Partial Tree-Gauging of Second Order Edge Element Vector Potential Formulations

Oszkár Bíró, *Member, IEEE*, and Kurt Preis

Institute of Fundamentals and Theory in Electrical Engineering, Graz University of Technology, 8010 Graz, Austria, biro@tugraz.at

A technique to partially gauge vector potential formulations represented by second order edge basis functions is presented. The eliminated edges are locally defined and form a part of a tree of the graph defined by the finite element mesh. Hence, the curl of any vector potential spanned by the edges of the mesh can be represented as the curl a vector potential function supported by the retained edges. The resulting finite element equation system with a reduced number of degrees of freedom is well conditioned and, as illustrated by a numerical example, its iterative solution is significantly faster than that of the full system.

***Index Terms*—Computational electromagnetics, finite element analysis, edge elements, tree-gauge.**

I. Introduction

The algebraic equation systems resulting from the discretization of vector potential curl-curl equations by edge basis functions are known to be singular. The reason is that the curls of the edge basis functions are linearly dependent.

The usual way to eliminate the singularity is to set the degrees of freedom corresponding to the edges belonging to any spanning tree of the graph defined by the finite element mesh to zero (tree-gauge, [1]). However, the resulting system is well known to be ill conditioned, making the application of iterative Krylov-types methods infeasible [2]. This is in contrast to the singular full system which can be solved relatively fast by conjugate gradient methods, provided the right hand side is consistent [3].

In the present paper, a technique of setting less degrees of freedom to zero than the number of tree edges is proposed for the case of second order edge based functions being used. The resulting set of equations is still singular, but, in addition to involving less degrees of freedom than the full system, it is well conditioned and can be solved fast using iterative techniques.

II. Tree-gauge

In order to establish the notations, the idea of tree-gauging is reviewed briefly [1]. Let E denote the set of all n^e edges e_j in a finite element mesh. Any vector potential function $\mathbf{A}$ can be approximated on the mesh as

$$\mathbf{A}_h = \sum_{e_j \in E} A_j \mathbf{N}_j \tag{1}$$

with A_j denoting the degree of freedom corresponding to e_j, i.e.

$$A_j = \int_{e_j} \mathbf{A} \cdot d\mathbf{r} , \tag{2}$$

and $\mathbf{N}_j$ being the edge basis function associated with e_j, i.e.

$$\int_{e_j} \mathbf{N}_i \cdot d\mathbf{r} = \delta_{ij} \tag{3}$$

where δ_{ij} is the Kronecker symbol. Selecting a spanning tree T in the graph defined by E, e_j can be classified as a tree edge: $e_j \in T$ or as a co-tree edge: $e_j \in C$ where C is the set of co-tree edges, i.e. $T \cup C = E$. $\mathbf{A}_h$ can now be split as

$$\mathbf{A}_h = \mathbf{A}_h^{tree} + \mathbf{A}_h^{co-tree} = \sum_{e_j \in T} A_j \mathbf{N}_j + \sum_{e_j \in C} A_j \mathbf{N}_j \ . \tag{4}$$

Obviously, $\mathbf{A}_h^{tree}$ is a gradient function, i.e. there exists a scalar function u so that

$$\mathbf{A}_h^{tree} = \nabla u \ . \tag{5}$$

The appropriate function u can be constructed as $u = \sum_{n_i \in N} u_i N_i$ where N is the set of all n^n nodes n_i in the finite element mesh, u_i are the nodal values and N_i the node basis functions satisfying $N_i\left(n_k\right) = \delta_{ik}$. Indeed, the values u_i can be set so that $A_j = u_{j_2} - u_{j_1}$ for all $e_j \in T$ pointing from node j_1 to j_2, since T connects all nodes without forming a loop. In view of (5), setting $\mathbf{A}_h^{tree}$ to zero does not change the curl of $\mathbf{A}_h$, i.e. $\mathbf{A}_h = \mathbf{A}_h^{co-tree}$ is an appropriate gauge. In addition, the curls of the edge basis functions associated with the co-tree edges (numbering n^e-(n^n-1)) are linearly independent and, hence, the finite element system matrix is nonsingular.

III. Partial Tree-gauge

The advantage of tree-gauging is that it reduces the number of degrees of freedom and, thus, potentially leads to lower computational demand. However, as mentioned, the resulting system matrix is ill conditioned and solving the algebraic equations by Krylov-type techniques like the method of conjugate gradients requires such an enormous number of iterations that the gain due to the lower number of degrees of freedom is more than offset by the resulting high computing time.

One reason for the deterioration of the conditioning is that the condition number of the reduced element matrices is much higher than for the full element matrix as illustrated for second order hexahedral elements in [4]. In addition, the number of edges belonging to a spanning tree is very different for each finite element, further worsening the overall conditioning.

The number of degrees of freedom can be reduced by not eliminating all edges belonging to a tree, but only some of them. If this is done in a way that, on the one hand, the condition number of the reduced element matrix is relatively low, and, on the other hand, the number of eliminated edges is the same for all elements, the resulting reduced system matrix will be only slightly worse conditioned than the original full one.

In case of second order edge elements, it is customary to introduce mid-side nodes in addition to the corner nodes arising in first order elements. This leads to two edges being present along a geometrical edge (side) of the element. As an example, consider second order hexahedral element with 20 nodes and 36 edges shown in Fig. 1a [5]. The situation is similar in case of the tetrahedral elements with 10 nodes and 20 edges presented in [6]. For such elements it is straightforward to eliminate one of the edges along each element side. The remaining 24 edges for the element shown in Fig 1a are indicated by thick lines in Fig. 1b. The number of edges retained in case of the of the tetrahedral elements of [6] is 14. The elimination of these half-side edges can easily be carried out globally so that the number of eliminated edges in each finite element of the mesh is the same. Indeed, one half of each global side will be retained and the other half eliminated.

It is easy to see that the half-side edges thus eliminated form a part of a global tree, since no loop in the graph can exist which is entirely made up of them. Denoting the set of the eliminated edges by H, and that of the retained edges by R, the approximation of the vector potential can be split as

$$\mathbf{A}_h = \mathbf{A}_h^{half-side} + \mathbf{A}_h^{remaining} = \sum_{e_j \in H} A_j \mathbf{N}_j + \sum_{e_j \in R} A_j \mathbf{N}_j \ . \qquad (6)$$

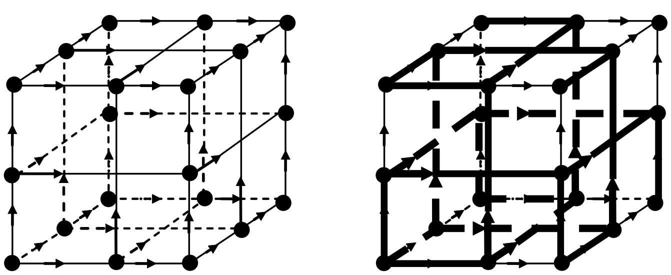

Fig. 1. A second order hexahedral element.
a.: 20 nodes and 36 edges. b.: Remaining edges thick.

The partial tree-gauge amounts to setting $\mathbf{A}_h^{half-side}$ to zero which, obviously, does not restrict the curl of the vector potential. However, the curls of the edge basis functions corresponding to the edges in R are not entirely independent, i.e. the resulting system matrix will be still singular. Considering a single element, the rank of the 24x24 element matrix in the hexahedral case is 17 (n^e=36, n^n=20, n^e-(n^n-1)=17) and that of the 14x14 element matrix in the tetrahedral case is 11 (n^e=20, n^n=10, n^e-(n^n-1)=11). As presented in [4], the condition number of the element matrix of an undistorted second order element shown in Fig. 1a is 6.6625 if no edges are eliminated (19 zero eigenvalues), but is as high as 143.07 if the co-tree edges are retained only (no zero eigenvalue). In contrast, the element matrix of order 24 obtained by eliminating the half-side edges has a condition number with the modest value of 15.242 (7 zero eigenvalues).

IV. Numerical Example

To illustrate the effectiveness of using the partial tree-gauge, TEAM Workshop Problem No. 7 has been meshed using the elements shown in Fig. 1a with their number ranging between 2,890 and 184,960. The $\mathbf{A}_r$,V-$\mathbf{A}_r$ formulation has been employed with three different gauging strategies of no gauge, partial-tree gauge and tree-gauge. The resulting equations have been solved by the ICCG method to achieve the norm of the residual vector normalized by that of the right hand side vector to fall below 10^{-6}. The computations have been carried out on an Intel(R) i7 CPU@2.93 GHz platform. The computational data shown in Table I indicate that the partial gauge is superior to using no gauge and much better than full tree-gauging.

TABLE I
Computational Data

Elements (Hexahedral)	Gauge	No. DoF	Iterations	CPU time/s for solution
	No	31,826	135	2.2
2,890	**Partial**	**24,370**	**154**	**1.2**
	Tree	22,066	838	5.2
	No	267,256	228	33.8
23,120	**Partial**	**202,840**	**430**	**30.0**
	Tree	182,149	2,634	144.0
	No	916,674	413	203.6
78,030	**Partial**	**693,774**	**463**	**111.3**
	Tree	621,274	5,938	1,123.9
	No	2,190,464	545	643.7
184,960	**Partial**	**1,655,536**	**635**	**380.8**
	Tree	1,480,465	28,244	12,602.4

V. Acknowledgment

The authors wish to thank Joachim Schöberl for the discussions leading to the idea presented.

References

[1] R. Albanese, G. Rubinacci, "Magnetostatic field computations in terms of two-component vector potentials," *Int. J. Num. Meths*, vol. 29, pp. 515-532, March 1990.

[2] O. Bíró, K. Preis, K. R. Richter, "On the use of the magnetic vector potential in the nodal end edge finite element analysis of 3D magnetostatic fields," *IEEE Trans. Magn.*, vol. 32, pp. 651-654, May 1996.

[3] Z. Ren, "Influence of the RHS on the convergence behaviour of the curl-curl equation," *IEEE Trans. Magn.,* vol. 32, pp. 655-658, May 1996.

[4] I. Tičar, O, Bíró, K. Preis, "Vector potential expanded by edge basis functions associated with loops on finite-element facets," *IEEE Trans. Magn.,* vol. 38, pp. 437-440, March 2002.

[5] A. Kameari, "Calculation of transient 3D eddy current using edge-elements," *IEEE Trans. Magn.,* vol. 24, pp. 118-121, March 1990.

[6] J. F. Lee, D. K. Sun, Z. J. Cendes, "Tangential vector finite elements for electromagnetic field computation," *IEEE Trans. Magn.,* vol. 27, pp. 4032-4035, September 1991.