

Proceedings of the 12th International Workshop on High-Level Parallel Programming Models and Supportive Environments

Long Beach, California USA

March 26th, 2007

The 12th HIPS workshop is a full-day meeting held at the IPDPS 2007 focusing on high-level programming of chip multi-processors (multi-core PCs), computing clusters, and massively-parallel machines. Like its predecessors, the workshop seeks cross-fertilizing research in areas of parallel applications, language design, compilers, run-time systems, and programming tools. It provides a timely and lightweight forum for scientists and engineers to present the latest ideas, findings, and tools in these rapidly changing fields.

HIPS 2007 features two invited talks and presentations of eleven refereed papers, which were selected out of fifteen submissions. At least three reviews for each paper were made by PC members, except for one which was deemed outside the scope by the program chair. Some papers received up to five reviews. The program committee held discussions through email. The program chair wish to acknowledge the generous contribution by the PC members. All completed reviews on time. Many interrupted their vacation or travel to give detailed and constructive reviews and to participate in group discussion. Two held review sessions with their students.

For the first time, the HIPS workshop has added a session (the last session) on programming models and tools for reliable parallel and distributed systems. The session is organized by Cristian Tapus from California Institute of Technology with a separate program committee. There is a significant overlap between the interests of HIPS and those of TOPMoDRS, so we join hands as an experiment. We renamed the workshop to HIPS-TOPMoDRS (the second part stands for tools, operating systems and programming models for developing reliable systems).

The chair wishes to thank the HIPS steering committee, Rudolf Eigenmann, Michael Gerndt, Frank Mueller, Craig Rasmussen, Martin Schulz for their support and help, the workshop chair of IPDPS, Allan Sussman, for his assistance with the organization, and Xiaoming Gu for administering the workshop web site and the review process. *Special thanks go to the workshop invited speakers, Rudolf Eigenmann of Purdue and Barbara Chapman of University of Houston.*

Summary of Accepted Papers

Naci presents the use of a number of techniques to support inter-loop-nest data locality optimization, including loop transformations and integer linear programming.

Quenette et al. study a role-based conceptual model, application hierarchical decomposition, and aspect oriented features for developing large-scale extensible scientific code, based on experience with programming real code in geophysics.

Li and Malony extend a performance modeling framework for automated performance problem detection for complex applications with composite parallel patterns. The authors analyze a large, real application.

Jamali and Liu present a quality of service (QoS) strategy for grids that takes into account resource ownership in determining usage. The results are presented for small scale systems.

Suh et al. describe and evaluate a two-level compiler that implements the Stream Virtual Machine programming model on a special purpose multicore experimental microprocessor (the Raw research microprocessor).

Reinhardt and Karypis describe an OpenMP implementation of an algorithm that finds, in a single large labeled undirected sparse graph, the connected sub-graphs with a given minimum number of edge-disjoint embedding. It shows good scalability on up to 30 processors.

Panuganti et al. present a MATLAB toolbox that provides an optimized implementation of PNL Global Array library for use by both library writers and MATLAB end users. They compare its performance on the NAS benchmarks to the standard C and Fortran implementations using MPI.

Patarasuk and Yuan give an bandwidth-efficient algorithm for all reduce operation. The algorithm is optimal because it has the lowest communication and it is contention free. The experimental results on three clusters and two topologies show superior performance over the machine naive library and two other existing implementations.

Gabriel and Huang present a dynamic adaptive library called the Abstract Data and Communication Library (ADCL) that permits a programmer to express communication at a high level, and then uses the run time support to choose the best communication mechanism.

Vishnu et al. describe the design of an MPI communication substrate over a 12x Infini-Band network and its evaluation using both micro- and application benchmarks. The design takes advantage of the parallel communication paths and hardware send/receive engines available on the network cards.

Grelck et al. combine data- and task-parallelism using the functional array-language SaC and the stream processing language S-Net. The approach is illustrated using a program to solve the Sudoku puzzle.

HIPS 2007 Program Committee

Arun Chauhan	Indiana University, USA.
Daniel G. Chavarria	Pacific Northwest Natl. Lab, USA.
Li Chen	Chinese Academy of Sciences, China.
Guang R. Gao	University of Delaware, USA.
Michael Gerndt	Technische Uni. München, Germany
Mary W. Hall	USC Information Sciences Inst., USA.
Ami Marowka	Shenkar Col. of Eng. and Design, Israel
Daniel Quinlan	Lawrence Livermore Natl. Lab, USA.
Lawrence Rauchwerger	Texas A & M University, USA.
Sven-Bodo Scholz	University of Hertfordshire, UK.
Alan Snavely	Uni. of California, San Diego, USA
Rich Wolski	Uni. of California, Santa Barbara, USA
Yuan Xin	Florida State University, FL, USA

January 19, 2007

Chen Ding
HIPS 2007 Chair
Rochester, New York, USA