Lecture Notes in Computer Science 1068
Edited by G. Goos, J. Hartmanis and J. van Leeuwen

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

Takayasu Ito Robert H. Halstead, Jr.
Christian Queinnec (Eds.)

Parallel Symbolic Languages and Systems

International Workshop PSLS'95
Beaune, France, October 2-4, 1995
Proceedings

Springer

Volume Editors

Takayasu Ito
Department of Computer and Mathematical Sciences
Graduate School of Information Sciences, Tohoku University
Sendai 980, Japan

Robert H. Halstead, Jr.
Cambridge Research Lab, Digital Equipment Corporation
One Kendall Square, Bldg. 700, Cambridge, MA 02139, USA

Christian Queinnec
Laboratoire d'Informatique de L'Ecole Polytechnique
F-91128 Palaiseau Cedex, France

Cataloging-in-Publication data applied for

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Parallel symbolic languages and systems : proceedings / International Workshop PSLS '95, Beaune, France, October 2 - 4, 1995. Takayasu Ito ... (ed.). - Berlin ; Heidelberg ; New York ; Barcelona ; Budapest ; Hong Kong ; London ; Milan ; Paris ; Santa Clara ; Singapore ; Tokyo : Springer, 1996
(Lecture notes in computer science ; Vol. 1068)
ISBN 3-540-61143-6
NE: Itō, Takayasū [Hrsg.]; PSLS <1995, Beaune>; GT

CR Subject Classification (1991): D.1.3, D.3.3-4, D.2.2, C.1.2, C.2.4, E.1

ISBN 3-540-61143-6 Springer-Verlag Berlin Heidelberg New York

Typesetting: Camera-ready by author
SPIN 10512847 06/3142 – 5 4 3 2 1 0 Printed on acid-free paper

Preface

Parallel symbolic computing is a basic yet challenging area with wide application to high-performance computing. Parallel symbolic computing is characterized by the use of irregular dynamic data structures and irregular data-driven operations on them, frequently involving list structure, recursion, and garbage collection. Parallel Lisp languages and concurrent logic programming languages have been designed and implemented to serve as programming tools for parallel symbolic computing; in recent years many parallel symbolic computing applications have been implemented using C, C++, and their parallelized extensions.

To explore the issues and research directions concerning applications and programming tools for parallel symbolic computing, an international workshop on Parallel Symbolic Languages and Systems (PSLS'95) was held in Beaune, France, on October 2–4, 1995. This workshop is the third in a series that began with the Sendai workshop on Parallel Lisp in 1989 and the Cambridge workshop on Parallel Symbolic Computing in 1992; the proceedings of these workshops were published as Springer LNCS volumes 441 and 748, respectively. We are pleased to publish the proceedings of the Beaune workshop as another volume in the Springer LNCS series. This volume contains 21 papers representing the work that was presented and discussed at the workshop.

Workshop participants submitted draft manuscripts describing their research, and preliminary proceedings containing them were distributed at the workshop. At the workshop, each presentation stimulated fruitful and flourishing discussions, with many questions and comments. After the workshop we worked to obtain more comments on each paper, in some cases asking for reviews from external experts, and we forwarded these comments to the authors to guide them in improving their papers. Taking into account the discussions at the workshop and the review comments after the workshop, participants revised their manuscripts and contributed them to the proceedings. Contributions in the proceedings are of two kinds: full papers of 15–30 pages and extended abstracts of 6–9 pages. Extended abstracts were contributed in cases where the work reported was at too early a stage for a full paper, because it was being published in more detail elsewhere, or for other reasons. Some of the full papers are high-quality original contributions and some are detailed overviews of ongoing projects.

The papers and extended abstracts in the proceedings are organized by subject into the following five groups.

Part I: Evaluation Strategies

The first two papers in this group discuss evaluation strategies that extend lazy task creation. The paper by M. Feeley describes a new way to use lazy task creation—"lazy remote procedure call"— in a parallel variant of C, and its effectiveness is shown by some experimental results on Cray T3D. The paper by T. Ito gives an efficient evaluation strategy called "steal-based evaluation" for structured concurrency constructs in parallel Scheme systems; several extensions

of the strategy are also proposed, including the steal-based lazy evaluation of `delay` and stream computation. The paper by L. Moreau and D. Ribbens gives an operational semantics of `pcall` and `fork` in the presence of first-class continuations and side-effects, using the approach pioneered by Flanagan and Felleisen to give an operational semantics for `future`.

Part II: Programming Tools

The paper by R. Halstead describes a performance-visualization tool "Vista" for visualizing core performance data sets—histograms and event histories—of parallel executions; Vista has grown from a performance-visualization tool for Multilisp, but it is expected to be applicable to various parallel languages. The paper by T. Kamada and A. Yonezawa introduces an efficient debugging scheme for programs on massively parallel processors, and it discusses how replay and race-detection mechanisms can be implemented efficiently. The extended abstract by P. Cox, H. Glaser, and B. Lanaspre reports prototype work on Distributed Prograph, which is a distributed implementation of the visualization language Prograph based on the data flow computation model. The paper by K. Ueda describes his experiences with a mode analyzer for the concurrent logic programming language Flat GHC.

Part III: Irregular Data Structures and Applications

The first two papers in this group discuss parallel software tools for irregular data applications. The paper by K. Yelick et al. explains Multipol, an interesting parallel data structure library, and reports on the use of Multipol to build several parallel symbolic applications, including electromagnetic wave propagation in three dimensions, discrete event simulation, and parallel Gröbner basis computation for systems of nonlinear equations. The paper by M. Raghavachari and A. Rogers analyzes some irregular parallel applications to determine the kind of language support that is useful in developing those applications, and systematically evaluates several parallel programming languages against these criteria. The paper by L. Kale, B. Richards, and T. Allen discusses parallel graph coloring using prioritization techniques along with a large set of search-pruning heuristics; performance results show the effectiveness of this approach. The extended abstract by J.-L. Giavitto, O. Michel, and J.-P. Sansonnet explains an ongoing research project on "group-based" shapes and fields for data-parallel and data-flow processing of streams and collections.

Part IV: Systems

The paper by Nikhil presents a detailed overview of Cid, an extension of C to support parallel symbolic applications, reports preliminary performance results, and compares Cid to other parallel versions of C and C++. The extended abstract by Y. Ishikawa et al. gives an overview of MPC++, an extension of C++ that

supports a meta-level architecture facilitating language definition and modification in a massively parallel computer system. The paper by A. Nguyen-Tuong, A. Grimshaw, and J. Karpovich explains Mentat (an object-oriented parallel system based on C++) and presents and evaluates an approach to fault-tolerance that is based on Mentat's data flow model and a replication mechanism. The paper by H. Bal, K. Langendoen, and R. Bhoedjang explains Orca and some symbolic application programs in Orca, gives performance results, and explains Orca's usefulness for parallel symbolic applications, although Orca lacks support for several important parallel symbolic computing mechanisms. The extended abstract by Chikayama gives an overview of KLIC, a portable parallel implementation of the concurrent logic programming language KL1, which served as a kernel language for the Japanese Fifth Generation Computer Systems project.

Part V: Distributed Models and Systems

The papers in this group represent European activities on distributed Lisp systems and related topics. The paper by C. Queinnec gives an overview of an ongoing project on DMERRON, a data model for a coherent distributed shared memory, and addresses the importance of interoperability and the application programming interface in the design of DMEROON. The paper by J. Piquer explains a new implementation of TransPive based on the one he did at INRIA during his Ph.D. directed by C. Queinnec. The implementation is running on TCP/IP networking, using PVM as a communication layer. The extended abstract by D. de Roure explains preliminary experiences in the use of distributed Lisp techniques for distributed hypermedia information systems, and suggests the advantages and increased usefulness that can be realized by using distributed Lisp techniques. The extended abstract by J. Padget describes the attractiveness of a "virtual multicomputer" for machine-independent development of parallel software and discusses the primary requirements for virtual multicomputers. The paper by B. Berthomieu gives an overview of an ongoing project on LCS, which is an extension of Standard ML equipped with some concurrency and communication primitives derived from Milner's CCS.

The idea of holding this international workshop occurred to us right after the Cambridge workshop in 1992. C. Queinnec proposed to take care of the local arrangements in France, obtaining partial support from INRIA (Institut National de Recherche en Informatique et en Automatique). The workshop was held at the hotel NOVOTEL in Beaune, which is an old historic city located in the heart of Burgundy. The workshop was held with a small number of participants in a very friendly atmosphere conducive to the development of stimulating discussions. In addition to the formal workshop sessions, the participants deepened their friendship at a reception and a banquet, enjoying delicious French cuisine and Burgundy wines. Also, they had opportunities to visit the Hospices de Beaune and the downtown of Beaune for shopping, dining, and sightseeing expeditions

exploring the historic old city and the wine cellars in town. We would like to express our sincere appreciation of the support provided by INRIA and the local arrangements made by Ms. Claudie Thénault (INRIA) who served as the workshop secretary.

We provided reviewing comments to the authors of the proceedings with help of the following people: Saumya Debray, Robert Cori, Jean-Pierre Talpin, and Takashi Chikayama. We would like to express our sincere gratitude to them.

Finally, we thank Alfred Hofmann and Anna Kramer of Springer-Verlag for their assistance in the publication of this volume, and we also thank all those who helped organize, and participated in, the workshop for their invaluable contributions.

February 1996

Takayasu Ito
Robert H. Halstead, Jr.
Christian Queinnec

Table of Contents

Part IV: Systems

Part V: Distributed Models and Systems