers which was held in Ithaca, NY August 8th-10th, 1994. This workshop series has traditionally been a forum for the presentation of state-of-the-art research in languages, restructuring compilers and runtime systems by leading groups in the US, Europe and Japan. This year, we received about 45 submissions in response to the call for papers. Because of time constraints, we were unable to accept all papers, and these proceedings contain the 32 papers that were selected for presentation.

Thanks are due to many people for making the workshop a success. The members of the standing program committee — Utpal Banerjee, David Gelernter, Alex Nicolau and David Padua — were a great source of advice and information. Helene Croft put the entire database of names and addresses online, which should make it easier to run this workshop in future years. She was also relentless in tracking down tardy authors, referees, speakers and attendees, which made the workshop run like clockwork! Thanks are due to the referees: Utpal Banerjee, Carrie Brownhill, A. Capitanio, Sudeep Gupta, David Gelernter, Laurie Hendren, J. Hummel, Richard Johnson, Indu Kodukula, David Kolson, Vladimir Kotlyar, Wei Li, Mayan Moudgill, Alex Nicolau, Rishiyur Nikhil, Steve Novack, F. Onion, David Padua, Anne Rogers, Paul Stodghill, Thorsten von Eicken, H. Wang and Richard Zippel. Finally, the Cornell Department of Computer Science, and the Cornell Theory Center gave generous financial grants, which allowed us to reduce the registration fee for participants.

October, 1994

Keshav Pingali
Program Chair

Table of Contents

Starting Small: Fine-Grain Parallelism

Fine-Grain Scheduling under Resource Constraints 1

Paul Feautrier
Université de Versailles, France

Mutation Scheduling: A Unified Approach to Compiling for Fine-Grain Parallelism 16

Steven Novack, Alexandru Nicolau
University of California at Irvine

Compiler Techniques for Fine-Grain Execution on Workstation Clusters Using PAPERS 31

H.G. Dietz, W.E. Cohen, T. Muhammad, T.I. Mattox
Purdue University, West Lafayette, Indiana

Getting Your Ducks in a Row: Alignment and Distribution

Solving Alignment Using Elementary Linear Algebra 46

David Bau, Induprakas Kodukula, Vladimir Kotlyar, Keshav Pingali, Paul Stodghill
Cornell University, Ithaca, New York

Detecting and Using Affinity in an Automatic Data Distribution Tool 61

Eduard Ayguadé, Jordi Garcia, Mercè Gironés, Jesús Labarta, Jordi Torres, Mateo Valero
Polytechnic University of Catalunya, Barcelona, Spain

Array Distribution in Data-Parallel Programs 76

Siddhartha Chatterjee, Robert Schreiber, Thomas J. Sheffler
NASA Ames Research Center, Moffett Field, California
John R. Gilbert
Xerox Research Center, Palo Alto, California

Postlinear Loop Transformations

Communication-Free Parallelization via Affine Transformations . 92

Amy W. Lim, Monica S. Lam
Stanford University, California

Finding Legal Reordering Transformations Using Mappings . . . 107

Wayne Kelly, William Pugh
University of Maryland, College Park

A New Algorithm for Global Optimization for Parallelism and Locality 125

Bill Appelbe, Srinivas Doddapaneni, Charles Hardnett
Georgia Institute of Technology, Atlanta

What Next?

Polaris: Improving the Effectiveness of Parallelizing Compilers . 141

William Blume, Rudolf Eigenmann, Keith Faigin, John Grout, Jay Hoeflinger, David Padua, Paul Petersen, William Pottenger, Lawrence Rauchwerger, Peng Tu, Stephen Weatherford
University of Illinois at Urbana

A Formal Approach to the Compilation of Data-Parallel Languages 155

J.A. Trescher, L.C. Breebaart, P.F.G. Dechering, A.B. Poelman, J.P.M. de Vreught, H.J. Sips
Delft University of Technology, The Netherlands

The Data Partitioning Graph: Extending Data and Control Dependencies for Data Partitioning 170

Tsuneo Nakanishi, Kazuki Joe, Akira Fukuda, Keijiro Araki
Nara Institute of Science and Technology, Japan
Hideki Saito, Constantine D. Polychronopoulos
University of Illinois at Urbana

Back to Basics: Program Analysis

Detecting Value-Based Scalar Dependence 186

Eric Stoltz, Michael Wolfe
Oregon Graduate Institute of Science & Technology, Portland

Minimal Data Dependence Abstractions for Loop Transformations 201

Yi-Qing Yang, Corinne Ancourt, François Irigoin
Ecole des Mines de Paris, France

Differences in Algorithmic Parallelism in Control Flow and Call Multigraphs 217

Vincent Sgro, Barbara G. Ryder
Rutgers University, New Brunswick, New Jersey

Flow-Insensitive Interprocedural Alias Analysis in the Presence of Pointers 234

Michael Burke, Paul Carini, Jong-Deok Choi
IBM Watson Research Center, Yorktown Heights, NY
Michael Hind
State University of New York at New Paltz, and
IBM Watson Research Center, Yorktown Heights, NY

How to Communicate Better

Incremental Generation of Index Sets for Array Statement Execution on Distributed-Memory Machines . . . 251

S.D. Kaushik, C.-H. Huang, P. Sadayappan
The Ohio State University, Columbus

A Unified Data-Flow Framework for Optimizing Communication 266

Manish Gupta, Edith Schonberg, Harini Srinivasan
IBM Watson Research Center, Yorktown Heights, NY

Interprocedural Communication Optimizations for Distributed Memory Compilation 283

Gagan Agrawal, Joel Saltz
University of Maryland, College Park

Automatic Parallelization Considered Unnecessary

Analysis of Event Synchronization in Parallel Programs 300

J. Ramanujam, A. Mathew
Louisiana State University, Baton Rouge

Computing Communication Sets for Control Parallel Programs . 316

Jeanne Ferrante
IBM Watson Research Center, Yorktown Heights, NY
Dirk Grunwald, Harini Srinivasan
University of Colorado, Boulder

Optimizing Parallel SPMD Programs 331

Arvind Krishnamurthy, Katherine Yelick
University of California at Berkeley

Languages for Parallelism: Something Borrow, Something New

An Overview of the Opus Language and Runtime System 346

Piyush Mehrotra, Matthew Haines
Institute for Computer Applications in Science and Engineering, Hampton, Virginia

SIMPLE Performance Results in ZPL 361

Calvin Lin, Lawrence Snyder
University of Washington, Seattle

***Cid*: A Parallel, "Shared-Memory" C for Distributed-Memory Machines** 376

Rishiyur S. Nikhil
Digital Equipment Corporation, Cambridge, Massachusetts

EQ: Overview of a New Language Approach for Prototyping Scientific Computation 391

Thomas Derby, Robert Schnabel, Benjamin Zorn
University of Colorado at Boulder

And Now, for Something Completely Different

Reshaping Access Patterns for Generating Sparse Codes 406

Aart J.C. Bik, Peter M.W. Knijnenburg, Harry A.G. Wijshoff
Leiden University, The Netherlands

Evaluating Two Loop Transformations for Reducing Multiple-Writer False Sharing 423

François Bodin, Thierry Montaut
IRISA, Campus de Beaulieu, Rennes, France
Elana D. Granston
Rice University, Houston, Texas

Parallelizing Tree Algorithms: Overhead vs. Parallelism 440

Jon A. Solworth, Bryan B. Reagan
University of Illinois at Chicago

When Your Program Runs (Finally)

Autoscheduling in a Distributed Shared-Memory Environment . 455

José E. Moreira, Constantine D. Polychronopoulos
University of Illinois at Urbana-Champaign

Optimizing Array Distributions in Data-Parallel Programs 472

Krishna Kunchithapadam, Barton P. Miller
University of Wisconsin at Madison

Automatic Reduction Tree Generation for Fine-Grain Parallel Architectures when Iteration Count is Unknown 487

Kei Hiraki
University of Tokyo, Japan
Satoshi Sekiguchi
Electrotechnical Laboratory, MITI, Ibaraki, Japan