1

Summary of the Open European Network for High-Performance Computing in Complex Environments

Emmanuel Jeannot
Inria Bordeaux Sud-Ouest, Talence, France

Julius Žilinskas
Vilnius University, Vilnius, Lithuania

In this chapter, we describe the COST Action IC0805 entitled "Open European Network for High-Performance Computing on Complex Environments." This Action had representation from more than 20 countries and lasted from 2009 to 2013. We outline the scientific focus of this Action, its organization, and its main outcomes. The chapter concludes by presenting the structure of the book and its different chapters.

High-Performance Computing on Complex Environments, First Edition.
Edited by Emmanuel Jeannot and Julius Žilinskas.

1.1 INTRODUCTION AND VISION

In recent years, the evolution and growth of the techniques and platforms commonly used for high-performance computing (HPC) in the context of different application domains has been truly astonishing. While parallel computing systems have now achieved certain maturity thanks to high-level libraries (such as ScaLAPACK) or runtime libraries (such as MPI), recent advances in these technologies pose several challenging research issues. Indeed, current HPC-oriented environments are extremely complex and very difficult to manage, particularly for extreme-scale application problems.

At the very low level, the latest generation CPUs are made of multicore processors that can be general-purpose or highly specialized in nature. On the other hand, several processors can be assembled into a so-called symmetrical multiprocessor (SMP) which can also have access to powerful specialized processors, such as graphics processing units (GPUs), that are now increasingly being used for programmable computing resulting from their advent in the video-game industry, which has significantly reduced their cost and availability. Modern HPC-oriented parallel computers are typically composed of several SMP nodes interconnected by a network. This kind of infrastructure is hierarchical and represents a first class of heterogeneous system in which the communication time between two processing units is different, depending on whether the units are on the same chip, on the same node, or not. Moreover, current hardware trends anticipate a further increase in the number of cores (in a hierarchical way) inside the chip, thus increasing the overall heterogeneity, even more toward building extreme-scale systems.

At a higher level, the emergence of heterogeneous computing now allows groups of users to benefit from networks of processors that are already available in their research laboratories. This is a second type of infrastructure where both the network and the processing units are heterogeneous in nature. Specifically, here the goal is to deal with networks that interconnect a (often high) number of heterogeneous computers that can significantly differ from one another in terms of their hardware and software architecture, including different types of CPUs operating at different clock speeds and under different design paradigms, and also different memory sizes, caching strategies, and operating systems.

At the high level, computers are increasingly interconnected together throughout wide area networks to form large-scale distributed systems with high computing capacity. Furthermore, computers located in different laboratories can collaborate in the solution of a common problem. Therefore, the current trends of HPC are clearly oriented toward extreme-scale, complex infrastructures with a great deal of intrinsic heterogeneity and many different hierarchical levels.

It is important to note that all the heterogeneity levels mentioned above are tightly linked. First of all, some of the nodes in computational distributed environments may be multicore SMP clusters. Second, multicore chips will soon be fully heterogeneous with special-purpose cores (e.g., multimedia, recognition, networking) and not only

GPUs mixed with general-purpose ones. Third, these different levels share many common problems such as efficient programming, scalability, and latency management. Hence, it is very important to conduct research targeting the heterogeneity at all presented hardware levels. Moreover, it is also important to take special care of the scalability issues, which form a key dimension in the complexity of today environment. The extreme scale of this environment comes from every level:

1. *Low Level*: number of CPUs, number of cores per processor;
2. *Medium Level*: number of nodes (e.g., with memory);
3. *High Level*: distributed/large-scale (geography dispersion, latency, etc.);
4. *Application*: extreme-scale problem size (e.g., calculation-intensive or data-intensive).

In 2008, the knowledge on how to efficiently use program or scale applications on such infrastructures was still vague. This was one of the main challenges that researchers wanted to take on. Therefore, at that time, we decided to launch the COST Action for high-performance and extreme-scale computing in such complex environments entitled "*Open European Network for High-Performance Computing in Complex Environments*." The main reasons were as follows:

- There was a huge demand in terms of computational power for scientific and data-intensive applications;
- The architectural advances offered the potential to meet the application requirements;
- None of the state-of-the-art solutions in HPC at that time allowed exploitation to this potential level;
- Most of the research carried out in this area was fragmented and scattered across different research teams without any coordination.

COST[1] was indeed an appropriate framework for the proposed Action. The main goal of this Action was to overcome the actual research fragmentation on this very hot topic by gathering the most relevant European research teams involved in all the scientific areas described above (from the CPU core to the scientific applications) and coordinate their research.

Summarizing, this project within the COST framework allowed us to expect some potential benefits such as high-level scientific results in the very important domain of high-performance and extreme-scale computing in complex environment; strong coordination between different research teams with significant expertise on this subject; a better visibility of the European research in this area; and a strong impact on other scientists and high-performance applications.

[1]European Cooperation in Science and Technology: http://www.cost.eu.

1.2 SCIENTIFIC ORGANIZATION

1.2.1 Scientific Focus

The expected scientific impacts of the project were to encourage the specific community to focus research on hot topics and applications of interest for the EU, to propagate the collaboration of research groups with the industry, to stimulate the formation of new groups in new EU countries, and to facilitate the solution of highly computationally demanding scientific problems as mentioned above. For this, the groups involved in this Action collaborated with several scientific and industrial groups that could benefit from the advances made by this Action, and prompted the incorporation of new groups to the network.

To achieve the research tasks, different leading European research teams participated in the concrete activities detailed in Section 1.3.

1.2.2 Working Groups

Four working groups were set up to coordinate the scientific research:

- numerical analysis for hierarchical and heterogeneous and multicore systems;
- libraries for the efficient use of complex systems with emphasis on computational library and communication library;
- algorithms and tools for mapping and executing applications onto distributed and heterogeneous systems;
- applications of hierarchical-heterogeneous systems.

It is important to note that these working groups targeted vertical aspects of the architectural structure outlined in the previous section. For instance, the Action's goal was to carry out work on numerical analysis at the multicore level, at the heterogeneous system level, as well as at the large-scale level. The last working group (Applications) was expected to benefit from research of the other three groups.

1.3 ACTIVITIES OF THE PROJECT

To achieve the goal of this Action, the following concrete activities were proposed. The main goal was to promote collaboration through science meetings, workshops, schools, and internships. This allowed interchange of ideas and mobility of researchers.

1.3.1 Spring Schools

The goal was to provide young researchers with a good opportunity to share information and knowledge and to present their current research. These schools contributed to the expansion of the computing community and spread of EU knowledge.

1.3.2 International Workshops

The goal of these meetings was to take the opportunity during international conferences to meet the attendees and other researchers by co-locating workshops.

1.3.3 Working Groups Meetings

The scientific work plan was divided among different working groups. Each working group had substantial autonomy in terms of research projects. A leader nominated by the Management Committee led each working group. Members of a given working group met once or twice a year to discuss and exchange specific scientific issues and problems.

1.3.4 Management Committee Meetings

These meetings were devoted to the organization of the network and ensured the scientific quality of the network.

1.3.5 Short-Term Scientific Missions

The goal of short-term scientific missions (STSMs) was to enable visits by early stage researchers to foreign laboratories and departments. This was mainly targeted at young researchers to receive cross-disciplinary training and to take advantage of the existing resources. The goal was to increase the competitiveness and career development of those scientists in this rapidly developing field through cutting-edge collaborative research on the topic.

1.4 MAIN OUTCOMES OF THE ACTION

We believe that this COST Action was a great success. It gathered 26 European countries and 2 non-COST countries (Russia and South Africa). We have held 12 meetings and 2 spring schools. Fifty-two STSMs have been carried out. We have a new FP7 project coming from this Action (HOST). We have edited a book, and more than 100 papers have been published thanks to this Action.

We have set up an application catalog that gathers applications from the Action members. Its goal is to gather a set of HPC applications that can be used as test cases or benchmarks for researchers in the HPC field. The applications catalog is available at https://complexhpc-catalogue.bordeaux.inria.fr.

In total, the Action gathered more than 250 participants over the four years of the project.

We have sent a survey to the Action members. From this survey, it clearly appears that one of the greatest successes of the Action is the continuous strengthening of the network for many of its members both in terms of research teams and research domains. Many STSMs have been done through new network connections. Spring schools are seen as a major success, as they helped many young researchers to share and exchange knowledge and gain new connections. Many PhD theses have been defended during the course of the Action, and some of the management committee members have been invited on the defense board of some of these PhDs. Moreover, many presentations given during the meeting are considered very useful and have opened new research directions for other attendees.

We had four goals in this Action:

1. to train new generations of scientists in high-performance and heterogeneous computing;
2. to overcome research fragmentation, and foster HPC efforts to increase Europe's competitiveness;
3. to tackle the problem at every level (from cores to large-scale environment);
4. vertical integration to provide new integrated solutions for large-scale computing for future platforms.

Goal 1 has exceeded our expectations. The spring schools have been a great success. We had many STSMs, and the number of early stage researchers attending the meeting was always very high. We had great response from young researchers.

Goal 2 has also been achieved satisfactorily. Thanks to the Action, many joint researches have been carried out, and we have created a nice network of researchers within our Action. Moreover, many top-level publications have been made thanks to the Action.

Goal 3 has also been achieved. We have scientific results that cover the core level and the distributed infrastructure, as well as results that cover the intermediate layers. This is due to the fact that the consortium was made of researchers from different areas. This was very fruitful.

Goal 4 has not been achieved. The main reason is the fact that providing integrated solutions requires more research and development than a COST Action can provide. It goes far beyond the networking activities of COST Action.

1.5 CONTENTS OF THE BOOK

This book presents some of the main results, in terms of research, of the COST Action presented in this chapter. We are very proud to share this with the interested reader. We have structured the book according to the following parts in order to have a good balance between each part:

1. Numerical Analysis for Heterogeneous and Multicore Systems (Chapters 2, 3, and 4);
2. Communication and Storage Considerations in High-Performance Computing (Chapters 5, 6, 7, and 8);
3. Efficient Exploitation of Heterogeneous Architectures (Chapters 9, 10, 11, and 12);
4. CPU + GPU coprocessing (Chapters 13, 14, and 15);

5. Efficient Exploitation of Distributed Systems (Chapters 16 and 17);
6. Energy Awareness in High-Performance Computing (Chapters 18, 19, and 20);
7. Applications of Heterogeneous High-Performance Computing (Chapters 21, 22, and 23).

Chapter 2 discusses the redesign of the iterative solution algorithm in order to efficiently execute them on heterogeneous architectures. Chapter 3 studies the performance of a meshless numerical partial differential equation (PDE) solver, parallelized with OpenMP. The results depend on the way the computations are distributed and the way the cache is used. Chapter 4 presents the development of three parallel numerical algorithms for the solution of parabolic problems on graphs with a theoretical and experimental study of their scalability.

Chapter 5 surveys different techniques for mapping processes to computing units in order to optimize communication cost and reduce execution time. Chapter 6 offers a comprehensive overview of how to implement topology- and performance-aware collective communications. Chapter 7 analyzes the many-core architecture using a new model (K-model) in order to estimate the complexity of a given algorithm designed for such an architecture. Chapter 8 presents a scalable I/O storage system for the hierarchical architecture of Blue Gene computers featuring buffering and asynchronous I/O.

Chapter 9 describes algorithmic techniques for offline scheduling of independent workflows in order to satisfy user's quality of service. Chapter 10 investigates the advantage of using modern heterogeneous architecture for the efficient implementation of the Reed–Solomon erasure code. Chapter 11 analyzes the factors that enable the development of efficient parallel programs on modern many-core parallel architecture. Chapter 12 studies efficient solutions for electromagnetism applications in clusters of CPU + GPU nodes.

Chapter 13 describes how the functional performance model can be used to optimize the performance of scientific applications for heterogeneous and hierarchical platform. Chapter 14 presents algorithms for multilevel load-balancing on multicore and multi-GPU environments. Chapter 15 faces the all-pair shortest path problem for sparse graph. Different scheduling strategies are studied to efficiently solve such problems on heterogeneous systems.

Chapter 16 surveys different resource management systems and scheduling algorithms for HPC for clouds. Chapter 17 discusses different approaches for performing resource discovery in large-scale distributed systems.

Chapter 18 focuses on how to optimize and adapt software solution to improve energy efficiency in the context of HPC application. Chapter 19 studies energy-aware scheduling policies for three scenarios of federated cloud dealing with energy awareness. Chapter 20 explores the use of heterogeneous chip multiprocessors for network security and strategy to improve energy consumption in such contexts.

Chapter 21 describes the "jungle computing paradigm," which consists in gathering a complex hierarchical collection of heterogeneous computing hardware with an application to hyperspectral remote sensing. Chapter 22 presents a new model for image and video processing based on parallel and heterogeneous platforms in order to improve the performance of the application when dealing with high-definition images. Chapter 23 applies load-balancing techniques to efficiently execute tomographic reconstruction using hybrid GPU + CPU systems.

As you can see, this covers a large spectrum of results and topics on HPC and heterogeneous systems.

We wish you a fruitful and enjoyable time with this book.

ACKNOWLEDGMENT

This publication is supported by COST.