Journal of Computational Science 14 (2016) 1–4

Contents lists available at ScienceDirect

Journal of Computational Science

journal homepage: www.elsevier.com/locate/jocs

Editorial

Route to exascale: Novel mathematical methods, scalable algorithms and Computational Science skills

ARTICLE INFO

Keywords:
Novel mathematical methods
Scalable algorithms
Exascale computing
Computational Science research methods
HPC

ABSTRACT

This editorial outlines the research context, the needs and challenges on the route to exascale. In particular the focus is on novel mathematical methods and mathematical modeling approaches together with scalable scientific algorithms that are needed to enable key science applications at extreme-scale. This is especially true as HPC systems continue to scale up in compute node and processor core count. These extreme-scale systems require novel mathematical methods to be developed that lead to scalable scientific algorithms to hide network and memory latency, have very high computation/communication overlap, have minimal communication, have fewer synchronization points. It stresses the need of scalability at all levels, starting from mathematical methods level through algorithmic level, and down to systems level in order to achieve overall scalability. It also points out that with the advances of Data Science in the past few years the need of such scalable mathematical methods and algorithms able to handle data and compute intensive applications at scale becomes even more important. The papers in the special issue are selected to address one or several key challenges on the route to exascale.

1. Research context, challenges and environment

Computational Science and High-Performance Computing (HPC) are key strategic assets for the developed economies such as the USA [1,6,7,19], EU and its innovative capacity [5,12,13,15], Japan and the BRICS countries – China, Russia, Brazil, Mexico, India, South Africa. Large scale computing in science and industry has become an indispensable way to tackle societal and scientific grand challenges, and to address the needs of industry to innovate in products and services in the developed economies and now propagating to BRICS and further a field. Computational approaches to scientific grand challenge problems such as the detection and treatment of diseases like cancer, modeling of the human brain, and climate forecasting are beginning to bear fruit. Computational Science, an interdisciplinary field that melds basic sciences, mathematical modeling, quantitative analysis techniques and HPC techniques, is proving integral in addressing the big problems in industries ranging from manufacturing and aerospace, to drug design and risk management.

According to the latest IDC Report [13], in Europe, 97% of companies that adopted HPC stated that they “could no longer compete or survive without it.” From the study it is also evident that Computational Science and HPC are of increasing importance for manufacturing (smart manufacturing), drug design, health and medicine, resource exploration (oil and gas, etc), weather forecasting and global climate change [13]. The areas according to IDC where Europe has an advantage, provides substantial contribution and can provide leadership, can be identified as: weather and climate research, clean and sustainable energy, automotive and aerospace design, bio-life sciences, particle physics and related fields, modeling of materials/molecular dynamics.

Many of the above areas represent also major societal drivers/challenges Worldwide, such as Energy, Climate Change, Urbanization, etc. [17] and scientific methods, in particular Computational Science and HPC are crucial to address these challenges.

Computational Science as an interdisciplinary field is undergoing rapid change. Scientific instruments from particle accelerators to DNA sequencers are generating petabytes of data faster than our current capacity to analyze it. ‘Big Data’ approaches to grand challenges are increasingly involving the integration of massive and many sources of data. Hardware designers, backed by governments from the EU and US to India and China, are in a global race to create supercomputers capable of exascale performance (1000 times the performance of the very large supercomputers of today). New computational models such as CPU-GPUs and many-core architectures, and software models such as clouds and highly virtualized infrastructures, and data models such as schema-less unstructured data and high-throughput sensor streams, are increasing the number of design approaches and are fundamentally changing the range of problems that can be effectively addressed. In particular, Data and Compute Intensive science at scale poses an interesting challenge in combining the advances in both Data Science and Computational Science [6,12,22].

We have identified key societal and scientific grand challenges, in many areas we have tools for amassing the required analytical

http://dx.doi.org/10.1016/j.jocs.2016.04.014
1877-7503/

data, and our computing capacity continues to scale-up. However, there is an important ingredient which needs to be present, an adequate supply of highly trained computational scientists with the ability to understand complex scientific problems and the skills in mathematical modeling, simulation, Data Science techniques, and HPC to address them. Indeed the focus on training capable scientists and capable workforce with the right mix of skills to address the above challenges is growing both in Europe and the USA [4]. The latest executive order of the US President Obama (July 2015) has focused on long term National Strategic Computing Initiative [20,21] that in addition to focusing on ensuring US leadership in HPC also states as key component education and training in order to educate and train scientists for the "next generation of HPC systems, covering fundamental concepts in modeling, simulation and data analytics, as well as the ability to formulate and solve problems using advanced computing [18]. An editorial focusing on Computational Science skills for exascale together with several papers focusing on education and training for exascale can be found in a special section with the same title in this special issue.

2. Novel mathematical methods and scalable algorithms on the route to exascale: needs, approaches and examples

There are several strategic documents on exascale computing [1,5,6,8,11,12,14,15] outlining the important role of mathematical methods and algorithms together with the co-design approach in producing robust applications that can leverage the emerging exascale architectures. In particular the ones with focus on novel mathematics for exascale are ESSI II report [11], the US DOE report on Applied Mathematics Research for Exascale Computing [8], and key components of the ETP4HPC Strategic Research Agenda [12]. The Key challenges to be tackled and key recommendations to be followed on the road to exascale that are identified [8,11,12] include:

- "A significant ... development of new models, discretizations, and algorithms implemented in new science application codes is required in order to fully leverage the significant advances in computational capability that will be available at the exascale. Many existing algorithms and implementations that have relied on steady clock speed improvements cannot exploit the performance trends of future systems."
- "... an intensive co-design effort is essential for success, where computer scientists, applied mathematicians, and application scientists work closely together to produce a Computational Science discovery environment able to exploit the computational resources that will be available at the exascale."
- The development of efficient, massively parallel and energy-efficient applications is highly dependent on the capabilities of the programming environment.
- Developing and "defining new programming APIs and languages for expressing heterogeneous massive parallelism in a way that provides an abstraction of the system architecture, promotes high performance and efficiency, and interoperates with energy and resiliency management systems". As well as developing "scalable tools and techniques for practicable analysis of the entire program performance".
- Development of "libraries and components that exploit massively parallel heterogeneous computing resources and are capable of adapting to changing execution contexts (caused, for example, by faults)".
- Developing "Energy-Efficient Basic Algorithmic Motifs. The vast majority of HPC applications can be broken down into a small number of basic algorithmic motifs such as dense linear algebra kernels, for solving linear systems and least-squares problems (i.e., LU and QR decompositions), sparse matrix–vector multiplications, etc. Investigation and the development of energy-efficient implementations of these motifs" is essential.

In fact, while tackling all the above challenges we have to have in mind the characteristics of the targeted exascale architectures [8] while developing the methods, algorithms, programming models, libraries and targeted applications as well as to apply active co-design approach to ensure scalability at all levels: mathematical, algorithmic and system levels [2,3,8] in order to achieve overall scalability.

The papers selected in this special issue address one or several of the above outlined challenges:

For example, Emad et al. [10], discusses the ability to exploit emerging exascale computational systems. It is envisaged that such systems will require a careful review and redesign of core numerical algorithms and their implementations in order to fully exploit multiple levels of concurrency, hierarchical memory structures and heterogeneous processing units that will become available in the exascale computational platforms. This paper presents the "unite and conquer" approach to solve linear systems of equations and eigenvalue problems on extreme scale computing systems. Indeed, there are two ways to optimize the execution of a restarted method on a large-scale distributed system. The first one is to optimize the number of floating point operations per restart cycle through maximizing the concurrency inside a restart cycle while minimizing latencies. The second way is to accelerate/improve the rate of convergence for a given computational scheme. The unite and conquer restarted approach focuses on decreasing the number of restart cycles by coupling either synchronously or asynchronously several restarted methods called also co-methods. In the end of a restart cycle, each co-method locally gathers available results of all collaborating co-methods and selects the best one in order to create its restarting information. Consequently this permits the global reduction of the number of cycles to convergence. The unite and conquer restarted methods are heterogeneous, fault tolerant, support asynchronous communications and present a big potential of load balancing. Due to these properties, they are well adapted to large-scale multi-level parallel architectures. The paper also shows the relevant programming paradigms that allow multi-level parallel expression of these methods and how the software engineering technology can contribute significantly in achieving performance scalability. It presents some experiments validating the approach for unite and conquer restarted Krylov methods on several parallel and distributed platforms.

Liu et al. [16] present an efficient mathematical approach for multi-objective multi constrained optimization for classification. They propose a novel sparse least squares support vector machine, named ramp loss least squares support vector machine (RLSSVM), for binary classification. By introducing a non-convex and non-differentiable loss function based on the ε-insensitive loss function, RLSSVM has several important advantages compared with the plain LSSVM: Firstly, it has the sparseness which is controlled by the ramp loss, leading to its better scaling properties; secondly, it can explicitly incorporate noise and outlier suppression in the training process, and thirdly, the non-convexity of RLSSVM can be used to find efficiently solution by the Concave-Convex Procedure (CCCP). Experimental results on several benchmark datasets show the effectiveness of the proposed method. Efficient parallelization approaches are also outlined thus prompting how efficient mathematical methods developments can lead to efficient parallel algorithms down the stack.

Song et al. [24] present highly scalable algorithm for parallel 3-D FFT. Many scientific applications use parallel 3-D FFT and it is important to achieve high performance on large-scale systems with many thousands of computing cores. This paper describes