

## Preface to the Special Issue on Sequential Code Parallelization

Chao Wang<sup>1</sup> · Nadia Nedjah<sup>2</sup> ·  
Luiza M. Mourelle<sup>3</sup> · Aili Wang<sup>4</sup>

Published online: 16 July 2016

© Springer Science+Business Media New York 2016

We are delighted to present this special issue of International Journal of Parallel Programming on Sequential Code Parallelization.

Parallelization is now the standard way in computing. In order to bridge the gap between parallel hardware resources and the performance/cost requirements, programmers must reform applications taking into account knowledge about the target parallel architecture. Elaborating parallel programs is much more challenging and unnatural than writing sequential equivalent codes. However, the traditional automatic instruction-level parallelism does not make full use of the state-of-the-art hardware architectures, especially for cutting-edge applications and future trends in parallel computing for various platforms. The aim of this special issue is to provide a way

---

✉ Chao Wang  
cswang@ustc.edu.cn

Nadia Nedjah  
nadia@eng.uerj.br

Luiza M. Mourelle  
ldmm@eng.uerj.br

Aili Wang  
wangal@ustc.edu.cn

- <sup>1</sup> Department of Computer Science, University of Science and Technology of China, Ren'ai Road 188, Suzhou 215123, Jiangsu, China
- <sup>2</sup> Department of Electronics Engineering and Telecommunications, State University of Rio de Janeiro, Rio de Janeiro, Brazil
- <sup>3</sup> Department of Systems Engineering and Computation, State University of Rio de Janeiro, Rio de Janeiro, Brazil
- <sup>4</sup> School of Software Engineering, University of Science and Technology of China, Ren'ai Road 188, Suzhou 215123, Jiangsu, China

of exploring aspects of sequential code parallelization. This issue includes 10 papers briefly discussed as follows.

In “A Parallel Yet Pipelined Architecture for Efficient Implementation of the Advanced Encryption Standard Algorithm on Reconfigurable Hardware,” the authors present an efficient pipelined hardware implementation of the AES-128 algorithm. Besides, the authors show that if the required number of rounds must increase to defeat attackers, the proposed implementation stays efficient.

In “A Cross-ISA Kernelized High-Performance Parallel Emulator,” the authors propose a retargetable and high-performance system named xKEMU. By reusing QEMU and Linux kernel as components, xKEMU could improve QEMU performance by a factor of about 1.24x on integer and memory performance for x86 emulation. The performance of virtual devices is also enhanced due to pass-through in the emulation of peripheral devices.

In “Towards Scalable Java HPC with Hybrid and Native Communication Devices in MPJ Express,” the authors extend MPJ Express software to provide two new communication devices, including hybrid and native implementations. The hybrid application enables MPJ Express to exploit hybrid parallelism on a cluster of multicore processors by sitting on top of existing shared memory and network communication devices. The native application uses JNI wrappers in interfacing MPJ Express to native MPI implementations like MPICH and Open MPI.

In “Parallel Implementations of the Cooperative Particle Swarm Optimization on Many-Core and Multi-Core Architectures,” the authors propose a cooperative strategy, which consists of subdividing an optimization problem into many simpler sub-problems. The optimization work for all the selected sub-problems can run in parallel. The authors map the work onto four different parallel high-performance multiprocessors, based on Multi- and Manycore architectures.

In “Transparent Speculative Parallelization of Discrete Event Simulation Applications Using Global Variables,” the authors present speculative PDES systems that run on top of multi-core machines, where simulation objects can concurrently process their events with no guarantee of causal consistency, and actual violations of causality rules are recovered through rollback/recovery schemes.

In “Czip: a Fast Lossless Compression Algorithm for Climate Data,” the authors propose a lossless compression algorithm for climate data, named czip. The authors efficiently eliminate data redundancy through several new methods, including adaptive prediction, eXclusive OR (XOR) differencing, multiway compression and static regions.

In “Combining Data and Computation Distribution Directives for Hybrid Parallel Programming: A Transformation System,” the authors describe dSTEP, a directive-based programming model for hybrid shared and distributed memory machines. The originality of our work is the definition and an implementation of a unified high-level programming model addressing both data, and computation distributions, providing a particularly fine control of the calculation.

In “A Parallelization Approach for Hard Real-Time Systems and its Application to two Industrial Programs,” the authors present a parallelization approach for hard real-time systems, which ensures a high reuse of legacy code and preserves timing

analysability. The authors create models of the legacy programs showing the potential of parallelism, optimize them and change the source codes accordingly.

In “Automatic Parallelization: Executing Sequential Programs on a Task-Based Parallel Runtime,” the authors create a new parallelizing compiler that analyzes the read and writes instructions and control flow modifications in programs to identify a set of dependencies between the instructions in the program.

In “Purge-Rehab: Eager Software Transactional Memory with High Performance under Contention,” the authors present a lightweight mechanism for controlling transaction restarts in eager validation to reduce livelock and thus improve throughput and reduce starvation. Purge-Rehab achieves higher performance in five benchmarks, similar in four, and is lower in only one, showing that eager validation can reach high performance under contention. Purge-Rehab is implemented in word-based RSTM but applies to any STM.

The editors strongly believe that the papers as selected to appear in this special issue are an accurate representation of current topics in the selected areas. They hope these research work will stimulate further development of the subject. The guest editors are very grateful to the authors of this particular section and to the reviewers for their tremendous service by critically reviewing the submitted papers. The editors would also like to thank the Editor-in-Chief of International Journal of Parallel Programming, Prof. Alexandru Nicolau, Renuka Nidhi, and Joyce Yanya Chen for the editorial assistance and excellent cooperative collaboration to produce this scientific work. We hope you enjoy this particular issue and take some inspiration from it for your future research. Enjoy reading it!

Chao Wang, Nadia Nedjah, Luiza M. Mourelle, and Aili Wang  
July 7, 2016