A Near Zone Preconditioner with Sparse Approximate Inverses for Solving Finite Element Equations

Yuqing Xie, and Lin Li, *Member, IEEE*

State Key Laboratory of Alternate Electrical Power System with Renewable Energy Sources,
North China Electric Power University, Beijing, 102206, China, yuqing_xie@ncepu.edu.cn

A physically motivated near zone preconditioner is presented for solving the equations obtained from finite element method. Different from common sparse approximate inverse (SPAI) preconditioner, the proposed one gives the sparsity pattern based on a physical approximation. And its sparseness can be adjusted in different applications. The process of the algorithm needs low memory and CPU time, and is inherently parallel. The preconditioner in conjunction with conjugate gradient method is used to calculate the electrical fields in a power transformer. And this application demonstrates the effectiveness of the approach.

***Index Terms*—Finite element method, Power transformer, Preconditioner, Sparsity pattern**

I. Introduction

The equations which are derived from engineering problems based on finite element method (FEM) are usually sparse, and also frequently badly conditioned. In such cases, it may make the convergence rate of the iteration low and the accuracy of the result bad. Hence, the original equation needs to be transferred into another equation which has lower condition number. The preconditioning techniques can realize the transformation. In other word, a preconditioner would make the spectral properties of the coefficient matrix better.

The most popular preconditioner is incomplete Cholesky (IC) factorization preconditioners for symmetric and definite matrices [1], or incomplete LU (ILU) factorization preconditioners for general matrices [2]. The IC or ILU factorization apply some recursive algorithms to form a sparse lower triangular matrix and a sparse upper triangular matrix, which is difficult to be computed in parallel.

A common parallelized preconditioner is approximate inverse preconditioner, which was the first to be proposed in the 1970s [3]. The main difficulty of the approximate inverse techniques to form the preconditioners is how to pick a suitable sparsity pattern for a good preconditioner. To overcome the difficulty, the SPAI preconditioner is proposed, whose sparsity pattern is captured automatically [4]. But for some problems, the performed sparsity pattern may not be very perfect.

The major purpose of this paper is to introduce a sparsity pattern to construct an approximate inverse preconditioner in parallel. The proposed sparsity pattern is based on the fact that the solution of one node in finite element method is directly affected by the properties of near zone nodes. Similar ideas are also used in the method of moment solutions for integral equations [5].

II. Derivation of the Preconditioner

In the numerical analysis for some engineering problems, the discretized matrix equations derived from finite element method generally has the following form

$$Ax = b \tag{1}$$

The coefficient matrix A is sparse and commonly has a large condition number, which result in a high computational cost for the iterative solver. In order to improve the situation, we consider a preconditioning matrix M, and apply it into (1). Then, the preconditioned equation can be described as

$$MAx = Mb \tag{2}$$

The matrix M should have a good approximation of A^{-1}, and it also should have a good sparsity pattern to decrease the computational cost. Considering the features of M, a method to obtain the preconditioner is to minimize the 2-norm of the residual matrix I-AM

$$\min_{M\in S}\left\|AM-I\right\|_2^2=\min_{m_k\in s_k}\sum_{k=1}^{n}\left\|Am_k-e_k\right\|_2^2 \tag{3}$$

The vector m_k is the kth column of the matrix M, and e_k is the unit vector. The set S is the sparsity pattern of the matrix M, and the set s_k is the sparsity pattern of the vector m_k. And n is the order of matrix A. The solution of (3) is in equivalent to n independent least square problems

$$\min_{m_k\in s_k}\left\|Am_k-e_k\right\|_2^2 \qquad k=1,2,\cdots,n \tag{4}$$

The process to solve (4) is to solve n independent equations, which can be solved completely in parallel. The remaining problem is to obtain a suitable sparsity pattern for M. The basic idea to determine the sparsity pattern for M is as follows.

Based on the characteristics of the stiff matrix arises from finite element method, the elements in A is directly affected by the connection between nodes. Fig. 1 shows the correlation between node 1 and the surrounded nodes. And it is just part of the whole physical domain divided by the quadratic element.

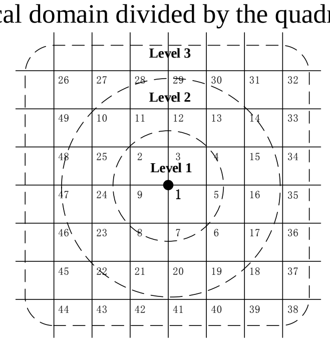

Fig. 1. Correlation between node 1 and the surrounded nodes.

In Fig. 1, node 1 has a strong relevance with node 1~9, which is labelled as level 1. The nodes of level 2, that is node 10~25, has a lower relevance than level 1, which can be labeled as level 2. This relevance can be extended to the whole domain.

The way to analyze the correlation degree between node 1 and its surrounded nodes can also be applied to other nodes. And to some extent, the correlation degree between nodes reflects the strength of coupling in physical fields. Hence, we can take strong coupling elements as the nonzero elements in M, which can determine the sparsity pattern of M. For (4) when $k=1$, the nonzero elements in vector m_1 may belong to the set S_{11} ($S_{11}=\{m_1, m_2, \ldots, m_9\}$), and the subscript of the elements in S_{11} is the same as the node number in the level 1. If changing the nonzero elements set S_{11} to S_{21} ($S_{21}=\{m_1, m_2, \ldots, m_{25}\}$), the vector m_1 can be calculated more accurate and less sparse. And if the nonzero elements set including all of the elements, which is corresponding to the fact that node 1 have relevance with the whole nodes in the domain, the solution of m_1 is exactly equal to the fist column vector of A^{-1}.

The above has showed the strategy to obtain the sparsity pattern of m_1, and other column of M can also be achieved with the same technique. Supposing the nonzero elements of m_k belongs to the set S_k, and J is a set which equals the subscripts of the elements in S_k, a new submatrix $A(I,J)$ can be defined, where the set I is obtained when $A(:,J)$ is not equal to zero. With the new defined submatrix, problems (4) can be reduced to the following forms

$$\tilde{A}_k \tilde{m}_k = \tilde{e}_k \qquad k = 1, 2, \cdots n \tag{5}$$

with

$$\tilde{A}_k = A(I,J)\ , \quad \tilde{m}_k = M(J,k)\ , \quad \tilde{e}_k = e(k)$$

Since coefficient A_k is small, equation (5) can be easily solved through least squared method.

III. Numerical Experiment

A problem of computing electrical fields in a power transformer is used to test the proposed algorithm. The calculation model is showed in Fig.2. The model is 2D equivalent model from part of an actual power transformer. In the model, the left and bottom is symmetric, and the top and right is the tank, which is grounded. The coil is connected to the voltage excitation. The dielectric constants of oil and paper in the model are $\varepsilon_{oil} = 2.2\varepsilon_0$, $\varepsilon_{paper} = 4.4\varepsilon_0$.

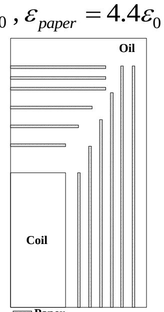

Fig. 2. Calculation model of power transformer.

The model is divided into 16000 quadratic elements with 16301 FEM unknowns. And the condition number of the stiff matrix is 3.35×10^{20} , while the number of the none zero entries is 144901.The tradition CG method, ICCG method and preconditioning conjugate gradient (PCG) method with preconditioners proposed in section Ⅱ are applied to calculate the electrical fields of the power transformer. Fig.3 displays the convergence curves with different methods when the voltage applied to the coil is 10 KV. In the figure, the label "lev1_PCG" means PCG method, in which the sparsity pattern of the preconditioner is derived from the nodes of level 1 in Fig.1 . And the other labels have the same meaning. Obviously, the preconditioner in "lev4_PCG" is more complicated and less sparse than in "lev1_PCG".

Fig.3 reveals the main features of the preconditioners proposed in this paper. On one hand, it's clear that the more complicated of the preconditioner, the faster of the convergence in the iteration solver of PCG method. On the other hand, the PCG method is faster than traditional CG method within a certain residual error. And it has a similar iteration speed with ICCG method, while the proposed method can be implemented in parallel.

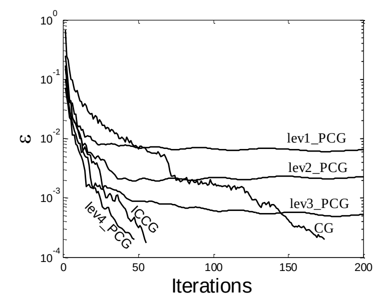

Fig. 3. Convergence curves with different methods, where $\varepsilon=\|Ax_k-b\|_2/\|x_k\|_2$.

IV. Conclusions

A strategy to compute the preconditioner for solving the finite element equations has been presented in this paper. It needs low computational cost and can be realized in parallel. Different sparseness of the preconditioner has also been described to be used in applications with different levels of complexity. The features of the preconditioner are displayed through an industrial application of calculating the electrical fields in a power transformer.

Acknowledgment

This project is supported by the project of National Natural Science Foundation of China. (No.51277064)

References

[1] J. A. Meijerink and H. A. van der Vorst, "An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix," Mathematics of Computation, vol.137, no.137, pp. 148-162, Jan. 1977.

[2] E. Chow and Y. Saad, "Experimental study of ILU preconditioners for indefinite matrices," Journal of Computational and Applied Mathematics, vol. 86, no.2, pp. 387-414, Dec. 1997.

[3] M.Benzi, "Preconditioning techniques for large linear systems: a survey," Journal of Computational Physics, vol. 182, no.2, pp. 418-477, Nov. 2002.

[4] M. J. Grote and T. Huckle, "Parallel preconditioning with sparse approximate inverses," SIAM Journal on Scientific Computing, vol. 18, no.3, pp. 838-853, 1997.

[5] M. Carr, M. Bleszynski and J.L.Volakis, "A near-field preconditioner and its performance in conjunction with the BiCGstab (ell) solver," IEEE Antennas and Propagation Magazine, vol. 46, no.2, pp. 23-30, Apr. 2004.