Parallel Preconditioning Method for Iterative Linear Solvers on Multicore/Manycore Clusters

Kengo Nakajima*

* Information Technology Center, The University of Tokyo
2-11-16 Yayoi, Bunkyo-ku, Tokyo 1138658, Japan
e-mail: nakajima@cc.u-tokyo.ac.jp, web page: http://nkl.cc.u-tokyo.ac.jp

ABSTRACT

Solving linear equations is the most important and expensive process in various types of applications of scientific computing. Recently, Krylov-type iterative solvers are widely used on massively parallel computers. It is widely known that preconditioning is essential for robust and efficient convergence of iterative solvers. This presentation overviews recent works in this area by the speaker, focusing on examples on multicore/manycore clusters.

In the present work, optimization of ICCG (Conjugate Gradient Method with Incomplete Cholesky preconditioning) solver has been conducted on a single node of Intel Xeon Phi (Knights Landing (KNL) and Knights Corner (KNC)) using OpenMP. Target application (Poisson3D-OMP) solves SPD matrices derived from 3D FVM with 7-point stencil for Poisson's equations. ICCG is used for solving linear equations with sparse matrices in a wide range of applications of science and engineering. Because IC/ILU (incomplete LU factorization) preconditioning consists of processes with data dependency, reordering is needed for achievement of parallel computing using OpenMP on muticore/manycore processors. Poisson3D-OMP adopts CM-RCM reordering method, which provides efficient and robust convergence of iterative solvers with IC/ILU preconditioning. ELL-type storage format for sparse matrices is one of the optimum methods for SpMV operations. In the present work, extended version of ELL storage format (ELL/Sliced ELL/ SELL-C-σ) is implemented to multithreaded ICCG solver on KNL and KNC. SELL-C-σ [1] is further extension of Sliced ELL for SIMD processors. C (=chunk size) corresponds to SIMD width, and "σ" (sorting scope) is specified according to the distribution of number of non-zero off-diagonal components at each row of the sparse matrix. Chunk size (C) of SELL-C-σ is set to 8, which corresponds to SIMD width (512 bit) of KNL and KNC for 64-bit operations. This work is the first case where the SELL-C-σ is applied to IC/ILU preconditioning method with data dependency. Proposed method with various types of numbering and outer-loop control has been evaluated. The "Flat/Quadrant" mode is applied to KNL, where only MCDRAM has been utilized. Programs with ELL-type matrix storage provide 2x performance of those with CRS format. Generally, KNL is three times as fast as KNC. This number corresponds to ratio of peak performance and CPU speed between KNL and KNC. Because SELL-C-σ consists of extra computations for padding compared to original ELL approach, SELL-C-σ is generally slower than the optimum ELL case on KNC. On KNL, effects of vectorization are significant, and SELL-C-σ is slightly faster than ELL. In the present work, number of operations for non-zero off-diagonal components at each is very few (less than 3), because it is FVM application with 7-point stencils. We can expect more significant effects of SELL-C-σ for FEM-type applications, where number of non-zero components at each row is larger. In the present work, we have 6 operations for each row, if number of color for CM-RCM is 2. This is the only case where SELL-C-σ is faster than ELL on KNC. Because this is the very preliminary study, further optimization of SELL-C-σ should be applied. Moreover, we may make contributions to SELL-C-σ capability for IC/ILU preconditioning in GHOST library under collaborations for ESSEX-II project with FAU/Erlangen [2].

REFERENCES

[1] M. Kreutzer, et al., A unified sparse matrix data format for efficient general sparse matrix-vector multiply on modern processors with wide SIMD units, SIAM Journal on Scientific Computing 36:5, C401-C423, 2014.

[2] https://blogs.fau.de/essex/