

Application Performance of Physical System Simulations

Vladimir GETOV^{a,1}, Peter M. KOGGE^b and Thomas M. CONTE^c

^aUniversity of Westminster, London, UK

^bUniversity of Notre Dame, Indiana, USA

^cGeorgia Institute of Technology, Atlanta, Georgia, USA

Abstract. Various parallel computer benchmarking projects have been around since early 1990s but the adopted so far approaches for performance analysis require a significant revision in view of the recent developments of both the relevant application domains and the underlying computer technologies. This paper presents a novel performance evaluation methodology based on assessing the processing rate of two orthogonal use cases — dense and sparse physical systems — as well as the energy efficiency for both. Evaluation results with two popular codes — HPL and HPCG — validate our approach and demonstrate its use for analysis and interpretation in order to identify and confirm current technological challenges as well as to track and roadmap the future application performance of physical system simulations.

Keywords. Performance and energy efficiency analysis, Peta- and Exa-scale systems, Performance evaluation methodology

1. Introduction

Computer simulation of physical real-world phenomena emerged with the invention of electronic digital computing and has been increasingly adopted as one of the most successful modern methods for scientific discovery. Arguably, the main reasons for this success has been the rapid development of novel computer technologies that has led to the creation of powerful supercomputers, large distributed systems, high-performance computing frameworks with access to huge data sets, and high throughput communications. In addition, unique and sophisticated scientific instruments and facilities, such as giant electronic microscopes, nuclear physics accelerators, or sophisticated equipment for medical imaging are becoming integral parts of those complex computing infrastructures. Subsequently, the term ‘e-science’ was quickly embraced by the professional community to capture these new revolutionary methods for scientific discovery via computer simulations of physical systems [1].

Focusing on the application domain for physical system simulations, this paper explains in detail our performance evaluation methodology with the most-recent results, analysis and interpretation based on the relevant technical report [2] produced by the Applications Benchmarking (AB) International Focus Team (IFT) as part of the IEEE

¹ Corresponding Author, School of Computer Science and Engineering, University of Westminster, 115 New Cavendish Street, London W1W 6UW, United Kingdom; E-mail: V.S.Getov@westminster.ac.uk.

International Roadmap for Devices and Systems (IRDS) initiative². Since 2015, IRDS is the successor of the International Technology Roadmap for Semiconductors (ITRS), which used to be provided by the Semiconductor Industry Association [3]. The mission of AB IFT is to identify key application areas, and to track and roadmap the performance of these applications for the next 15 years. Given a list of market drivers from the Systems and Architectures IFT, the AB IFT investigates and applies long-term analysis to identify the important or critical application areas for different user communities. Table 1 summarizes the ones that are under consideration at present.

Table 1. Application areas.

Application area	Description
Big data analytics	Data mining to identify nodes in a large graph that satisfy a given feature or features.
Feature recognition	Graphical dynamic moving image (movie) recognition of a class of targets (e.g. face, car). This can include neuromorphic / deep learning approaches such as deep neural networks.
Discrete event simulation	Large discrete event simulation of a discretized-time system. (e.g., large computer system simulation) Generally used to model engineered systems. Computation is integer-based.
Physical system simulation	Simulation of physical real-world phenomena. Typically, finite-element based. Examples include fluid flow, weather prediction, thermo-evolution. Computation is floating-point-based.
Optimization	Integer NP-hard optimization problems, often solved with near-optimal approximation techniques.
Graphics, augmented reality, virtual reality.	Large scale, real-time photorealistic rendering driven by physical world models. Examples include interactive gaming, augmented reality, virtual reality.

In order to track these areas, the AB IFT relies upon existing standard benchmarks where available. These benchmarks should fulfil two criteria:

- **Benchmark Code Availability:** There are several sets of benchmark codes available that cover each application area. However, many of these benchmarks either cover only a portion of an application area or cover more than one application area.
- **Benchmark Results Availability:** In order for benchmarks to be useful for projecting a trend in performance vs. time, there must be a sufficiently long history of benchmark scores. At a minimum, AB IFT believes that at least 4 years prior to the current day of results should be available.

The most important application codes for physical system simulations are typically based on finite-element algorithms — such as boundary element method, N-body problem, fast multipole method, hierarchical matrices, iterative stencil computations — while the computations constitute heavy workloads that conventionally are dominated by floating-point arithmetic. Example applications include areas such as climate modelling, plasma physics (fusion), medical imaging, fluid flow, and thermo-evolution. In addition, physical system simulation is critical to product design in the automobile and aerospace industries as well as for obtaining more accurate climate modelling and prediction. Our results confirm that:

- The area of physical system simulations requires innovative computer architectures because the data locality we have been expecting from our

² <https://irds.ieee.org/>

applications for three decades is disappearing. Novel solutions that can help addressing the “3rd Locality Wall” challenge [4] are urgently needed.

- Since the application area of physical system simulations is based predominantly on floating-point arithmetic, novel architecture proposals that address floating-point processing challenges are also expected to have substantial impact, particularly for dense system computations.
- Energy efficiency indicators need urgent improvements by at least an order of magnitude. This is equally valid for both homogeneous and heterogeneous architectures including accelerators and FPGAs.

The rest of this paper is organized as follows. Section 2 provides a review of previous work in the area. Section 3 introduces our novel approach and methodology while Section 4 presents experimental results with corresponding discussions. Section 5 outlines some of the important technological challenges. Finally, Section 6 concludes the paper.

2. Background

Taking the viewpoint of application programmers and end-users, this section outlines the major benchmarking efforts that have been part of the developments in this field over the years.

2.1. NAS Parallel Benchmarks

The NAS Parallel Benchmarks (NPB) include the descriptions of several (initially eight) “pencil and paper” algorithms [5]. Realistically, all of them are computational kernels although the authors claim that the suite includes three “simulated applications” but this claim is from the early 90s and it does not sound convincingly today. The NPB benchmarking methodology does not involve any hierarchy and each of the kernels is to be used individually for performance measurements. The codes cover only the Computational Fluid Dynamics (CFD) application domain which is of primary interest for NASA.

2.2. GENESIS Distributed-Memory Benchmarks

The GENESIS codes [6] were developed in a 3-layer hierarchy — low-level micro-benchmarks, kernels, and compact applications. This was intended to express the performance of higher-level codes via a composition of performance results produced by the kernels in the layer below. However, this proved to be a difficult task, particularly when including sufficiently broad set of computational science codes in the compact applications layer.

2.3. PARKBENCH Committee

The PARKBENCH Public International Benchmarks for Parallel Computers [7]. This was an ambitious international effort to glue together the most popular parallel benchmarks at that time — NPB, GENESIS, and several kernels including LINPACK

[8]. The PARKBENCH suite adopted the hierarchical approach from GENESIS, thus inheriting the same difficulties described above.

2.4. SPEC

All major machine vendors have participated in the development of SPEC HPG (High Performance Group), since achieving portability across all involved platforms has been an important concern in the development process [9]. The goal was to achieve both functional and performance portability. Functional portability ensured that the makefiles and run tools worked properly on all systems, and that the benchmarks ran and validated consistently. To achieve performance portability, SPEC accommodated several requests by individual participants to add small code modifications that took advantage of key features of their machines. There are many SPEC HPG benchmarking results available, but their main role is to confirm that new hardware products and platforms have been validated by the vendors.

2.5. Dwarfs — Computational Patterns

Another more recent “pencil and paper” parallel benchmark suite is the Dwarfs Mine based on the initial “Seven Dwarfs” proposal (2004) by Phillip Colella. The Dwarfs (computational patterns) are described as well-defined targets from algorithmic, software, and architecture standpoints. The number of Dwarfs (which are really kernels with some of them mapped to NPB) was then extended to 13 in the “View from Berkeley” Technical Report [10]. The report confirms “presence” of the 13 Dwarfs in 6 broad application domains — embedded computing, general-purpose computing, machine learning, graphics/games, databases and RMS (recognition/mining/synthesis) codes. Some recent studies suggest that more Dwarfs should be added for other application domains, while it is also not clear if the existing ones are sufficient for the domains described in the “View from Berkeley” Technical Report. The Dwarfs Mine description adopts a bottom-up hierarchical approach like GENESIS and then PARKBENCH. Although more systematic, it suffers from the same benchmarking hierarchy difficulties. Furthermore, the availability of benchmarking codes and results is very limited but even more importantly, the application domains are different from the ones selected by the AB IFT in the IEEE IRDS initiative.

3. Methodology

Over the years, the relevant benchmarking projects described in Section 2 above, have covered predominantly dense physical system simulations, in which high computational intensity carries over when parallel implementations are built to solve bigger problems faster. As long as emphasis was on dense problems, this approach resulted in systems with increasing computational performance and was the presumption behind the selection of the LINPACK benchmark [8] for the very popular semi-annual TOP500 rankings of supercomputers [11].

Many new applications with very high economic potential — such as big data analytics, machine learning, real-time feature recognition, recommendation systems, and even physical simulations - have been emerging in the last 10-15 years. However,

these codes typically feature irregular or dynamic solution grids and spend much more of their computation in non-floating-point operations such as address computations and comparisons, with addresses that are no longer regular or cache-friendly. The computational intensity of such programs is far less than for dense kernels, and the result is that for many real codes today, even those in traditional scientific cases, the efficiency of the floating-point units that have become the focal point of modern core architectures has dropped from the >90% to <5%. This emergence of applications with data-intensive characteristics — e.g. with execution times dominated by data access and data movement — has been recognized recently as the “3rd Locality Wall” for advances in computer architecture [4].

To highlight the inefficiencies described above, and to identify architectures which may be more efficient, a new evaluation code was introduced in 2014 called HPCG³ (High Performance Conjugate Gradient) benchmark [12]. HPCG also solves $Ax=b$ problems, but where A is a very sparse matrix — normally, with 27 non-zeros in rows that may be millions of elements in width. On current systems, floating point efficiency mirrors that seen in full scientific codes. For example, one of the fastest supercomputers in the world in terms of dense linear algebra is the Chinese TaihuLight, but that same supercomputer can achieve only 0.4% of its peak floating-point capability on the sparse HPCG benchmark. Detailed analysis lead to the conclusion that HPCG performance in terms of useful floating-point operations is dominated by memory bandwidth to the point that the number of cores and their floating-point capabilities are irrelevant [13]. There are of course application codes with highly irregular and latency-bound memory access that deliver significantly lower performance, but they are uncommon. While HPCG does not represent the worst-case scenario, it has been widely accepted as a typical performance yardstick for memory-bound applications.

Therefore, our selected benchmark codes that cover the “Physical System Simulation” application area of interest are the High-Performance LINPACK (HPL) and the HPCG. Both are very popular codes with very good regularity of results since June 2014. Another very important reason for selecting HPL and HPCG is that they represent different types of real-world phenomena — the HPL models dense physical systems while the HPCG models sparse physical systems. Therefore, the available benchmarking results provide excellent opportunities for comparisons and interpretation, as well as lay out a relatively well-balanced overall picture of the whole domain for physical system simulation applications. Our approach is to explore a 3-dimensional space — dense systems performance, sparse systems performance, and energy efficiency for both cases.

4. Performance Results

With HPL as the representative of dense system performance and HPCG as the representative for sparse systems, there are readily available performance and energy results published twice per year (June and November) with rankings of up to 500 systems for those two benchmarks since June 2014.

³ <http://www.hpcg-benchmark.org/>

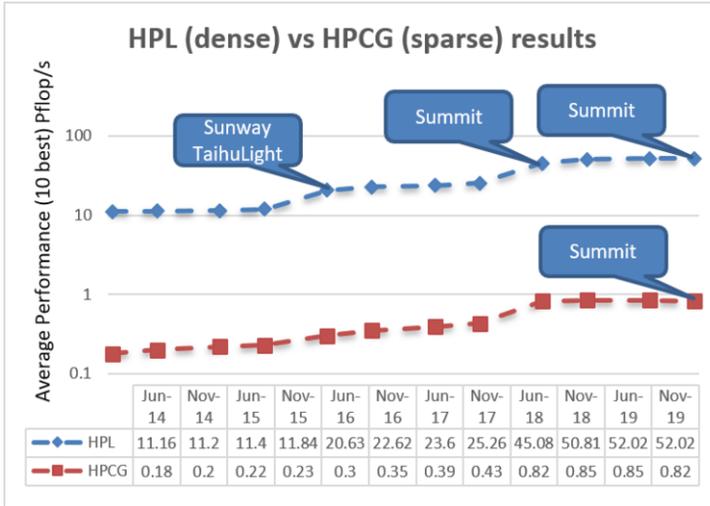


Figure 1. Average performance of HPL (dense systems) vs. HPCG (sparse systems).

We have further decided to use the average of the top 10 performance and energy results for each of these two benchmarks. This latter choice could be a point for further discussion and optimization of the benchmarking approach for this application domain. We have selected the 10 best only (rather than a larger number) because of the very limited HPCG results in the early years of publicly available HPCG measurements.

Figure 1 shows a significant performance gap of nearly 2 orders of magnitude between HPL and HPCG results in the last several years. The increase of the average HPL performance since June 2016 is because of the introduction of the Chinese Sunway TaihuLight system. The most recent increase of both HPL and HPCG performance is visible since June 2018 after the installation of the Summit supercomputer at ORNL. An optimistic expectation here would be to observe that the gap keeps closing and then assess the rate of this progress. Unfortunately, we do not have any evidence that the observed performance gap is in fact closing to any degree. Thus, we can draw the conclusion that one of the main challenges ahead will be to significantly increase sparse systems performance with any future computing systems designed for this application domain. While it is clear that reaching Eflop/s performance with HPL will happen soon, it is equally clear that this achievement will leave this significant gap between dense and sparse system performance unchanged.

Figure 2 complements the above analysis by showing a similar gap of approximately 2 orders of magnitude for the fraction of peak performance results between HPL and HPCG. This provides clear evidence of something we have known for years — our production codes, which usually implement sparse system simulations, are unable to deliver more than a few percent of the peak system performance that HPL results would seem to promise. The figure shows that this gap has not been reducing, and further points out the need to address sparse system performance in the next generation of computer architectures designed for this application domain.

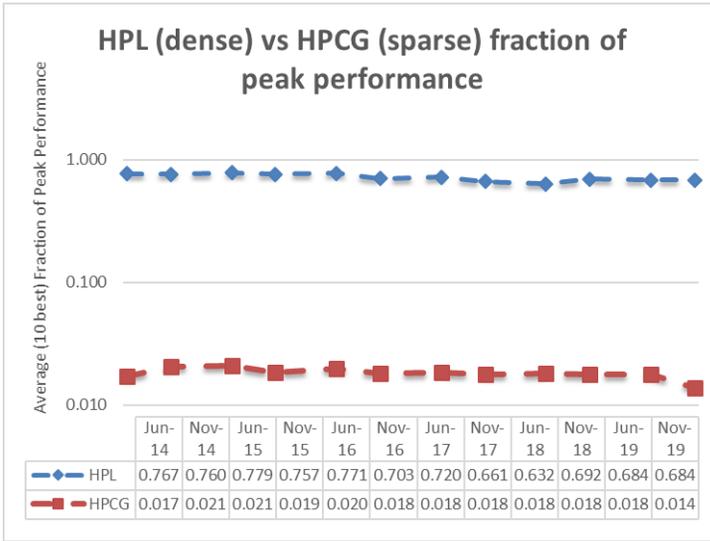


Figure 2. Fraction of peak performance for HPL (dense systems) vs. HPCG (sparse systems).

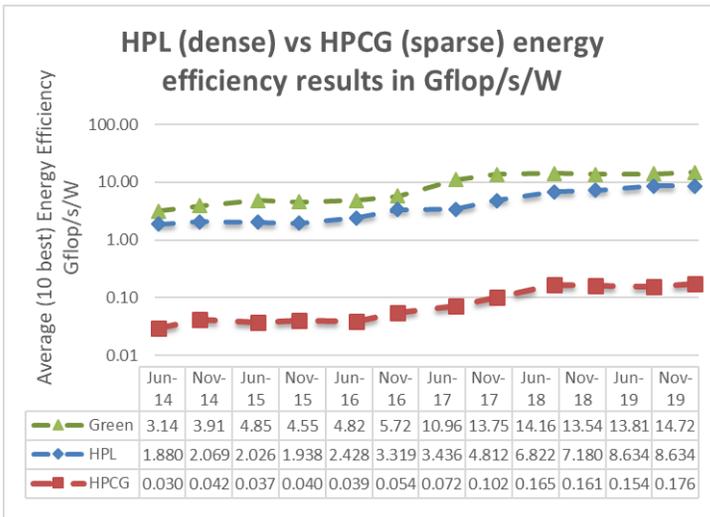


Figure 3. HPL (dense systems) vs. HPCG (sparse systems) vs. the most energy-efficient supercomputers on the Green 500 list.

The energy efficiency dimension of our evaluation is depicted in Figure 3. The current supercomputing designs appear to be able to scale up to 200 Pflop/s while remaining within the recommended 20 MW system power consumption envelope. An optimistic estimate based on this would require five times improvements in energy efficiency, and seven times improvements in the HPL performance currently delivered by the Summit supercomputer. However, such improvements are not realistic, since the best energy efficiency results and rankings are different from the HPL ranking (see comments above about the top 10 ranked results). Therefore, a more realistic projection

based on the current (end of 2019) Summit results is that one needs ten times energy efficiency improvement and ten times higher HPL performance to reach the Eflop/s barrier. Unfortunately, this would only fulfil the desired performance and energy efficiency for the computation of dense physical systems such as the HPL benchmark.

Similar performance versus energy efficiency analysis and projections for sparse systems based on the HPCG results look much more pessimistic. Here the two orders of magnitude lower performance delivered for sparse systems by the current supercomputing architectures strongly impact the energy efficiency.

5. Technological Challenges

Following the results and the discussion presented in the previous section, the main technological challenges that could help drive the future developments and improvements in the field of physical system simulation are summarised briefly below.

5.1. Reduced Data Movement

Since the late 1980s, reducing significantly the data movement has been one of the most important challenges towards achieving higher computer performance. Achieving higher bandwidth and lower latency for accessing and moving data — both locally (memory systems) and remotely (interconnection networks) — are key challenges towards building supercomputers at Eflop/s level and beyond. Breakthrough architecture solutions addressing those challenges could potentially enable up to two orders of magnitude higher performance particularly for sparse physical system simulations. More specifically, forthcoming designs of High Bandwidth Memory (HBM) such as HBM3+ and HBM4 expected to be released between 2022 and 2024, are likely to change substantially the application performance landscape for future supercomputers.

5.2. Efficient Floating-Point Arithmetic

Established in 1985, the IEEE 754 Standard for Floating-Point Arithmetic was renewed again in July 2019 [14]. However, the level of interest in this standard has been declining following critical comments about various important aspects of IEEE 754 including wasted cycles, energy inefficiencies, and accuracy. Unfortunately, the path forward is unclear at present and may involve keeping this standard as an option at least for backward compatibility while developing and implementing novel and more efficient solutions. Several efforts to address these problems follow two main approaches.

- Analysis of specific algorithms and re-writing of existing codes in order to improve the performance by using lower or mixed floating-point precision without compromising accuracy. This approach has been shown to work well but only for specific algorithms/codes, and with significant dedicated efforts for each case [15].
- More radical approaches proposing new solutions have been under development including the Posit Arithmetic proposal [16]. This work

introduces a new data type — posit — as a replacement for the traditional floating-point data type because of its advantages. For example, posits guarantee higher accuracy and bitwise identical results across different systems which have been recognized as the main weaknesses of the IEEE 754 Standard. In addition, they enable more economical design with high efficiency which lowers the cost and the consumed power while providing higher bandwidth and lower latency for memory access.

5.3. Low Consumed Power

During the last two decades, further developments of computer architecture and microprocessor hardware have been hitting the so-called “Energy Wall” because of their excessive demands for more energy. Subsequently, we have been ushering in a new era with electric power and temperature as the primary concerns for scalable computing. This is a very difficult and complex problem which requires revolutionary disruptive methods with a stronger integration among hardware features, system software and applications. Equally important are the capabilities for fine-grained spatial and temporal instrumentation, measurement and dynamic optimization, in order to facilitate energy-efficient computing across all layers of current and future computer systems. Moreover, the interplay between power, temperature and performance add another layer of complexity to this already difficult group of challenges.

Existing approaches for energy efficient computing rely heavily on power efficient hardware in isolation which is far from acceptable for the emerging challenges. Furthermore, hardware techniques, like dynamic voltage and frequency scaling, are often limited by their granularity (very coarse power management) or by their scope (a very limited system view). More specifically, recent developments of multi-core processors recognize energy monitoring and tuning as one of the main challenges towards achieving higher performance, given the growing power and temperature constraints. To address these challenges, one needs both suitable energy abstraction and corresponding instrumentation which are amongst the core topics of ongoing research and development work. Another approach is the use of application-specific accelerators to improve the application performance, while reducing the total consumed power which in turn minimises the overall thermal energy dissipation.

6. Conclusions

The application area of physical system simulations urgently needs novel and innovative architectures that provide solutions resolving the 3rd Locality Wall challenge. This includes both novel memory systems and interconnection networks offering much higher bandwidth and lower latency. Energy efficiency indicators also need urgent improvements by at least an order of magnitude. This requirement is equally valid for both homogeneous and heterogeneous architectures (including accelerators and FPGAs) that need further comparisons and analysis. Since the application area of physical system simulations is based predominantly on floating-point arithmetic, novel architecture proposals that address floating-point processing challenges can also be expected to have substantial impact, particularly for dense system computations.

7. Acknowledgements

We gratefully acknowledge the publicly available HPL and HPCG performance and power consumption results which enabled the presentation and analysis in Section 4 above. In addition, we would also like to thank Geoffrey Burr (IBM), Satoshi Matsuoka (RIKEN) and Takeshi Iwashita (Hokkaido University) for their useful comments as well as Linda Wilson (IEEE IRDS) for all her help and support.

References

- [1] V. Getov, e-Science: The Added Value for Modern Discovery, *Computer* **41**(8), 30–31, IEEE Computer Society, 2008.
- [2] G. Burr, T. Conte, V. Getov, P.M. Kogge, M. Levy, S. Matsuoka, D. Sunwoo, J. Torrellas, *Application Benchmarking*, The IEEE International Roadmap for Devices and Systems, IEEE, 2019.
- [3] F.D. Wright, T.M. Conte, Standards: Roadmapping Computer Technology Trends Enlightens Industry, *Computer* **51**, (6), 100–103, IEEE Computer Society, 2018.
- [4] P.M. Kogge, Data Intensive Computing, the 3rd Wall, and the Need for Innovation in Architecture, *Argonne Training Program on Extreme-Scale Computing*, August 2017, slides: http://extremecomputingtraining.anl.gov/files/2017/08/ATPESC_2017_Dinner_Talk_6_8-4_Kogge-Data_Intensive_Computing.pdf, video: <https://youtu.be/ut9sBnwF6Kw>.
- [5] D.H. Bailey, E. Barszcz, J.T. Barton, D.S. Browning, R.L. Carter, L. Dagum, R.A. Fatoohi, P.O. Frederickson, T.A. Lasinski, R.S. Schreiber, H.D. Simon, V. Venkatakrisnan, S.K. Weeratunga. The NAS Parallel Benchmarks, *Int. J. of Supercomputer Applications* **5**(3), 63–73, 1991, <http://www.davidhbailey.com/dhbpapers/benijsa.pdf>.
- [6] C. Addison, V. Getov, A. Hey, R. Hockney, I. Wolton. The GENESIS Distributed-Memory Benchmarks. In: J. Dongarra and W. Gentzsch (Eds.) *Computer Benchmarks, Advances in Parallel Computing* **8**, 257–271, Elsevier Science Publishers, 1993.
- [7] D.H. Bailey, M. Berry, J. Dongarra, V. Getov, T. Haupt, T. Hey, R.W. Hockney, D. Walker, “PARKBENCH Report-1: Public International Benchmarks for Parallel Computers”, TR UT-CS-93-213, *Scientific Programming* **3**(2), 101–146, 1994, <http://www.davidhbailey.com/dhbpapers/parkbench.pdf>.
- [8] J. Dongarra, P. Luszczek, A. Pettit, The LINPACK Benchmark: Past Present and Future, *Concurrency and Computation: Practice and Experience* **15**(9), 803–820, 2003.
- [9] R. Henschel, S. Wienke, B. Wang, S. Chandrasekaran, G. Juckeland, J. Li, V.G. Vergara Larrea, Using the SPEC HPG Benchmarks for Better Analysis and Evaluation of Current and Future HPC Systems, Tutorial at ISC18, Frankfurt, Germany, June 2018, slides: http://pages.iu.edu/~henschel/ISC18/SPECBenchmarksTutorial-Presentation_ISC2018.pdf.
- [10] K. Asanovic, R. Bodik, B.C. Catanzaro, J. J. Gebis, P. Husbands, K. Keutzer, D.A. Patterson, W.L. Plishker, J. Shalf, S. W. Williams, K. A. Yelick, The Landscape of Parallel Computing Research: A View from Berkeley, *TR UCB/EECS-2006-183*, University of California, Berkeley, Dec. 2006, <http://www.eecs.berkeley.edu/Pubs/TechRpts/2006/EECS-2006-183.pdf>.
- [11] E. Strohmaier, H.W. Meuer, J. Dongarra, H.D. Simon, The TOP500 List and Progress in High-Performance Computing, *Computer* **48**(11), 42–49, IEEE Computer Society, 2015.
- [12] J. Dongarra, M.A. Heroux, P. Luszczek, High-Performance Conjugate-Gradient Benchmark: A New Metric for Ranking High Performance Computing Systems, *Int. J. of High-Performance Computing Applications* **30**(1), 3–10, 2016.
- [13] V. Marjanović, J. Gracia, C. W. Glass, Performance modeling of the HPCG benchmark, *Proc. Int. Workshop on Performance Modeling, Benchmarking and Simulation of High-Performance Computer Systems*, pp. 172–192, Springer, 2014.
- [14] *IEEE 754-2019 – IEEE Standard for Floating-Point Arithmetic*, IEEE xPlore, July 2019.
- [15] A. Haidar, P. Wu, S. Tomov, J. Dongarra, Investigating Half Precision Arithmetic to Accelerate Dense Linear System Solvers, *Proc. Scala Workshop at SC'17*, pp. 1–8, ACM, 2017.
- [16] J.L. Gustafson, I.T. Yonemoto, Beating Floating Point at its Own Game: Posit Arithmetic, *Supercomputing Frontiers and Innovations* **4**(2), 71–86, 2017.