

MGRIT preconditioned Krylov subspace method

Ryo Yoda
Kogakuin University
Shinjuku-ku, Tokyo
em18017@ns.kogakuin.ac.jp

Akihiro Fujii
Kogakuin University
Shinjuku-ku, Tokyo
fujii@cc.kogakuin.ac.jp

Teruo Tanaka
Kogakuin University
Shinjuku-ku, Tokyo
teru@cc.kogakuin.ac.jp

ABSTRACT

MGRIT re-discretize the problem with larger time-step width at the coarse-levels, which often cause unstable convergence. We propose a Krylov subspace method with MGRIT preconditioning as a more stable solver. For unstable problems, MGRIT preconditioned Krylov subspace method performed better than MGRIT in terms of the number of iterations. The contributions of the paper are organized as follows. We showed the matrix form of MGRIT operations, and the improvement of eigenvalue or singular-value distribution. We exemplified MGRIT with Krylov subspace method reaching convergence faster than MGRIT.

CCS CONCEPTS

• **Mathematics of computing** → *Solvers*;

KEYWORDS

parallel-in-time, linear solver, MGRIT, Krylov-subspace method, preconditioning

ACM Reference Format:

Ryo Yoda, Akihiro Fujii, and Teruo Tanaka. 2018. MGRIT preconditioned Krylov subspace method. In *Proceedings of Super Computing 2018 (SC'18)*. ACM, New York, NY, USA, 3 pages.

1 INTRODUCTION

In massively parallel environment, Multigrid Reduction in Time (MGRIT) [1][2][5] has gained attentions as a scalable solver of the time evolution problem. Because this method construct a multilevel structure enlarging the time-step width, the coarser-level problem becomes unstable. Here, we propose a Krylov subspace method with MGRIT as preconditioning to enhance the stability of the solver. The proposed method inherits the scalability of MGRIT, and can be performed with the same computation complexity as MGRIT and with the additional memory cost of the restart vectors. Because the time evolution matrix is nonsynmmetric, we used Generalized Minimum RESidual (GMRES) [8][10][11][12] as the Krylov subspace method in this study.

The paper is organized as follows. Section 2 describes MGRIT algorithm overview. Section 3 describes the MGRIT preconditioning matrix, and its application. Section 4 describes experiments to confirm eigenvalue distribution and singular-value distribution, and number of iterations.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

SC'18, November 2018, Dallas, Texas USA

© 2018 Copyright held by the owner/author(s).

2 MGRIT

When the differential equation is discretized by the implicit method, the relation between unknowns of time-steps are described as in Eq. (1). The matrix A constructed by a simulation length T from Eq. (1), is called a time evolution matrix.

$$\begin{cases} g^0 &= u^0 \\ \Phi u^{i+1} &= u^i + g^{i+1} \quad (i = 0, 1, \dots, T-1) \end{cases} \quad (1)$$

$$Au = \begin{bmatrix} I & & & \\ -I & \Phi & & \\ & \ddots & \ddots & \\ & & -I & \Phi \end{bmatrix} \begin{bmatrix} u^0 \\ u^1 \\ \vdots \\ u^T \end{bmatrix} = \begin{bmatrix} g^0 \\ g^1 \\ \vdots \\ g^T \end{bmatrix} = g \quad (2)$$

Let m be the coarsening ratio. For the above problem $Au = g$, MGRIT [2] constructs a coarse-level Δ with a wider time-step width equal to $m \times \Delta t$. It recursively constructs coarser-levels and solves the problem using the coarser-level solutions.

3 MGRIT PRECONDITIONED GMRES

3.1 Preconditioning matrix

When MGRIT is employed as a preconditioner, it is necessary to derive a stationary iterative matrix. We described all MGRIT operations in the matrix form.

As shown below, MGRIT divides the time-steps into C-points (red) and F-points (black). We derived the matrix form of C-relaxation, which updates the C-points, and that of F-relaxation, which updates the F-points. Each relaxation can be written, by defining operations I_c and I_f to extract only the C/F-points.

- C-relaxation: Update C-points from F-points in the previous step.

$$- M_c u = I_f u + I_c g$$

$$- u = M_c^{-1} I_f u + M_c^{-1} I_c g$$

$$\begin{pmatrix} I & & & \\ -I & & & \\ & I & & \\ & & I & \\ & -I & \Phi & \\ & & & I \end{pmatrix} \begin{pmatrix} u_0 \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{pmatrix} = \begin{pmatrix} I & & & \\ & I & & \\ & & I & \\ & & & I \end{pmatrix} \begin{pmatrix} u_0 \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{pmatrix} + \begin{pmatrix} I & & & \\ & I & & \\ & & I & \\ & & & I \end{pmatrix} \begin{pmatrix} g_0 \\ g_1 \\ g_2 \\ g_3 \\ g_4 \\ g_5 \end{pmatrix}$$

- F-relaxation: Update F-points sequentially from C-points.

$$- M_f u = I_c u + I_f g$$

$$- u = M_f^{-1} I_c u + M_f^{-1} I_f g$$

$$\begin{pmatrix} I & & & \\ -I & \Phi & & \\ & -I & \Phi & \\ & & I & \\ & & & -I & \Phi & \\ & & & & -I & \Phi \end{pmatrix} \begin{pmatrix} u_0 \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{pmatrix} = \begin{pmatrix} I & & & \\ & I & & \\ & & I & \\ & & & I \end{pmatrix} \begin{pmatrix} u_0 \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{pmatrix} + \begin{pmatrix} I & & & \\ & I & & \\ & & I & \\ & & & I \end{pmatrix} \begin{pmatrix} g_0 \\ g_1 \\ g_2 \\ g_3 \\ g_4 \\ g_5 \end{pmatrix}$$

The restriction matrix R sets I at the C-points and reduces the dimension to a coarse-level. The interpolation matrix P is R^T , which expands the dimension to a fine-level. When the pre-smoothing and post-smoothing are expressed as $\mathbf{u} = S_1\mathbf{u} + T_1\mathbf{g}$ and $\mathbf{u} = S_2\mathbf{u} + T_2\mathbf{g}$ respectively. The preconditioning matrix M at two levels becomes $(S_2T_1 + T_2 + S_2PA_{\Delta}^{-1}R(I - AT_1))$.

3.2 Preconditioned GMRES

GMRES allows application of the right preconditioning [11][12]. The pseudocode is represented as follows. The highlight are the preconditioning part. MGRIT preconditioned GMRES (MGRIT-GMRES) uses the MGRIT iteration matrix M as a preconditioning in the pseudocode.

```

 $\mathbf{v}_0 = \mathbf{r}_0 / \|\mathbf{r}_0\|_2$ 
for j = 0, k # k is a restart number.
   $\mathbf{z}_j = M^{-1}\mathbf{v}_j$ 
   $\mathbf{v}_{j+1} = A\mathbf{z}_j$ 
  for i = 0, j
     $h[i][j] = (\mathbf{v}_{j+1}, \mathbf{v}_i)$ 
     $\mathbf{v}_{j+1} = \mathbf{v}_{j+1} - h[i][j]*\mathbf{v}_{j+1}$ 
  end
   $h[j+1][j] = \|\mathbf{v}_{j+1}\|_2$ 
   $\mathbf{v}_{j+1} = \mathbf{v}_{j+1}/h[j+1][j]$ 
end
Compute  $\mathbf{x}_k = \mathbf{x}_0 + Z_k\mathbf{y}_k$  #  $Z_k = [\mathbf{z}_0, \mathbf{z}_1, \dots, \mathbf{z}_k]$ 

```

Since it keeps orthogonal basis vectors for k iterations, MGRIT-GMRES requires more memory space than MGRIT. As for calculation cost, the most cost-intensive process in GMRES is SpMV. This has the same computation complexity as that of $A\mathbf{u}$, which is the operation required for MGRIT residual calculation.

4 NUMERICAL EXPERIMENTS

We used Oakforest-PACS supercomputer (JCAHPC) [17]. As an example of a time evolution problem, we considered the two dimensional linear diffusion-problem with a 5-point difference. The problem discretized by the implicit method was regarded as a stable problem, while that discretized by the explicit method with large Δt was regarded as an unstable problem.

The iteration number of the Krylov subspace method depends on the eigenvalue distribution A . Therefore, first, we checked whether MGRIT preconditioning improved the eigenvalues distribution of the time evolution matrix of A . In the case of the explicit method, because all the diagonal elements of A are 1, all the eigenvalues are 1. Therefore, we see the improvement of the singular value distribution of the problem.

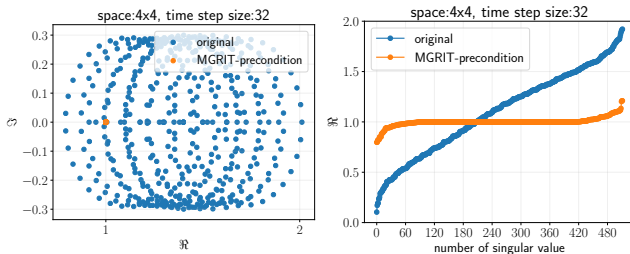


Fig. 1. Implicit: eigen value

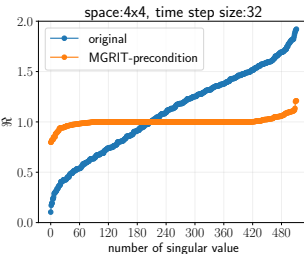


Fig. 2. Explicit: singular value

As shown in Fig. 1, the eigenvalue distribution improved considerably upon MGRIT preconditioning. Similarly, as shown in Fig. 2, the singular value distribution improved considerably upon MGRIT preconditioning.

We implemented GMRES, MGRIT and MGRIT-GMRES (the proposed method) in both time-direction and space-direction parallelization with Flat MPI. The data distribution is shown in Fig. 3 and Fig. 4. In this case, there are 16 processes (4 time direction $\times$ 4 space direction) and 4 colors (red, blue, green, and orange) in Fig. 3, and Fig. 4 correspond to 4 MPI communicators in space direction.

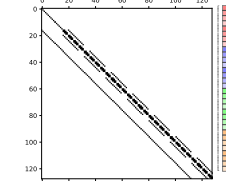


Fig. 3. the relation to A and each process

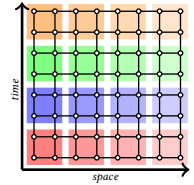


Fig. 4. the status in space and time direction

We measured the number of iterations when the problem was discretized by the implicit or explicit method, for GMRES, MGRIT, and MGRIT-GMRES.

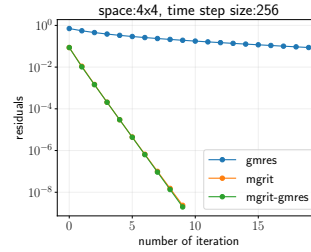


Fig. 5. Implicit: the iteration num.

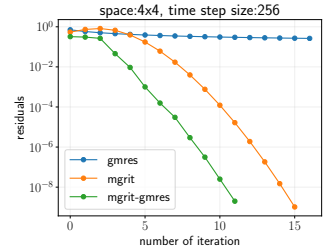


Fig. 6. Explicit: the iteration num.

As shown in Fig. 5, the residual histories of MGRIT and MGRIT-GMRES agreed approximately. Because the coarse-level correction with implicit method for this problem was correct enough. As shown in Fig. 6, the MGRIT residual norm increased for a few initial iterations. This indicates that a coarse-grid correction degrades because of larger time-step width. In this case, MGRIT-GMRES monotonically decreased the residuals and reduced the number of iteration.

5 CONCLUSION

In massively parallel environments, MGRIT attracts attentions as a scalable solver of time evolution problems. When constructing a multi-level in the time direction, coarsening of the time-step width poses a problem. We proposed the MGRIT preconditioned Krylov subspace method, and we evaluated it through eigenvalue distribution and the iteration number for convergence. MGRIT preconditioned GMRES reduces the number of iteration, with almost same computation complexity and additional memory amount for restart vectors. Krylov subspace method enhances the stability of MGRIT method. MGRIT preconditioning doesn't depend on Krylov subspace method. We will combine the another Krylov subspace method, and evaluate the more large time-step width or more unstable problems, discretized from another governing equation.

ACKNOWLEDGMENTS

This work was partially supported by JSPS KAKENHI Grant Numbers JP15K15998 and "Joint Usage/Research Center for Interdisciplinary Large-scale Information Infrastructures" and "High Performance Computing Infrastructure" in Japan (Project ID: jh180022-NAHI).

REFERENCES

- [1] R. D. Falgout, S. Friedhoff, Tz. V. Kolev, S. P. MacLachlan, and J. B. Schroder, *Parallel Time Integration with Multigrid*, SIAM J. Sci. Comput, 36, pp.C635-C661, (2014)
- [2] R. D. Falgout, A. Katz, Tz. V. Kolev, J. B. Schroder, A. M. Wissink, U. M. Yan: *Parallel Time Integration with Multigrid Reduction for a Compressible Fluid Dynamics Application*, Journal of Computational Physics. (2014)
- [3] H. Gahvari, V. A. Dobrev, R. D. Falgout, Tz. V. Kolev, J. B. Schroder, M. Schulz and U. M. Yang, *A Performance Model for Allocating the Parallelism in a Multigrid-in-Time Solver*, The 7th International Workshop on Performance Modeling, Benchmarking and Simulation of High Performance Computer Systems (PMBS16), Supercomputing 16 (2016)
- [4] R. D. Falgout, S. Friedhoff, Tz. V. Kolev, S. P. MacLachlan, J. B. Schroder and S. Vandewalle, *Multigrid Methods with Space-Time Concurrency*, Computing and Visualization in Science, Springer, (2017)
- [5] V. Dobrev, Tz. Kolev, N. A. Petersson, and J. B. Schroder, *Two-level convergence theory for Multigrid Reduction in Time (MGRIT)*, SIAM J. Sci. Comput., 39, pp. 501-527, (2017)
- [6] M. J. Gander and S. Vandewalle, *Analysis of the parareal time-parallel time-integration method*, SIAM J. Sci. Comput., 29, pp. 556-578, (2007)
- [7] M. J. Gander, *50 years of time parallel time integration*, in Multiple Shooting and Time Domain Decomposition Methods, Springer, Cham, pp. 69-113, (2015)
- [8] Y. Saad, and Martin H. Schultz. *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Comput., 7, pp. 856-869, (1986)
- [9] Y. Saad, *Krylov Subspace Methods on Supercomputers*, SIAM J. Sci. Comput, 10, pp.1200-1232, (1989)
- [10] H. A. van der Vorst and C. Vuik, *The superlinear convergence behaviour of GMRES*, SIAM J. Comput. Appl., 48, pp. 327-341, (1993)
- [11] J. Baglama, D. Calvetti, G. H. Golub, and L. Reiche, *Adaptively preconditioned GMRES algorithms*, SIAM J. Sci. Comput, 20, pp. 243-279, (1998)
- [12] Y. Saad, *Iterative Methods for Sparse Linear Systems*, SIAM, second edition, (2003)
- [13] W. L. Briggs, V. E. Henson, and S. F. McCormick, *A Multigrid Tutorial Second Edition*, SIAM, (2000)
- [14] I. Yavneh, *Why Multigrid Methods Are So Efficient*, Computing in Science and Engineering, 8, pp. 12-22, (2006)
- [15] O. Tatebe, *The Multigrid Preconditioned Conjugate Gradient Method*, 6th Copper Mountain Conference on Multigrid Methods, pp. 621-634, (1993)
- [16] K. Nakajima, *Parallel Iterative Linear Solvers with Preconditioning for Large Scale*, (2002)
- [17] <https://www.cc.u-tokyo.ac.jp/index-e.html> (accessed on July 31, 2018)