Viewpoint (Communications of the ACM) Draft

Michael J. Flynn,
Oskar Mencer,
Veljko Milutinovic,
Goran Rakocevic,
Per Stenstrom,
Roman Trobec,
Mateo Valero,

Has the time come to move from petaflops (on simple benchmarks) to petadata per unit of time and power (on sophisticated benchmarks)?

The race to build ever-faster supercomputers is on, with more contenders than ever before. But, the current goals set for this race may not lead to the fastest computation for particular applications.

Introduction

The supercomputer community is now facing an interesting situation: Systems do exist, which, for some sophisticated applications, and some relevant performance measures, demonstrate an order of magnitude higher performance [Weston2011, Lindtjorn2011, Oriato2010], compared to the top systems from the Top 500 Supercomputers list [Top500web1994], but are not on that list, because their LINPACK performance is poor.

Typical applications of such systems are: (a) geo-mechanical simulations based on sparse matrix algorithms, which do not scale beyond a few nodes on conventional systems [Lindtjor2011], or (b) financial stochastic PDEs [Weston2011], and (c) high resolution (70Hz) seismic modeling in Oil&Gas industry [Oriato2010].

Relevant performance measures for which the above mentioned performance ratio improvements apply are: (a) performance per watt, (b) performance per cubic foot, or (c) performance per monetary unit (dollar, yen, yuan, euro, etc.).

The above-mentioned systems are often times based on a kind of dataflow approach.

A creator of the Top 500 Supercomputers list rightfully claimed that this list sheds light on only one dimension of modern supercomputing [ACM2011a], which is a relatively narrow one. This paper tries to induce thinking about alternative performance measures for ranking, possibly ones with a much wider scope [ACM2011b]. This short communication is not offering a solution; it is offering a theme for brainstorming.

Having said all the above, the remaining text concentrates on the following issues: (a) rationales (what are the evolutionary achievements that may justify a possible paradigm shift in the ranking domain), (b) justification (what are the numerical measurements that

require rethinking), (c) suggestions (what are the possible avenues leading to potential improvements of the ranking paradigm). As usual, we conclude by: (a) restating the contribution of this short paper, (b) specifying to whom all this might be of benefit, and (c) opening possible directions for future research.

Rationales

For data flow systems, utilization of a relatively slow clock (even if deep pipelining is used) is typical, while the entire data flow is completed more efficiently. This is because the data flow approach enables low clock frequency, which allows for low power dissipation. Slow clock is not a problem for Big Data computations, since the speed of computation depends on pin throughput and local memory size/bandwidth inside the computational chip. In fact, opposite to popular belief, even if data flow is implemented using FPGA chips, in spite of the fact that general purpose connections inside FPGA chips bring a slowdown, the clock would not be slow because of the use of FPGAs - pin throughput and local memory size/bandwidth are the problem; not the computational unit speed. Therefore, if counting is oriented to performance measures correlated with clock speed, these systems perform poorly. However, if counting is oriented to performance measures sensitive to the amount of data processed, these systems may perform richly (see Figure 1). This is the first issue of importance.

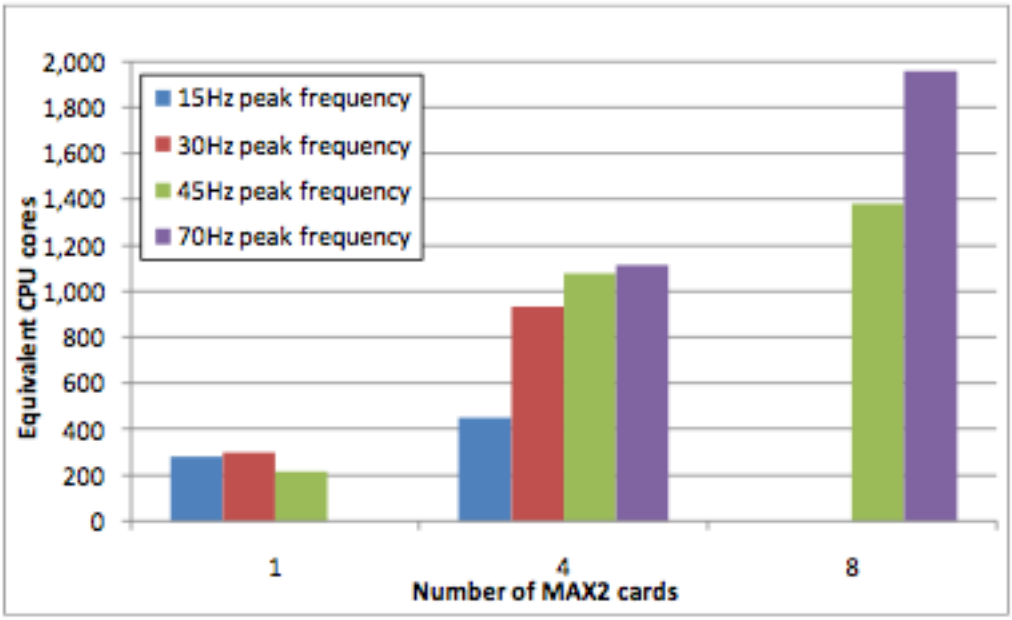

Figure 1: Performance of Maxeler-accelerated Finite Difference Modeling, using a large scale system based on MAX2 cards. Source: [Oriato2010]

The second issue of importance is related to the fact that, due to their lower clock speed, systems based on a kind of a data flow approach consume less power, less space, and less money, compared to systems driven by a fast clock (see Table 1 for support of this claim).

In addition to the above said, the third issue of importance is that systems based on a kind of data flow approach perform poorly on relatively simple benchmarks, which are typically not rich in the amount and variety of data structures. However, they perform fairly well on relatively sophisticated benchmarks, rich in the amount and variety of data structures (see Table 2).

All these three issues are further elaborated in the text to follow.

Streaming dataflow systems have the potential to retire a result every clock cycle. As such, given a certain number of output pins on a chip, dataflow computation can generate a large amount of results for a given clock frequency. Conversely, in order to achieve a certain performance level, dataflow can result in very low clock frequency for the given speed of computation. Consequently, if counting is oriented to performance measures correlated with clock speed, data flow computing looks very unappealing. However, if counting were oriented to performance measures sensitive to the amount of processed data, the conclusions would end up being different.

The sheer magnitude of the data flow parallelism can be used to overcome initial speed disadvantage. Indeed operating at lower frequency has the advantage of reducing the overall power. In order to achieve maximum acceleration the kernel application is compiled into a dataflow engine. Optimization creates a static dataflow machine by unrolling loops and inserting pipeline points at each stage of the data flow. The resultant array structure nowadays can be 500 pipeline stages deep, or even deeper, in future. Ideally in the static dataflow form, data can enter each stage of the pipeline every cycle. If, after this instantiation of the data flow engine, there is still additional silicon and pin bandwidth available on the accelerator, it may be possible to realize a second, third, or fourth instantiation on the same accelerator directly increasing the parallelism and the performance, for less power.

Low clock frequency results in low power consumption, and [Weston2011] shows that the measured speedups (31x and 37x) were achieved while reducing the power consumption of the 1U compute node (see Tables 1 and 2). Combining power and performance measures is a challenge that is already starting to be addressed by the Green500 list. However, evaluating radically different models of computation such as dataflow, remains yet to be addressed, and especially in the context of total cost of ownership.

Platform	Idle	Processing
Dual Xeon L5430 2.66GHz Quad Core 48GB DDR DRAM	185W	255W
(as above) with MAX2-4412C Dual Xilinx SX240T, 24GB DDR DRAM	210W	240W

Table 1: Power usage published by J. P. Morgan [Weston2011] for 1U compute nodes when idle (no program running) and while processing (the credit derivatives risk calculation). The essential point here is that the accelerated system runs X times faster (X denotes the speedup), and even takes a bit less power.

Precision	Speedup
Full Precision	31x
Reduced Precision	37x

Table 2: MaxNode-1821 vs. Eight Core Xeon Server speedup (referred to as X in Table 1), taken from a J.P. Morgan study [Weston2011].

Justification

Performance of an HPC system depends on the adaption of a computational algorithm to a scientific problem, discretization of the problem, mapping onto data structures, mapping onto representable numbers, the dataset size, the quality of the implementation, and the suitability of the underlying architecture compared to all the other choices in the spectrum of design options. In light of all these choices, how does one evaluate a computer system's suitability for a particular task such as climate modeling or genetic sequencing?

To shed more light on the above question, one can start from the following statement: If one runs LINPACK (a relatively simple benchmark dealing with matrix/vector multiplication) on a highly ranked Top 500 system (for example, one that offers a limited public remote access [Tianhe2011]), one obtains the performance of P Petaflops. If one runs the same benchmark on a modern dataflow system (for example, one used by a number of banking, geo-physics, and petrol companies - one such example, but not the only one, is [Maxeler2011], which is programmed in a variant of the Java language), one obtains the performance of P/M Petaflops, where M could be a relatively large number, greater than one. If one recalculates the obtained results for another performance measure (the amount of data processed (D) per unit of time and unit of power and unit of money, or similar), one obtains the performance ratio of D/m, where M>m, and m is also greater than one. In other words, for LINPACK, an Intel node might have an equal or better performance (measured in Petaflops) than a MaxNode; however, for several real applications, execution time for the same MaxNode is significantly smaller.

If one runs a relatively data intensive workload (e.g., order of gigabytes) used in banking environments [Weston2011]), and compares the same two systems for the same performance measure (data, per unit of time, per unit of power) the advantage is in favor of a modern data flow system, in the ratio of η:1, where η is a relatively small number higher than one.

If one runs a highly data intensive workload (e.g., order of terabytes) used by geophysicists [Lindtjorn2011]), and compares the same two systems for the same performance measure (data, per unit of time, per unit of power, or per unit of money) the

advantage is again in favor of the modern data flow system, in the ratio of n:1, where n is a relatively large number higher than one (n>η).

If one runs an extremely data intensive workload (e.g., order of petabytes) used by petrol companies [Oriato2010]), and compares the same two systems for the same performance measure (data, per unit of time, per unit of power) the advantage is considerably in favor of the modern data flow system, in the ratio of N:1, where N is a relatively large number higher than one (N>n>η).

Obviously, the dataflow approach is favourized if a more complex performance measure is used. It is further on favourized if more complex benchmarks are used. The first issue refers to major user concerns (processing duration and electricity bill). The second issue refers to major user needs (complex applications and purchase costs).

Yet at the end of the day, a decision has to be made as to which computer system to construct for a given scientific challenge, or more typically, for a wide range of scientific challenges. If indeed one machine needs to serve the entire scientific community of a country, and the machine is evaluated based on LINPACK, it may well be suboptimal for many of the scientific challenges. Even if we assume that it is the optimal machine given the set of applications, the question is whether a set of smaller, and perhaps to some extent specialized, custom machines for each algorithmic domain would not be able to serve the community in a more efficient way and give the tax payer more scientific progress for less money.

Suggestions

This short communication does not suggest that the Petaflops count be eliminated, but rather that a data centric measure could shed some more light on other aspects of HPC systems. One idea is to look at Petabytes per second per cubic foot per Watt for a particular algorithm and dataset size.

Of course, financial considerations play a major role in computing. However, it is unreasonable to include non-transparent and ever negotiated pricing information into an engineering measure. We know that the cost of computer systems is dictated by the cost of the chips and the cost of the chips is a function of chip area.
So, adding a measure of performance per computational chip area could encapsulate intrinsic underlying costs of the various approaches.

Finally, the real test of a computer system lies in the hands of users. However, typically such users do not get to tell the whole story when publishing papers, due to obvious restrictions. If there were a way to capture the user experience in an objective fashion, this could really help with concise evaluation of computer systems. Yet, there does exist a solution to address the user satisfaction challenge: simply, if one purchases a system

that later on turns out not to be the right one, one does not purchase a new system of that same sort, and looks around for novel solutions.

Conclusions

The findings of this paper are of interest to those supercomputing users who wish to minimize not only the purchase costs, but also the maintenance costs, for a given performance requirement. Also to those manufacturers of supercomputing oriented systems who are able to deliver more for less, but are using unconventional architectures [Stojanovic2011].

Topics for future research include the ways to incorporate the price/complexity issues and also the satisfaction/profile issues. The ability issues (availability, reliability, extensibility, partition ability, programmability, portability, etc.) are also of importance for future of any ranking effort.

In conclusion, whenever a paradigm shift happens in computer technology, computer architecture, or computer applications, a new approach has to be introduced. The same type of thinking, as the one presented in this paper, happened at the time when GaAs technology was introduced for high-radiation environments, and had to be compared with silicon technology, for a new set of relevant architectural issues. Solutions which ranked high until that moment, suddenly obtained new and relatively low ranking positions [HelbigMilutinovic1989].

Box

If indeed one machine needs to serve the entire scientific community of a country, and the machine is evaluated based on LINPACK, it may well be suboptimal for many of the scientific challenges.

References

[Weston2011] Weston, S., (JP Morgan) Spooner, J., Racaniere, S., Mencer O., (Imperial College London), “Rapid Computation of Value and Risk for Derivatives Portfolio,”

Concurrency and Computation: Practice and Experience, Special Issue Paper, July 2007, doi: 10.1002/cpe.1778.

[Lindtjorn2011] Lindtjorn, O., (Schlumberger), Clapp R., (Stanford University) Pell, O., Mencer, O., Flynn, M., (Stanford University), Fu, H., (Tsinghua University), "Beyond Traditional Microprocessors for Geoscience High-Performance Computing Applications," IEEE Micro, vol. 31, no. 2, March/April 2011.

[Oriato2010] Oriato, D., O. Pell, O., Andreoletti, C., Bienati, N., "Finite Difference modeling beyond 70Hz with FPGA acceleration," SEG 2010, HPC Workshop, Denver, USA, October 2010.

[HelbigMilutinovic1989] Helbig, W., Milutinovic, V., "The RCA's DCFL E/D MESFET GaAs 32-bit Experimental RISC Machine," IEEE Transactions on Computers, Vol. 36, No. 2, February 1989, pp. 263-274.

[ACM2011a] Geller, T., "Supercomputing's Exaflop Target," Communications of the ACM, Vol. 54, No. 8, Aug. 2011, pp. 16 - 18.

[ACM2011b] Singh, S., "Computing without Processors," Communications of the ACM, Vol. 54, No. 8, August 2011, pp. 46 - 54.

[Maxeler2011] Maxeler Technologies, http://www.maxeler.com/content/frontpage/, October 20, 2011.

[Top500web1994] Dongarra, J., Meuer, H., and Strohmaier, E., "Top500 supercomputer sites". http://www.netlib.org/benchmark/top500.html. (updated every 6 months).

[Tianhe2011] Top500.org, "National Supercomputing Center in Tianjin: Tianhe-1A," http://www.top500.org/system/10587, October 20, 2011.

[Stojanovic2012] Sasa Stojanovic *et. al*. "A Comparative Study of Selected Hybrid and Reconfigurable Architectures," Proceedings of the IEEE ICIT Conference, Kos, Greece, March 2012.

About the Authors

Michael J. Flynn is with the Stanford University, US
Oskar Mencer is with the Imperial College, London, UK
Veljko Milutinovic is with the University of Belgrade, Serbia
Goran Rakocevic is with the Mathematical Institute, Belgrade, Serbia
Per Stenstrom is with the Chalmers University of Technology, Sweden
Roman Trobec is with the Jozef Stefan Institute, Slovenia
Mateo Valero is with the Barcelona Supercomputing Centre, Spain

Acknowledgments

This research was supported by discussions at the Barcelona Supercomputing Centre, during the FP7 EESI Final Project Meeting. The strategic framework for this work was inspired by Robert Madelin and Mario Campolargo of the EC, and was presented in the keynote of the EESI Final Project Meeting.