Reducing the memory bandwidth usage of Krylov methods with polynomial preconditioning.

Bram Reps[1] Simplice Donfack[2] Olaf Schenk[2]
Wim Vanroose[1]

[1] University of Antwerp, Belgium,
[2] Università della Svizzera italiana, Switzerland

Krylov subspace methods perform poorly on many HPC systems. They attain only a fraction of the available peak performance due to the limited available memory bandwidth. Indeed, Krylov methods have a low arithmetic intensity since they do only a few flops with every double that is read from the main memory. So they rapidly stall because they cannot read fast enough the input from the memory.

There is an emerging set of tools to reduce the bandwidth usage of stencil code and take advantage of SIMD vectorization on modern hardware. These tools, such as Pochoir, PATUS, MODESTO, PLUTO and many others can significantly increase the performance of the user's stencil code, by applying temporal and spatial blocking, where possible.

In this talk we show how polynomial preconditioning can be exploited to take advantage of this available set of automatic tools from an active community. The effective bandwidth usage of Krylov methods is significantly reduced. And so, in addition to the well-known positive effect on the performance on distributed memory machines, preconditioning can lead to better scalability in the number of threads on shared memory machines.