

# Generalized Finite-Element Method for Magnetized Nanoparticles

Alexander Plaks, Igor Tsukerman, *Senior Member, IEEE*, Gary Friedman, and Benjamin Yellen

**Abstract**—The generalized finite-element method is applied to model self-assembly of magnetized nanoparticles. Only a regular hexahedral grid is used. The particles need not be meshed and are represented by additional basis functions approximating derivative jumps of the potential at particle boundaries.

**Index Terms**—Generalized finite-element method (GFEM), magnetized particles, nanotechnology, partition of unity (PU).

## I. INTRODUCTION

MULTIPARTICLE self-assembly processes are of great interest in a number of nanotechnology applications, such as microarrays for high-resolution chemical or biological sensing and DNA sequencing, photonic bandgap materials, novel magnetic storage media. As proposed by one of the authors, self-assembly of nanoparticles can be guided by magnetized templates on a substrate [1]. This approach has several advantages: magnetic forces are relatively long-range, easily tuneable, and do not require any permanent supply of power.

This paper presents a novel computational technique based on the generalized finite-element method (GFEM) by partition of unity (PU) developed in [2] and [3] (see also [4]–[6]) and previously applied by one of the authors [7]–[9] to electromagnetic problems.

For multiple particles with varying positions, structured finite-element (FE) meshes would be impractical. GFEM, unlike the standard FEM, does not necessarily require structured meshes and can employ nonpolynomial basis functions to approximate derivative jumps at material interfaces. Particles need not even be meshed and can be simply “pasted up” on top of a regular background grid representing the substrate and air regions. The PU algorithm seamlessly merges the field inside and outside the particles.

## II. GFEM

The GFEM-PU method has been described in [2]–[9] and is outlined here very briefly for completeness. A system of functions  $\{\varphi_i\}$ ,  $1 \leq i \leq n_{\text{patches}}$  forms a *partition of unity* over a

set of overlapping “patches” (subdomains) covering the computational domain  $\Omega$  if

$$\sum_{i=1}^{n_{\text{patches}}} \varphi_i \equiv 1 \quad \text{in } \Omega, \quad \text{supp}(\varphi_i) = \Omega_i. \quad (1)$$

To construct a partition of unity in practice, one starts with a set of functions  $\tilde{\varphi}_i$  satisfying the second part of condition (1),  $\text{supp}(\tilde{\varphi}_i) = \Omega_i$ , and then normalizes these functions to satisfy the first part of (1) [2]–[8]

$$\varphi_i = \frac{\tilde{\varphi}_i}{\sum_{k=1}^{n_{\text{patches}}} \tilde{\varphi}_k}. \quad (2)$$

If solution  $u$  is approximated *locally*, i.e., over each patch  $\Omega_i$ , by a certain function  $u_{hi}$ , these local approximations can be merged using the  $\varphi$ 's as functional weights

$$u = \sum_{i=1}^{n_{\text{patches}}} u \varphi_i \quad u_h = \sum_{i=1}^{n_{\text{patches}}} u_{hi} \varphi_i. \quad (3)$$

This way of merging local approximations is the central point of the whole computational procedure, as it ensures that the global approximation accuracy over the whole domain is commensurate with the local accuracy over each subdomain.

It is this guarantee of the approximation accuracy that distinguishes the PU-based GFEM from various other “meshless” methods in different engineering applications [10] and electromagnetics [11], [12]. Indeed, if (3) is applied, the approximation error over the whole computational domain  $\Omega$  can be estimated as (see [2] and [3] for further details)

$$\begin{aligned} \varepsilon_a &\equiv \|\nabla(u_h - u)\|_{L_2(\Omega)} \\ &= \left\| \sum_{i=1}^{n_{\text{patches}}} \nabla[(u_{hi} - u)\varphi_i] \right\|_{L_2(\Omega)} \\ &\leq \sum_{i=1}^{n_{\text{patches}}} \|\nabla[u_{hi} - u]\|_{L_2(\Omega_i)} \|\varphi_i\|_{\infty} \\ &\quad + \sum_{i=1}^{n_{\text{patches}}} \|u_{hi} - u\|_{L_2(\Omega_i)} \|\nabla\varphi_i\|_{\infty}. \end{aligned} \quad (4)$$

Thus, under the assumption that the above norms of  $\varphi_i$  are bounded, the global approximation essentially reduces, due to the partition of unity, to local approximations over each patch. Please see [2]–[6] for rigorous mathematical analysis.

Locally, within each patch  $\Omega_i$ , the numerical solution  $u_{hi}$  is sought as a linear combination of some approximating functions

Manuscript received June 18, 2002. This work was supported in part by the National Science Foundation and by the Defense Advanced Research Projects Agency.

A. Plaks is with the PlasoTech Corporation, Encino, CA 91436 USA (e-mail: alex@plassotech.com).

I. Tsukerman is with the Department of Electrical and Computer Engineering, University of Akron, Akron, OH 44325-3904 USA (e-mail: igor@uakron.edu).

G. Friedman and B. Yellen are with the Electrical and Computer Engineering Department, Drexel University, Philadelphia, PA 19104 USA (e-mail: gary@ece.drexel.edu; Benjamin.Biron.Yellen@drexel.edu).

Digital Object Identifier 10.1109/TMAG.2003.810408

$$g_m^{(i)} \quad u_{hi} = \sum_m a_m^{(i)} g_m^{(i)}. \quad (5)$$

The final system of approximating functions  $\{\psi_m^{(i)}\}$  is built using elements of the partition of unity  $\varphi_i$  as weight functions

$$\psi_m^{(i)} = g_m^{(i)} \varphi_i. \quad (6)$$

The global approximation  $u_h$  is a linear combination of the  $\psi$ 's

$$u_h = \sum_{m,i} c_m^{(i)} \psi_m^{(i)} \quad (7)$$

and the standard Galerkin procedure can be applied.

### III. APPLICATION OF GFEM TO MULTIPARTICLE PROBLEMS

#### A. Brief Description of a Generic Problem

A class of problems under consideration consists in finding the distribution of the magnetic field around a system of magnetic particles being deposited on a magnetized substrate. The standard FE analysis in this case faces serious difficulties. For multiple spherical or spheroidal particles in a volume, FE meshes would be geometrically very complex and difficult to construct, and it would be almost impossible to avoid “flat” elements with large approximation errors [13], [14]. Moreover, changing positions of the particles would lead to mesh distortion and related numerical artifacts, especially in force computation [16], [17].

Fast multipole methods (FMMs) (see [18] and references therein) may be suitable for this type of problems, although the fact that the magnetic moments of particles are not known *a priori* and that the substrate is a continuum may complicate the matter. We focus on GFEM as a promising alternative, as it allows for any approximating functions in the particles and, in the future, for representation of other types of physical interactions (van der Waals, Derjaguin–Landau–Verwey–Overbeek, etc.)

#### B. Options Available for GFEM-PU

In GFEM, it is natural to consider a very simple, uniform background mesh, with the particles “pasted up” on top of it as separate subdomains (“patches”). For brick elements in the background mesh, each patch can be chosen to contain the eight elements sharing a common node, and the shape function  $\varphi$  for this patch could be piecewise-trilinear.

Thus the most obvious set of overlapping patches for the problem would include the bricks and, in addition, the particles. However, this approach is not satisfactory. Estimate (4) shows that global approximation is good *provided that all* patch-wise approximations are good. Within each particle, approximation of the field can indeed be made as accurate as needed by a suitable choice of the discrete approximation space. But for the brick patches of the background mesh the situation is not as simple: the standard piecewise-polynomial space would not suffice because of the derivative jumps of the potential at particle boundaries.

This leads to a somewhat different approach. The set of patches does *not* include the particles and consists only of the

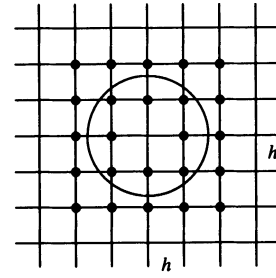


Fig. 1. Additional approximating functions on the particle boundary.

brick patches in the background mesh. Each particle contributes not a patch but rather a system of approximating functions  $g_m^{(i)}$  that, as described in Section II, get multiplied by the brick-element shape functions  $\varphi$  as in (6). It is essential that the local approximating functions  $g_m^{(i)}$  include derivative jumps on particle boundaries, as explained in more detail below. This approach is similar to the treatment of singularities in GFEM (e.g., [9]), except that the “field jump” functions rather than singular approximating functions are added to the basis.

#### C. Details of the Algorithm

For simplicity but without any essential loss of generality, consider spherical particles in the unit cube with a hexahedral mesh of size  $h = h_x = h_y = h_z$ . Each patch corresponds to a mesh node and contains eight elements sharing that node (patches at the boundary may contain fewer elements). The patch shape function is piecewise-trilinear, as in the standard FEM:  $\varphi_i(x, y, z) = (1 - |\xi|)(1 - |\eta|)(1 - |\zeta|)$ , where  $\xi = (x - x_i)/h$ ,  $\eta = (y - y_i)/h$ ,  $\zeta = (z - z_i)/h$ , and  $x_i, y_i, z_i$  are the coordinates of the node.

The local approximating functions  $g_m^{(i)}$  contain two separate subsets, the first one due to the background brick mesh and the second one due to the particles. It is the second subset that accounts for the field jump on the particle boundary. In brick patches that intersect the boundary, these additional approximating functions are, in the simplest case

$$g_j^p = \begin{cases} 0, & \text{if } R \leq R_j \\ R - R_j, & \text{if } R > R_j \end{cases} \quad (8)$$

where  $R_j$  is the radius of the  $j$ th particle, and  $R$  is the distance from the particle center. Alternatively, the second subset of  $g_m^{(i)}$  can be constructed by a coordinate mapping [20], as explained in [21]. In any event, the additional approximation functions are introduced in all patches centered on the black dots in Fig. 1.

The stiffness matrix is assembled in the usual way, by integrating  $\nabla\psi_i \cdot \nabla\psi_j$  over each element and summing up the individual element contributions. The integrals involving trilinear functions are standard and can be found analytically. Integrals containing the jump functions (8) require numerical quadratures because of the shape complexity (Fig. 2).

For numerical integration, we subdivide each of the hexahedral elements into  $m \times m \times m$  subcubes (typically, with  $m = 4$ , as in Fig. 2) and apply the  $3 \times 3 \times 3$  Gaussian quadrature in each of these subcubes. The same procedure is used for the computation of the right-hand side: the “hexahedral” contributions are integrated analytically and the “spherical” ones—numerically.

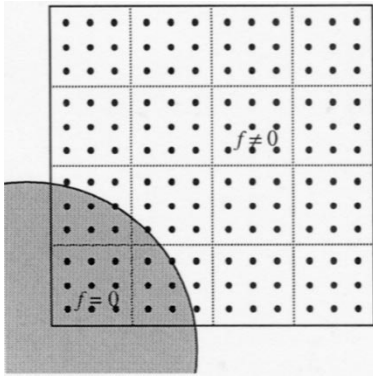


Fig. 2. Integration points for numerical quadratures.

This integration scheme has been adopted for now due to its relative simplicity; better alternatives may exist [4]–[6].

Boundary conditions for the trilinear functions are treated in exactly the same way as in the standard FEM. Since the domain boundary has to be moved sufficiently far away from the particles to avoid truncation errors, boundary conditions do not affect the treatment of the additional jump functions.

#### IV. PRACTICAL APPLICATIONS AND EXPERIMENTAL SETUP

One important application where the motion and positioning of magnetic particles needs to be controlled is self-assembly of particles into biochemical micro-arrays. The assembly process considered in this paper is driven by magnetic templates prepatterned on the substrate [1].

There is growing interest in fabrication of high-resolution DNA and other biochemical arrays for purposes ranging from biochemical sensing to gene sequencing. Application of such arrays to sensing is based on the following principle. Different biochemical agents in the array act as different molecular probes. By locating each molecular probe in a different part of the array substrate, the position of each probe essentially encodes the probe type. The resulting array can be called a position-encoded biochemical array. When binding events between different molecular probes in the array with molecules of an analyte occur, they can be usually detected by fluorescent labels attached to the analyte molecules. This permits identification of the analyte molecules by simply observing the set of positions on the array substrate emitting the fluorescent signal.

Currently, the two main methods of fabricating position-encoded arrays are direct synthesis of array elements on a substrate using sequential masking techniques and printing of presynthesized molecules onto different array spots. These methods are producing biochemical arrays with a number of elements ranging from several hundred to several thousand.

While both types of array fabrication technologies are in commercial use and have yielded a large amount of information, significant challenges remain. One of the main challenges for directly synthesized micro-arrays is their reliability and cost for longer molecular sequences. For this reason, printing of presynthesized chemical labels is sometimes preferred. However, the spot size in mechanically printed arrays is difficult to reduce below 100 square micrometers. Reducing the physical size of the array would permit to decrease the volume of the analyte

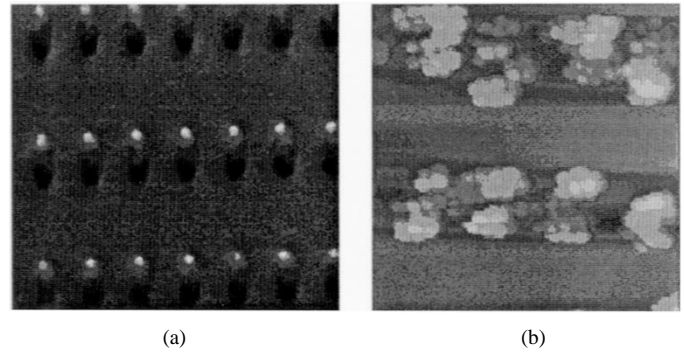


Fig. 3. (a) Magnetic force microscopy image of single-domain 500-nm-long and 50-nm-thick NiFe islands used to precipitate superparamagnetic nanoparticles. (b) Atomic force microscopy image of superparamagnetic nanoparticles (137 nm in diameter, produced at the Chemistry Department, University of Pittsburgh by S. Asher) aggregating on top of a two-dimensional array of NiFe islands shown in (a).

and to cut the cost and time of the analysis. Another problem with mechanical printing is variability in the number and distribution of molecular probes over the substrate.

The use of nano-particles as carriers of different molecular probes for the array can improve the uniformity of biochemical distribution over the array substrate and reduce the spot size. We are developing a magnetographic printing process whereby superparamagnetic beads (particles) acting as carriers of different molecular probes can be accurately printed onto different areas on the substrate by using magnetized templates.

In the current experimental setup, magnetized templates are formed by lithographic patterning of thin (50 nm) Cobalt film into islands of various shapes and sizes. The beads are placed into deionized water together with the magnetically patterned substrate, and the solution is mildly agitated. An example of the resulting decoration of the substrate by the superparamagnetic beads is shown in Fig. 3.

#### V. NUMERICAL RESULTS

The magnetostatic field is computed by GFEM-PU, as described in the theoretical sections above. The infinite domain is truncated and homogeneous Dirichlet conditions imposed on the boundary. The size of the cubic domain is normalized to unity. Three spherical particles, each with the normalized radius  $r = 0.05$  (Fig. 4), are located at  $(0.3, 0.65, 0.5)$ ,  $(0.5, 0.65, 0.5)$ , and  $(0.7, 0.65, 0.5)$ . The magnetized island occupies the region  $0.4 \leq x, z \leq 0.6$ ,  $0.45 \leq y \leq 0.5$  and is magnetized along the  $x$  axis.

The magnetostatic equation is well known

$$\nabla \cdot \mu \nabla u = \nabla \cdot \mu \mathbf{M} \quad (9)$$

where  $\mathbf{M}$  is a given magnetization of the island. The weak formulation of (9) is also standard

$$(\mu \nabla u, \nabla u') = (\mu \mathbf{M}, \nabla u') \quad \forall u' \in H_0^1(\Omega)$$

where parentheses stand for the inner product in  $L_2(\Omega)^3$ .

The numerical solution is sought in the discrete space spanned by the  $\psi$  functions as in (7). A regular hexahedral three-dimensional mesh with  $41 \times 41 \times 41$  nodes was used. The overall CPU time, including matrix assembly and the

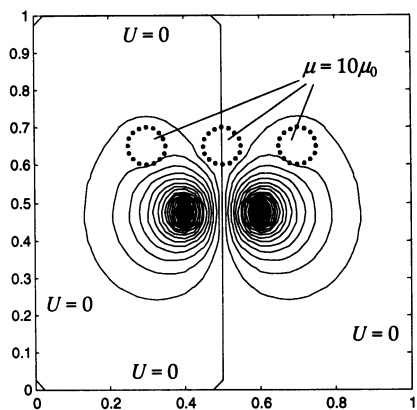


Fig. 4. Lines of equal scalar potential in the two-dimensional cross section for the sample multiparticle problem at  $z = 0.5$ .

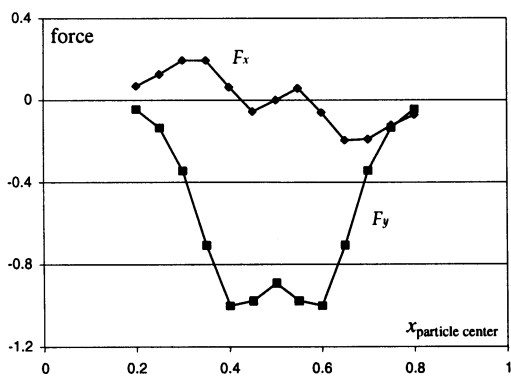


Fig. 5. Forces on a particle as functions of position.

linear system solver, was about 4 min on a 1.6-GHz Pentium. For the time being, only a simple ICCG solver was employed; more advanced methods, such as multigrid, should reduce the solution time drastically (see [19], as well as references therein, for the authors' implementation of multilevel preconditioners for related problems).

Forces acting on the particles were computed using the Maxwell stress tensor. Normalized forces as functions of the particle position along the  $x$  axis are plotted in Fig. 5. The local minimum of  $F_x$  at  $x = 0.45$  and the local maximum at  $x = 0.55$  show that the particle is attracted to the edges of the magnetized island.

In standard FEM, force computation is a notoriously difficult problem [16], [17]. For multiple particles with changing positions, standard FE analysis of forces would be prone to large numerical errors due to mesh distortion. In GFEM-PU, there is no distortion; the motion of particles is reflected only in the changing definition of the additional approximating functions (8). This is similar to the case of motion with fixed grids [15], [16], where high accuracy of force computation has been validated both theoretically and numerically [16]. It is hoped that further studies will confirm the high accuracy of GFEM in force calculation.

## VI. CONCLUSION

The feasibility of the GFEM-PU method for multiparticle problems in nanotechnology applications has been demon-

strated. Only very simple regular meshes are needed and no mesh distortion occurs when particles change their positions. Implementation difficulties are shifted from generation of complex FE meshes and minimization of mesh distortion to numerical integration.

GFEM-PU has been applied to several sample multiparticle problems. Its advantages and limitations, especially implementation details and the accuracy of force computation, need to be thoroughly evaluated in future studies. Cross validation of numerical versus experimental results is another major challenge for future work.

## REFERENCES

- [1] B. Yellen, G. Friedman, and A. Feinerman, "Analysis of interactions for magnetic particles assembling on magnetic templates," *J. Appl. Phys.*, vol. 91, no. 10, pp. 8552–8554, 2002.
- [2] J. M. Melenk and I. Babuška, "The partition of unity finite element method: Basic theory and applications," *Comput. Methods Appl. Mech. Eng.*, vol. 139, pp. 289–314, 1996.
- [3] I. Babuška and J. M. Melenk, "The partition of unity method," *Int. J. Numer. Methods Eng.*, vol. 40, no. 4, pp. 727–758, 1997.
- [4] C. A. Duarte, I. Babuška, and J. T. Oden, "Generalized finite element methods for three-dimensional structural mechanics problems," *Comput. Struct.*, vol. 77, no. 2, pp. 215–232, 2000.
- [5] T. Strouboulis, I. Babuška, and K. L. Copps, "The design and analysis of the generalized finite element method," *Comput. Methods Appl. Mech. Eng.*, vol. 181, no. 1–3, pp. 43–69, 2000.
- [6] T. Strouboulis, K. Copps, and I. Babuška, "The generalized finite element method," *Comput. Methods Appl. Mech. Eng.*, vol. 190, no. 32–33, pp. 4081–4193, 2001.
- [7] L. Proekt and I. Tsukerman, "Method of overlapping patches for electromagnetic computation," *IEEE Trans. Magn.*, vol. 38, pp. 741–744, Mar. 2002.
- [8] I. Tsukerman and L. B. Proekt, "Generalized scalar and vector elements for electromagnetic computation," in *Proc. 11th Int. Symp. Theoretical Electrical Engineering*, Linz, Austria, Aug. 2001.
- [9] L. Proekt, S. Yuferev, I. Tsukerman, and N. Ida, "Method of overlapping patches for electromagnetic computation near imperfectly conducting cusps and edges," *IEEE Trans. Magn.*, vol. 38, pp. 649–652, Mar. 2002.
- [10] T. Belytschko, Y. Krongauz, D. Organ, M. Fleming, and P. Krysl, "Meshless methods: An overview and recent developments," *Comput. Methods Appl. Mech. Eng.*, vol. 139, no. 1–4, pp. 3–47, 1996.
- [11] Y. Maréchal, "Some meshless methods for electromagnetic field computations," *IEEE Trans. Magn.*, vol. 34, pp. 3351–3354, Sept. 1998.
- [12] Y. Maréchal, J. L. Coulomb, G. Meunier, and G. Touzot, "Use of the diffuse approximation method for electromagnetic field computation," *IEEE Trans. Magn.*, vol. 30, pp. 3558–3561, Sept. 1994.
- [13] I. A. Tsukerman, "Approximation of conservative fields and the element 'edge shape matrix'," *IEEE Trans. Magn.*, vol. 34, pp. 3248–3251, Sept. 1998.
- [14] I. A. Tsukerman and A. Plaks, "Comparison of accuracy criteria for approximation of conservative fields on tetrahedra," *IEEE Trans. Magn.*, vol. 34, pp. 3252–3255, Sept. 1998.
- [15] I. A. Tsukerman, "Overlapping finite elements for problems with movement," *IEEE Trans. Magn.*, vol. 28, pp. 2247–2249, Sept. 1992.
- [16] —, "Accurate computation of 'ripple solutions' on moving finite element meshes," *IEEE Trans. Magn.*, vol. 31, pp. 1472–1475, May 1995.
- [17] W. Müller, "Comparison of different methods of force calculation," *IEEE Trans. Magn.*, vol. 26, pp. 1058–1061, Mar. 1990.
- [18] L. Greengard, V. Rokhlin, and H. Cheng, "A fast adaptive multipole algorithm in three dimensions," *J. Comput. Phys.*, vol. 155, no. 2, pp. 468–498, 1999.
- [19] I. Tsukerman and A. Plaks, "Hierarchical basis multilevel preconditioners for 3-D magnetostatic problems," *IEEE Trans. Magn.*, vol. 35, pp. 1143–1146, May 1999.
- [20] I. Babuška, G. Caloz, and J. E. Osborn, "Special finite-element methods for a class of 2nd-order elliptic problems with rough coefficients," *SIAM J. Num. Anal.*, vol. 31, no. 4, pp. 945–981, 1994.
- [21] I. Tsukerman, "Finite element difference schemes for electro- and magnetostatics," in *Proc. Compumag*, 2003, to be published.