Int J Parallel Prog (2014) 42:525–528
DOI 10.1007/s10766-013-0297-y

EDITORIAL

Guest Editorial for High-level Parallel Programming and Applications

Gaetan Hains · Youry Khmelevsky

Received: 17 October 2013 / Accepted: 17 October 2013 / Published online: 28 November 2013
© Springer Science+Business Media New York 2013

As processor and system manufacturers increase the amount of both inter- and intra-chip parallelism it becomes crucial to provide the software industry with high-level, clean and efficient tools for parallel programming. Parallel and distributed programming methodologies are currently dominated by low-level techniques such as send/receive message passing, or equivalently unstructured shared memory mechanisms. Higher-level, structured approaches offer many possible advantages and have a key role to play in the scalable exploitation of ubiquitous parallelism.

Since 2001 the HLPP series of meetings has been a forum for researchers developing state-of-the-art concepts, tools and applications for high-level parallel programming. Its general emphasis is on software quality, programming productivity and high-level performance models. The 2013 symposium was a self-contained event, held on the 1st and 2nd of July at the *Institut Henri Poincaré* (IHP) in Paris. The invited talk was given by Leslie Valiant inventor of the BSP paradigm and 2010 Turing award on*A Bridging Model for Multi-Core Computing.* Twenty nine papers were submitted and evaluated by three or more reviewers and among those twelve were selected for presentation at HLPP2013 and publication in this Journal. This special issue of IJPP presents the first eight selected papers that we now summarize.

Compilation and program transformation techniques are at the hearth of high-performance computing and three papers are devoted to them. Automatic generation of parallel code is generalized and made more efficient with dynamic support as described in "**Dynamic and Speculative Polyhedral Parallelization Using Compiler-Generated Skeletons"** by Philippe Clauss, Alexandra Jimborean,

G. Hains (✉)
LACL, UPEC, Creteil, France
e-mail: gaetan.hains@u-pec.fr

Y. Khmelevsky
Computer Science Department, Okanagan College, Kelowna, BC, Canada
e-mail: ykhmelevsky@okanagan.bc.ca

Jean-François Dollinger, Vincent Loechner and Juan Manuel Martinez Caamano. Declarative code on using algorithmic skeletons is automatically optimized by general and adaptive loop fusions techniques in "**An Automatic Fusion Mechanism for Variable-Length List Skeletons in SkeTo"** by Kento Emoto and Kiminori Matsuzaki. Another paper in this category presents expressive refactoring techniques based on a cost (performance) model for declarative communicating programs: "**Cost-Directed Refactoring for Parallel Erlang Programs"** by Christopher Brown, Marco Danelutto, Kevin Hammond, Peter Kilpatrick and Archibald Elliott presents.

Much attention has been given recently to the evolution towards heterogeneous architectures based on mixtures of classical processors and accelerators. Declarative and efficient GPGPU programming for the OCaml language is described in "**Efficient Abstractions for GPGPU Programming"** by Mathias Bourgoin, Emmanuel Chailloux and Jean-Luc Lamotte. Then a multi-GPU implementation of the generic all-pairs algorithmic skeleton is presented in "**Introducing and Implementing the Allpairs Skeleton for GPU Systems"** by Michel Steuwer, Malte Friese, Sebastian Albers and Sergei Gorlatch.

The Bulk-Synchronous Parallel (BSP) algorithms and cost model has drawn the attention of many high-level parallel programmers for its promise of *immortal algorithms* (in the words of L. Valiant) whose performance will scale and adapt to any combination of machine parameters. One of the most recent tools for programming BSP algorithms is the new C library presented in "**MulticoreBSP for C: a high-performance library for shared-memory parallel programming"** by Albert-Jan Nicholas Yzelman, Rob Bisseling, Dirk Roose and Karl Meerbergen.

Declarative high-level techniques not only make programming more efficient but allow for program proof and verification techniques to fully certify parallel and distributed behavior. Such an application of the Coq theorem prover is presented in "**Bringing Coq Into the World of GCM Distributed Applications"** by Nuno Gaspar, Ludovic Henrio and Eric Madelaine.

Finally, an original application of algorithmic skeletons to sensor networks programming is introduced in "**Engineering energy efficient visual sensor network applications using skeletons"** by Nicoletta Triolo, Susanna Pelagatti and Stefano Chessa.

Four more papers have been selected for publication and will appear later in IJPP. Three of them cover new progress on "classical" topics for HLPP on high-level programming and run-time support for it:

- **"An Efficient Scalable Work-Stealing Runtime System for S-Net"** by Clemens Grelck and Bert Gijsbers
- **"An Algorithm Template for Domain-Based Parallel Irregular Algorithms"** by Carlos Hugo Gonzalez Vazquez
- **"A scalable Farm skeleton for hybrid parallel and distributed programming"** by Steffen Ernsting and Herbert Kuchen.

and a fourth one represents an opening towards parallel software engineering:

- **"Design patterns percolating to parallel programming framework implementation"** by Massimo Torquati, Marco Danelutto, Sonia Campa, Peter Kilpatrick and Marco Aldinucci.

The organizers of HLPP2013 wish to thank:

- Our guest speaker Prof. Leslie Valiant who made us the honor of accepting our invitation.
- Institut Henri Poincaré for its generous provision of both Amphithéatres, catering space, WiFi and prestigious location.
- Springer and the International Journal of Parallel Programming for their editorial assistance and publication of the final proceedings for HLPP2013.
- The steering committee for decisive comments when faced with key decisions.
- The authors of all submitted papers for their interest in our series of events and their confidence in our review process.
- Our sponsor Rogue Wave Software for financial support and informative documentation of their HPC software tools.
- EXQIM S.A.S. for hosting the HLPP accounts.
- Lydie and Daria for helping with hotels, restaurant and the welcome desk.
- Last but not least the program committee and all reviewers for their reliable support and core contribution to our scientific event and publication:

PC members:

Mostafa Bamha (Université d'Orléans, France)
Rob Bisseling (Utrecht University, Netherlands)
Murray Cole (University of Edimburgh, United Kingdom)
Kento Emoto (University of Tokyo, Japan)
Frédéric Gava (Université Paris-Est, France)
Alexandros Gerbessiotis (New Jersey Institute of Technology, United States)
Andy Gill (University of Kansas, United States)
Clemens Grelck (University of Amsterdam, Netherlands)
Christoph Kessler (Linköping University, Sweden)
Herbert Kuchen (Universität Münster, Germany)
Ramon Lawrence (University of British Columbia, Canada)
Quentin Miller (Somerville College Oxford, United Kingdom)
Susanna Pelagatti (University of Pisa, Italy)
Alexander Tiskin (University of Warwick, United Kingdom)

Other reviewers:

Wadoud Bousdira (Université d'Orléans, France)
Mathias Bourgoin (Université Pierre et Marie Curie, France)
Emmanuel Chailloux (Université Pierre et Marie Curie, France)
Albert Cohen (INRIA, France)
Usman Dastgeer (Linköping University, Sweden)
Marco Danelutto (University of Pisa, Italy)
Steffen Ernsting (Universität Münster, Germany)
Joël Falcou (Metascale, France)
Jean Fortin (Université Paris-Est, France)
Cécile Germain-Renaud (Université Paris-Sud, France)
Kevin Hammond (University of St-Andrews, United Kingdom)
Noman Javed (University of Edinburgh, United Kingdom)

Sylvain Jubertie (Université d'Orléans, France)
Chong Li (Université Paris-Est, France)
Frédéric Loulergue (Université d'Orléans, France)
Fabrice Mourlin (Université Paris-Est, France)
Francesco Nidito (Microsoft, United Kingdom)
Julien Tesson (Université Paris-Est, France)
Nicoleta Triolo (University of Pisa, Italy)

Paris (France) and Kelowna, BC (Canada) July 2013.
Gaétan Hains and Youry Khmelevsky
Co-editors of the IJPP special issue and co-organizers of HLPP2013.