J Supercomput (2014) 70:511–513
DOI 10.1007/s11227-014-1279-6

High performance computing: an essential tool for science and engineering breakthroughs

J. Ranilla · E. M. Garzón · J. Vigo-Aguiar

Published online: 20 August 2014

High-performance computing (HPC) has become an essential tool in both scientific and industrial communities because it provides an excellent environment for solving a wide range of problems in modern science and engineering. Nowadays, HPC clusters (HPCC) are, in fact, the main architecture for supercomputers due to the high performance of commodity microprocessors and networks and to the lower price/performance ratio. Moreover, current standalone computers present tremendous power and they could be considered as desktop HPC platforms if their resources are appropriately exploited.

Today's HPC platforms are based on clusters of multicore nodes, which include accelerator devices such as GPUs or Xeon Phi coprocessors. Hence, the heterogeneity characterizes modern HPC platforms. Recent technological advances had a strong impact in the computational power of a wide range of platforms. Recently, HPC has become even more popular due to the rise of computational resources into devices as extended as smartphones, tablets, laptop and stand-alone computers, since all of them are based on multicore processors which can include coprocessors devices.

This special issue is devoted to HPC focusing on twofold: the rise of computationally demanding scientific models of practical interest and the evolution of HPC technology, which can meet these high-performance requirements in a wide diversity of scenarios.

J. Ranilla
Universidad de Oviedo, Gijón, Spain
e-mail: ranilla@uniovi.es

E. M. Garzón
Universidad de Almería, Almería, Spain
e-mail: gmartin@ual.es

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

For this, a set of representative applications that need HPC capabilities for their development is analyzed. Specifically, they are focused on the following fields: optimization of the injected electrical power into the grid and the energy in communication systems with multiple transmitter and receiver antennas; simulation of aircraft noise scattering, forest fire evolution and fluid vibration; and finally, video and hyperspectral image processing.

The heterogeneity is also related to the parallel programming interfaces. Therefore, parallel codes should express every parallelism level by different interfaces. It means that programmers have to combine several kinds of data structures, programming models and load distributions to take advantage of the heterogeneity of current computers. Thereby, in this scenario the design of tools, resources and methodologies which help programmers and scientists to develop HPC software is a goal of interest. This issue includes nine works which can be classified in this line. They focus their interest on multicore, GPUs, combination of both and mobile devices. So, a set of tools for analyzing memory management of codes in modern multicore systems with complex memory hierarchies is described; an extensive analysis of the impact of fusing vector operations on the performance of the GPU is developed; a useful strategy to systematically select the appropriate GPU configuration parameters is described; an extension of directives of parallel task programming paradigm to GPUs is proposed; SSE and AVX instructions in multicore CPUs and GPU computing for solid and fluid vibration problems are evaluated in terms of performance; auto-tuning techniques on multicores and multi-GPU systems for two-dimensional polynomial regression models are proposed; a collection of benchmark applications to evaluate the performance on mobile devices is analyzed; the performance of collaborative augmented reality (CAR) systems based on feature tracking when using mobile phones is evaluated; and a decentralized protocol for mobile control access is described.

A classical approach to facilitate software development on HPC architectures is based on the use of basic routines or libraries which compute the operations of most interest in different application contexts and are designed to optimally exploit specific HPC architectures. Several works in this issue follow this model. So, implementations of evolutionary global optimization solvers on multicore-cluster and GPU are analyzed; a new hybrid OpenMP/MPI solution for the PageRanking problem is proposed; an adaptive algorithm for time-invariant Differential matrix Riccati equations is developed on GPUs; multicore non-negative matrix factorization using Alternating Least Squares algorithms is evaluated; a new modified block algorithm for the lattice reduction technique multicore+gpus; an in-memory application-level checkpoint-based migration solution for MPI codes is proposed; how to distribute the queries over the cluster to obtain the best performance with heuristics.

This special issue gathers the most relevant contributions to the fifth minisymposium *High Performance Computing applied to Computational Problems in Science and Engineering* as a part of the *13th International Conference on Computational and Mathematical Methods in Science and Engineering* (CMMSE 2013), which was held during 23–27 June 2013, in Almeria. It was made possible thanks to the authors of the contributions and also to a significant number of reviewers that contributed the final result with their comments and suggestions for improvements. We would like to conclude with our acknowledgments to all of them and we hope that the bottom line will

be of your interest. We would also like to acknowledge to the “Ministerio de Educación y Ciencia” of Spain, for its support to the Spanish CAPAP-H4 network (HPC in Heterogeneous Systems, TIN2011-15734-E), and to the “Ministerio de Economía y Competitividad” from Spain/FEDER for supporting grants TIN2012-37483-C03-03 and TEC2012-38142-C04-04.