Performance Improvement of Block Red-Black MILU(0) Preconditioner with Relaxation on GPU

Akemi Shioya
The University of Electro-Communications
Tokyo, Japan
akemi.shioya@uec.ac.jp

Yusaku Yamamoto
The University of Electro-Communications
Tokyo, Japan
yusaku.yamamoto@uec.ac.jp

1 INTRODUCTION

The block red-black ordering [1] is often used to parallelize the incomplete LU preprocessing. In this ordering strategy, the computational grid is divided into blocks and then each block is colored in the same way as the red-block ordering. The advantage of this method is that the synchronization cost is smaller because of the small number of colors, and the convergence is better than the nodal red-black method because the nodes are grouped into blocks. The MILU(0) factorization compensates for the fill-in elements dropped during the ILU factorization process, thereby reducing the influence of dropping fill-ins is reduced. However, zero pivots occur when the MILU(0) factorization is combined with the red-black ordering. This situation can be avoided by using the relaxation coefficient α for compensation in the MILU(0) factorization [2]. However, this method has only been evaluated on CPUs. In this poster, we parallelize a Krylov subspace solver with this preconditioner and evaluate its performance on a modern GPU with high computing performance.

2 IMPLEMENTATION

We parallelized a BiCGSTAB solver combined with block red-black ordering and relaxed MILU(0) factorization for GPUs using OpenACC and optimized with the following:

- **Realizing coalesced memory access in the forward/backward substitution**: In the natural storage format, the elements of each block are stored separately, so the efficiency of data access is deteriorated. Therefore, all the matrices and vectors used in the block parallelization are stored in such a way that data accessed by adjacent threads are adjacent in the array.
- **Exploiting the single precision performance of the GPU**: There is a difference of several times between single-precision and double-precision computation speed of the GPU. The relationship between the computed solution and the residual is maintained for the right preconditioning even if the substitution calculation contains errors [3]. Therefore, the calculation is performed by the single precision operation in substitution.

3 PERFORMANCE TEST

We use the Poisson equation arising from the Particle-In-Cell (PIC) plasma simulation as a test problem. The environment is a GDEP MAS-i7WF workstation equipped with a Quadro GP100. We used the CUDA 9.1 and PGI Fortran 17.4 compilers with -O3 optimization.

To speed up the calculation time, the number of blocks should be (1) in multiples of 32 and (2) 32×6 or more per SM. Fig. 1 shows a comparison of the computation time between the naive implementation and the improved implementation. These implementation techniques and optimizations have greatly improved the computation speed compared to naive implementations. In addition, the proposed method achieved shorter solution times compared to the BiCGSTAB implementations with ILU(0) or AMG preconditioning included in the GPU libraries cuSPARSE, MAGMA, and ViennaCL.

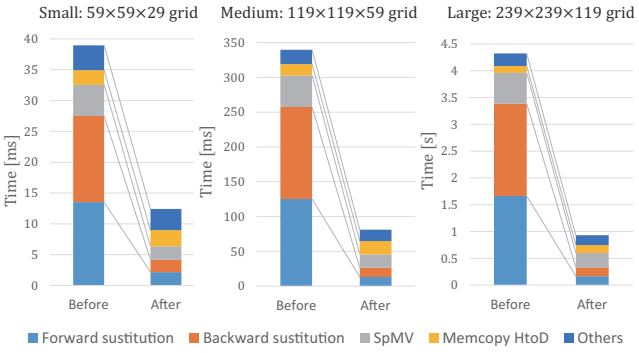

Figure 1: Profiling results of of BiCGSTAB iteration.

4 CONCLUSION

We parallelized the solution of linear simultaneous equations with the coefficient matrix ordered by the block red-black ordering and preconditioned by the relaxed MILU(0) and optimized the parameters for parallel computation on GPUs. The optimization significantly improved the execution speed compared to the naive implementation, and the solution was obtained faster than by the BiCGSTAB routines in existing libraries. In the future, we will further examine the block division conditions and evaluate them in other environments.

REFERENCES

[1] Iwashita T. and Shimasaki M. 2003. Block red-black ordering: A new ordering strategy for parallelization of ICCG method. *International Journal of Parallel Programming* 31, 1 (2003), 55–75.
[2] Shioya A. and Yamamoto Y. 2018. The danger of combining block red-black ordering with modified incomplete factorizations and its remedy by perturbation or relaxation. *Japan Journal of Industrial and Applied Mathematics* 35, 1 (2018), 195–216.
[3] Tadano H. and Sakurai T. 2007. On single precision preconditioners for Krylov subspace iterative methods. In *In International Conference on Large-Scale Scientific Computing*. Springer, Berlin, Heidelberg, 721–728.