

## Guest Editorial: Special Issue on Computing Frontiers

**Antonino Tumeo<sup>1</sup> · Hubertus Franke<sup>2</sup> ·  
Gianluca Palermo<sup>3</sup> · John Feo<sup>4</sup>**

Published online: 5 February 2018

© Springer Science+Business Media, LLC, part of Springer Nature 2018

This special issue collects extended versions of the best papers of the 2016 edition of the ACM International Conference on Computing Frontiers. Held for the first time in the northern part of Italy, in Como, the 13th edition of the conference continued the tradition of pushing the boundaries of computing research (the “Computing Frontiers”), of a wide range of topics, including novel computing paradigms, computational models, algorithms, application paradigms, development environments, compilers, operating environments, computer architecture, hardware substrates, memory technologies, and smarter life applications. Among the 30 regular papers accepted at the conference (out of 96 total submissions) we have invited authors of the seven most representative, in terms of quality and topics, to submit extended versions for this special issue. From these seven papers, after a rigorous peer-review process according to the International

---

✉ Antonino Tumeo  
antonino.tumeo@pnnl.gov

Hubertus Franke  
frankeh@us.ibm.com

Gianluca Palermo  
gianluca.palermo@polimi.it

John Feo  
john.feo@pnnl.gov

- <sup>1</sup> High Performance Computing, Pacific Northwest National Laboratory (PNNL), 902 Battelle Blvd, MSIN J4-30, Richland, WA 99352, USA
- <sup>2</sup> IBM T.J. Watson Research Center, 1101 Kitchawan Road, Yorktown Heights, NY 10567, USA
- <sup>3</sup> Dipartimento di Elettronica, Informatica, e Bioingegneria (DEIB), Politecnico di Milano, Via Ponzio 34/5, 20133 Milan, MI, Italy
- <sup>4</sup> Northwest Institute for Advanced Computing (NIAC), Pacific Northwest National Laboratory (PNNL), 902 Battelle Blvd, MSIN J4-30, Richland, WA 99352, USA

Journal of Parallel Programming standards, we have selected five to appear in this Special Issue.

The first article, “Accelerating Data Analytics on Integrated GPU Platforms via Runtime Specialization” by Naila Farooqui, Indrajit Ro, Yuan Chen, Vanish Talwar, Rajkishore Barik, Brian Lewis, Tatiana Shpeisman, and Karsten Schwan, discusses how to match regular and irregular application behaviors to the underlying heterogeneous resources, including GPUs, through runtime specialization.

In the second article, “Hierarchical Pattern Mining with the Automata Processor”, Ke Wang, Elaheh Sadredini, and Kevin Skadron propose a flexible, hardware-accelerated framework for mining hierarchical patterns with Apriori-based algorithms. These lead to multi-pass pruning strategies but expose massive parallelism. Under this framework, the authors implemented two hierarchical pattern mining techniques on the Micron Automata Processor.

The original versions of these two articles also won the best paper award at the conference.

In “Graph Programming Interface (GPI): A Linear Algebra Programming Model for Large Scale Graph Computations”, W P Horn, Manoj Kumar, Joefon Jann, Jose Moreira, Pratap Pattnaik, Mauricio Serrano, Gabriel Tanase, and Hao Yu present a model for implementing graph algorithms using a linear algebra based specification. They first specify the set of linear algebra primitives that enable describing graph algorithms through their composition, and then describe a high-performance C++ implementation of these primitives and their integration in the Spark framework to achieve scalability.

In “Fast Automated Processing and Evaluation of Identity Leaks”, David Jaeger, Hendrik Graupner, Chris Pelchen, Feng Cheng, and Christoph Meinel present a fully automated approach for the processing and analysis of a large number of internet identity leaks. The proposed prototype implementation of the leak processing workflow includes the extraction of digital identities from structured and unstructured leak-files, the identification of hash routines, and a quality control to ensure leak authenticity. Through parallel and distributed computing, it also makes leaks almost immediately available for analysis and notification after they have been published.

Finally, the article “Automated Compiler Optimization of Multiple Vector Loads/Stores”, by Viacheslav Zakharin, Farhana Aleen, Rakesh Krishnaiyer, Garima Gupta, David Kreitzer, and Chang-Sun Jr Lin discusses two automated compiler transformations to enable more effective vectorization of code with irregular data accesses. The first optimization improves the performance for a group of adjacent gathers/scatters. The second optimization improves performance for a group of stencil vector accesses using more efficient Single Instruction Multiple Data (SIMD) instructions. Both optimizations are implemented in a pre-release of the Intel Compiler.

These 5 contributions target data analytics, large scale approaches, architecture specific optimizations, runtime optimizations, security, and compilers. They are a snapshot of the best works that appeared at the conference. They clearly demonstrate how Computing Frontiers is able to address a breadth of computing topics with the

required depth. We hope you will enjoy reading them as much as we enjoyed preparing this special issue.

The Guest Editors,  
Antonino Tumeo, Hubertus Franke  
Gianluca Palermo, John Feo