

Foreword: Parallelism in Algorithms and Architectures

Geppino Pucci · Victor Luchangco ·
Rajmohan Rajaraman

Published online: 8 January 2014
© Springer Science+Business Media New York 2014

This special issue contains 6 selected papers whose preliminary versions appeared in the Proceedings of the 23rd Annual ACM Symposium on Parallelism in Algorithms and Architectures (SPAA), held June 2011, in San Jose, California, USA. These papers were selected by the special issue co-editors from 35 papers that were presented at the conference. The authors were invited to submit full versions of their papers, which were then fully refereed according to the usual standards of *Theory of Computing Systems*. The selected papers are representative of the breadth and depth of the research in parallelism in algorithms and architectures that was presented at SPAA 2011.

The first paper *Tight Bounds on Adopt-Commit Objects*, by James Aspnes and Faith Ellen, presents matching upper and lower bounds for adopt-commit objects, which form an important building block for agreement and consensus protocols.

The second paper *Highly-Efficient Wait-Free Synchronization*, by Panagiota Fatourou and Nikolaos Kallimanis, presents a new universal construction for wait-free objects with good theoretical properties but unrealistic system assumptions, and then shows how to adapt this technique to produce a practical wait-free concurrent object implementations that perform well on real machines (though with weaker theoretical guarantees).

G. Pucci
University of Padua, Padua, Italy
e-mail: geppino.pucci@unipd.it

V. Luchangco
Oracle, Burlington, MA, USA
e-mail: victor.luchangco@oracle.com

R. Rajaraman (✉)
Northeastern University, Boston, USA
e-mail: rraj@ccs.neu.edu

In *Nearly-Linear Work Parallel SDD Solvers, Low-Diameter Decomposition, and Low-Stretch Subgraphs*, Guy E. Blelloch, Anupam Gupta, Ioannis Koutis, Gary L. Miller, Richard Peng, and Kanat Tangwongsan provide the first nearly work-optimal parallel solver for Symmetric Diagonally Dominant (SDD) linear systems by leveraging on a parallel decomposition technique to generate low-stretch spanning subgraphs, and present its applications to improved parallel graph algorithms.

Software Engineering with Transactional Memory versus Locks in Practice, by Victor Pankratius and Ali-Reza Adl-Tabatabai, presents the results of a small but detailed user study comparing programmers writing a concurrent program using transactional memory and locks, showing the potential of transactional memory and also some possible pitfalls.

In *Re-Chord: A Self-stabilizing Chord Overlay Network*, by Sebastian Kniesburges, Andreas Koutsopoulos, and Christian Scheideler, the authors present a self-stabilizing extension of the Chord peer-to-peer network, devised to recover the network from any initial state where a number of peers are nearly connected, in a number of rounds which is quasi linear in this number.

The final paper *On a Local Protocol for Concurrent File Transfers*, by MohammadTaghi Hajiaghayi, Rohit Khandekar, Guy Kortsarz, and Vahid Liaghat, studies the complexity of transferring files between pairs of hosts in a distributed environment, and presents constant factor approximation algorithms for average completion time and makespan metrics.

Each paper was extensively reviewed and most underwent multiple revisions. We would like to thank all people who have contributed to the special issue, including the authors and anonymous referees, the TOCS editor-in-chief Alan Selman, and Springer staff members Vithyaa Bathman and Melissa Fearon. We hope the readers will enjoy the result of this endeavor.