Parallel Realization of Element by Element Analysis of Eddy Current Field Based on Graphic Processing Unit

Dongyang Wu, Xiuke Yan, Renyuan Tang, Dexin Xie, and Ziyan Ren

Department of Electrical Engineering
Shenyang University of Technology, Liaoyang, Liaoning 110870, China
shineast_521@163.com, yanxke@126.com, sgdtds@sina.com, xiedx2010@163.com, and rzyhenan@163.com

Abstract **─** The element by element parallel finite element method (EbE-PFEM) applied to engineering eddy current problem is presented in this paper. Unlike classical finite element method (FEM), only element matrix is needed to store for EbE method. Thereby more storage memory saved. Element by element conjugated gradient (EbE-CG) method is used to solve the equations which are discretized from elements level. Considering the ill-conditioned character of system equations, highly parallel Jacobi preconditioned (JP) method is used to accelerate the convergence. Besides, the process of dealing with boundary condition based on EbE theory is introduced. To validate the method, a 2D eddy current problem in complex frequency domain is used. The numerical analysis is carried out on the graphic processing units (GPU) with a compute unified device architecture (CUDA) parallel programming model to accelerate the convergence. And the results demonstrate that the JP method and GPU platform are effective in solving eddy current field with improved convergence.

Index Terms **─** Eddy current filed, element by element method, graphic processing unit, parallel computing.

I. INTRODUCTION

Due to the computer resource requirements of classical FEM for solving the electromagnetic problems, the parallel finite element method (PFEM) has become increasingly popular in recent years. Element by element (EbE) method [1] is a PFEM which can execute the parallelism on the elements level. The advantage of EbE method compared to classical FEM is that it does not need assembling and storing system matrix. Its key idea is to decouple the element solution by directly solving element equations instead of whole equations. The solving process is executed in parallel, and only intermittent communication is needed. Initially, EbE method was used for heat conduction problem and then expanded to the field of mechanics. More recently however, with the development of general purpose on graphic processing unit (GPGPU), EbE method has received increasing attention as it is very suitable for parallel processing and with the GPU[2]-[4] being a multi-core device, parallel processing at element level on different cores can be achieved. Some good results have been obtained with electrostatic problem, as in [5], [6].

In author's previous work, firstly EbB-CG method is directly used to solve 2D eddy current problem parallelly on the GPU, and 3.4 times speed up rate achieved compared with that of serial calculation with CPU [7]. Furthermore, TEAM problem 7 is taken as an example to validate the EbE method and GPU are effective for 3D linear eddy current problem, and the results have a good agreement with experiment data [8]. The purpose of this paper is to broaden the JP method to 2D eddy current analysis with two different medium in solving domain, and a comparing analysis is fulfilled between EbE-CG method and EbE-JPCG method.

II. EBE METHOD AND GPU IMPLEMENTATION

A. Node connection matrix

The key function of node connection matrix (NCM) is to transit the node information between local variables and global variables.

Now, assume $\boldsymbol{x}$ is global solution vector (GSV), $\boldsymbol{x}^e$ is the local elements solution vector (LESV), $\boldsymbol{x}^{(e)}$ is global elements solution vector (GESV), E is the total number of elements, $\boldsymbol{Q}$ is NCM. Then consider three type operations of NCM as below:

$$\boldsymbol{Q}\boldsymbol{x} = \boldsymbol{x}^{(e)}, \tag{1}$$

where $\boldsymbol{x}^{(e)} = \left(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(E)}\right)^T$, this operation achieves the alternation from GSV to GESV according to the node number of each element:

$$\boldsymbol{Q}^T \boldsymbol{x}^e = \boldsymbol{x}, \tag{2}$$

where $\boldsymbol{x}^e = \left(\boldsymbol{x}^1, \boldsymbol{x}^2, \ldots, \boldsymbol{x}^E\right)^T$, this operation achieves the summation of LESV which have the same node number. This process alternates the LESV to GSV:

$$\boldsymbol{Q}\boldsymbol{Q}^T \boldsymbol{x}^e = \boldsymbol{x}^{(e)}. \tag{3}$$

Submitted On: November 16, 2015
Accepted On: May 10, 2016

Equation (3) achieves the alternation from LESV to GESV.

NCM also can be operated with the element matrix $\boldsymbol{K}^e$, and the relationship between system matrix $\boldsymbol{K}$ and the element matrix $\boldsymbol{K}^e$ can be given as follows:

$$\boldsymbol{K} = \boldsymbol{Q}^T \boldsymbol{K}^e \boldsymbol{Q}. \tag{4}$$

Equations (1) to (4) provide the theoretical foundation for fulfilling the parallel EbE technique.

B. EbE-CG method

For the traditional FEM, the system matrix $\boldsymbol{K}$ and right hand side (RHS) vector $\boldsymbol{b}$ must be assembled from the element matrix $\boldsymbol{K}^e$ and element RHS $\boldsymbol{b}^e$, while for the EbE-PFEM, considering (1)-(4) the assemble process can be deduced as follows:

$$\boldsymbol{b} = \boldsymbol{Q}^T \boldsymbol{b}^e = \boldsymbol{K}\boldsymbol{x} = \boldsymbol{Q}^T \boldsymbol{K}^e \boldsymbol{Q}\boldsymbol{x} = \boldsymbol{Q}^T \boldsymbol{K}^e \boldsymbol{x}^{(e)}. \tag{5}$$

As shown in (5), the product of assembling the element vector is equivalent with the product of assembling element matrix. So, we can solve the element equations parallelly as below:

$$\boldsymbol{K}^e \boldsymbol{x}^{(e)} = \boldsymbol{b}^e. \tag{6}$$

As we know, CG method mainly contains two types of inner product calculations, i.e., $(\boldsymbol{r},\boldsymbol{r})$ and $(\boldsymbol{p},\boldsymbol{A}\boldsymbol{p})$ which can be calculated by EbE method as follows:

$$(\boldsymbol{r},\boldsymbol{r}) = \boldsymbol{r}^T\boldsymbol{r} = (\boldsymbol{r}^e)^T \boldsymbol{Q}\boldsymbol{Q}^T \boldsymbol{r}^e = \sum (\boldsymbol{r}^e)^T \boldsymbol{r}^{(e)}, \tag{7}$$

where $\boldsymbol{r}^{(e)} = \boldsymbol{r}^e \oplus \sum_{j\in adj(e)} \boldsymbol{r}^j$, $\boldsymbol{r}$ is the global residual, $\boldsymbol{r}^e$ is the local element residual, $\boldsymbol{Q}$ is the NCM. $\boldsymbol{r}^{(e)}$ is the sum of $\boldsymbol{r}^e$ and $\boldsymbol{r}^j$ which are relative with $\boldsymbol{r}^e$. So this process needs the solution information of adjacent nodes. The calculation of $(\boldsymbol{p},\boldsymbol{A}\boldsymbol{p})$ is similar with $(\boldsymbol{r},\boldsymbol{r})$.

C. Dealing with boundary condition

It is not necessary to assemble the system matrix for EbE method, so the boundary condition (BC) has to be applied on the elements level. Now, taking an example of 2D with triangular subdivision (Fig. 1), and assume the value of first kind BC is $\boldsymbol{U}_0$.

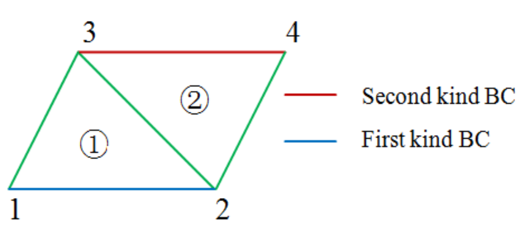

Fig. 1. Partial subdivision of 2D model.

Based on traditional FEM idea, we can get the element matrix equation of ①, as described in (8):

$$\begin{bmatrix} K_{11}^1 & K_{12}^1 & K_{13}^1 \\ K_{21}^1 & K_{22}^1 & K_{23}^1 \\ K_{31}^1 & K_{32}^1 & K_{33}^1 \end{bmatrix} \begin{bmatrix} x_1^{(1)} \\ x_2^{(1)} \\ x_3^{(1)} \end{bmatrix} = \begin{bmatrix} b_1^1 \\ b_2^1 \\ b_3^1 \end{bmatrix}. \tag{8}$$

Differ from classical FEM, the element matrix and right hand side vector must be modified with weights simultaneously. Taking element ① as an example, we can get the modified element Equation (9):

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & \dfrac{K_{22}^1}{K_{22}^1 + K_{22}^2} & 0 \\ 0 & 0 & K_{33}^{(1)} \end{bmatrix} \begin{bmatrix} x_1^{(1)} \\ x_2^{(1)} \\ x_3^{(1)} \end{bmatrix} = \begin{bmatrix} U_0 \\ \dfrac{K_{22}^1}{K_{22}^1 + K_{22}^2} U_0 \\ b_3^1 \end{bmatrix}. \tag{9}$$

In contrast to first kind BC, the second kind BC (node 3 and 4) can be applied on elements directly. For the 2D eddy current problem, the current density is easily applied to the elements level during the element analysis of RHSV.

III. NUMERICAL EXPERIMENT

In this work, a conductor in an open slot of motor is taken as an example to analyze the skin effect. Two models are considered to verify the validity of the proposed method. Model I is shown in Fig. 2, it is a current-carrying conductor in an open slot, for which the analytical solution is available [9], and the domain contains only one conducting medium. And its mathematical model is shown as below:

$$\begin{cases} \dfrac{\partial^2 A}{\partial y^2} = \mathrm{j}\omega\sigma\mu_0 A - \mu_0 J_e = p^2 A - \mu_0 J_e \quad in\,\Omega \\ \dfrac{\partial A}{\partial n} = 0 \quad (on\, AB, CD, BD) \\ \dfrac{\partial A}{\partial y} = \mu_0 H_x = -\dfrac{\mu_0 I_\mathrm{m}}{b} \quad (on\, AC) \end{cases}, \tag{10}$$

where $\boldsymbol{A}$ is vector magnetic potential, ω is angular frequency, σ electrical conductivity, μ_0 is magnetic conductivity, J_e is electrical current density, Ω is solving domain, H_x is tangential component of magnetic field intensity, I_m is magnitude of current and b is width of open slot. And the analytical solution of current density (J) is shown as follows:

$$J = -\mathrm{j}\omega\sigma A + J_e = \frac{pI_\mathrm{m}}{b\mathrm{sh}ph} \bullet \mathrm{ch}py. \tag{11}$$

Additional, in order to validate the proposed method for eddy current problem with different mediums, Model II is established in this paper (as shown in Fig. 3). For Model II, there is 1 mm width air gap surrounding the conductor, for which the condition number of its system matrix becomes greater than that of Model I, and convergence of solving the equations also becomes worse. Both of two models are under the complex excited current $\boldsymbol{I}_m = (10000 + j0)$ A.

To test the accelerating performance of proposed method on different computation scales, Model I and

Model II have been meshed into three different sizes, shown in Table 1 and Table 2. The mesh of Model II in size B is shown in Fig. 4, and its magnetic field distribution is shown in Fig. 5. Furthermore, the convergence of equations solved using CG and JPCG has been researched. Both of EbE-CG and EbE-JPCG methods are implemented with CPU and GPU separately.

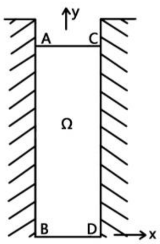

Fig. 2. Current-carrying conductor in an open slot with air surrounded.

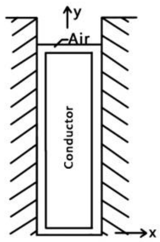

Fig. 3. Current-carrying conductor in an open slot.

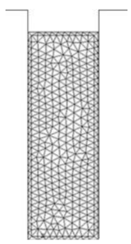

Fig. 4. The mesh of Model II in size B.

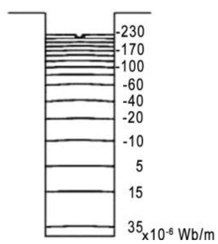

Fig. 5. Magnetic field distribution of Model II.

All the numerical computations are carried out on a server with NVIDIA GTX 660 GPU clocked at 1.0 GHz with 960 cores and 2G DDR5 global memory, and an Intel Xeon E3-1230 CPU 3.3 GHz with 8G global memory. Programming is in C++, and compiled by Visual Studio 2010 and CUDA 5.5.

To reduce the communication cost between CPU and GPU, the whole elements information is transferred to GPU global memory initially. The solving process is operated parallelly on GPU until computation results meet the convergence criterion, then result data is transferred from GPU to CPU. The GPU calculation is fulfilled on different blocks, and the threads on the same block are parallel running. But different block cannot communicate. However, during the CG iteration process, some kinds of steps such as the calculation of $\boldsymbol{r}^{(e)}$ need the information of other relative elements which are not in the same block. To overcome this, if the nodes on the boundary of memory block, the node information is stored on both sides concurrently. A little more memory needed, but high parallelism obtained. For other steps, all the read and write instructions for threads within same warp (a cluster of threads) are operated in the aligned and coalesced way to improve parallel performance.

The calculation results are shown in Table 1 and Table 2. Table 3 is shown the comparison of memory required. Figure 6 is the current density comparison between analytical and numerical solution of Model I. From Fig. 6, we can see that the result calculated using the proposed correlates well with analytical solution, which validates the method.

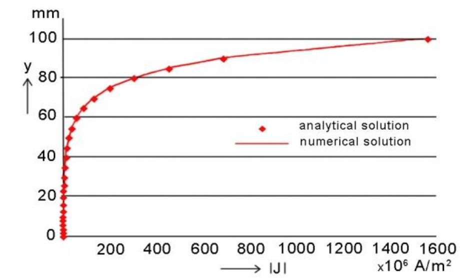

Fig. 6. Comparison of current density with EbE-CG method (Model I).

Table 1: The numerical results of Model I

Mesh Size	Node	Element	Iterations		CPU Time (ms)		GPU Time (ms)	
			CG	JPCG	CG	JPCG	CG	JPCG
A	90	138	56	33	78	62	23	18
B	342	594	100	46	485	359	87	65
C	1080	1953	175	68	1549	1231	239	173

Table 2: The numerical results of Model II

Mesh Size	Node	Element	Iterations		CPU Time (ms)		GPU Time (ms)	
			CG	JPCG	CG	JPCG	CG	JPCG
A	580	683	85	73	2578	1927	753	557
B	905	1511	134	96	6987	5125	1215	843
C	1384	2235	201	137	9768	7254	1441	935

Table 3: Comparison of memory required (Model II)

Mesh Size	Memory Required (kB)		Memory Saved (%)
	EbE	FEM	CG
A	32	79	59.5
B	72	145	50.3
C	107	218	50.9

The distribution of current density in Model II is shown in Fig. 7, which also shows that accurate results can be obtained using EbE-JPCG to eddy current problem with different medium. From the results shown in Table 1 and Table 2, we can see that the convergence of equations solving using JPCG is better than that using CG. For the same model, the GPU processor is faster than CPU due to its high parallelism.

Figure 8 shows three different mesh size level's speed up rate comparison of EbE-CG and EbE-JPCG methods which are fulfilled on GPU for Model II. Figure 9 shows the speed up rate comparison of EbE-JPCG method fulfilled on GPU for two models.

Both EbE-CG method and EbE-JPCG method are applied to Model II which contains two materials. As shown in Table 2, the time consumed is much more than Model I, however, results indicate overall improved convergence and processing time with increasing mesh size as shown in Figs. 8-9.

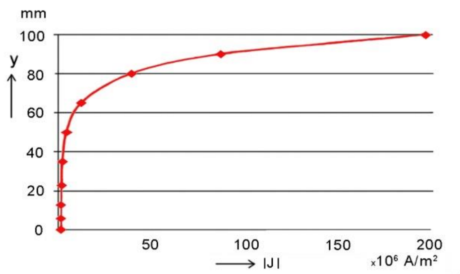

Fig. 7. Distribution of current density for EbE-JPCG method (Model II).

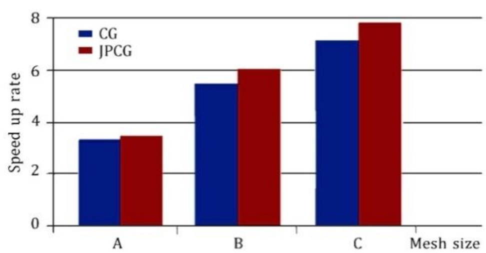

Fig. 8. Comparison of speed up rate for EbE-CG method and EbE-JPCG method.

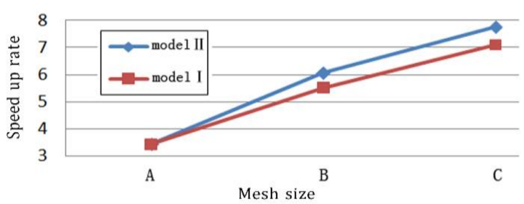

Fig. 9. Comparison of speed up rate for EBE-JPCG method implementation on GPU (Model II).

IV. CONCLUSION

The EbE-JPCG technique and GPU parallel computing platform applied to eddy current problems are the main contributions of this work. This paper presents a comparative analysis of the performance of EbE-CG method and EbE-JPCG method which are fulfilled on CPU and GPU. As shown in Table 1, Table 2 and Fig. 8, EbE-JPCG method converges more quickly than the EbE-CG method. As well, GPU acceleration becomes more effective with increasing mesh size. The numerical results demonstrate that JP method is effective for EbE method and parallel computing. As shown in Table 3, EbE method can save approximately 50% memory space, it is an important contribution for GPU platform which just has a few GB memory. Another contribution of this paper is to provide basis for solving of 3D eddy current problem, as in [8]. The future work currently in progress includes applying the EbE technique and GPU parallel platform to 3D eddy current losses calculation of large power transformer. Considering its serious ill-conditioned, JP method will be ineffective. So a new improved JP method which is also convenient for parallel EbE implementation is included in the ongoing work.

ACKNOWLEDGMENT

This work is supported by National Natural Science Funds (51507105), Science Foundation of Education Department of Liaoning Province, P.R. China (L2013046) and Natural Science Foundation of Liaoning Province, P.R. China (2015020087).

REFERENCES

[1] T. J. R. Hughus, I. Levit, and J. Winget, "An element-by-element solution algorithm for problems of structural and solid mechanics," *Computer Methods in Applied Mechanics and Engineering*, vol. 36, pp. 241-254, 1983.

[2] A. F. P. Camargos and V. C. Silva, "Performance analysis of multi-GPU implementations of Krylov-subspace methods applied to FEA of electromagnetic phenomena," *IEEE Transactions on Magnetics*, vol. 51, no. 3, March 2015.

[3] T. Okimura, T. Sasayama, and N. Takahashi, "Parallelization of finite element analysis of nonlinear magnetic fields using GPU," *IEEE Transactions on Magnetics*, vol. 49, no. 5, May 2013.

[4] O. Bottauscio, M. Chiampi, J. Hand, et al., "A GPU computational code for eddy-current problems in voxel-based anatomy," *IEEE Transactions on Magnetics*, vol. 51, no. 3, March 2015.

[5] D. M. Fernández, M. M. Dehnavi, and W. J. Gross, "Alternate parallel processing approach for FEM," *IEEE Transactions on Magnetics*, vol. 48, no. 2, pp. 399-402, 2012.

[6] I. Kiss, S. Gyimóthy, Z. Badics, et al., "Parallel realization of the element-by-element FEM technique by CUDA," *IEEE Transactions on Magnetics*, vol. 48, no. 2, pp. 507-510, 2012.

[7] D. Y. Wu, R. Y. Tang, and D. X. Xie, "*Element by element finite element method applied to eddy current problem*," *CEFC'2014*, France, May 25-28, 2014.

[8] D. Y. Wu, X. K. Yan, R. Y. Tang, D. X. Xie, and Z. W. Chen, "GPU acceleration of EBE method for 3-D linear steady eddy current field," *ICEMS'2015*, Thailand, October 25-28, 2015.

[9] D. X. Xie, "Finite element method applied to the calculation of skin effect of current-carrying conductor in an open slot," *HIET Journal*, vol. 4, no. 1, pp. 6-23, 1981. (In Chinese).