3. Infrastructure requirement for software development scientific and high-performance computing

Balance in components performances need to be seriously taken into consideration. Currently existing HPC systems based on GPU offers disproportional amount of floating point computing power relative to memory bandwidth and interconnect bandwidth, more specifically, inter-node interconnect bandwidth.

Accordingly, software algorithms that has intrinsically low arithmetic intensity (need to move large amount of data from memory to computing unit or between memories on different nodes) cannot benefit at all from the GPU-CPU hybrid architectures that are available now (ex. Lassen/Sierra) and likely to be the case in near future (El Capitan/Frontier).

There are at least a few types of simulations that heavily rely on three-dimensional Fast Fourier Transform (FFT) that has been known to be low arithmetic intensity. The reason has been very well known. It achieves low cpu cost scaling, from N2 to N log N, by reducing number of floating operations using the symmetry of operation pattern, however, the fact that one must perform operation on every single data. Therefore, FFT algorithm, which offers significant cpu cost reduction, cannot benefit from a HPC architecture that does not have sufficient memory bandwidth/interconnect bandwidth.

As an example, heFFTe library, while it shows perfect scaling on Summit GPU HPC system, ***achieves only 0.15% of peak floating point performance***, clearing showing that interconnect bandwidth is the bottleneck and the advertised computing power of GPU is not usable.

It is well known that density functional theory (DFT) simulations consume significant amount of HPC cpu cycles, not only in the US but globally. For instance, at HPC centers in the US, it was reported that about 15% of cpu cycles are consumed by DFT simulations, which heavily relies on 3D-FFT whose performance severely suffers from insufficient interconnect bandwidth relative to its computing power.

It is my understanding that classical MD codes typically use 3D-FFT in calculating coulomb interaction. Naturally, performance of such code will also be limited by the interconnect bandwidth, not by computing power of GPU.

Abundant misleading information: there are many DFT software that advertise “GPU ready”. Please be aware that currently existing GPU ready DFT software offers GPU performance only for limited type of calculations such as GW, hybrid exchange correlation, configuration interaction (CI), which have very high arithmetic intensity, or k-point parallization, which is not for simulations of large system (no high computing power necessary).

Please be aware the extent of impact to subfield of scientific communities not only in the US but other countries such as EU where nearly 100% of HPC systems are GPU based systems installed by the US vendors. Scientific communities that uses DFT or classical MD that rely on 3D-FFT all suffers from this imbalance between computing power and memory/interconnect bandwidth.

Also, in my opinion, dual memory architecture (CPU memory and GPU) memory should have been avoided since it will required additional set of interconnect for direct GPU-GPU communication. Please keep in mind that fast interconnect occupies significant portion of cost of HPC system. If it requires two sets, it would likely to end up compromising something due to the overall budget constraint.

My suggestion will be to wait until manufacturers become able to squeeze CPU, GPU and memory in one chip (SOC) so that there is only one memory space, only one set of interconnect and no need of moving data from CPU memory to GPU memory, no need of using proprietary programing language. I suppose RISC-V based SOC chip will have the potential. Ad-on GPU that relies on proprietary programming language should be avoided.

4. Developing and maintaining community software

Lack of stability in long term financial support. Open source software development is rather community service that the authors do not receive credit until the software gain sufficient support from scientific community. In other words, it has much long incubation time than conventional scientific research project. Too short evaluation cycle and/or too harsh financial penalty do not encourage software developer to server for community.

5. Challenges in building a diverse workforce and maintaining an inclusive professional environment

Too short evaluation cycle and/or too harsh financial penalty (reduction of budget etc) are too much adversity for individual’s family planning in particular to female due to biological constraint.

6. Requirements, barriers, and challenges to technology transfer, and building communities around software projects, including forming consortia and other non-profit organizations

Current HPC systems rely too much on proprietary technologies. Note: GPU programing language is proprietary, which majority of scientists do not understand at all. In addition, nothing is guaranteed for the programing environment used in the next generation HPC machine making it impossible to design “portable” software.

This was not the case when HPC system required compliers compatible with community standard such as IEEE.

7. Overall scope of the stewardship effort

I would like to see less emphasis on getting number one position on TOP500 list and more emphasis on usability, which will help more scientific outcome. Current HPC systems are nearly impossible for scientist to take full advantage of it.

8. Management and oversight structure of the stewardship effort

Structure looks ok, however, execution is not.

Co-design center is supposed to facilitate information sharing among of the stakeholders so as to develop the HPC system that should maximize scientific, engineering output and breakthrough. This must mean that a popularly used software such as DFT must be able to perform well and development of such software must not meet exceedingly difficult challenge: the conditions that are very poorly met.

This is clear evidence that communication between ECP, procurement team, hardware design and development team were not effective.

As examples: I have a BES funded software center in developing a DFT code started in 2019. We immediately contacted a few ECP supported library development projects, only to find out that none of them were usable for our software development due to the specific design of software interface. In addition, some of them told us that they have high priority customers, which does not include DFT code development, so that we will not be able to use these libraries in near future.

Furthermore, I learned from heFFTe development team that they were fully aware that the ratio between GPU computing power and interconnect bandwidth of Frontier made it impossible to take full advantage of GPU computing power (only 0.15% of peak flops performance). However, they were not aware at all about the larger implication of this limitation: DFT code and classical MD code with coulomb interaction will not be able to take full advantage of GPU performance.

It seems to me that the organization became too large so that effective communication became impossible.

I'd suggest that, instead of having one gigantic HPC development project, develop multiple medium sized projects with slightly different emphasis (ex. hardware architectures focusing on high arithmetic intensity software vs low arithmetic intensity software).

9. Assessment and criteria for success for the stewardship effort

This is an extremely challenging subject. We need continuous debate on this subject involving all the stakeholders.

From the viewpoint of software developer, ultimately it needs to provide environment that accelerate use of software, which requires, easy to use and easy to program (users are scientists not expert programmer). I will argue that if we are to have such environment, scientists will be able to take full advantage of what HPC system will offer, which in turn will increase probability of having scientific breakthrough in future.

At last, the words of wisdom of Warren Buffet, who is arguably one of the most successful individuals in private sector, answering to the question what is the secret of his success?

Invest only long term. Invest on companies that love what they are doing over gaining a lot of money. In other words, he will not invest on people whose priority is money making over content of their business.

I believe the notion of "private sector mange research organizations such as DOE lab better than public sector such as a university" has been completely misinterpreted and misused.

We must ask ourselves if we are supporting researchers who cares more about research than getting more funding.