

Performance of a Nonlinear 2D and 3D Magnetostatic Volume Integral Formulation

Kimmo Forsman, Lauri Kettunen and Jukka-Pekka Nuutinen
Tampere University of Technology
Laboratory of Electricity and Magnetism
P.O.Box. 692, FIN-33101 TAMPERE, FINLAND

Abstract – The characteristics and performance of an h -type method using volume integral equations and edge elements to calculate nonlinear 2D and 3D magnetostatic fields are studied. Recent developments in the GFUNET package, which implements a volume integral method, have significantly improved the efficiency of the code. This is borne out by recent results.

I. INTRODUCTION

Numerical approximations for static or low frequency electromagnetic problems can be computed with either partial differential equations (PDEs) or integral equations (IEs). IEs have some advantages over PDEs; for example, air regions can be excluded, and the exterior boundary condition (i.e. the magnetic field vanishes in infinity) is satisfied automatically. One of the main disadvantages of IEs is that it requires the solution of a dense set of linear equations; PDEs typically yield sparse sets. Direct methods need $O(n^3)$ operations to solve a dense set. However, by using iterative solvers and parallel computers the problems with dense matrices can be reduced [1]. Therefore, it seems reasonable to study IEs especially in shape optimization problems and in eddy current problems with moving objects.

In this paper an h -type volume integral formulation is used to accurately solve some 2D and 3D nonlinear magnetostatic problems. The software in which such a method is adopted is called GFUNET. A concise description of the method for 3D problems is presented; the major differences between the 2D and 3D versions are discussed. Theoretical aspects of the formulation are discussed in more detail elsewhere [1], and this paper concentrates on the accuracy and the speed of GFUNET.

II. AN h -TYPE INTEGRAL FORMULATION

A. Approximation of magnetic field strength

In GFUNET [1]-[2] the magnetic field strength $\mathbf{H}$ is

Manuscript received July 6, 1994. This work was supported in part by the Academy of Finland.

approximated in the space W^1 spanned by Whitney elements of order $p = 1$ (i.e. edge elements). The unknown coefficients h_i are the line integrals of $\mathbf{H}$ along edges of a tetrahedral mesh (triangular in 2D) such that

$$\mathbf{H}^1 = \sum_{i=1}^{N_e} h_i \mathbf{w}_i, \quad (1)$$

where $\mathbf{w}_i$ is a basis function of W^1 related to edge e_i and N_e is the number of edges in the mesh.

In a simply connected region V where there are no source currents, all closed line integrals of $\mathbf{H}$ vanish, and therefore the coefficients h_i are not independent. An independent set of coefficients can be found by forming a tree from the graph of all edges in the mesh [1]-[3]. Thus any h_i can be expressed with the tree edges h_j^t as

$$h_i = \sum_{j=1}^{N_{te}} R_{ij} h_j^t, \quad (2)$$

where N_{te} is the number of tree edges and R is an $N_e \times N_{te}$ incidence matrix entries of which are all -1, 0, or 1. By combining (1) and (2) $\mathbf{H}^1$ can be expressed in terms of h_j^t along tree edges:

$$\begin{aligned} \mathbf{H}^1 &= \sum_{i=1}^{N_e} h_i \mathbf{w}_i = \sum_{i=1}^{N_e} \left(\sum_{j=1}^{N_{te}} R_{ij} h_j^t \right) \mathbf{w}_i \\ &= \sum_{i=1}^{N_{te}} h_i^t \left(\sum_{j=1}^{N_e} R_{ji} \mathbf{w}_j \right) = \sum_{i=1}^{N_{te}} h_i^t \mathbf{v}_i, \end{aligned} \quad (3)$$

where $\mathbf{v}_i = \sum_{j=1}^{N_e} R_{ji} \mathbf{w}_j$ is a basis function of $\text{grad } W^0 = W^1 \cap \ker(\text{curl})$ associated to tree edge e_i [1].

B. Volume integral formulation

The magnetic field $\mathbf{H}$ can be decomposed into the part due to the source currents, $\mathbf{H}_s$, and that due to magnetization, $\mathbf{H}_m$. Multiplying $\nabla \cdot \mathbf{B} = 0$ with a test function, integrating over V , and applying integral relationships a variational equation can be derived [1]:

$$\int_V \mu \mathbf{H} \cdot \mathbf{v}_i dv = \int_V \mu \mathbf{H}_s \cdot \mathbf{v}_i dv + \int_V \mu \mathbf{H}_m \cdot \mathbf{v}_i dv, \quad \forall \mathbf{v}_i. \quad (4)$$

If $\mathbf{H}_s$ and $\mathbf{H}_m$ are also approximated in $W^1 \cap \ker(\text{curl})$ (i.e. $\mathbf{H}_s^1 = \sum h_i^s \mathbf{v}_i$ and $\mathbf{H}_m^1 = \sum h_i^m \mathbf{v}_i$), the solution of (4) can be shown to be equivalent to that of

$$h_i = h_i^s + h_i^m, \quad \forall i = 1, \dots, N_{te}, \quad (5)$$

where the superscript t , referring to tree edges, has been dropped [1].

In (5), h_i^m can be calculated as a difference in scalar potential between the end nodes of e_i . Moreover, by presenting the magnetization as $\mathbf{M} = \chi(\|\mathbf{H}^1\|)\mathbf{H}^1$ and by linearizing the nonlinear problem, (5) can be given in matrix form,

$$\sum_{j=1}^{N_{te}} C_{ij} h_j = h_i^s, \quad \forall i = 1, \dots, N_{te}. \quad (6)$$

The elements of matrix C are

$$C_{ij} = \delta_{ij} + \sum_{k=1}^2 \frac{(-1)^k}{4\pi} \int_V \chi(\mathbf{r}') \frac{\mathbf{v}_j(\mathbf{r}') \cdot (\mathbf{r}_{i_k} - \mathbf{r}')}{\|\mathbf{r}_{i_k} - \mathbf{r}'\|^3} dv', \quad (7)$$

where $\mathbf{r}_{i_k}$ is the location of k^{th} node (n_{i_k}) of e_i . The nonlinearity of the susceptibility χ is taken into account by iteratively solving (6). The right hand side (RHS), h_i^s , can be found using the Biot-Savart law.

III. IMPLEMENTATION

A. Generation and solving of matrix equation

For practical reasons it is convenient to define a local tree in each tetrahedron and to compute the volume integral in (7) element by element. The contributions of local tree edges e_i should be added to those of global tree edges forming a path that connects the end nodes of e_i [2]. These paths correspond to the non-zero entries of incidence matrix R , though R is never explicitly formed.

A tree is formed by arbitrarily selecting a root node and by applying a breadth-first search (BFS) to the graph of all edges in the mesh. The BFS provides the father for each node and also ensures that for a given root, the tree depth is minimized. Thus a path from any node to the root node is easily obtained, and by combining two of those paths, a path between any pair of nodes is quickly generated. The CPU-time required for tree and path generation is negligible compared with the total execution time. When compared to an arbitrary tree generation, use of the BFS reduced the total number of tree edges in the paths by about 30–40%. The lengths of paths and the numbering of tree edges become more important when using iterative solvers [1].

Since $\mathbf{M}$ is approximated by a constant vector in each element, geometry dependent volume integrals

$$\int \frac{\mathbf{r} - \mathbf{r}'}{\|\mathbf{r} - \mathbf{r}'\|^3} dv', \quad (8)$$

are computed only once and stored into a file. There are $N_t \times N_n$ of these integrals, where N_t and N_n are the number of tetrahedra and nodes respectively. Only the self field terms, where the field node belongs to the source element, are integrated analytically and the rest can be calculated numerically without loss of accuracy.

To date the dense matrix equation (6) has been solved using LU decomposition. Even among LU solvers there have been significant differences: the LU solver of the Numerical Recipes library [4] was found to be about ten times slower than that of LAPACK library [5] using BLAS routines tuned for the DEC Alpha 3000 workstations in which the computational tests have been done.

B. Solving of nonlinear equations

The first 5–10 nonlinear iterations are done so that χ is simply updated using the previous solution of $\mathbf{H}^1$. When a sufficiently accurate approximate solution is obtained, GFUNET switches to Newton-Raphson iterations. The total number of iterations can further be reduced by inserting an initial guess from a previously computed solution with a coarser mesh. In previous versions of GFUNET, when only a simple update procedure (with relaxation) was employed, the number of nonlinear iterations was typically about 30–50 [2],[6],[7]. The total number of nonlinear iterations is now in most cases 10–15.

C. Differences in 2D version

The structure of GFUNET-2d is the same as GFUNET-3d. In 2D, the elements are triangles, all vectors have only two components, etc. The main difference is that all 2D integrals, those corresponding to integrals (8) in 3D, are computed analytically [8].

IV. TEST RESULTS

A. General remarks

The results presented in this paper were computed on a DEC 3000-600 AXP workstation. In both 2D and 3D, the mesh generator tries to generate as regular elements as possible. It also allows the user to refine the mesh smoothly close to regions of interest [9].

The improved GFUNET-3d is typically about 30 times faster than previous versions [2],[6],[7]. There are several reasons for this: use of the Newton-Raphson iteration, the option of inserting an initial guess, a more efficient LU solver and a significantly faster path generation. The results given here were chosen to indicate typical characteristics of GFUNET:

- If the mesh is strongly refined in regions where the spatial distribution of $\mathbf{M}$ varies greatly (i.e. near corners and air gaps), moderately accurate results

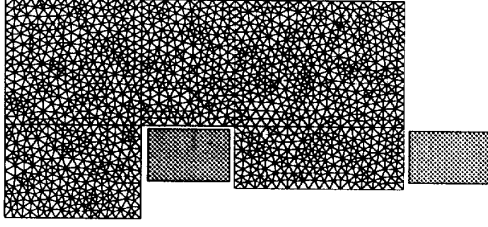


Fig. 1. One half of a 2D dipole magnet. Case 1: a uniform mesh with 1419 nodes and 2692 elements. (The width of the pole is 200 mm.)

are obtained quickly with coarse meshes, and more accurate results are available within a reasonable time.

- Even if the approximation of $\mathbf{M}$ is constant in each element, high accuracy of magnetic field in the air regions can be obtained.
- Magnetic forces can also be calculated.
- In cases with highly saturated ferromagnetic materials, $\mathbf{H}_m$ (not only $\mathbf{H} = \mathbf{H}_m + \mathbf{H}_s$) in air regions is very accurate with low number of elements.

B. 2D dipole magnet

The importance of mesh refinement is indicated by results obtained on a 2D dipole magnet, where the field homogeneity in the air gap can be judged from the harmonic coefficients. The geometry and two meshes of the magnet are shown in Fig. 1 and Fig. 2. Results from this magnet are presented in Table I; the Case 3 mesh is refined in a similar way as that of Case 2. In Table I also results obtained using the OPERA-2d PDE package [10] are presented. Using the OPERA-2d results as a reference, it can be seen that accurate magnetic fields in the air gap are computed when the mesh near the air gap and the inner corners of the magnet is significantly finer than elsewhere, as in Case 2 and 3. Case 1 shows that adding elements in other areas of the magnet does not improve the accuracy.

C. TEAM problem 20

The force acting on an object can be calculated by interpreting $\mathbf{M}$ with equivalent magnetization currents: $\mathbf{J}_m = \nabla \times \mathbf{M}$ and $\mathbf{K}_m = \mathbf{M} \times \mathbf{n}$. Since $\mathbf{M}$ is a constant vector in each element, $\mathbf{J}_m = 0$, and therefore only the surface currents $\mathbf{K}_m$ need to be taken into account. Thus the force $\mathbf{F}$ can be expressed as

$$\mathbf{F} = \int_S \mathbf{K}_m \times \mathbf{B} da, \quad (9)$$

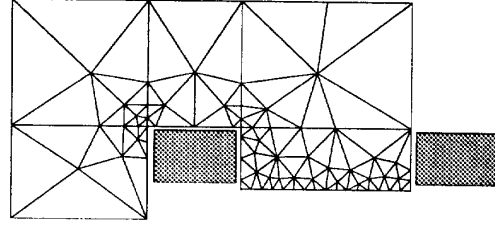


Fig. 2. One half of a 2D dipole magnet. Case 2: a refined mesh with 80 nodes and 110 elements.

TABLE I
RESULTS OF 2D DIPOLE MAGNET

Case	1	2	3
Number of nodes	1419	80	2055
Number of elements	2692	110	3854
CPU-time (s)	311	0.8	1210

Harmonic (Fourier) coefficients of B_y (T) ($\tau_0 = 20$ mm)				
n	Case 1	Case 2	Case 3	OPERA-2d
0	0.30855	0.30854	0.30828	0.30827
1	$-9.81 \cdot 10^{-5}$	$-1.01 \cdot 10^{-4}$	$-1.53 \cdot 10^{-5}$	$-1.34 \cdot 10^{-5}$
2	$1.69 \cdot 10^{-5}$	$1.60 \cdot 10^{-5}$	$-4.75 \cdot 10^{-5}$	$-4.97 \cdot 10^{-5}$
3	$-9.86 \cdot 10^{-7}$	$-4.20 \cdot 10^{-7}$	$2.09 \cdot 10^{-7}$	$3.12 \cdot 10^{-7}$
4	$-7.23 \cdot 10^{-6}$	$-7.25 \cdot 10^{-6}$	$-1.03 \cdot 10^{-5}$	$-1.00 \cdot 10^{-5}$
5	$1.00 \cdot 10^{-7}$	$-1.63 \cdot 10^{-7}$	$2.10 \cdot 10^{-8}$	$-9.70 \cdot 10^{-9}$
6	$-8.65 \cdot 10^{-7}$	$-8.02 \cdot 10^{-7}$	$-1.11 \cdot 10^{-6}$	$-1.10 \cdot 10^{-6}$
7	$5.39 \cdot 10^{-8}$	$-1.71 \cdot 10^{-7}$	$1.10 \cdot 10^{-9}$	$1.01 \cdot 10^{-9}$

where the surface S contains all facets in the object.

Table II contains some GFUNET-3d results for TEAM problem 20 [11]. The meshes are refined near the air gaps and at corners of the steel parts. Measured values are also shown in Table II. It can be seen that even with the coarse mesh of Case 1 the computed magnetic field is quite accurate, although the force is more inaccurate. Increasing the number of elements increases the accuracy of force computation. However, the computed force seems to converge few per cent above the measured value.

Table III has timing results of the main parts of GFUNET for meshes of TEAM problem 20. In small problems the integration of the RHS of (6) is the most time consuming part. When the problem size increases, the matrix generation and the LU solver start to dominate.

TABLE II
RESULTS OF TEAM PROBLEM 20 (1000 AT)

Case	1	2	3	measured
Nodes	155	1013	3986	
Elements	366	3815	17953	
$B_z(P_1)$ (T)	0.311	0.328	0.323	0.36
$B_z(P_2)$ (T)	0.258	0.272	0.256	0.24
$(B_z)_{ave}(\alpha - \beta)$ (T)	0.783	0.746	0.734	0.72
$(B_z)_{ave}(\gamma - \delta)$ (T)	0.153	0.142	0.138	0.13
F_z (N)	9.31	8.63	8.53	8.1

TABLE III
TIMING OF THE MAIN PARTS OF GFUNET
(TEAM PROBLEM 20 (1000 AT) WITH THREE MESHES)

Case	1	2	3
Number of nodes	155	1013	3986
Number of elements	366	3815	17953
Number of equations	153	1011	3984
Tree generation (s)	0.01	0.3	3.7
Path generation (s)	0.001	0.008	0.03
RHS generation (s)	8.1	52.8	220
Matrix generation (s)	3.2	190	3500
Matrix update (s)	0.06	8.5	207
Jacobian generation (s)	0.1	8.0	322
LU solver (s)	0.05	13.9	438
Number of iterations	16	12	11
Total CPU-time (s)	13.6	491	11367

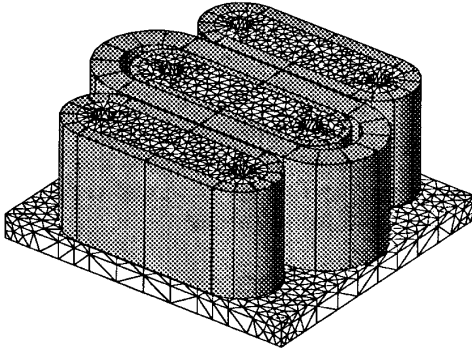


Fig. 3. One half of a high field wiggler magnet. Mesh of Case 2 with 2122 nodes and 8802 elements (in one-eighth of the magnet).

D. High field wiggler magnet

The last example is a high field wiggler magnet designed and built at the Tampere University of Technology [12] demonstrating that accurate results can be obtained in a highly saturated case. One half of the magnet is presented in Fig. 3, even if fields need only be computed in one-eighth of the magnet. In Fig. 4 the measurements of the central field are compared to computed values for two meshes: Case 1 had 929 nodes and 3831 elements and Case 2 had 2122 nodes and 8802 elements. The CPU-times for Cases 1 and 2 were 757 s and 3429 s respectively. Due to complex geometry of the magnet – there are holes in the poles – results can not easily be computed for meshes with less elements than in Case 1.

V. CONCLUSION

By using the h -type volume integral formulation presented here, approximate solutions to magnetostatic problems can be obtained on a powerful workstation in under a minute, and more accurate results are available within

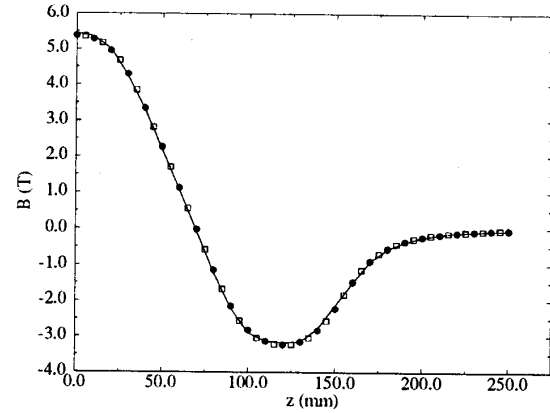


Fig. 4. The central field of the wiggler magnet. Case 1: solid circles, Case 2: squares, measured: solid line.

a reasonable time (from several minutes to a few hours) assuming meshes are properly refined. The results presented clearly indicate that the method is competitive.

REFERENCES

- [1] L. Kettunen, K. Forsman, D. Levine, and W. Gropp, "Integral equations and nonlinear 3D magnetostatics," submitted to Int. J. Numer. Methods Eng., 1994.
- [2] L. Kettunen and L. Turner, "A volume integral formulation for nonlinear magnetostatics and eddy currents using edge elements," *IEEE Trans. Magn.*, vol. 28, pp. 1639–1642, 1992.
- [3] R. Albanese and G. Rubinacci, "Integral formulation for 3D eddy-current computation using edge elements," *IEE Proc.*, vol. 135, Pt. A, no. 7, pp. 457–462, 1988.
- [4] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes, The Art of Scientific Computing*. Cambridge University Press, 1986.
- [5] E. Anderson et al., *LAPACK Users's Guide*. SIAM, 1992.
- [6] L. Kettunen, K. Forsman, D. Levine, and W. Gropp, "Solutions of TEAM problem #13 using integral equations in a sequential and parallel computing environment," in *Proc. of the Fourth Int. TEAM Workshop*, (Miami, U.S.A.), pp. 41–43, 1994.
- [7] K. Forsman and L. Kettunen, "Solutions of TEAM problem #20 using integral equations," in *Proc. of the Fourth Int. TEAM Workshop*, (Miami, U.S.A.), pp. 109–111, 1994.
- [8] R. Beth, "Complex representation and computation of two-dimensional magnetic fields," *J. of Applied Physics*, vol. 37, no. 7, pp. 2568–2571, 1966.
- [9] K. Forsman and L. Kettunen, "Tetrahedral mesh generation in convex primitives by maximizing solid angles," *IEEE Trans. Magn.*, vol. 30, pp. 3535–3538, September 1994.
- [10] Vector Fields Ltd., 24 Bankside, Kidlington, Oxford, U.K.
- [11] T. Nakata, N. Takahashi, and H. Morishige, "Analysis of 3-d static force problem," in *Proc. of TEAM Workshop on Computation of Applied Electromagnetics in Materials*, (Sapporo, Japan), 1993.
- [12] J.T. Eriksson et al., "Stability and training of a high field superconducting wiggler," *IEEE Trans. Appl. Superconduct.*, vol. 3, no. 1, pp. 817–820, 1993.