Journal of Southeast University (English Edition) Vol. 25, No. 3, pp. 362 – 366 Sept. 2009 ISSN 1003—7985

Preconditioned BiCGSTAB algorithm and its applications to eddy current solutions

Zhu Faxi Yu Haitao Hu Minqiang

(School of Electrical Engineering, Southeast University, Nanjing 210096, China)

Abstract: A new favorable iterative algorithm named as PBiCGSTAB (preconditioned bi-conjugate gradient stabilized) algorithm is presented for solving large sparse complex systems. Based on the orthogonal list, the special technique of only storing non-zero elements is carried out. The incomplete LU factorization without fill-ins is adopted to reduce the condition number of the coefficient matrix. The BiCGSTAB algorithm is extended from the real system to the complex system and it is used to solve the preconditioned complex linear equations. The locked-rotor state of a single-sided linear induction machine is simulated by the software programmed with the finite element method and the PBiCGSTAB algorithm. Then the results are compared with those from the commercial software ANSYS, showing the validation of the proposed software. The iterative steps required for the proposed algorithm are reduced to about one-third, when compared to the BiCG method, therefore the algorithm is fast.

Key words: preconditioned bi-conjugate gradient stabilized (BiCGSTAB) algorithm; incomplete LU decomposition; orthogonal list; finite element method (FEM); eddy current

The finite element analysis of sinusoidal steady-state eddy current problems will ultimately be attributed to the solution of large, sparse complex linear equations[1]. However, the calculation of complex linear equations is very complicated in practical engineering problems. Although one can change the complex system into the equivalent real system of double the order, dramatically increased computation costs often exceeds the level of tolerance. For many systems this leads to strong coupling between the equations for the real and imaginary components which may make solutions difficult. That is why this method is usually not under consideration[2]. In previous papers, the complex linear equations were solved using the direct algorithm of Gaussian elimination. However, the costs in terms of computer operations and storage is unbearable when the number of equations is too large. Iterative methods have become the mainstream method for complex linear systems, especially for the large, sparse ones. There are many iterative methods used for complex systems, such as the BiCG method and the GMRES method. The BiCG method is usually presented for non symmetric positive definite (SPD) equations. However, breakdowns sometimes occur in the solution when the BiCG method is used. Because there is no minimization of a norm as in the CG method, the convergence behavior is irregular. The biggest disadvantage is that the transpose of the coefficient matrix is needed, and this matrix is not always readily available[3-4]. The GMRES method has monotonic convergence and only involves one matrix-vector product with the coefficient matrix, but the amount of storage and operation count increases as the iteration progresses[4]. Therefore, these methods have considerable limitations in practical applications.

Equations with a large condition number of coefficient matrices formed from discretization by finite element methods in the electromagnetic field (EMF) problems may encounter slow convergence when iterative methods are applied directly to them. In order to improve convergence performance and speed up the convergence rate of the iterative methods, preconditioning techniques have been widely used. A preconditioning operation is simply a means of transforming the original linear system into one having the same solution, but it is likely to be easier to solve with an iterative solver. In fact, the matrix might have a very large range of eigenvalues, causing poor convergence behavior or leading to loss of convergence. When preconditioners are used, a new matrix is produced with eigenvalues closer to unity. A variety of preconditioning methods have been elaborated in Refs. [5 – 6]. So far, preconditioners used frequently include the incomplete Cholesky method (IC), the symmetrical successive over-relaxation method (SSOR), and the incomplete LU decomposition method (ILU[p]). Although IC decomposition has been successfully used in real systems, the precondition effect is not very good in the complex system. The biggest drawback of the IC method is that it is only suitable for symmetric systems[7]. The precondition effect of the SSOR method intensively depends on the choice of a relaxing factor ω[8], but how to choose the appropriate factor ω is still a difficult problem. The ILU[p] factorization is widely used and newer and better variants frequently emerge[9-10], where parameter p, positive integer, denotes the order of fill-in elements.

The major research work in this paper includes building up the complex incomplete LU decomposition, implementing the algorithmic program and solving large-scaled complex linear equation groups by the complex incomplete LU factorization bi-conjugate gradient stabilized method. The orthogonal list technology is introduced for the storage of the coefficient matrix and the application of the PBiCGSTAB algorithm to eddy current problems in the linear induction motor is discussed. Results show that the proposed method is precise and effective.

1 PBiCGSTAB Algorithm

Let us consider a set of complex linear equations written as follows:

$$\boldsymbol{Q}\boldsymbol{x}=\boldsymbol{b} \tag{1}$$

Received 2009-03-12.

Biographies: Zhu Faxi (1984—), male, graduate; Yu Haitao (corresponding author), male, doctor, professor, htyu@seu.edu.cn.

Citation: Zhu Faxi, Yu Haitao, Hu Minqiang. Preconditioned BiCGSTAB algorithm and its applications to eddy current solutions[J]. Journal of Southeast University (English Edition), 2009, 25(3): 362 – 366.

where $\boldsymbol{Q}$ is a complex square matrix of order n; $\boldsymbol{x}$ is an unknown complex vector to be solved; $\boldsymbol{b}$ is the constant complex vector.

1.1 Incomplete LU decomposition

As mentioned above, the convergence speed is strongly dependent on the coefficient matrix. In order to achieve good convergence characteristics, it is a wise choice to employ precondition technology. Eq. (1) is transformed into an equation system as

$$\boldsymbol{M}^{-1}\boldsymbol{Q}\boldsymbol{x} = \boldsymbol{M}^{-1}\boldsymbol{b} \tag{2}$$

where $\boldsymbol{M}$ is a nonsingular matrix of order n. Here transformed system (2) can be easier to solve, compared with solving Eq. (1). In this context, the matrix $\boldsymbol{M}$ is called a preconditioner.

One of the simplest but effective ways of defining a preconditioner is to perform an incomplete factorization as the original matrix $\boldsymbol{Q}$. This entails a decomposition of the form given as

$$\boldsymbol{Q} = \boldsymbol{L}\boldsymbol{U} + \boldsymbol{E} \tag{3}$$

where $\boldsymbol{L}(\boldsymbol{U})$ is lower (upper) triangular matrix with the same sparse structure as the original coefficient matrix $\boldsymbol{Q}$, and $\boldsymbol{E}$ is the residual or error of the factorization. This incomplete factorization known as ILU[0] is rather easy and inexpensive to compute. Thus, preconditioner matrix $\boldsymbol{M}$ can be calculated by

$$\boldsymbol{M} = \boldsymbol{L}\boldsymbol{U} \tag{4}$$

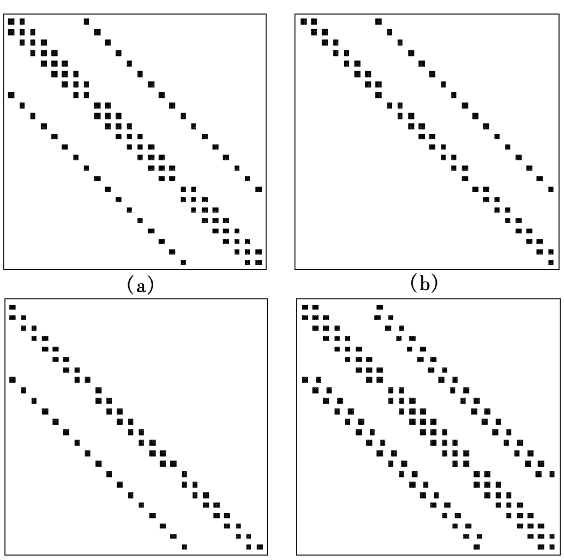

Fig. 1 ILU[0] factorization. (a) $\boldsymbol{Q}$; (b) $\boldsymbol{U}$; (c) $\boldsymbol{L}$; (d) $\boldsymbol{LU}$

The incomplete factorization (ILU[0]) principle can be best illustrated by Fig. 1. Consider one matrix $\boldsymbol{Q}$ as illustrated in Fig. 1(a). As shown in Figs. 1(b) and (c), matrix $\boldsymbol{L}(\boldsymbol{U})$ is easy to be found. If the product $\boldsymbol{LU}$ is performed, the resulting matrix will have the pattern shown in Fig. 1 (d). It is impossible in general to match $\boldsymbol{Q}$ with this product for any $\boldsymbol{L}$ and $\boldsymbol{U}$. This is due to the extra diagonals produced in the product, namely, the fill-in elements. If these fill-in elements are ignored, then it is possible to find $\boldsymbol{L}$ and $\boldsymbol{U}$ so that their product is equal to $\boldsymbol{Q}$ in the other diagonals. This defines the ILU[0] factorization in general terms.

Let $S(\boldsymbol{M})$ denote the nonzero pattern of indices in $\boldsymbol{M}$, indicating the indices of row and column of all non-zero coefficients $(i, j) \in S(\boldsymbol{M})$. Matrix $\boldsymbol{M}$ is implemented to keep members of $\boldsymbol{L}$ and $\boldsymbol{U}$. The ILU[0] algorithm is listed as follows[5]:

```
Set m_ij = q_ij
    for i = 2, ..., n
        for k = 1, ..., i - 1 and (i, k) ∈ S(M)
            m_ik = m_ik / m_kk
            for j = k + 1, ..., n and (i, j) ∈ S(M)
                m_ij = m_ij - m_ik m_kj
        end j
    end k
end i
```

where m_{ij} is the element in matrix $\boldsymbol{M}$; q_{ij} is the element in matrix $\boldsymbol{Q}$.

1.2 BiCGSTAB algorithm

The BiCGSTAB algorithm is a variant of the BiCG method. The algorithm can be thought of as a product of the BiCG method and the GMRES method. The residual vector is minimized locally by the GMRES method. The BiCGSTAB algorithm does not use the transpose of the matrix $\boldsymbol{Q}$ in the calculation of the recurrences. This is advantageous for cases where the transpose of the matrix $\boldsymbol{Q}$ is not readily available. Ref. [11] indicates that it is hard to obtain a very good result by the algorithm for which the convergence behavior is severely irregular. As a result, BiCGSTAB algorithms can achieve more precise results, when compared with algorithms such as the BiCG and the CGS. However, when the GMRES step stagnates, the residual vector is not minimized and the algorithm breaks down.

The definitions of inner products and norms in the complex systems are given as follows:

$$(\boldsymbol{X}, \boldsymbol{Y}) = \boldsymbol{X}^{\mathrm{H}}\boldsymbol{Y} \tag{5}$$

$$\|\boldsymbol{X}\|_2 = \sqrt{\sum_{i=1}^{n} |x_i|^2} \tag{6}$$

$$\|\boldsymbol{X}\|_\infty = \max(|x_i|) \tag{7}$$

where $\boldsymbol{X}, \boldsymbol{Y}$ denote complex vectors; sign H denotes conjugate and transpose, that is, $\boldsymbol{X}^{\mathrm{H}} = (\boldsymbol{X}^*)^{\mathrm{T}}$; x_i is the element of complex vector $\boldsymbol{X}$; n is the dimension of the vector $\boldsymbol{X}$; $|x_i|$ is the absolute value of a complex number.

Based on the definitions of inner products and norms above, the iterative formula of the precondition BiCGSTAB method can be written as follows:

1) Input matrix $\boldsymbol{Q}$, right vector $\boldsymbol{b}$, original vector $\boldsymbol{x}_0$ and error ε, compute $\boldsymbol{r}_0 = \boldsymbol{b} - \boldsymbol{Q}\boldsymbol{x}_0$, $\overline{\boldsymbol{r}_0} = \boldsymbol{r}_0$, $\boldsymbol{p}_0 = \boldsymbol{r}_0$;

2) Iteration for $i = 1, 2, \ldots$,

$$\boldsymbol{M}\boldsymbol{Y}_j = \boldsymbol{p}_j, \quad \alpha_j = \frac{(\overline{\boldsymbol{r}_j}, \boldsymbol{r}_i)}{(\overline{\boldsymbol{r}_j}, \boldsymbol{Q}\boldsymbol{Y}_j)}, \quad \boldsymbol{s}_j = \boldsymbol{r}_j - \alpha_j \boldsymbol{Q}\boldsymbol{Y}_j$$

$$\boldsymbol{M}\boldsymbol{Z}_j = \boldsymbol{s}_j, \quad \omega_j = \frac{(\boldsymbol{Q}\boldsymbol{Z}_j, \boldsymbol{s}_j)}{(\boldsymbol{Q}\boldsymbol{Z}_j, \boldsymbol{Q}\boldsymbol{Z}_j)}, \quad \boldsymbol{x}_{j+1} = \boldsymbol{x}_j + \alpha_j \boldsymbol{Y}_j + \omega_j \boldsymbol{Z}_j$$

$$\boldsymbol{r}_{j+1} = \boldsymbol{s}_j j - \omega_j \boldsymbol{QZ}_j, \quad \beta_{j+1} = \frac{\alpha_j}{\omega_j} \frac{(\overline{\boldsymbol{r}_{j+1}}, \boldsymbol{r}_{j+1})}{(\overline{\boldsymbol{r}_j}, \boldsymbol{r}_j)}$$

$$\boldsymbol{p}_{j+1} = \boldsymbol{r}_{j+1} + \beta_j (\boldsymbol{p}_j - \omega_j \boldsymbol{QZ}_j)$$

3) If $\|\boldsymbol{r}_{j+1}\|_2 / \|\boldsymbol{r}_0\|_2 < \varepsilon$, then stop; otherwise $j = j + 1$, begin a new iteration, and go back to 2).

1.3 Data storage technique

The coefficient matrix obtained by the FEM in eddy current problems is a large scale of a sparse one. In general, there are fewer than 10 non-zero elements in each row. As a consequence, it is costly to store every element in the coefficient matrix. Based on the orthogonal list, the special technique of storing only non-zero elements is used in this paper.

In fact, not only a lot of memory is saved but also modification operations can be carried out conveniently under this data storage strategy. In the orthogonal list, each cell is a data structure. The connections between cells are linked by a pointer. The context of cells is illustrated as follows:

i	j	comp	rp	dp

where i, j respectively denote the indices of rows and columns; comp is a complex structure in which the real components and the imaginary components of complex numbers are stored. rp is a pointer pointing to the next cell in the same row. If the current cell is the last in the row, rp points to NULL. dp is a pointer pointing to the next cell in the same column. If the current cell is the last in the column, dp points to NULL.

2 Simulation Model

Eddy current problems produced in a linear induction machine (LIM) are essentially three-dimensional. However, direct three-dimensional field analysis of the problem may cause a huge amount of calculation. As a matter of fact, two-dimensional analyses are accurate enough for most practical problems, and also easy and economical to apply.

In order to simplify practical problems into two dimensions, the assumptions made in this analysis are enumerated as follows:

1) The source current density and the magnetic vector potential have only z directed components, which are invariant in that direction, and vary sinusoidally with time;

2) The current-carrying conductors are assumed to have infinite conductivity and the eddy currents within them can be ignored;

3) The iron parts are of finite single valued permeability and resistivity. Hysterisis, magnetic saturation and temperature effects of resistivity are ignored;

4) The field is assumed quasi-stationary, so that displacement currents are absent.

In view of the foregoing assumptions[12-13], the convective diffusion equation for eddy current problems can be depicted as follows:

$$\nabla \times (\gamma \nabla \times \boldsymbol{A}) + \sigma \frac{\partial \boldsymbol{A}}{\partial t} = \boldsymbol{J} \tag{8}$$

where γ is the reluctivity; σ is the conductivity; $\boldsymbol{J}$ is the ampere density (complex); and $\boldsymbol{A}$ is the magnetic vector potential (complex).

The computational model in two-dimension is established in Fig. 2. As we can see, the entire motor needs to be studied and one pole pitch is extended at each end of a primary core so as to consider longitudinal end effects[14]. According to the computational model, boundary value problems (BVP) are given by

$$\left.\begin{aligned} \gamma \frac{\partial^2 \dot{\boldsymbol{A}}}{\partial x^2} + \gamma \frac{\partial^2 \dot{\boldsymbol{A}}}{\partial y^2} = \mathrm{j}\sigma\omega\dot{\boldsymbol{A}} - \dot{\boldsymbol{J}}_s \\ \dot{\boldsymbol{A}}_{AB} = \dot{\boldsymbol{A}}_{CD} = \dot{\boldsymbol{A}}_{AD} = \dot{\boldsymbol{A}}_{BC} = \boldsymbol{0} \end{aligned}\right\} \tag{9}$$

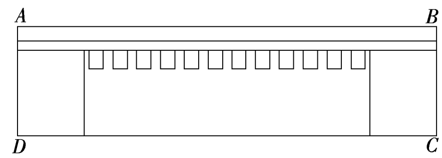

Fig. 2 Computational model of LIM

By using the Galerkin method, Eq. (9) becomes an integral equation as

$$\int_\Omega W\left\{\left(\gamma \frac{\partial^2 \dot{\boldsymbol{A}}}{\partial x^2} + \gamma \frac{\partial^2 \dot{\boldsymbol{A}}}{\partial y^2}\right) + \dot{\boldsymbol{J}}_s - \mathrm{j}\omega\dot{\boldsymbol{A}}\right\} \mathrm{d}_\Omega = 0 \tag{10}$$

where the weighting function W is set to be equal to an interpolation function and Ω denotes the boundary enclosing the whole computation region. One can obtain the large sparse complex linear equations after discretizing (10) and imposing the boundary conditions,

$$\boldsymbol{KA} = \boldsymbol{f} \tag{11}$$

where $\boldsymbol{K}$ is a coefficient matrix of order n; $\boldsymbol{A}$ is an unknown vector to be solved; $\boldsymbol{f}$ is the source on the right hand side.

3 Results and Comparison

Based upon the finite element method and the PBiCGSTAB solver, the FEM software used for performance analysis of a linear machine is programmed by C++. In order to prove the correctness of the programmed software, a locked-rotor state of a single-sided LIM is simulated. Then the results are compared with those from the commercial software ANSYS. Also, the convergence characteristics of the BiCG method, the BiCGSTAB method, and the PBiCGSTAB method are compared.

The flux densities in the centerline of the gap are given in Fig. 3 and Fig. 4. In fact, only the real component of the Y-axis component of the flux density B_y is provided, mainly due to the fact that flux densities are complex and two-dimensional. Through the comparison of FEM and ANSYS, the validity of FEM software is confirmed. Thus, the correctness of the PBiCGSTAB algorithm to solve complex equations is proved. Fig. 3 is plotted when the phase angle of the current in phase A is equal to 30°. However, the angle in Fig. 4 is 60°.

A comparison is made between the BiCGSTAB method and the BiCG method, which is commonly used in the solution of complex linear equations. Also, the comparison between the BiCGSTAB method and the PBiCGSTAB algorithm is carried out. In order to compare the rate of convergence and the convergence characteristics, convergence curves of various iterative methods are provided in Fig. 5 and Fig. 6. Here, a new criterion for judgment is presented. When the qualification $\|\boldsymbol{r}\|_{\infty} < 10^{-8}$ is satisfied, the algorithm is considered to have converged.

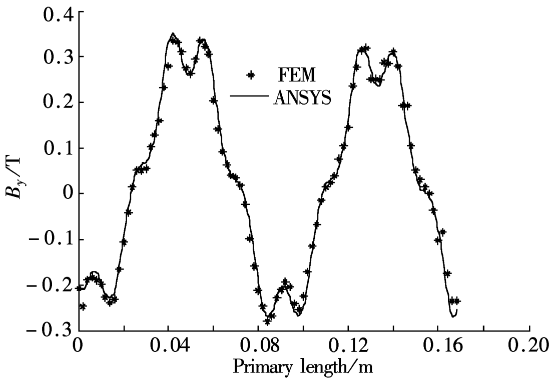

Fig. 3 Flux density along air gap (phase is 30°)

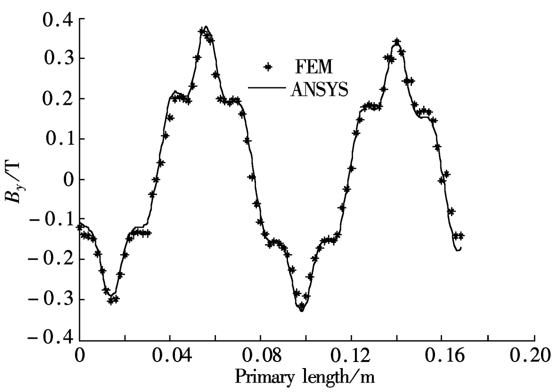

Fig. 4 Flux density along air gap (phase is 60°)

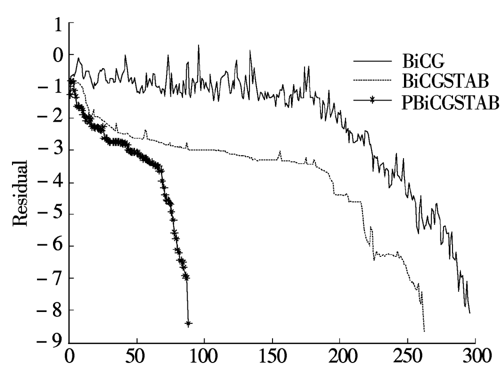

Fig. 5 Convergence curve of iterative method(1 925 nodes)

It can be seen in Fig. 5 and Fig. 6 that all the three algorithms converge to the final result; however, the required iterative steps are different. In order to obtain the same accuracy of results, the BiCG algorithm requires the most iterative steps. Irregular convergence behavior is another drawback of the BiCG method.

Oscillation accompanies the whole convergence process. This is mainly because there is no minimization of a norm as in the CG method. As we can see, the BiCGSTAB algorithm can reach very good accuracy in fewer steps and its convergence curve is much smoother in comparison with the BiCG method. The best method mentioned above is the PBiCGSTAB method which converges rapidly to the precise results

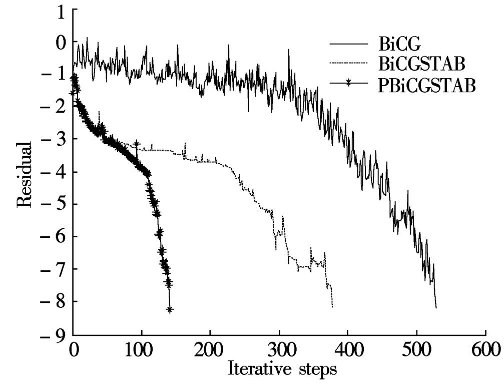

Fig. 6 Convergence curve of iterative method(4 950 nodes)

without oscillation. Through the contrast between the BiCGSTAB method and the PBiCGSTAB method, preconditioning techniques have been proved to be of great importance in the solution of the large, sparse and complex systems.

4 Conclusion

The orthogonal list technology is suitable for the storage of a large sparse matrix, especially for a complex one. The BiCGSTAB algorithm can reach very good accuracy in fewer steps and its convergence curve is much smoother in comparison with other iterative methods in the solution of a complex system. Precondition technique of incomplete LU decomposition can greatly improve the efficiency of the algorithm. The combination of the BiCGSTAB method and incomplete LU decomposition is appropriate for the solution of complex equations generated in the finite element analysis.

References

[1] Chari M V K. Finite-element solution of the eddy-current problem in magnetic structures [J]. *IEEE Transactions on Power Apparatus and Systems*, 1974, **93**(1): 62 − 72.

[2] Jacobs D A H. A generalization of conjugate-gradient method to solve complex systems [J]. *IMA Journal of Numerical Analysis*, 1986, **6**(4): 447 − 452.

[3] Kleiner M A. Steady-state methods for simulation of RF and microwave circuits[D]. Medford: Tufts University, 2002: 48 − 51.

[4] Topsakal E, Kindt R, Sertel K, et al. Evaluation of the BICGSTAB(1) algorithm for the finite-element/boundary-integral method [J]. *Antennas and Propagation Magazine*, 2001, **43**(6): 124 − 131.

[5] Saad Y. *Iterative methods for sparse linear systems* [M]. New York: PWS Publishing, 1996: 265 − 319.

[6] Ke C. *Matrix preconditioning techniques and applications* [M]. London: Cambridge University Press, 2005.

[7] Zhang Yongjie, Sun Qin. Preconditioned matrix and its structure methods [J]. *Journal of Changchun University of Science and Technology*, 2006, **29**(4): 128 − 130. (in Chinese)

[8] Wen Daigang, Jiang Kexun, Huang Jian. Real type of ICCG method for complex algebraic equations of FEM [J]. *Transactions of China Electro-Technical Society*, 1996, **11**(4): 62 − 64. (in Chinese)

[9] Wang Kai, Kim Sangbae, Zhang Jun. Global and localized parallel preconditioning techniques for large scale solid earth simulations [J]. *Parallel and Distributed Processing Symposium*, 2003, **19**(4): 22 − 26.

[10] Sheen S C, Wu J L. Preconditioning techniques for the BiCGSTAB algorithm used in convection-diffusion problems [J]. *Numerical Heat Transfer*, 1998, **34**(2): 241 − 256.

[11] Cao Zhihao. *Iterative methods based on variation principle* [M]. Beijing: Science Press, 2005: 182 − 184. (in Chinese)
[12] Ye Yunyue. *The theory and application of linear machines* [M]. Beijing: China Machine Press, 2000: 26 − 31. (in Chinese)
[13] Poloujadoff M. *The theory of linear induction machinery* [M]. New York: Oxford University Press, 1980.
[14] Lu Qinfen, Ye Yunyue. Performance study on linear induction motor used by amusement equipment based on FEM [C]// *Proceedings of the Eighth International Conference on Electrical Machines and Systems*. Nanjing, China, 2005: 748 − 751.

预处理的 BiCGSTAB 算法及其在涡流求解中的应用

朱发熙　余海涛　胡敏强

(东南大学电气工程学院,南京 210096)

摘要:针对涡流场有限元分析形成的大型稀疏复线性方程组,提出了预处理稳定双共轭梯度法(PBiCGSTAB). 利用二维正交链表结构实现系数矩阵的全稀疏存储,采用无填充的不完全 LU 分解对方程组进行预处理以降低系数矩阵的条件数. 将稳定双共轭梯度法从实数领域扩展到复数领域,并利用它求解预优过的复线性方程组. 基于有限元法和 PBiCGSTAB 算法编制直线电机性能分析软件,并对一电机堵转的情况进行仿真,将计算结果和 ANSYS 计算出的结果进行了比较,证明了该软件的正确性. 并且通过求解器比较发现,在相同精度条件下 PBiCGSTAB 算法只需要 BiCG 算法三分之一的迭代步数,证明了该算法的快速性.

关键词:预处理稳定双共轭梯度法;不完全 LU 分解;正交链表;有限元法;涡流
中图分类号:TM154