Commun. Comput. Phys.
doi: 10.4208/cicp.OA-2023-0072

Vol. **36**, No. 1, pp. 200-220
July 2024

AutoAMG(θ): An Auto-tuned AMG Method Based on Deep Learning for Strong Threshold

Haifeng Zou[1,2], Xiaowen Xu[3,*], Chen-Song Zhang[4] and Zeyao Mo[3]

[1] *Graduate School of China Academy of Engineering Physics, China Academy of Engineering Physics, P.R. China.*
[2] *Shenzhen International Center for Industrial and Applied Mathematics, Shenzhen Research Institute of Big Data, P.R. China.*
[3] *Laboratory of Computational Physics, Institute of Applied Physics and Computational Mathematics, P.R. China.*
[4] *Academy of Mathematics and Systems Science, Chinese Academy of Sciences, P.R. China.*

Received 8 March 2023; Accepted (in revised version) 30 November 2023

Abstract. Algebraic Multigrid (AMG) is one of the most widely used iterative algorithms for solving large sparse linear equations $Ax=b$. In AMG, the coarse grid is a key component that affects the efficiency of the algorithm, the construction of which relies on the strong threshold parameter θ. This parameter is generally chosen empirically, with a default value in many current AMG solvers of 0.25 for 2D problems and 0.5 for 3D problems. However, for many practical problems, the quality of the coarse grid and the efficiency of the AMG algorithm are sensitive to θ; the default value is rarely optimal, and sometimes is far from it. Therefore, how to choose a better θ is an important question. In this paper, we propose a deep learning based auto-tuning method, AutoAMG(θ) for multiscale sparse linear equations, which are common in practical problems. The method uses Graph Neural Network (GNN) to extract matrix features, and a Multilayer Perceptron (MLP) to build the mapping between matrix features and the optimal θ, which can adaptively predict θ values for different matrices. Numerical experiments show that AutoAMG(θ) can achieve significant speedup compared to the default θ value.

AMS subject classifications: 65F08, 65F10, 65N55, 68T05

Key words: AMG, strong threshold, graph neural network, auto-tuning, multiscale matrix.

1 Introduction

Solving sparse linear equations $Ax=b$ is ubiquitous in numerical simulations, and is a major bottleneck affecting computational efficiency. Owing to its good generality and

*Corresponding author. *Email addresses:* zou_haifeng@foxmail.com (H. Zou), xwxu@iapcm.ac.cn (X. Xu), zhangcs@lsec.cc.ac.cn (C.-S. Zhang), zeyao_mo@iapcm.ac.cn (Z. Mo)

optimal computational complexity, the AMG algorithm [1–3] is one of the most widely used algorithms for large-scale sparse linear equations, which uses only information from the matrix to construct components, including coarsening, interpolation, and restriction operators. During the coarsening procedure, a subset of points from the adjacency matrix A is selected as points in the coarse grid, which is the basis for constructing a coarse grid matrix A_c. Different coarsening strategies will result in different coarse matrices A_c. In the classical AMG algorithm, points in the subset are selected based on the strong threshold θ and the strength of the connectivity between points, which is calculated by the value of the matrix entries. Hence the value of θ directly affects the grid coarsening results, and is a key factor affecting the algorithm's efficiency.

In the classical AMG algorithm, most coarsening algorithms are based on heuristic strategies for coarse grid construction. A basic principle is to perform coarsening along the direction of strong connectivity to accommodate the property that algebraic errors are smoothed or relaxed along the same direction. If the strong threshold θ is large, then the number of points in the corresponding coarse grid is large, which means the AMG algorithm has high complexity. If θ is small, although the number of points in the coarse grid is smaller, the residuals may decrease more slowly, requiring more iterations to converge. Since there is no strict theoretical guarantee on the size of the optimal coarse grid, the current value of θ can only be chosen empirically. For example, in the HYPRE AMG solver [4], depending on the physical dimension of the sparse matrix, θ equals 0.25 for 2D problems and 0.5 for 3D problems. However Vakili [5] and Nikola [6] utilize the incompressible Navier Stokes equation and linear poroelasticity equation, respectively, as the test cases, both of which show the increase of θ along with the monotone decrease of time. Here, we take the diffusion problem as the example, and find that the number of iterations changes irregularly with the increase of θ. If the diffusion coefficients are isotropic, the default values of θ can achieve the desired convergence rate. If the diffusion coefficients are anisotropic, which means there are significant differences in the strength of connectivity between points, then the default values of θ maybe far from the optimal. Notably, small changes in θ may have a large impact on the construction of the coarse grid, thus affecting the convergence rate and efficiency of AMG. In particular, we focus on the so-called multiscale sparse matrices [7]. In some typical test cases, the number of iterations of the default θ is 10 times larger than the minimum number of iterations obtained by grid search (see Section 2.3 for detail).

The above problem can be summarized as follows: how to choose an appropriate θ for any given sparse matrix. Considering that the properties of the input matrix may vary dynamically, the automatic selection of a suitable θ for different linear systems is a crucial and challenging task, since there is no theoretical guarantee yet. Machine learning and deep learning algorithms provide a feasible approach. Paola F [8] used a Convolutional Neural Network (CNN) to extract matrix features, and built a regression model with those features. The inputs of the regression model are matrix features, strong threshold θ, and $-\log_2 h$ (h is the edge length in the mesh), and the output y is an approximated convergence factor. After training, the regression model is used to optimize θ. There are