Multilevel Variable-Block Schur-Complement Based Preconditioning on Accelerators

B. Carpentieri^{1,a)}, M. Sosonkina² and J. Liao³

¹School of Science and Technology, Nottingham Trent University, Clifton Campus, Nottingham, NG11 8NS, UK

²Modeling, Simulation & Visualization Engineering, Old Dominion University, 1318 Engr. and Comp. Sci. Bldg, Norfolk, VA 23529

³Johann Bernoulli Institute for Mathematics and Computer Science, University of Groningen, Nijenborgh, 9, 9747 AG, Groningen, The Netherlands

^{a)}Corresponding author, Presenting author: bcarpentieri@gmail.com

ABSTRACT

Sparse matrices arising from the solution of systems of partial differential equations may often exhibit a fine-grained block structure when several unknown physical quantities are associated with the same grid point. Examples arise in the finite element discretization of the Navier-Stokes equations for turbulent flows analysis, in the boundary element discretization of the Helmholtz equations in electromagnetic scattering applications and in the study of the mechanisms underlying cardiac electrical dynamics modelled by the Bidomain equations, to name a few. If variables assigned to each grid point are numbered consecutively, the matrix arising from the discretization may have a block structure, with the presence of small and usually dense nonzero blocks in the pattern, due to the mutual coupling of the variables at the same node. We refer to this form of blocking as a perfect block ordering. On the other hand, when the matrix is general unstructured, it is sometimes possible to compute imperfect block orderings by treating some zero entries of the matrix as nonzero elements, with a little sacrifice of memory, and grouping together sets of rows and columns having a similar nonzero structure.

In all these situations, it is natural to consider block forms of multi-elimination methods that can exploit any available block structure, either perfect or imperfect. A clear advantage is to store the matrix as a collection of blocks using the variable block compressed sparse row (VBCSR) format, saving column indices and pointers for the block entries. On indefinite problems, computing with blocks instead of single elements enables a better control of pivot breakdowns, near singularities, and other possible sources of numerical instabilities. Block Incomplete LU (ILU) solvers may be used instead of pointwise ILU methods as local solvers. A full block implementation may be unravelled based on higher level optimized Basic Linear Algebra Subroutines (BLAS), having better flops to memory ratios on modern cache-based computer architectures. Finally, grouping variables in clusters, the Schur complement is smaller and the last reduced system may be better conditioned and easier to solve.

Our recently developed variable block algebraic recursive multilevel solver (VBARMS) incorporates compression techniques during the factorization to detect fine-grained dense structures in the linear system automatically, without any user's knowledge of the underlying problem, and exploits them to improve the overall robustness and throughput of the multilevel iterative solver [1]. Exposing dense matrix blocks during the factorization may lead to more efficient and numerically stable parallel solvers [2].

In this talk we present a performance analysis of VBARMS against the parallel implementation of the ARMS method provided in the pARMS package [3]. We illustrate its remarkable efficiency for solving block structured linear systems arising from an implicit Newton-Krylov formulation of the Reynolds Averaged Navier Stokes equations in turbulent incompressible flow analysis past a three-dimensional wing (the mesh is depicted in Fig. 1), in combination with conventional parallel global solvers such as in particular the Restricted Additive Schwarz preconditioner. We show how the performance of VBARMS improves on hardware accelerators (Intel Xeon Phi, graphics processing units) by revealing a high-degree of the parallelism. Finally, we report on ongoing experiments that use a different compression for the coefficient matrix and for the Schur Complement matrix, to improve the robustness and to decrease the factorization costs of the VBARMS method.



Figure 1. 3D steady incompressible turbulent flow past the DPW3 Wing: the mesh.

Keywords: Multilevel iterative solvers, Incomplete factorization methods, Block algorithms, Compression techniques, Hardware accelerators.

References

- [1] Carpentieri B., Liao, J., Sosonkina, M. (2014) VBARMS: A variable block algebraic recursive multilevel solver for sparse linear systems. *Journal of Computational and Applied Mathematics* **259**, **Part A**, 164–173.
- [2] Carpentieri, B., Liao, J., Sosonkina, M., Bonfiglioli, A. and Baars, S. (2016) Using the VBARMS method in parallel computing. *Parallel Computing*, In Press, doi:10.1016/j.parco.2016.01.005.
- [3] Li, Z., Saad, Y. and Sosonkina, M. (2003) pARMS: a parallel version of the algebraic recursive multilevel solver. *Numerical Linear Algebra with Applications* **10**, 485–509.