More information about this series at http://www.springer.com/series/7407
This volume contains the papers presented at LCPC 2016: the 29th International Workshop on Languages and Compilers for Parallel Computing held during September 27–29, 2016, in Rochester, New York.

Since its founding in 1988, the LCPC workshop has been a leading venue for research on parallelizing compilers and related topics in concurrency, parallel languages, parallel programming models, runtime systems, and tools. The workshop spans the spectrum from foundational principles to practical experience, and from early ideas to polished results. LCPC encourages submissions that go outside the scope of scientific computing and enable parallel programming in new areas, such as mobile computing and data centers. The value of LCPC stems largely from its focused topics and personal interaction. This year’s location, in Rochester, NY, was both scenic and convenient. September weather is beautiful, as is the university campus, located at the confluence of the Genesee River and the historic Erie Canal.

Specific topics of LCPC 2016 included:

- Compiling for parallelism and parallel compilers
- Static, dynamic, and adaptive optimization of parallel programs
- Parallel programming models and languages
- Formal analysis and verification of parallel programs
- Parallel runtime systems and libraries
- Performance analysis and debugging tools for concurrency and parallelism
- Parallel algorithms and concurrent data structures
- Parallel applications
- Synchronization and concurrency control
- Software engineering for parallel programs
- Fault tolerance for parallel systems
- Parallel programming and compiling for heterogeneous systems

There were 26 submissions. Each submission was reviewed by at least three, and on average 3.5, Program Committee members. The committee decided to accept 23 papers, of which 20 are regular papers (up to 15 pages) and three are short papers (up to five pages).

The workshop program includes three keynotes:

- “Parallel Computation Models and Systems, Dataflow, Coelets, and Beyond” by Guang R. Gao of University of Delaware
- “Towards High-Level High-Performance Software Development” by P. (Saday) Sadayappan of Ohio State University
- “The Multi-core Problem as an Algorithmic Problem” by Leslie Valiant of Harvard University
There was also one invited talk on “Tapir: Embedding Fork-Join Parallelism into LLVM’s Intermediate Representation” by Tao Schardl of MIT.

We would like to thank Pengcheng Li for creating the workshop website at http://www.cs.rochester.edu/u/cding/lcpc2016/ and compiling the final publication package, and the computer science staff for the help in organizing the workshop and the financial support from Huawei, IBM, as well as the Goergen Institute of Data Science and Department of Computer Science at University of Rochester. The generation of the proceedings was assisted by the EasyChair conference system.

December 2016

Chen Ding
John Criswell
Peng Wu
Organization

Program Committee

Ayon Basumallik  MathWorks Inc., USA
James Brodman    Intel, USA
Arun Chauhan     Indiana University, USA
John Criswell    University of Rochester, USA
Chen Ding        University of Rochester, USA
Matthew Fluet    Rochester Institute of Technology, USA
Jeff Huang       Texas A&M University, USA
Hironori Kasahara Waseda University, Japan
Frank Mueller    North Carolina State University, USA
P. Sadayapan     Ohio State University, USA
Xipeng Shen      North Carolina State University, USA
Michelle Strout  University of Arizona, USA
Peng Tu          Intel, USA
James Tuck       North Carolina State University, USA
Peng Wu          Huawei US Research Lab, USA
Contents

Large Scale Parallelism

QUARC: An Array Programming Approach to High Performance Computing .................................................. 3
   Diptorup Deb, Robert J. Fowler, and Allan Porterfield

Utilizing Concurrency: A New Theory for Memory Wall .............................................................. 18
   Xian-He Sun and Yu-Hang Liu

ParFuse: Parallel and Compositional Analysis of Message Passing Programs ..................................... 24
   Sriram Aananthakrishnan, Greg Bronevetsky, Mark Baranowski, and Ganesh Gopalakrishnan

Fast Approximate Distance Queries in Unweighted Graphs Using Bounded Asynchrony .......................... 40
   Adam Fidel, Francisco Coral Sabido, Colton Riedel, Nancy M. Amato, and Lawrence Rauchwerger

Energy Avoiding Matrix Multiply ........................................................................................................... 55
   Kelly Livingston, Aaron Landwehr, José Monsalve, Stéphane Zuckerman, Benoit Meister, and Guang R. Gao

Resilience and Persistence

Language Support for Reliable Memory Regions ................................................................................. 73
   Saurabh Hukerikar and Christian Engelmann

Harnessing Parallelism in Multicore Systems to Expedite and Improve Function Approximation .......... 88
   Aurangzeb and Rudolf Eigenmann

Adaptive Software Caching for Efficient NVRAM Data Persistence ...................................................... 93
   Pengcheng Li and Dhruva R. Chakrabarti

Compiler Analysis and Optimization

Polyhedral Compiler Technology in Collaboration with Autotuning Important to Domain-Specific Frameworks for HPC ................................................................. 101
   Mary Hall and Protonu Basu
Contents

An Extended Polyhedral Model for SPMD Programs and Its Use in Static Data Race Detection .................................................. 106  
Prasanth Chatarasi, Jun Shirako, Martin Kong, and Vivek Sarkar

Polygonal Iteration Space Partitioning ............................................. 121  
Aniket Shivam, Alexandru Nicolau, Alexander V. Veidenbaum,  
Mario Mango Furnari, and Rosario Cammarota

Automatically Optimizing Stencil Computations on Many-Core NUMA Architectures ................................................................. 137  
Pei-Hung Lin, Qing Yi, Daniel Quinlan, Chunhua Liao,  
and Yongqing Yan

Formalizing Structured Control Flow Graphs .................................... 153  
Amit Sabne, Putt Sakdhnagool, and Rudolf Eigenmann

Dynamic Computation and Languages

Automatic Vectorization for MATLAB ............................................... 171  
Hanfeng Chen, Alexander Krolik, Erick Lavoie, and Laurie Hendren

Analyzing Parallel Programming Models for Magnetic Resonance Imaging ................................................................. 188  
Forest Danford, Eric Welch, Julio Cárcenas-Rodriguez,  
and Michelle Mills Strout

The Importance of Efficient Fine-Grain Synchronization
for Many-Core Systems ................................................................ 203  
Tongsheng Geng, Stéphane Zuckerman, José Monsalve,  
Alfredo Goldman, Sami Habib, Jean-Luc Gaudiot, and Guang R. Gao

Optimizing LOBPCG: Sparse Matrix Loop and Data Transformations in Action ................................................................. 218  
Khalid Ahmad, Anand Venkat, and Mary Hall

GPUs and Private Memory

LightHouse: An Automatic Code Generator for Graph Algorithms on GPUs ................................................................. 235  
G. Shashidhar and Rupesh Nasre

Locality-Aware Task-Parallel Execution on GPUs ............................. 250  
Jad Hbeika and Milind Kulkarni

Automatic Copying of Pointer-Based Data Structures ...................... 265  
Tong Chen, Zehra Sura, and Hyojin Sung
<table>
<thead>
<tr>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Automatic Local Memory Management for Multicores Having Global Address Space</td>
<td>282</td>
</tr>
<tr>
<td><em>Kouhei Yamamoto, Tomoya Shirakawa, Yoshitake Oki, Akimasa Yoshida, Keiji Kimura, and Hironori Kasahara</em></td>
<td></td>
</tr>
</tbody>
</table>

**Run-time and Performance Analysis**

<table>
<thead>
<tr>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mapping Medley: Adaptive Parallelism Mapping with Varying Optimization Goals</td>
<td>299</td>
</tr>
<tr>
<td><em>Murali Krishna Emani</em></td>
<td></td>
</tr>
<tr>
<td>The Contention Avoiding Concurrent Priority Queue</td>
<td>314</td>
</tr>
<tr>
<td><em>Konstantinos Sagonas and Kjell Winblad</em></td>
<td></td>
</tr>
<tr>
<td>Evaluating Performance of Task and Data Coarsening in Concurrent Collections</td>
<td>331</td>
</tr>
<tr>
<td><em>Chenyang Liu and Milind Kulkarni</em></td>
<td></td>
</tr>
</tbody>
</table>

**Author Index** | 347  |