Comparison of Geometric and Algebraic Multigrid Methods in Edge-Based Finite-Element Analysis

K. Watanabe, H. Igarashi, and T. Honma, *Member, IEEE*

Division of Systems Science and Informatics, Graduate School of Information Science and Technology, Hokkaido University, Sapporo 060-0814, Japan

This paper discusses the comparison between the geometric multigrid (GMG) method and the algebraic multigrid (AMG) method in edge-based finite-element (FE) analysis. The GMG method requires the hierarchical meshes. On the other hand, the AMG method requires the only a single mesh information. The system matrices of the coarse grids are generated using algebraic operation in AMG. The numerical results show that both multigrid methods are faster than the conventional solvers in large-scale analysis. Although multigrid methods require the setup procedures, the calculation time of these procedures is comparatively short and increase linearly with the number of unknowns.

Index Terms—**Algebraic multigrid (AMG), edge element, finite-element method (FEM), geometric multigrid (GMG).**

I. Introduction

THE MULTIGRID method has been applied to electromagnetic field problems so far, to show that it can significantly reduce computational time in comparison with conventional linear solvers such as incomplete Cholesky conjugate gradient (ICCG). The geometric multigrid (GMG) method requires the hierarchical meshes. The nested GMG method [1], in which the finer meshes are automatically obtained by dividing each coarse element into several finer elements, is used in our study. This lightens the load of mesh generators. On the other hand, the algebraic multigrid (AMG) method requires the only a single mesh information. The system matrices on the coarse meshes are generated using algebraic operations in AMG. We compare the computation time of the two methods in a finite-element (FE) analysis to show the effectiveness for large-scale analysis. In this paper, we evaluate these setup process of multigrid in large-scale FE analysis.

II. Formulation

A. Magnetostatic Problem

Let us consider magnetostatic field governed by

$$\nabla \times \nu \nabla \times \boldsymbol{A} = \boldsymbol{J}_0 \tag{1}$$

$$\nabla \cdot \boldsymbol{J}_0 = 0 \tag{2}$$

where ν is the magnetic reluctivity, $\boldsymbol{A}$ is the vector potential, and $\boldsymbol{J}_0$ is the current density. The current vector potential

$$\nabla \times \boldsymbol{T} = \boldsymbol{J}_0 \tag{3}$$

is introduced for satisfaction of (2). Equation (1) now leads to

$$\nabla \times \nu \nabla \times \boldsymbol{A} = \nabla \times \boldsymbol{T}. \tag{4}$$

FE discretization of (4) results in the system of linear equations

$$[\mathrm{K}]\{x\} = \{b\} \tag{5}$$

where $[\mathrm{K}]$ is a positively semidefinite matrix which is the discrete counterpart of the operator in the left side of (4), $\{x\}$ and $\{b\}$ denote column vectors corresponding to $\boldsymbol{A}$ and $\boldsymbol{T}$, respectively.

B. GMG Method

It is known that the linear solvers such as Gauss–Seidel and CG methods tend to eliminate the high-frequency components of the residue in (5) more rapidly than the low-frequency components. The multigrid method is based on this property, that is, the high-frequency residual components are eliminated on a fine mesh by small numbers of iterations of the linear solver (smoother). The remained residual components are then projected onto a more coarse mesh, in which they now have high frequency that can again be eliminated by small numbers of the iterations. The multigrid method solves (5) successively performing these processes. This procedure is usually called the coarse grid correction. Although there are many variations in multigrid method, all these variations are based on the coarse grid correction. The procedure of the two-grid V-cycle method that is the simplest multigrid method is described in the following.

Step 1 (Smoothing): The smoothing operation is applied to the system equation

$$[\mathrm{K_f}]\{x\} = \{b\} \tag{6}$$

for the fine mesh to obtain approximate solution $\{\tilde{x}\}$, where $[\mathrm{K_f}]$ denotes the system matrix defined on the fine mesh. In this step, the high-frequency components in the solution error are eliminated.

Step 2: The residual vector $\{r_f\}$ corresponding to the approximate solution $\{\tilde{x}\}$, is calculated

$$\{r_f\} = \{b\} - [\mathrm{K_f}]\{\tilde{x}\}. \tag{7}$$

Digital Object Identifier 10.1109/TMAG.2005.846092

Step 3 (Restriction): The residual vector is projected onto a coarser mesh using the restriction matrix [R]

$$\{r_c\} = [\mathrm{R}]\{r_f\} \tag{8}$$

where the component of the matrix [R] is obtained by the following integration:

$$R_{ij} = \int_{E_j^f} \boldsymbol{w}_i^c \cdot d\boldsymbol{l} \tag{9}$$

where E_j^f denotes jth edge in fine mesh, $\boldsymbol{w}_i^c$ denotes the interpolation function corresponding to ith edge in the coarse mesh. In the nested multigrid method, in which the finer meshes are obtained by dividing each coarse element, we can construct the restriction matrix using geometric relationship between coarse edges and fine edges [1]. Fig. 1 shows an example of the construction of restriction matrix in tetrahedral edge elements. The finer meshes are obtained by dividing each coarse element into eight finer elements. The component of restriction matrix corresponding to a coarse element E_i^c shown in Fig. 1(a) is obtained

$$R_{ij} = \begin{cases} 0.5 : j = 1, 2 \\ 0.25 : j = 3, 7 \\ -0.25 : j = 4, 5, 6, 8, 9 \\ 0 : \text{otherwise} \end{cases}. \tag{10}$$

Step 4: The residual equation in coarse mesh is solved to obtain the error vector $\{e_c\}$ corresponding to the residual vector $\{r_c\}$

$$[\mathrm{K_c}]\{e_c\} = \{r_c\} \tag{11}$$

where $[\mathrm{K_c}]$ is the system matrix defined in the coarse mesh. It takes a short time to solve (11) because there is a small number of unknowns in (11). Equation (11) cannot be solved by the direct solver such as Gauss-elimination method, because $[\mathrm{K}_c]$ is singular. For this reason, the CG or ICCG method is used in this study.

Step 5 (Prolongation): The error vector is projected onto the fine mesh using the prolongation matrix [P]

$$\{e_f\} = [\mathrm{P}]\{e_c\} \tag{12}$$

where [P] is usually chosen as the transpose of [R].

Step 6: The solution $\{\tilde{x}\}$ obtained in *Step 1* is corrected using error vector $\{e_f\}$

$$\{x\}_{\mathrm{new}} = \{\tilde{x}\} + \{e_f\}. \tag{13}$$

Step 7 (Post-Smoothing): The smoothing operation is applied to the system equation again. This procedure is called post-smoothing. After post-smoothing, the convergence of the solution is tested. If convergence condition is not satisfied, we go back to *Step 2*.

C. AMG Method

The AMG method is an alternative multigrid method. GMG methods require the hierarchical meshes. On the other hand, the AMG method requires only single grid information. It means that the AMG method can easily replace classical solvers such as ICCG method in FE analysis. To accomplish the coarse grid correction in AMG, the system matrix $[\mathrm{K_c}]$ corresponding to the virtual coarse mesh is obtained by

$$[\mathrm{K_c}] = [\mathrm{R}][\mathrm{K_f}][\mathrm{R}]^{\mathrm{T}}. \tag{14}$$

The classical AMG approaches [2] cannot apply to the electromagnetic FE analysis with edge elements because of the singularity of system matrix. To overcome this difficulty, the appropriate restriction matrices and smoothers have to be defined. In our study, the virtual coarse mesh is constructed using the following algorithm [3] (see Fig. 2).

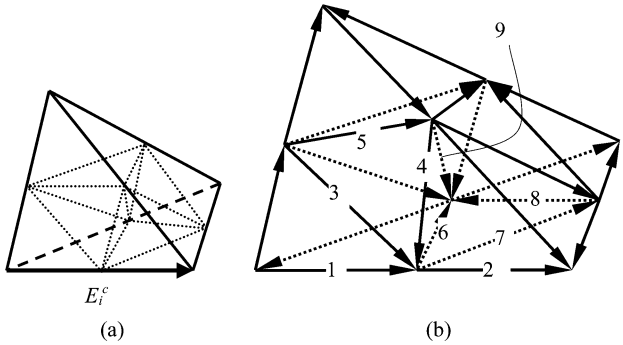

Fig. 1. Division to make the fine mesh in nested GMG. (a) Coarse element. (b) Fine elements.

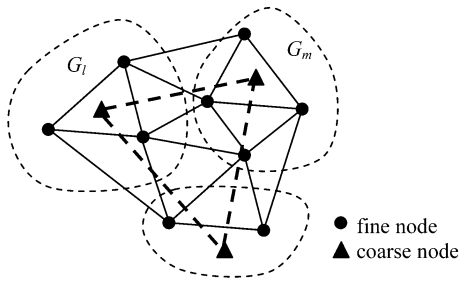

Fig. 2. Construction of virtual coarse grid.

D. Algorithm for Construction of Coarse Grid

1) The nodes on fine grid are divided into the node-groups G_i which denote the coarse node i.
2) The coarse edges are connected between coarse node k and l only if there is at lease one fine edge connecting the node-group G_k and G_l.

The component of restriction matrix R_{ki} corresponding to the coarse edge $\mathrm{E}_k^{\mathrm{c}} = (l, m)$ and fine edge $\mathrm{E}_i^{\mathrm{f}} = (i_1, i_2)$ is defined by

$$R_{ki} = \begin{cases} 1 & \text{if } i_1 \in G_l \wedge i_2 \in G_m \\ -1 & \text{if } i_1 \in G_m \wedge i_2 \in G_l \\ 0 & : \text{otherwise} \end{cases}. \tag{15}$$

In general, the calculation time of (14) is increase in order $o(n^2)$ or higher order, where n is the dimension of matrix. However, we can calculate (14) in order $o(n)$ using the property that the number of nonzero components in any column of matrix [R] is 1 or 0 [4].

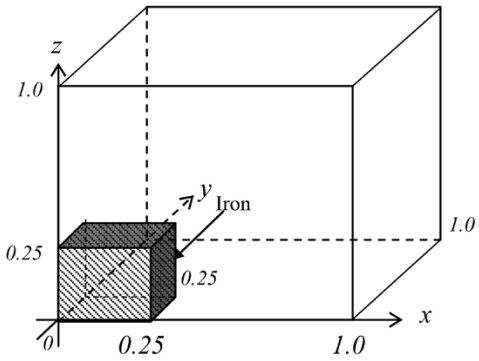

Fig. 3. Simple analysis model $(1/8)$.

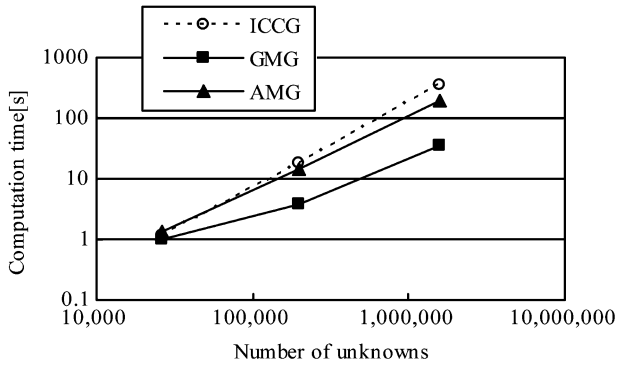

Fig. 4. Comparison of CPU time.

TABLE I
COMPARISON OF SMOOTHERS IN AMG

Num. of unknowns		point Gauss-Seidel	Block Gauss-Seidel
200,800	set up time [s]	0.7	2.3
	total time [s]	14.43	26.36
	total num. of smoothing	132	42
1,566,912	set up time [s]	6.26	20.12
	total time [s]	189.06	288.9
	total num. of smoothing	205	64

TABLE II
COMPARISON OF CALCULATION TIME

Number of unknowns		CPU time [s] GMG	AMG
200,800	1)hierarchical meshes	0.58	0.02
	2)construction of [R]	0.45	0.04
	3)construction of [Ac]	0.47	0.64
	total set up time	1.5	0.7
	iteration time	3.69	13.65
	total time	5.19	14.35
1,566,912	1)hierarchical meshes	5.02	0.17
	2)construction of [R]	3.91	0.24
	3)construction of [Ac]	3.81	5.85
	total set up time	12.74	6.26
	iteration time	35.6	189.06
	total time	48.34	195.32

It is also important to adopt the adequate smoother as well as the restriction matrix in edge based FE analysis. The block Gauss–Seidel smoother is suggested in [5] and [6] so as to overcome the bad convergence. The system matrix is divided into small submatrices to apply the block Gauss–Seidel smoother, which is the following multiplicative Schwarz procedure

$$\{x\}_i = \{x\}_{i-1} + \{r_s\}_i^T [\mathbf{K}_s]_i^{-1} \{r_s\}_i (\{b\} - [\mathbf{K}]\{x\}_{i-1}) \quad (i = 1, 2 \ldots, n_i) \quad (16)$$

where n_i is the number of nodes, $[\mathrm{K_s}]_i$ is constructed using all edges emanating from each node and $\{r_s\}_i$ is a prolongation operator between the whole matrix $[\mathrm{K}]$ and submatrix $[\mathrm{K_s}]_i$. The inverse operation of $[\mathrm{K_s}]_i$ have to be done only once and this can be done during the setup process of multigrid. In order to remove the singularity and enable to inverse the matrix $[\mathrm{K_s}]_i$, The AMG conjugate gradient method proposed in [7] is used in our study.

III. COMPARISON OF GMG AND AMG

To compare the AMG and GMG in large-scale FE analysis, we analyze a simple magnetostatic problem, in which the model consists of a cubic iron (relative permeability $= 1000$) and the surrounding air region (Fig. 3). Only $1/8$ of the model is considered due to the symmetry. The whole region is divided using tetrahedral edge elements.

Fig. 4 shows the calculation time of the two multigrid methods as well as ICCG method on an Intel Xeon 3 GHz PC. The point Gauss–Seidel smoother is applied to both multigrid methods. It is show that the GMG method is faster than the ICCG method in large number of unknowns. Although the AMG method is as fast as the ICCG method, the calculation time of AMG method increase slowly with the number of unknowns.

Next we compare the point Gauss–Seidel smoother and the block Gauss–Seidel smoother in AMG. Table I shows the convergence of the AMG with these smoothers. The setup time in Table I consists of the time to construct the virtual coarse meshes and the time to calculate (14). In the AMG with block Gauss–Seidel smoother, the time to calculate (16) is also included in the setup time. We can see that it takes a long CPU time to converge in the AMG with block Gauss–Seidel smoother even though there are a small number of iterations. This is because it takes long time to construct the submatrices as well as calculate (16).

Next we evaluate the GMG and AMG method from the viewpoint of a solver for FE analysis. Table II shows the setup time in GMG and AMG with point Gauss–Seidel smoother. The number of hierarchical meshes is 3 in both methods. The setup process of GMG and AMG consists of the following procedures.

A. Setup Procedures in GMG

1) Construction of the hierarchical meshes. The finer meshes are obtained by dividing each coarse element.
2) Construction of the restriction matrices $[\mathrm{R}]$ using (10).
3) Construction of the system matrices $[\mathrm{A_c}]$ which is obtained by the FE discretization of (4).

B. Setup Procedures in AMG

1) Construction of the virtual coarse meshes using the given fine mesh.
2) Construction of the restriction matrices $[\mathrm{R}]$ using (15).
3) Construction of the system matrices $[\mathrm{A_c}]$ using (14).

It is shown that the calculation time of these setup procedure is comparatively short in both method. Moreover these setup time increase linearly with the number of unknowns.

IV. Conclusion

This paper discusses the comparison between the GMG method and the AMG method in edge-based FE analysis. The GMG method requires the hierarchical meshes. On the other hand, the AMG method requires the only a single mesh information. The system matrices of the coarse grids are generated using algebraic operation in AMG. The numerical results show that both multigrid methods are faster than the conventional solvers such as ICCG method in large-scale analysis. Although multigrid methods require the setup procedures, the calculation time of these procedures is comparatively short and increase linearly with the number of unknowns.

Acknowledgment

The authors would like to thank Dr. T. Mifune for helpful discussions.

References

[1] M. Schinnerl, J. Schoberl, and M. Kaltenbacher, "Nested multigrid methods for the fast numerical computation of 3-D magnetic fields," *IEEE Trans. Magn.*, vol. 36, no. 4, pp. 1557–1560, Jul. 2000.
[2] J. Ruge and K. Stüben, "Algebraic multigrid," in *Multigrid Methods*, S. McCormick, Ed. Philadelphia, PA: SIAM, 1987, vol. 3, pp. 73–130. Frontiers in Applied Mathematics.
[3] S. Reitzinger and J. Schöberl, "An algebraic multigrid method for finite element discretizations with edge elements," *Numer. Linear Algebra Appl.*, vol. 9, pp. 223–238, 2002.
[4] T. Mifune, private communication, 2004.
[5] D. Arnold, R. Falk, and R. Winther, "Multigrid in H(div) and H(curl)," *Numer. Math.*, vol. 85, pp. 197–218, 2000.
[6] B. Weiß and O. Bíró, "Smoothing operators for edge element multigrid," *IEEE Trans. Magn.*, vol. 38, no. 2, pp. 397–400, Mar. 2002.
[7] T. Mifune, T. Iwashita, and M. Shimasaki, "New algebraic multigrid preconditioning for iterative solvers in electromagnetic finite edge-element analyzes," *IEEE Trans. Magn.*, vol. 39, no. 3, pp. 1677–1680, May 2003.

Manuscript received June 7, 2004.