Erik G. Boman
Parallel and Asynchronous Preconditioners for Manycore Architectures

Center for Computing Research
Sandia National Labs
Albuquerque
NM 87185
egboman@sandia.gov
Siva Rajamanickam

With the rise of manycore architectures, such as Xeon Phi and GPU, thread scalability on the node has become an important issue for solvers. Traditional preconditioners such as Gauss-Seidel and incomplete factorizations have limited parallelism. We present recent work to evaluate various types of multithreaded preconditioners across several manycore architectures. Our target are simple methods that may be used as smoothers in multigrid or inexact subdomain solvers.

First we examine Gauss-Seidel, a popular smoother. Convergence depends on the ordering. Orderings that work well in serial have limited parallelism. Graph coloring gives an ordering that provides a good compromise between convergence rate and parallelism. We use graph coloring and compare multi-colored Gauss-Seidel to asynchronous Gauss-Seidel, where the execution order is unknown.

Second, we explore row projection methods such as the Kaczmarz and Cimmino methods. These are also known as ART and SIRT, respectively, in image reconstruction. They are simple to implement on multithreaded architectures and avoid sparse triangular solves altogether. They are also more robust than Jacobi or Gauss-Seidel, especially on non-symmetric problems. We note that block versions of row projections exist and are known to have better convergence properties. These can be viewed as hybrid direct-iterative methods.

Our implementations are based on the Kokkos package for performance-portable manycore programming. This allows us to run the same code on CPU, Xeon Phi, and GPU, just with different compilation. We present preliminary results for model problems and matrices from the UF sparse matrix collection.