**Proceedings of the 7th GACM Colloquium on Computational Mechanics
for Young Scientists from Academia and Industry
October 11-13, 2017 in Stuttgart, Germany**

An HPC technology review with respect to large scale engineering applications

Ralf Schneider[1]*

Micro Abstract
An overview about the current state of the art in HPC-systems and technology will be given. Especially the current trends in accelerators, vector and many-core architectures, resulting from the stagnating scalar per core performance of classical CPUs will be addressed. In view of large scale engineering applications from the fields of structural mechanics and fluid dynamics the current and emerging bottlenecks as well as the technological and intellectual challenges will be discussed.

[1]Numerical Methods & Libraries, High Performance Computing Center Stuttgart, Stuttgart, Germany
***Corresponding author**: schneider@hlrs.de

Introduction

Taking a look at the TOP500 list[1], which twice a year ranks high performance computing (HPC) systems by the sustained performance they are able to achieve, one recognizes that since several years the increase in performance of HPC systems is no longer generated from an increased single core performance but instead from an increased system size which is expressed by an increased core count. In fact, when taking a look at so called accelerated systems, facilitating graphic processors or Intel many core architectures one even notices a decrease in CPU clock frequency. Expressed in a somewhat polemic way this means that nowadays HPC systems are not getting faster, they only get bigger.
One problem with this development is the fact, that the two essential subsystems system memory and network interconnect, which are, beside the single core performance, determining the sustained performance of a massive parallel application, have not kept pace with the increased on chip parallelism and performance.
In this contribution we will shed light on the implications the current technological developments in the area of HPC will have with respect to the solution of the next generation large scale engineering problems from the fields of structural mechanics and fluid dynamics. Further on it will be explained why a paradigm shift in software engineering and development has to happen if the next generation engineering problems should be solved by means of post peta-scale HPC systems that we will be seen in the coming years.

1 Trends in Performance and Efficiency

In figure 1 the development of the accumulated performance of the world's 500 fastest HPC systems, measured by the High-Performance Linpack Benchmark for Distributed-Memory Computers (HPL) [1,2] as published by the TOP500 list is shown. Additionally the performance of the two HPC systems ranked at position 1 and position 500 of the respective TOP500 list is depicted. As one can see from the red, solid regression line, it could have been expected during the last decades that the accumulated performance of the 500 fastest HPC systems exceeds the 10 $EFlop/s$ during the year 2016. Also the system ranked at position 500 could have been expected to achieve a sustained performance of 1 $PFlop/s$.

[1]https://www.top500.org

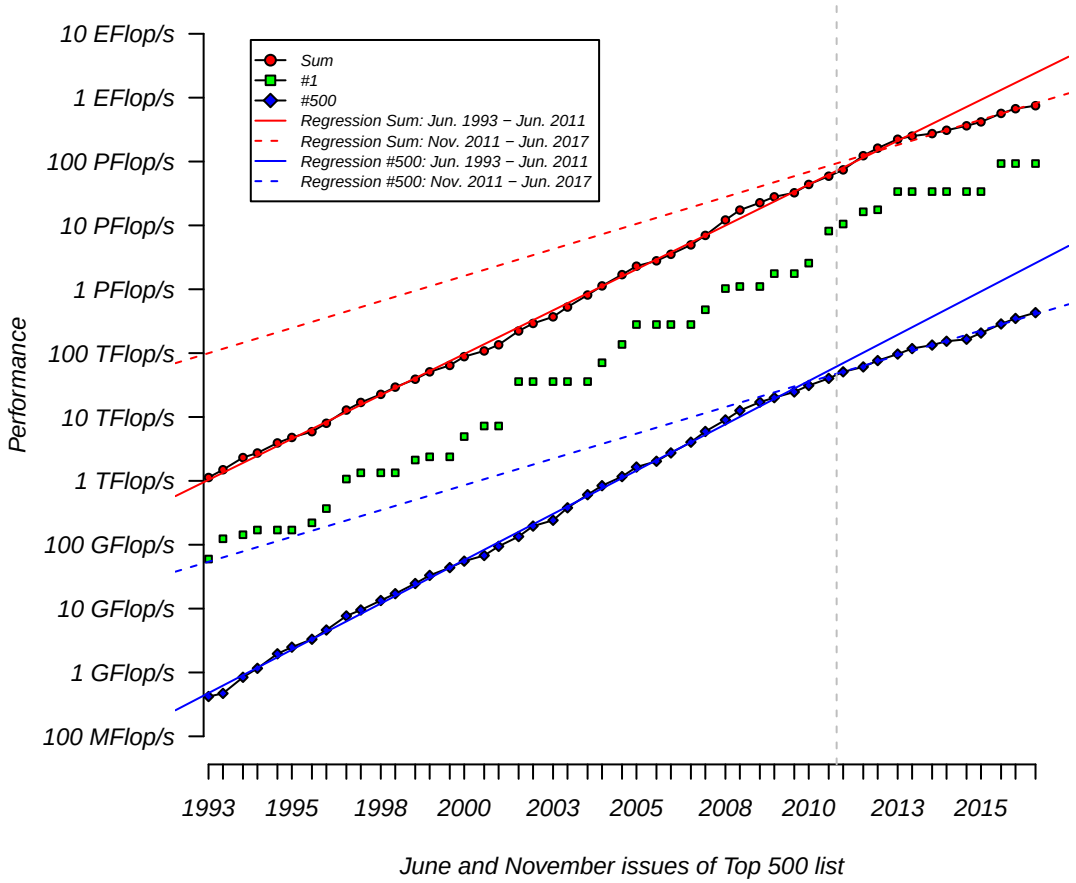

Figure 1. Performance development of HPC systems.

Today with the 2017 June issue of the TOP500 list published, these extrapolations have been proven wrong. Rather it can be seen from the two dashed regression lines, that since late 2013 the accumulated performance and already since 2009 the performance of the system ranked at position 500 increases with a much smaller slope than the expected one.
The vertical dashed grey line depicted in figure 1 marks the year 2011 when in November the first HPC system, the so called K computer located at the RIKEN Advanced Institute for Computational Science in Japan, was able to deliver a sustained performance R_{max} of 10.51 $PFlop/s$ measured by the HPL benchmark. If one takes a closer look to the architectures of the systems in the TOP500 list around the year 2011 it becomes obvious that starting in 2008 with the system Roadrunner the era of HPC systems accquering their perfromance by multi or many core CPUs begun. This trend is also shown in Table 1 where the architetures of the systems ranked at position one of the TOP500 list, which have always set the trend for the system architecture of the following years, is given. In addition to the increase of cores per CPU one can see from Table 1 that the the clock frequency of the single core remained almost constant at around 2GHz up to the present day. Another trend that was initiated by the Roadrunner system in 2008, besides breaking the 1 $PFlop/S$ sustained performance barrier by massively, increasing the system's core count, was the accelerator architecture. The system was equipped with general purpose dual core AMD Opteron CPUs that each were accelerated by an IBM PowerXCell 9 Core CPU. In this set-up The PowerXCell CPUs were not able to directly communicate with other CPUs in the cluster but were supplied with data by their host CPU. Today 5 out of the top 10 HPC systems are equipped with this accelerator technique, using cards of different kind even though this architecture makes a system much more complicated to program and by that reduces the possibility to use the system efficiently for today's general

purpose engineering software systems.
This statement can be proven to be true by a look to the extract of the June issue of the The High Performance Conjugate Gradients (HPCG) Benchmark list[2] presented in table 2. In the table the top three systems are listed along with the first one that features a pure x86_64 architecture and the first one that is build with the most recent vector CPUs. The HPCG benchmark is intended to complement the HPL benchmark in a way, that it tries to represent the essential and basic operations of applications executed nowadays on HPC systems. For that purpose it measures among other operations the performance of a sparse matrix-vector multiplication, vector updates and global dot products. These operations are quite different from the dense matrix solution techniques that are benchmarked by the HPL since they largely use indirect addressing off the data stored in memory. These data access patterns turn the solution of a system of equations into a compute problem whose performance is limited by the bandwidth that can be achieved for the transfer of data from memory to the CPU. While the memory bandwidth is already limited in modern general purpose x86_64 CPUs this is even more the case if the data have to be shifted to an accelerator from the host's memory to the accelerator's memory via the host CPU and the PCI-Express bus. This gets visible by the fraction of peak performance, the systems listed in table 2, are able to achieve in the HPCG benchmark. While the Earth-Simulator, a NEC-SX ACE vector system, as well as the K computer equipped with SPARC CPUs are able to harvest over 10% and 5% of their theoretical peak performance respectively, this value drops to 2.5% in the first system that appears in the HPCG list, which purely consists out of Intel Xeon E5 general purpose CPUs. The achievable sustained performance even drops further in the accelerated system Tianhe-2 close to 1% and drops below 0.5% in the Sunway many core architecture.

1.1 Options for the Future

Taking a look at the numbers presented in the two sections above, it becomes obvious, that HPC will no longer provide the same rate of acceleration that it did over the last decades. In accelerated many core systems it will get more and more difficult to achieve reasonable sustained performance with large scale integrated applications that depend on sparse matrix structures and indirect addressing techniques. In contrast, the architectures that can achieve reasonable sustained performance with engineering applications are no longer general purpose and software has to be especially for these systems. This last statement might also hold true for the accelerated many core systems with x86_64 architecture but only if new algorithms for the solution of engineering problems can be found that are able to cope with the memory bandwidth bottleneck.

[2] http://www.hpcg-benchmark.org/

Year	System	CPU architecture
2003	Earth-Simulator	NEC 1GHz Vector
2006	BlueGene/L	PowerPC 440 700MHz Dual Core
2008	Roadrunner	Opteron 1.4GHz Dual Core + Cell 3.2GHz 9-Cores
2010	Jaguar	Opteron 2.6 GHz 6 Cores
2011	K computer	SPARC 2GHz 8 Cores
2012	Sequoia	Power BQC 1.6GHz 16 Cores
2014	Tianhe-2	Intel Xeon E5 2.2GHz 12 Cores + Intel PHI 1.1GHz 57-Cores
2016	TaihuLight	Sunway SW26010 1.45GHz 260 Cores

Table 1. System architectures

Rank HPCG	System	Cores	HPL R_{max} [TFlop/s]	TOP500 Rank	HPCG [TFlop/s]	Fraction of Peak
1	K computer SPARC 8 C.	705024	10510	8	602.7	5.3
2	Tianhe-2 Intel PHI 57 C.	3120000	33863	2	580.1	1.1
3	TaihuLight SW26010 260 C.	10649600	93015	1	480.8	0.4
10	Pleiades Intel E5 10 C.	243008	5952	15	175.2	2.5
35	Earth-Simulator NEC-SX ACE 1 C.	8192	486	-	54.7	10.4

Table 2. Extract from the 2017 June issue of the HPCG list

In both cases one can not expect to continue with the software development like it was done during the last decades and still being able to get reasonable speed-up with simply executing the implementation on a bigger system. Instead of trying to develop engineering software in the most general purpose way towards program packages that are able to solve all kinds of engineering problems and that can be executed on every hardware platform, it will become more and more necessary to develop specialised software. These implementations will have to exploit inherent features of the posed engineering problem with respect to the targeted hardware platform on which the code is intended to be executed.

Conclusions

In Summary on can give the following three statements:

- Moore‘s law will come to an end by the mid of the next century.
- HPC will then no longer provide the same rate of acceleration that we got used to over the last decades.
- Algorithmic improvements along with specialised implementations have to take over

Which translate to the conclusion, that one has to invest in HPC application developments that are no longer directed towards general purpose codes able to run with limited Efficiency on every platform but directed towards the development of integrated, specialised applications, tailored to specialised systems. Only in that way it will be possible to harvest the performance which will be provided by HPC systems of the post peta-scale generation via extreme parallelism, specialised CPU architectures, accelerators, and hierarchical memory systems.

References

[1] J. Dongarra, C. Moler, J. Bunch, and G. Stewart. *LINPACK Users' Guide*. Society for Industrial and Applied Mathematics, 1979.

[2] A. Petitet, R. C. Whaley, R. C. Dongarra, and A. Cleary. Hpl - a portable implementation of the high-performance linpack benchmark for distributed-memory computers, Feb. 2016.