Available online at www.sciencedirect.com
ScienceDirect
Physica B 403 (2008) 368–371
PHYSICA B
www.elsevier.com/locate/physb

Magnetostatic solution by hybrid technique and fast multipole method

G. Gruosso[a], M. Repetto[b,*]

[a]Politecnico di Milano, Dipartimento di Elettronica e Informazione, I-20133 Milano, Italy
[b]Politecnico di Torino, Dipartimento di Ingegneria Elettrica, C.so Duca Abruzzi 24, I-10129 Torino, Italy

Abstract

The use of fast multipole method (FMM) in the solution of a magnetostatic problem is presented. The magnetostatic solution strategy is based on finite formulation of electromagnetic field coupled with an integral formulation for the definition of boundary conditions on the external surface of the unstructured mesh.

Due to the hypothesis of micromagnetic problem, the resulting matrix structure is sparse and integral terms are only on the RHS. Magnetic surface charge is used as source of these integral terms and is localized on the faces between tetrahedra. The computation of the integral terms can be performed by analytical formulas for the near field contributes and by FMM for far field ones.

Keywords: Micromagnetics; Fast multipole method; Cell method; Hybrid techniques

1. Hybrid magnetostatic solution

The use of an hybrid form of magnetostatic computation based on finite formulation of electromagnetic fields (FFEF), more often called cell method, and on a Green function applied to magnetization sources, has been presented [1,2]. In this formulation an integral boundary condition on the exterior surface of the mesh is related to the magnetization sources. Integral boundary conditions are used to terminate Ampere's law for all boundary edges expressing external magnetic field circulation as a potential difference evaluated by the Green function integrals at the center of each boundary face. Under the micromagnetic hypothesis, the magnetized region has permeability μ_0 and, for each time step of the dynamic analysis, magnetization distribution is supposed to be known.

The absence of magnetic polarization, that is of a magnetization-dependent field like $\mathbf{M} = \chi\mathbf{H}$, where χ is the magnetic susceptibility, eliminates magnetization contribution depending on configuration variables like magnetic vector potential. Thus all integral terms are known and can be carried on to the RHS. Another consequence of this fact is that there are no integral relations between matrix rows and thus matrix structure remains sparse as in the usual formulation of the cell method.

This numerical approach has been already applied to demagnetizing field computation on a structured hexahedral grid [3] and is now applied to a tetrahedral mesh. Magnetization is considered to be uniform in each tetrahedron so that magnetic sources are located only on triangular faces. The computation of magnetization-dependent integrals is performed by fully analytical formulas, nevertheless this phase is one of the most time-consuming parts of each magnetostatic solution. From these considerations stem the idea of using fast multipole method (FMM) to speed up the integral boundary conditions computation.

2. Fast multipole scheme for RHS computation

2.1. Multipole expansion

Magnetic scalar potential is computed by an integral operator applied to magnetization-dependent sources. The scalar potential ψ is expressed by

$$\psi(\mathbf{r}) = \int_{\Omega} \rho_{\mathrm{M}} G(\mathbf{r}, \mathbf{r}') \, \mathrm{d}\Omega_{\mathrm{M}}, \tag{1}$$

*Corresponding author. Tel.: +39 011 5647140.
E-mail address: maurizio.repetto@polito.it (M. Repetto).

0921-4526/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.physb.2007.08.051

where $G = 1/(4\pi|\mathbf{r}-\mathbf{r}'|)$ is the usual Green's function of three-dimensional static problems and ρ_M is the magnetic charge density obtained as $\rho_M = -\nabla\mathbf{M} - \nabla_S\mathbf{M}$, and Ω_M is the volume domain containing the magnetization. This term can be evaluated by considering two contributions to the potential: one due to near field and one due to the far field $\psi = \psi_{near} + \psi_{far}$. The first term [2] can be calculated by means of Eq. (1), while the last one can be obtained by resorting to an expansion in multipoles [4]:

$$\psi_{far}(r,\vartheta,\varphi) = \sum_{l=0}^{l}\sum_{m=-l}^{l}\frac{M_l^m}{r^{n+1}}Y_l^m(\vartheta,\varphi), \tag{2}$$

with

$$M_l^m = \sum_{i=1}^{N} q_i r_i^l Y_l^{-m}(\vartheta_i,\varphi_i), \tag{3}$$

where r_i, ϑ_i, and φ_i are spherical coordinates of ith charge $q_i = \sigma_i A_i$, and r, ϑ, and φ are the coordinates of field point. Y_l^m are the normalized spherical harmonics of degree l and order m given by the following:

$$Y_l^m(\theta,\varphi) = \sqrt{\frac{2l+1}{4\pi}\frac{(l-m)!}{(l+m)!}}P_l^m(\cos\vartheta)e^{jm\varphi}, \tag{4}$$

with the condition

$$Y_l^{-m}(\theta,\varphi) = (-1)^m Y_l^{m*}(\theta,\varphi), \tag{5}$$

where $P_l^m(\cos\vartheta)$ are the Legendre polynomials of degree l and order m [4].

2.2. O(N log(N)) fast multipole scheme

Several fast multipole schemes have been developed in order to increase the computational efficiency of the method [5–8]. All these methods are based on the same basic principle: in order to make systematic use of multipole expansions, a hierarchy of boxes which refines the computational domain into smaller and smaller regions will be defined. At refinement level 0, the whole computational domain will be considered. Refinement level $k+1$ is obtained recursively from level k by subdivision of each box into eight equal parts. This yields a natural tree structure, where the eight boxes at the $k+1$ level, obtained by subdivision of a box at level k, are considered its children.

By starting from the generated tree it is possible, for all sources which are "sufficiently far" from a given field point, using the multipole expansion, to summarize their total interaction. According to error estimation, a convenient definition of "sufficiently far" (or "well separated") is a relationship between the diagonal of each cube and the distance of field point from the center of the considered cube: $r' < 2\sqrt{(3)}w$, where r' is the distance of field point from the local origin and w the length of the edge of the cube. The first algorithm consists in evaluating the far field at all field points by considering the contribution of each cube well separated from them. The cost of this algorithm is approximately $N_f \log N_s$, where N_f is the number of evaluation points and N_s the number of the sources. Generally this approach is not enough to improve the computational efficiency and multilevel algorithms are introduced [7]; in the present formulation, the number of field points, located only on the surface of the active volumes, is much lesser than the number of sources located in the whole active volumes, therefore only the $O(N \log N)$ algorithm has been considered.

2.3. Data structures

In this section a data structure based on octree [8] subdivision of the domain combined with a *bit interleaving* techniques [9] will be described. This technique allows an efficient indexing of the cube in the hierarchical tree and then a robust and fast method to find relationship between cubes at each level of the octree. Briefly some characteristic aspects of the method will be outlined: considering a box at level l the whole number of children of this box is 8, numbered from 0 to 7, therefore the index of one of these boxes is unique and it is given by the following relationship:

$$\mathrm{IDX}(n,l) = (N_1, N_2, \ldots, N_j), \tag{6}$$

where l is the considered level, N_j the index of the box at level j given by $N_j = 0, \ldots, 7$ with $j = 1, \ldots, l$ and n is the global number of the cubes at level l with $n = N_1(8)^{(l-1)} + N_2(8)^{(l-2)} + \cdots + N_{l-1}(8) + N_l$. For example, by referring to the construction shown in Fig. 1,the cube $N_1 = 0$ and $N_2 = 5$ at level 2 is the cube $n = 5$ of the level 2 and so on; in this way the uniqueness of the indexing is guaranteed. In this way it is possible to obtain the father box (PRT) for each pair $(n,l)$by means of

$$\mathrm{PRT}(n,l-1) = (N_1, N_2, \ldots, N_{j'}), \tag{7}$$

with $j' = 1, \ldots, l-1$. From the previous statement it is possible to compute all the children (CHD) at level $l+1$:

$$\mathrm{CHD}(n,l+1) = (N_1, N_2, \ldots, N_{j+1}), \tag{8}$$

where $N_{j+1} = 0, \ldots, 7$. In other words the use of octree makes obtaining parent and children indices very convenient. Indeed the above operations are nothing but shift operations in the bit representation of n. Performing a right

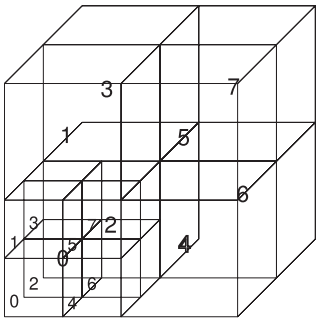

Fig. 1. Octree subdivision of the domain and cube hierarchy.