

High performance computing tools in science and engineering II

Enrique Quintana-Ortí · Jose Ranilla ·
Jesús Vigo-Aguiar

Published online: 11 May 2011
© Springer Science+Business Media, LLC 2011

This special issue collects research papers selected among those presented at the second minisymposium “HPC applied to Computational Problems in Science and Engineering” which was held in June 2010, in Almeria, Spain. This workshop was a special event organized within the framework of the “10th International Conference on Computational and Mathematical Methods in Science and Engineering”.

The papers in the issue can be classified into three main groups: parallel linear algebra algorithms, parallelization of applications, and tools and environments for parallel programming.

These are papers that deal with important computations in the linear algebra domain as, for example, using graphics processors to accelerate the computation of the matrix inverse or a parallel python library for nonlinear systems.

Science and engineering are an endless source of complexity. Therefore, it is not a surprise that so many researchers dedicate their time to develop parallel solvers or specific hardware designs for these applications. Along this line, a heterogeneous sample of problems are analyzed in the issue as, for example, a heterogeneous parallel solution for the fast multipole method, the analysis of magnetic resonance imaging using GPUs, or a many-core solution for real-time massive convolution for audio applications.

E. Quintana-Ortí
Universidad Jaime I, Castellón de la Plana, Spain

J. Ranilla
Universidad Oviedo, Oviedo, Spain

J. Vigo-Aguiar (✉)
Universidad de Salamanca, Salamanca, Spain
e-mail: jvigo@usal.es

Developing parallel codes that obtain the best performance from current architectures is often nontrivial. In response to this, a number of parallel experts work hard to ease this task, developing advanced tools, application programming interfaces, and environments for parallel programming. The issue is also sensitive to these contributions. Examples of this are a parameterised shared-memory scheme for parameterised metaheuristics, an automatic code generation for GPUs, accelerating network applications by distributed interfaces, a model for evaluating scalability in heterogeneous systems or power saving-aware prefetching for SSD-based systems.

We hope that the reader will find the papers in this issue both informative and inspiring. For the elaboration of this special issue, a new revision process was performed. We would like to thank the authors for their contributions and the reviewers for their work. We would also like to acknowledge support received from the Spanish Office of Science, namely the TIN2009-08058-E, TIN2010-14971 and MTM2010-08916-E and MTM2008-05489 grants.

March 2011