Co-designing numerical algorithms and software with emerging computer architectures

***Rong Tian**

Institute of Computing Technology, Chinese Academy of Sciences,
Kexueyuan Nanlu 6, Haidian, Beijing 100190, China

*Corresponding author: rongtian@ncic.ac.cn

In the HPC community, domain application experts commonly acknowledge one truth that even a finite element code delivers a nearly linear speedup, the performance of a computer that the code can utilize is in a rather small percent. On the other hand, the efficiency obtained for the linear algebraic package (LINPACK) benchmark test is more than 80% in general. A huge gap exists between the real performance and the "nominal"/benchmark performance.

The memory intensive S&E application does not simply benefit from the increase in the floating point performance. Simply increasing the number of cores reversely tends to widen the existing performance gap.

The core algorithms of the many S&E applications largely feature memory intensive. Following the current trend of hardware change, if no change is made to key numerical algorithms, the waste of the floating point capability seems unavoidable. Revisits to the key numerical algorithm seem critically necessary to closely catch up with the hardware change.

In this talk, we will bring about some co-design thinking with several examples of extreme-scale simulation, with a goal at scalability, fault tolerance, and hardware efficiency when developing new numerical algorithms and codes.

Keywords: petascale computing , finite element/meshfree methods, scalability, fault tolerance, hardware efficiency.