Point Interpolation Methods based on Weakened-Weak Formulations

Naísses Z. Lima
Graduate Program in Electrical Engineering - Federal University of Minas Gerais
Av. Antônio Carlos 6627, 31270-901, Belo Horizonte, MG, Brazil - naisseszoia@gmail.com
Renato C. Mesquita
Department of Electrical Engineering - Federal University of Minas Gerais
Av. Antônio Carlos 6627, 31270-010, Belo Horizonte, MG, Brazil - renato@ufmg.br

***Abstract*— This paper presents a study of meshless Point Interpolation Methods based on weakened-weak forms. The mathematical formulations of the methods are presented as well as the procedures for the support nodes selection called T-schemes. The numerical results are shown for four different types of electromagnetic static problems in order to ponctuate the characteristics of the approximation generated by these new methods.**

***Index Terms*—** Meshless methods, point interpolation method, smoothed gradient, weakened-weak form.

I. Introduction

Meshless methods are numerical techniques to solve boundary value problems. They were developed with the goal of eliminating the need of mesh generation, working only with nodes without a prescribed connectivity among them. [1]

Among the meshless methods developed so far, we highlight the Element-free Galerkin (EFG) [2], the Meshless Local Petrov-Galerkin (MLPG) [3] and the Point Interpolation Methods (PIM) [1]. These methods work with the problem mathematical formulation in its weak form. While EFG uses a global weak form, MLPG uses a local weak form and PIM works with both global and local ones. As weak forms present integral terms, an integration process must be carried out. Methods based on global formulations perform the integration of the weak form through a background mesh which is independent of the procedure for construction of the approximation functions (shape functions). On the other hand, methods based on local formulations execute the integration using local subdomains and no mesh is used, therefore they are called trully meshless methods.

In meshless methods, the most used procedures for shape function generation are the Moving Least Squares (MLS) and the Point Interpolation Method (PIM) [1]. The MLS approximation has a great feature that the approximated field function is continuous and smooth in the entire domain with the desired order of consistency, however it does not possess the Kronecker delta property. Unlike MLS, PIM produces approximations that have the Kronecker delta property, but that are nonconforming, i.e., they are discontinuous in some regions of the domain. The Kronecker delta property is interesting

received 21 Oct 2012; for review 7 Dec 2012; accepted 15 July 2013

ISSN 2179-1074

once the essential boundary conditions are naturally enforced, otherwise explicit methods must be used such as the Lagrange multipliers method and the penalty method, which bring the disadvantages of increasing the order of the equation system and imposing inexactly the boundary conditions, respectively, besides changing the condition number of the equation system matrix.

Weak formulations seek for functions in $\mathbb{H}^1$ space, meaning that both the function and its derivatives are square integrable. In theory, PIM using a global weak formulation does not produce functions with such requirements given its nonconforming characteristic. We can ignore this fact and yet use it to build the meshless shape functions with the drawback of decreasing the convergence rate of the method as shown in [4].

In order to overcome this problem, we can make use of the formulations based on weakened-weak (W^2) forms [5]. While the weak form reduces the degree of differentiability of the approximation function with respect to the strong form, the weakened-weak form reduces it with respect to the weak form. Given a problem represented by a second order partial differential equation, the solution must have second order derivatives. If the problem is represented by its weak form, then the solution must have square integrable first order derivatives. If we use the weakened-weak form, then just the function must be square integrable.

The W^2 formulation can be applied given the gradient smoothing operation and the $\mathbb{G}$ space theory, which can be seen in details in [5]. According to [1], the combination of PIM using W^2 formulations and the $\mathbb{G}$ space theory allows meshless methods to achieve higher convergence rates (superconvergence) and more accurate solutions than the finite element method (FEM) [6].

In this paper we present Point Interpolation Methods based on weakened-weak formulations and apply them to electromagnetics. The methods discussed here work with gradient smoothing based on nodes (NS-PIM), edges (ES-PIM) and cells (CS-PIM) and use T-schemes [1] to select the support nodes employed in the shape function generation. Numerical examples are shown and the performance of these methods are compared to the finite element method.

II. Mathematical Formulation

A. Weak Form

In electromagnetism, static 2D problems are expressed in a generalized form as follows: given k and f, determine the function $u : \Omega \subset \mathbb{R}^2 \to \mathbb{R}$ that satisfies

$$\begin{aligned} \nabla \cdot (k \, \nabla u) &= f && \text{on} \quad \Omega \\ u &= g && \text{on} \quad \Gamma_g \\ -k \frac{\partial u}{\partial n} &= h && \text{on} \quad \Gamma_h \end{aligned} \tag{1}$$

where k is related to the material physical characteristic and f to the source term; g e h are the Dirichlet and Neumann boundary conditions values, respectively, on the boundaries Γ_g and Γ_h, and

received 21 Oct 2012; for review 7 Dec 2012; accepted 15 July 2013

ISSN 2179-1074

$\partial\Omega \ = \Gamma \ = \Gamma_g \cup \Gamma_h$.

Using the weighted residual method, it can be shown that the weak form of the problem (1) is given by

$$\int_{\Omega} k\,\nabla w \cdot \nabla u \ \mathrm{d}\Omega = \ -\int_{\Omega} wf \ \mathrm{d}\Omega \ - \int_{\Gamma_h} wh \ \mathrm{d}\Gamma, \quad \forall w \in U \tag{2}$$

where w is the test function of the weighted residual method and U is the test function space, in which the function must belong to $\mathbb{H}^1(\Omega)$, i.e., its first order derivative must be square integrable over Ω.

B. Gradient Smoothing

A function can be approximated by an integral representation through a convolution with a kernel function $\widehat{W}$, also called smoothing function, over a predefined smoothing domain Ω_x^s, i.e., [1]

$$\hat{F} = \int_{\Omega_{\mathrm{x}}^{\mathrm{s}}} F\widehat{W} \ \mathrm{d}\Omega \tag{3}$$

where $\widehat{W}$ is continuously differentiable in Ω_x^s.

Using this idea, the integral representation of the gradient of a function u can be written as

$$\hat{\nabla} u = \int_{\Omega_{\mathrm{x}}^{\mathrm{s}}} \nabla u\widehat{W} \ \mathrm{d}\Omega = \int_{\Omega_{\mathrm{x}}^{\mathrm{s}}} \nabla(u\widehat{W}) \ \mathrm{d}\Omega - \int_{\Omega_{\mathrm{x}}^{\mathrm{s}}} u\nabla\widehat{W} \ \mathrm{d}\Omega \tag{4}$$

where it is assumed that u is continuous in Ω_x^s and hence at least piecewisely differentiable. Applying the gradient theorem in (4), we have

$$\hat{\nabla} u = \int_{\Gamma_{\mathrm{x}}^{\mathrm{s}}} u\widehat{W}\vec{n} \ \mathrm{d}\Gamma - \int_{\Omega_{\mathrm{x}}^{\mathrm{s}}} u\nabla\widehat{W} \ \mathrm{d}\Omega \tag{5}$$

where $\Gamma_x^s = \partial\Omega_x^s$ and $\vec{n}$ is the unit outwards normal on Γ_x^s.

The equality in (5) is no longer valid if the gradient does not exist in the whole subdomain Ω_x^s, i.e., if u is discontinuous there. However, the gradient of u still can be approximated by

$$\hat{\nabla} u \approx \int_{\Gamma_{\mathrm{x}}^{\mathrm{s}}} u\widehat{W}\vec{n} \ \mathrm{d}\Gamma - \int_{\Omega_{\mathrm{x}}^{\mathrm{s}}} u\nabla\widehat{W} \ \mathrm{d}\Omega\,. \tag{6}$$

Equation (6) is the generalized smoothing gradient operation [5]. This generalization is not rigorous in theory, but it is possible to be applied because no differentiation upon u is required in the right-hand side of (6) [1].

For simplicity, the smoothing function is defined as a local constant in Ω_x^s

$$\widehat{W}(x-\xi) = \overline{W}(x-\xi) = \begin{cases} 1/A_x^s & \text{if } \ \xi \ \in \ \Omega_x^s \\ 0 & \text{if } \ \xi \ \notin \ \Omega_x^s \end{cases} \tag{7}$$

where A_x^s is the area of the smoothing domain Ω_x^s at point x. Note that $\widehat{W}$ as defined in (7) satisfies the conditions of unity, positivity and decay, which are requirements of a kernel function for the integral representation (3) to be valid [1].

Using (7), (5) and (6) are written as

$$\hat{\nabla} u = \begin{cases} \dfrac{1}{A_x^s} \displaystyle\int_{\Omega_x^s} \nabla u \, \mathrm{d}\Omega = \dfrac{1}{A_x^s} \displaystyle\int_{\Gamma_x^s} u \vec{n} \, \mathrm{d}\Gamma & \text{if } u \in \mathrm{C}^0(\Omega_x^s) \\ \dfrac{1}{A_x^s} \displaystyle\int_{\Gamma_x^s} u \vec{n} \, \mathrm{d}\Gamma & \text{if } u \in C^{-1}(\Omega_x^s) \end{cases} \tag{8}$$

which is the smoothed gradient in a smoothing domain Ω_x^s.

C. Weakened-Weak Form (W^2)

Equation (8) provides a constant smooth approximation for the gradient in each subdomain Ω_x^s. If the problem domain is completely subdivided into N_s smoothing subdomains with no overlap, then the smoothed gradient can be calculated in the whole domain Ω. Rewriting (8) in terms of the gradient components, we have

$$\begin{aligned} \hat{\nabla} u = \begin{pmatrix} \dfrac{\partial u}{\partial x} & \dfrac{\partial u}{\partial y} \end{pmatrix} &= \left(\dfrac{1}{A_x^s} \int_{\Gamma_{\mathrm{x}}^{\mathrm{s}}} u n_x \ \mathrm{d}\Gamma \quad \dfrac{1}{A_x^s} \int_{\Gamma_{\mathrm{x}}^{\mathrm{s}}} u n_y \ \mathrm{d}\Gamma \right) \\ &= \left(\bar{g}_x(u) \quad \bar{g}_y(u) \right) \end{aligned} \tag{9}$$

where n_x and n_y are the x and y components of the unit outwards normal on Γ_x^s, respectively.

In order to approximate the gradient in the weak form (2) by the smoothed gradient, we use (9) and the bilinear term is rewritten as

$$\begin{aligned} \bar{a}_D(w,u) &= \sum_{i=1}^{N_s} k A_i^s \hat{\nabla} w \cdot \hat{\nabla} u \\ &= \sum_{i=1}^{N_s} k A_i^s \left(\bar{g}_x(w) \bar{g}_x(u) + \bar{g}_y(w) \bar{g}_y(u) \right) \end{aligned} \tag{10}$$

where A_i^s is the area of the ith smoothing subdomain. Thus we come to the weakened-weak form

$$\bar{a}_D(w,u) = -\int_{\Omega} wf \ \mathrm{d}\Omega - \int_{\Gamma_h} wh \ \mathrm{d}\Gamma, \quad \forall w \in U \tag{11}$$

where $\bar{a}_D(w,u)$ is defined in (10), and U is the test function space, which now must be in $\mathbb{G}^1$: only the function must be square integrable [5]. Note that in the weakened-weak form (11) the functions must be square integrable, while in the weak form (2) the first order derivatives of the function must be square integrable. Therefore, requirements on differentiability are weakened, which gives the name for this type of formulation. Besides that, while the integration of the weak form is performed using background cells, the integration of the W^2 is done using the constructed smoothing subdomains.

III. Point Interpolation Methods

In meshless methods, the approximation $u^h(\mathbf{x})$ for the field function u at a point $\mathbf{x}$ is expressed as

$$u^h(\mathbf{x}) = \sum_{i \in S_n} \phi_i(\mathbf{x}) u_i = \mathbf{\Phi}(\mathbf{x}) \mathbf{U}_S \tag{12}$$

where S_n is the support domain, a set of n nodes included in a compact local domain in the neighborhood of point $\mathbf{x}$, u_i is the nodal field variable at the ith node in the support domain and ϕ_i is the shape function of the ith node created using all the nodes in the support domain. Vectors $\mathbf{\Phi}$ and $\mathbf{U}_S$ hold the shape functions and nodal field variables at the n nodes in the support domain, respectively.

The meshless approximation properties are related to how the shape functions are constructed. In this work, the shape functions are generated using the Point Interpolation Method. PIM shape functions satisfy the Kronecker delta property, which allows a straightforward and exact imposition of Dirichlet boundary conditions, similar to the finite element method. Two versions of PIM are used: the first one uses exclusively polynomial basis functions to build its approximation, while the second one uses polynomial terms and radial basis functions (RBF).

A. Polynomial PIM

Polynomial PIM (PIMp) uses monomials as basis functions. The approximation u^h for the field function u at a point $\mathbf{x}$ is given by

$$u^h(\mathbf{x}) = \sum_{i=1}^{n} p_i(\mathbf{x})a_i = \mathbf{p}^{\mathrm{T}}(\mathbf{x})\mathbf{a} \tag{13}$$

where a_i is the coefficient for the ith polynomial term p_i, n is the number of nodes in the support domain of $\mathbf{x}$, and

$$\mathbf{a} = [a_1,\ a_2,\ \dots, a_{\mathrm{n}}\ ]^{\mathrm{T}} \tag{14}$$

$$\mathbf{p}(\mathbf{x}) = [p_1(\mathbf{x}),\ p_2(\mathbf{x}),\ \dots, p_{\mathrm{n}}(\mathbf{x})]^{\mathrm{T}}. \tag{15}$$

In 2D the polynomial basis are constructed from the Pascal triangle [1], and a complete basis of order m is

$$\mathbf{p}(\mathbf{x}) = \mathbf{p}(x, y) = [1,\ x,\ y,\ xy,\ x^2,\ y^2,\ \dots,\ x^m,\ y^m\ ]^{\mathrm{T}}. \tag{16}$$

The coefficients a_i in (13) are determined by enforcing (13) to be satisfied at the n support nodes. Thus, the shape functions are defined by

$$\mathbf{\Phi}(\mathbf{x}) = \mathbf{p}^{\mathrm{T}}(\mathbf{x})\mathbf{P}_Q^{-1} = [\phi_1(\mathbf{x}),\ \phi_2(\mathbf{x}),\ \dots,\ \phi_n(\mathbf{x})\ ] \tag{17}$$

where $\phi_i(\mathbf{x})$ is the shape function of the ith support node of $\mathbf{x}$, and $\mathbf{P}_Q$ is the moment matrix given by

$$\mathbf{P}_{\mathrm{Q}} = \begin{bmatrix} \mathbf{p}^{\mathrm{T}}(\mathbf{x}_1) \\ \mathbf{p}^{\mathrm{T}}(\mathbf{x}_2) \\ \vdots \\ \mathbf{p}^{\mathrm{T}}(\mathbf{x}_{\mathrm{n}}) \end{bmatrix} \tag{18}$$

The shape functions derivatives with respect to k, with $k = x, y$, are

$$\frac{\partial \mathbf{\Phi}(\mathbf{x})}{\partial k} = \frac{\partial \mathbf{p}^{\mathrm{T}}(\mathbf{x})}{\partial k}\mathbf{P}_Q^{-1}\ . \tag{19}$$

The PIMp approximations have consistency according to the polynomial basis used in (13): if a complete k order polynomial is used, then the shape functions possess C^k consistency [1].

Depending on the configuration of the nodes in the support domain, the moment matrix $\mathbf{P}_Q$ can be singular and break down the shape function construction process. One way to avoid the singularity of the moment matrix is to use radial basis functions together with polynomials in the basis, which leads to another variation of PIM, explored in the next section.

B. Radial PIM with Polynomials

Radial PIM with polynomials (RPIMp) combines radial basis functions and polynomials in the basis to construct the shape functions. The approximation u^h for the field function u at a point $\mathbf{x}$ is given by

$$u^h(\mathbf{x}) = \sum_{i=1}^{n} R_i(\mathbf{x})a_i + \sum_{j=1}^{m} p_j(\mathbf{x})b_j = \mathbf{R}^{\mathrm{T}}(\mathbf{x})\mathbf{a} + \mathbf{p}^{\mathrm{T}}(\mathbf{x})\mathbf{b} \tag{20}$$

where R_i is the radial basis function computed at $\mathbf{x}$ centered at the ith node in the support domain of $\mathbf{x}$, p_j is a monomial in the polynomial basis, a_i and b_j are the coefficients associated to the RBFs and polynomials, respectively, n is the number of nodes in the support domain of $\mathbf{x}$, m is the number of monomials in the polynomial basis and

$$\mathbf{a} = [a_1,\ a_2,\ \dots, a_n\ ]^{\mathrm{T}} \tag{21}$$

$$\mathbf{b} = [b_1,\ b_2,\ \dots, b_m\ ]^{\mathrm{T}} \tag{22}$$

$$\mathbf{R}(\mathbf{x}) = [R_1(\mathbf{x}),\ R_2(\mathbf{x}),\ \dots, R_n(\mathbf{x})]^{\mathrm{T}} \tag{23}$$

$$\mathbf{p}(\mathbf{x}) = [p_1(\mathbf{x}),\ p_2(\mathbf{x}),\ \dots, p_m(\mathbf{x})]^{\mathrm{T}}. \tag{24}$$

The coefficient vectors $\mathbf{a}$ and $\mathbf{b}$ can be determined satisfying (20) at the n nodes within the support domain and imposing constraints on the polynomial basis functions to guarantee a unique solution [7]. Thus two intermediate matrices $\mathbf{S}_a$ and $\mathbf{S}_b$ arise as follows

$$\mathbf{S}_b = \left[\mathbf{P}_m^{\mathrm{T}}\mathbf{R}_Q^{-1}\mathbf{P}_m\ \right]^{-1}\mathbf{P}_m^{\mathrm{T}}\mathbf{R}_Q^{-1} \tag{25}$$

$$\mathbf{S}_a = \mathbf{R}_Q^{-1} - \mathbf{R}_Q^{-1}\mathbf{P}_m\mathbf{S}_b \tag{26}$$

where $\mathbf{R}_Q$ is the moment matrix related to the radial basis functions, and $\mathbf{P}_m$ is the moment matrix associated to the polynomial terms, given by

$$\mathbf{R}_Q = \begin{bmatrix} R_1(\mathbf{x}_1) & R_2(\mathbf{x}_1) & \dots & R_n(\mathbf{x}_1) \\ R_1(\mathbf{x}_2) & R_2(\mathbf{x}_2) & \dots & R_n(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ R_1(\mathbf{x}_n) & R_2(\mathbf{x}_n) & \dots & R_n(\mathbf{x}_n) \end{bmatrix} \tag{27}$$

$$\mathbf{P}_m = \begin{bmatrix} p_1(\mathbf{x}_1) & p_2(\mathbf{x}_1) & \dots & p_{\mathrm{m}}(\mathbf{x}_1) \\ p_1(\mathbf{x}_2) & p_2(\mathbf{x}_2) & \dots & p_{\mathrm{m}}(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ p_1(\mathbf{x}_n) & p_2(\mathbf{x}_n) & \dots & p_m(\mathbf{x}_n) \end{bmatrix}. \tag{28}$$

From (25), (26) and (20), the shape functions can be expressed as

$$\mathbf{\Phi}(\mathbf{x}) = \mathbf{R}^{\mathrm{T}}(\mathbf{x})\mathbf{S}_a + \mathbf{p}^{\mathrm{T}}(\mathbf{x})\mathbf{S}_b = [\phi_1(\mathbf{x}),\ \phi_2(\mathbf{x}),\ \dots,\ \phi_n(\mathbf{x})\ ] \tag{29}$$

where $\phi_i(\mathbf{x})$ is the shape function of the ith support node of $\mathbf{x}$.

The shape functions derivatives with respect to the kth dimension are obtained differentiating $\mathbf{R}^{\mathrm{T}}$ and $\mathbf{p}^{\mathrm{T}}$ with respect to k:

$$\frac{\partial \mathbf{\Phi}(\mathbf{x})}{\partial k} = \frac{\partial \mathbf{R}^{\mathrm{T}}(\mathbf{x})}{\partial k}\mathbf{S}_a + \frac{\partial \mathbf{p}^{\mathrm{T}}(\mathbf{x})}{\partial k}\mathbf{S}_b \,. \tag{30}$$

As in the polynomial PIM, the aproximations generated by RPIMp possess consistency according to the polynomial basis used, and usually linear polynomials are employed. Unlike PIMp, the moment matrices of RPIMp do not suffer with singularity problems, thus it is always possible to build shape functions using this method. In the other hand, RPIMp is computationally more expensive than PIMp.

C. T-Schemes for Support Nodes Selection

To perform the integration of the weakened-weak form (11), an integration mesh is used, which is also called background mesh. Once such a mesh is already available, we can use it to select the support nodes for constructing the shape functions. If the mesh is composed of triangular cells, T-schemes [1] can be used for that purpose. The T-schemes select a set of nodes according to the available integration cells and work particularly well with the family of point interpolation methods [1].

In T3-scheme the support nodes are the cell vertices. It is used to construct linear PIM shape functions, as shown in Fig. 1. The PIM shape functions generated using T3-scheme are identical to those generated by the finite element method. As the number of support nodes is small, the computational performance of the numerical method will be higher. T3-scheme is the simplest of the T-schemes.

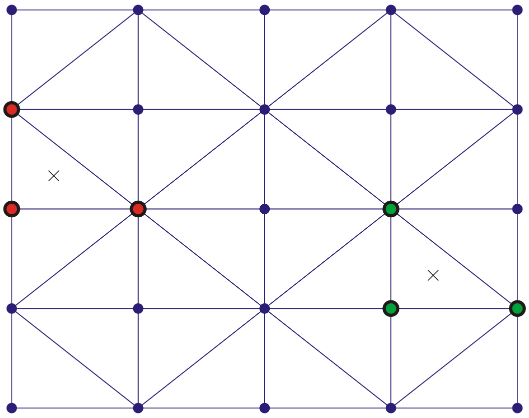

Fig. 1. T3-scheme for support nodes selection. In red are the support nodes for boundary cells and in green the ones for interior cells.

T6-scheme selects 6 nodes for a cell (Fig. 2). For an interior cell, it selects its 3 vertices plus 3 remote vertices of the three neighbouring cells. For a boundary cell, it selects the 3 vertices, 2 (or 1) remote vertices of neighbouring cells and 1 (or 2) node nearest to the centroid of the cell. This scheme is usually used for building PIM shape function with radial basis functions, aiming to generate approximations with more accuracy and efficiency.

received 21 Oct 2012; for review 7 Dec 2012; accepted 15 July 2013
© 2013 SBMO/SBMag ISSN 2179-1074

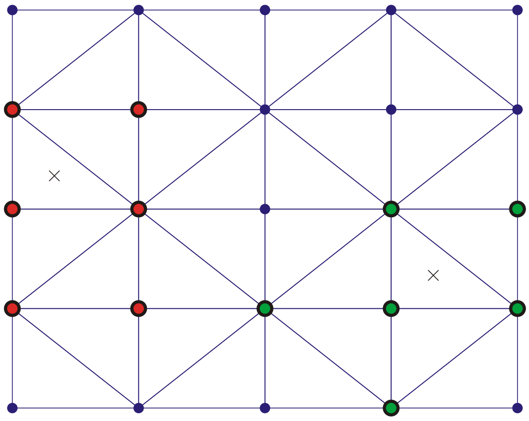

Fig. 2. T6-scheme for support nodes selection. In red are the support nodes for boundary cells and in green the ones for interior cells.

The T6/3-scheme is a combination of T6 and T3-schemes (Fig. 3). It selects 6 nodes for interpolation of interior cells in the same way as T6-scheme, and 3 nodes for boundary cells as in T3-scheme. T6/3-scheme is used to build high-order PIM shape functions, where linear interpolation is performed on boundary and quadratic interpolation is performed in the interior.

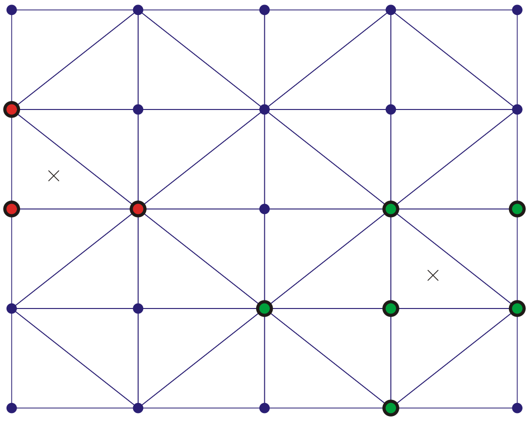

Fig. 3. T6/3-scheme for support nodes selection. In red are the support nodes for boundary cells and in green the ones for interior cells.

T2L-scheme selects two layer of nodes (Fig. 4). The first layer corresponds to the 3 vertices of the cell and the second layer holds the nodes that are directly connected to the first layer nodes. This scheme generally selects more nodes than T6-scheme, leading to bigger computational costs. T2L-scheme is used to build PIM shape functions with radial basis functions with higher order of consistency and when the nodal distribution is very irregular. T2L-scheme can also be employed to build MLS shape functions.

D. Construction of the Smoothing Domains

As seen in Section II-C, smoothed gradients are calculated based on smoothing domains. The procedure for construction of these domains leads to different numerical approximation characteristics. Two requirements are imposed on the smoothing domains: (I) the intersection between any two subdomains must be empty, i.e., there must be no overlap between them, and (II) the union of

received 21 Oct 2012; for review 7 Dec 2012; accepted 15 July 2013
© 2013 SBMO/SBMag *ISSN 2179-1074*

the subdomains must completely cover the problem domain. Three point interpolation methods based on smoothing gradients derive from the way that the smoothing domains are constructed: NS-PIM, ES-PIM and CS-PIM.

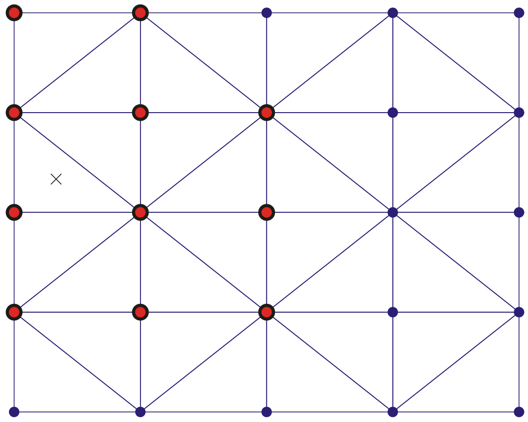

Fig. 4. T2L-scheme for support nodes selection.

The Node-Based Smoothing Point Interpolation Method (NS-PIM) [8] constructs the smoothing domains based on the nodes of the triangular mesh. For each node i, a smoothing domain Ω_i^s is created connecting sequentially the centroid of the cells which are incident to node i to the mid-edge-points, as it can be seen in Fig. 5. Therefore, the number of smoothing subdomains is equal to number of nodes in the mesh, and the requirements (I) and (II) are met.

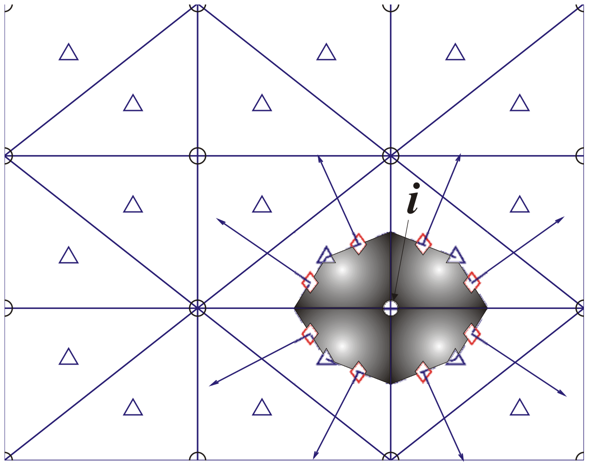

Fig. 5. Node based smoothing subdomain $\boldsymbol{\Omega}_i^s$. The centroids of the cells are represented by triangles and mid-edge-points by diamonds. The unit outwards normals on $\boldsymbol{\Gamma}_i^s$ are represented by vectors.

The Edge-Based Smoothing Point Interpolation Method (ES-PIM) [9] constructs its smoothing domains based on the edges of the triangular mesh. For each edge m, a smoothing domain Ω_m^s is created connecting the endpoints of edge m to the centroids of the adjacent cells, as shown in Fig. 6. Clearly, requirements (I) and (II) are met, and the number of smoothing domains corresponds to the number of edges in the mesh.

The Cell-Based Smoothing Point Interpolation Method (CS-PIM) [10] constructs its smoothing domains based on the cells of the triangular mesh, taking each triangle c as a smoothing domain Ω_c^s, as shown in Fig. 7. For that reason, CS-PIM is very similar to the finite element method. Requirements (I) and (II) are met and the number of smoothing domains matches the number of triangles in the mesh.

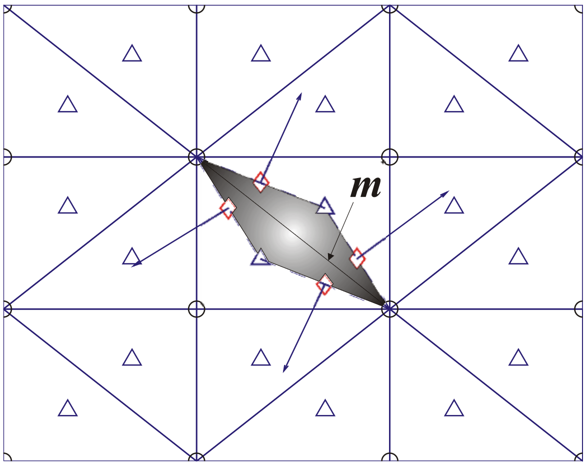

Fig. 6. Edge based smoothing subdomain $\boldsymbol{\Omega}_{\boldsymbol{m}}^{\boldsymbol{s}}$. The centroids of the cells are represented by triangles. The unit outwards normals on $\boldsymbol{\Gamma}_{\boldsymbol{m}}^{\boldsymbol{s}}$ are represented by vectors.

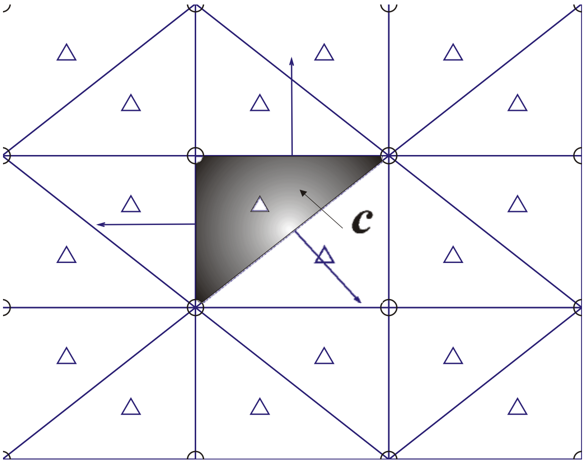

Fig. 7. Cell based smoothing subdomain $\boldsymbol{\Omega}_{\boldsymbol{c}}^{\boldsymbol{s}}$. The centroids of the cells are represented by triangles. The unit outwards normals on $\boldsymbol{\Gamma}_{\boldsymbol{c}}^{\boldsymbol{s}}$ are represented by vectors.

IV. Numerical Results

In this section we investigate the performance of the gradient smoothing methods presented in Section III. In the following problems, the errors on the approximated potentials are calculated using a set of sampled points. The error measure is defined as

$$e_p = \sqrt{\frac{\sum_{i=1}^{N_n}\left(u_i^a - u_i^h\right)^2}{\sum_{i=1}^{N_n}\left(u_i^a\right)^2}} \qquad (31)$$

where N_n is the number of sampled points, u_i^a is the analytical solution at the ith point and u_i^h is the numerical solution at the ith point.

received 21 Oct 2012; for review 7 Dec 2012; accepted 15 July 2013

ISSN 2179-1074

A. *Example 1: Parallel Plate Capacitor*

The first test problem is the parallel plate capacitor with one dielectric. The goal is to verify if the methods can reproduce a linear field, represented by the electric potential. The plates are located at $x = 0$ and $x = 1$ and are at voltages 100V and 10V, respectively. The problem analytical solution is

$$u^a = -90x + 100 \tag{32}$$

A distribution of 5 × 5 nodes is used in a square domain with the dimension of 1 × 1. We test the methods NS-PIM, ES-PIM and CS-PIM with T3-scheme. The norm of error e_p are 3.67×10^{-16}, 2.04×10^{-16} and 1.86×10^{-16}, respectively. This example shows numerically that the three methods can reproduce linear fields exactly (to machine accuracy), which ensures a second order convergence in $\mathbb{L}^2$ norm [1].

B. *Example 2: Parallel Plate Capacitor with Two Dielectric*

The problem of Example 1 is revisited, but with the presence of two dielectrics having relative permittivities of 1 and 5. The interface between the two dielectric is at $x = 0.5$. The purpose of this example is to test whether the methods can treat discontinuities in the solution due to the presence of different media in the domain. The problem analytical solution is

$$u^a = \begin{cases} -150x + 100 & \text{if } \ 0.0 \leq x < 0.5 \\ -30x + 40 & \text{if } \ 0.5 \leq x \leq 1.0 \end{cases} \tag{33}$$

Again, a distribution of 5 × 5 nodes is used. We test NS-PIM, ES-PIM and CS-PIM with T3-scheme. The norm of error e_p are 1.16×10^{-16}, 7.12×10^{-17} and 5.36×10^{-17}, respectively. The results show that the three methods can reproduce exactly (to machine accuracy) the discontinuity of the solution at the interface between the materials.

C. *Example 3: Square Domain with Senoidal Dirichlet Boundary Condition*

In this example we test the methods against a problem with a senoidal Dirichlet boundary condition. The problem domain is a square with the dimension of 1 × 1, as shown if Fig. 8. The boundaries at $x = 0$, $x = 1$ and $y = 0$ are at an electric potential of 10V. The top boundary ($y = 1$) has a potential given by a sine function with a maximum value of 100V. We test NS-PIM, ES-PIM and CS-PIM with T-schemes and shape functions generated by PIMp and RPIMp with Multiquadrics RBF [1]. The problem analytical solution is

$$u^a = \frac{90\sin(\pi x)\sinh(\pi y)}{\sinh(\pi)} + 10 \tag{34}$$

The first step is to verify if the models can approximate the electric potential correctly. For so, the solution is computed along the line $x = 0.5$ for the models NS-PIM with T6/3-scheme, ES-PIM with T3 and T6/3 schemes and, finally, ES-PIM with shape functions generated by RPIMp (ES-RPIM) and T2L-scheme for support nodes selection. Fig. 9 shows the analytical and numerical solutions for comparison. It can be seen that the results produced by all the models are in accordance with the analytical solution.

received 21 Oct 2012; for review 7 Dec 2012; accepted 15 July 2013

ISSN 2179-1074

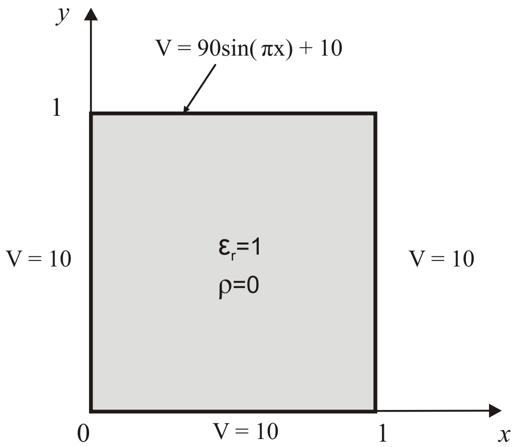

Fig. 8. Problem domain of Example 3.

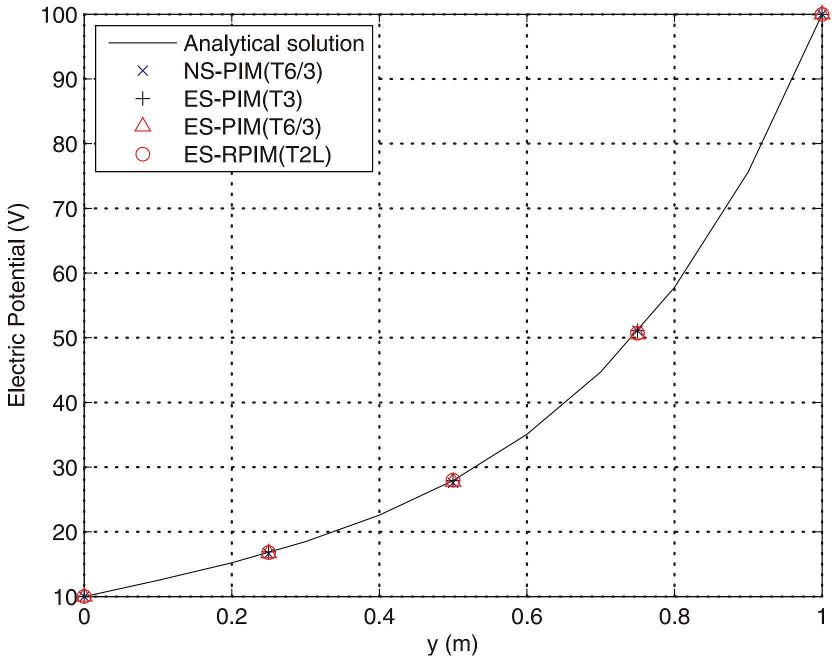

Fig. 9. Electric potential along $\boldsymbol{x = 0.5}$ calculated using different methods based on gradient smoothing.

In order to investigate the convergence rate of the methods to the exact solution, four nodes distributions are used to calculate the error in norm of e_p. We also test the finite element method for comparison. All models use the same sets of nodes and the same meshes. Fig. 10 shows the solution error on logarithm scale for different nodal densities represented by the nodal spacing h. As expected, FEM achieves a convergence rate around 2.0 (2.04), which is the theoretical value for linear models based on weak forms. CS-PIM produces the same numerical solution than FEM, which is also expected because its smoothing domains are the triangles of the mesh. Except the quadratic ES-PIM, that has a rate of 1.94, all the methods have convergence rates greater than the finite element method. It can be observed that the node based smoothing methods are less accurate than FEM. On the other hand, the edge based smoothing methods generate the best results, especially when they build their shape functions using radial basis functions. This is also expected once they use T6 and T2L-schemes, which select more support nodes for building the shape functions and increase the accuracy of the approximation.

The highest convergence rate (2.36) is obtained by ES-RPIM with T2L-scheme, indicating the presence of superconvergence [1]. Another model with a good rate (2.24) is the ES-PIM with T3-scheme, whose solution is also more accurate than FEM. This method is particularly interesting because it uses few support nodes, having thus a good computational efficiency in terms of memory usage and run time, and also it has no parameter to be adjusted as the models that use RBFs. According to the authors' experience, determining these parameters is a hard task because they can vary from problem to problem, meaning that there is not a deterministic way for calculating them, while their values impact significantly on the quality of the numerical solution.

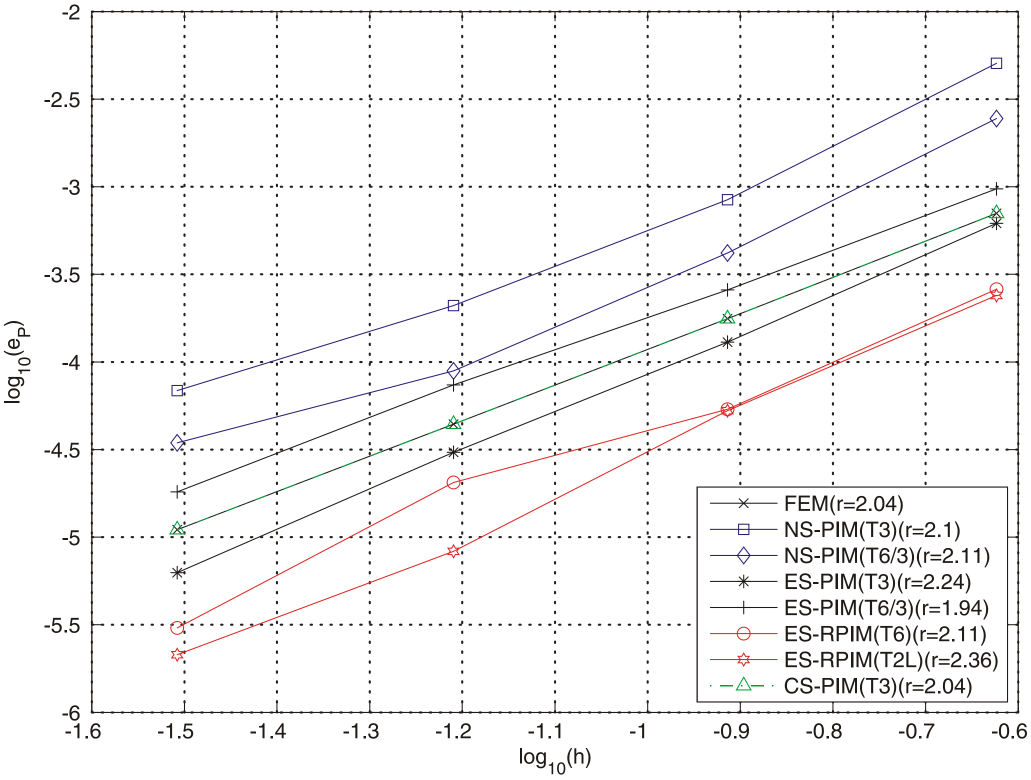

Fig. 10. Convergence rates in norm of error $\boldsymbol{e_p}$ for gradient smoothing methods and FEM.

Proceeding with the investigation of the characteristics of the smoothing methods, an analysis relative to the computational efficiency of the models is performed. Since the edge based smoothing models presented the best results for convergence rate and accuracy, they are the only ones that are put to the test.

It is known that the finite element method has a computational processing time considerably lower than the traditional meshless methods, namely EFG and MLPG, when considering that they use the same set of nodes. However, a fairer comparison is to evaluate the computational performance with respect to the accuracy of the numerical approximation. Following this line, the finite element method still overcomes the traditional meshless methods. This scenario changes when the former is compared

received 21 Oct 2012; for review 7 Dec 2012; accepted 15 July 2013

ISSN 2179-1074

against the gradient smoothing methods. The errors in norm of e_p as a function of processing time spent by the models are shown in Fig. 11. All the smoothing models, except ES-PIM (T6/3), have computational efficiency comparable to the finite element method. In particular, it can be seen that ES-PIM with T3-scheme is the most computational efficient in terms of run time, as previously pointed out, and it overcomes even the finite element method for the considered scenario. This result is due to the fact that ES-PIM (T3) generates more accurate solution than FEM for the same mesh with a not so much more processing time, that is, the difference between the processing time are compensated with a greater accuracy for the gradient smoothing method.

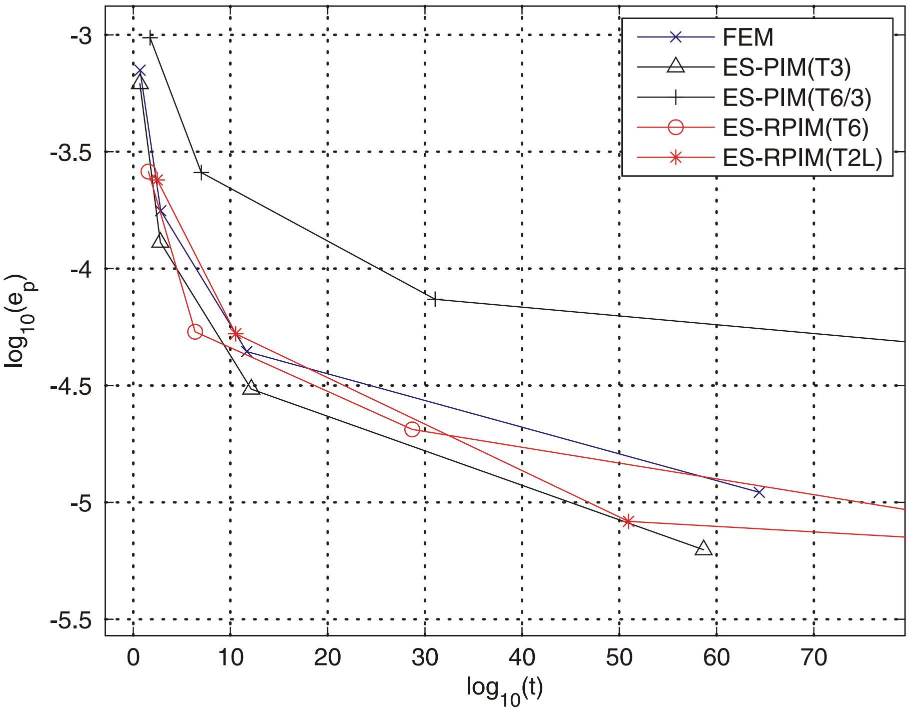

Fig. 11. Computational efficiency comparison for gradient smoothing methods and FEM. Error in norm of $\boldsymbol{e_p}$ *vs* processing time.

D. *Example 4: Radial Magnetic Bearing*

This example deals with an eight pole radial magnetic bearing taken from [11]. Fig. 12 shows the bearing geometry. In the original example, the ferromagnetic material has a nonlinear B-H curve. In this work, the nonlinearity is neglected for simplicity.

The example is a magnetostatic problem where the unknown is the magnetic vector potential A in z-direction. The magnetic flux density B is computed in the entire bearing using ES-PIM with T3-scheme and a mesh of 20513 nodes and 40464 triangular elements (Fig. 13). The flux lines are concentrated mostly in the ferromagnetic material and the levels of B are in accordance with [11].

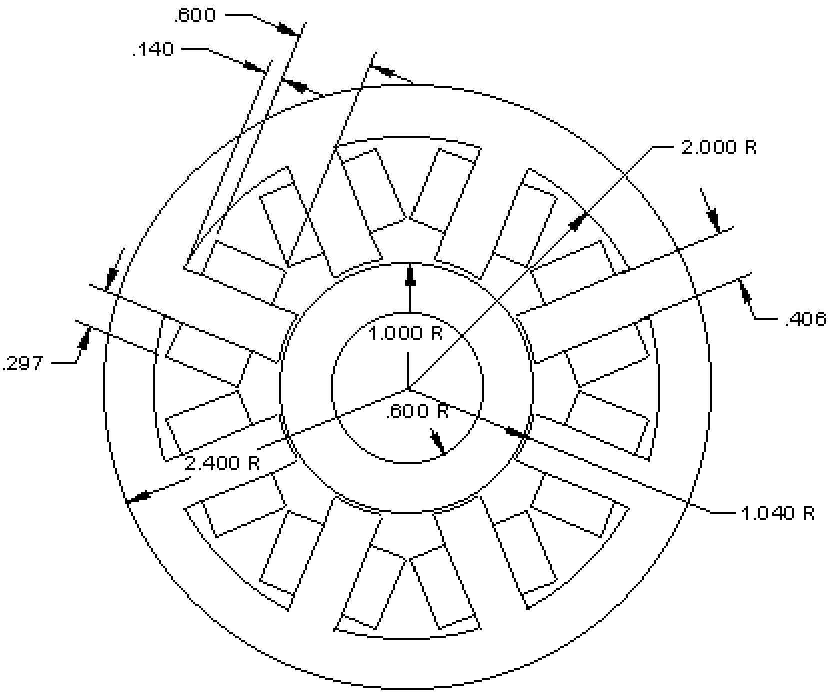

Fig. 12. Eight pole radial magnetic bearing. Dimensions are in inches. Figure taken from [11].

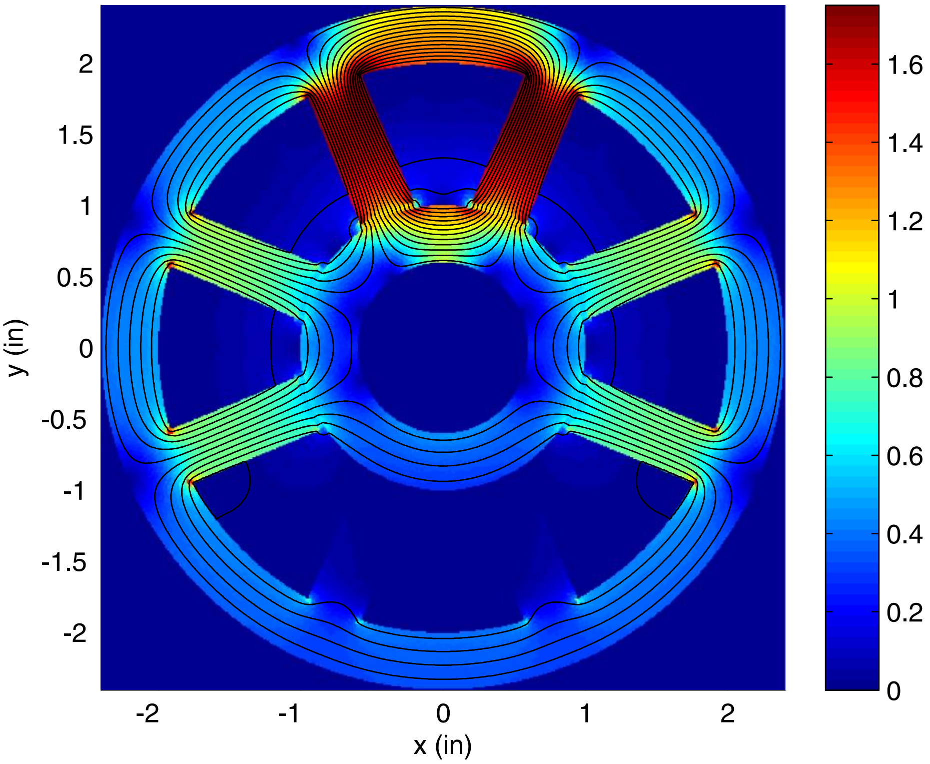

Fig. 13. Magnetic flux density (T) computed by ES-PIM with T3-scheme. Flux lines are also shown.

The previous mesh is used by ES-PIM and FEM to compute *B* at a radial segment in the upper left leg, from (0, 0) to (-0.9, 2.2). For comparison, a reference solution is adopted as the numerical

received 21 Oct 2012; for review 7 Dec 2012; accepted 15 July 2013

approximation generated by FEM using a mesh with 189492 nodes and 377677 triangular elements. The results are shown in Fig. 14 and 15 for ES-PIM and FEM, respectively. It can be seen that at the central region of the leg, ES-PIM is more accurate than FEM. The four maximum absolute percentual errors for each model are computed and indicated with red circles in the figures, and their values are listed in Table I.

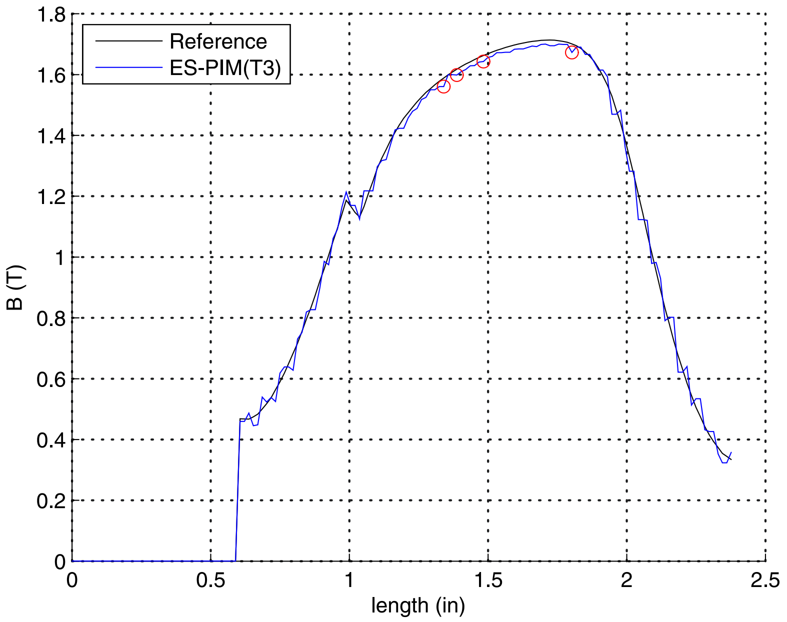

Fig. 14. Magnetic flux density (T) computed by ES-PIM with T3-scheme at a radial segment in the upper left leg. Red circles indicate where the four max percentual differences occur in the center region of the leg.

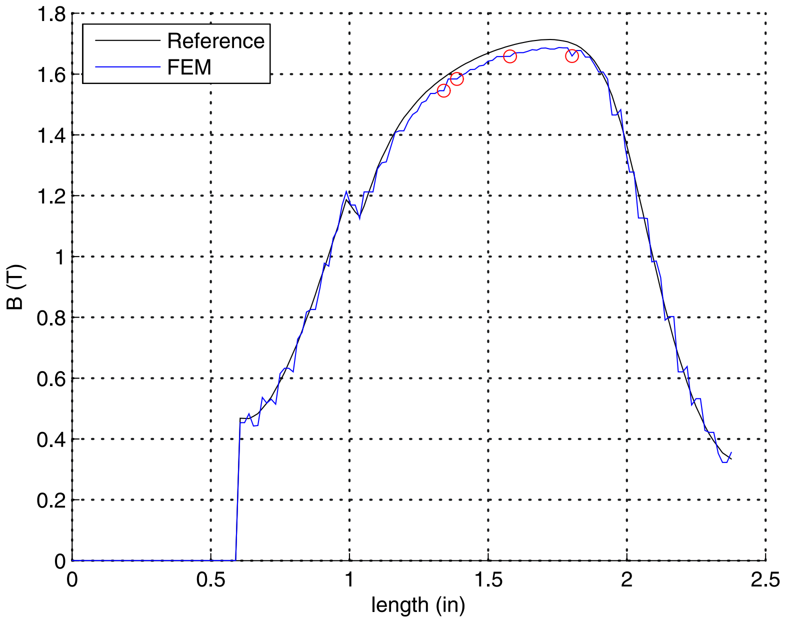

Fig. 15. Magnetic flux density (T) computed by FEM at a radial segment in the upper left leg. Red circles indicate where the four max percentual differences occur in the center region of the leg.

received 21 Oct 2012; for review 7 Dec 2012; accepted 15 July 2013

ISSN 2179-1074

TABLE I. MAXIMUM ABSOLUTE PERCENTUAL ERROR (%) FOR MAGNETIC DENSITY FLUX IN THE CENTRAL REGION OF THE UPPER LEFT LEG OF THE BEARING OF ES-PIM AND FEM NUMERICAL SOLUTIONS.

ES-PIM (T3)	**FEM**
1.77	2.71
1.66	2.44
1.21	2.12
1.20	2.07

V. CONCLUSION

An study of meshless point interpolation methods based on weakened-weak forms with gradient smoothing was presented. The methods NS-PIM, ES-PIM and CS-PIM and their formulations for static problems were derived. Four numerical examples were solved, covering both electrostatics and magnetostatics.

The numerical results showed that the gradient smoothing methods have convergence rates of the same order or higher than the finite element method, as suggested for models with W^2 formulations. Edge based smoothing methods have the best convergence rates and accuracies. In addition, these models presented comparable or higher computational efficiency than the finite element method when accuracy is considered.

An eight pole radial magnetic bearing problem was solved to test ES-PIM in a real and more complex problem. It was shown that the approximation generated by ES-PIM is also better than the one generated by FEM for this problem.

In general, edge based smoothing models proved to be the most attractive in terms of convergence rate, accuracy and computational efficiency. In particular, ES-PIM with T3-scheme is chosen by the author as the best gradient smoothing model.

ACKNOWLEDGMENT

This work was supported in part by the State of Minas Gerais Research Foundation - FAPEMIG, Brazil, the Brazilian government agency CAPES and National Council for Scientific and Technological Development - CNPq, Brazil.

REFERENCES

[1] G. R. Liu, Mesh Free Methods: Moving Beyond the Finite Element Method, 2nd ed. CRC Press, 2009.

[2] T. Belytschko, Y. Y. Lu, and L. Gu, “Element-free Galerkin methods,” International Journal for Numerical Methods in Engineering, vol. 37, no. 2, pp. 229 – 256, 1994.

[3] S. N. Atluri and S. Shen, “The meshless local Petrov-Galerkin (MLPG) method - A simple and less-costly alternative to the finite element and boundary element methods,” CMES - Computer Modeling in Engineering and Sciences, vol. 3, no. 1, pp. 11–51, 2002.

[4] N. Z. Lima, R. C. Mesquita, W. G. Facco, A. S. Moura, and E. J. Silva, “The nonconforming point interpolation method applied to electromagnetic problems,” IEEE Transactions on Magnetics, vol. 48, pp. 619–622, 2012.

[5] G. Liu, “A g space theory and a weakened weak (w2) form for a unified formulation of compatible and incompatible methods: Part i theory and part ii applications to solid mechanics problems,” International Journal for Numerical Methods in Engineering, vol. 81, pp. 1093–1126, 2010.

[6] T. J. R. Hughes, The Finite Element Method: Linear Static and Dynamic Finite Element Analysis. Dover Publications, 2000.

[7] J. G. Wang and G. R. Liu, “A point interpolation meshless method based on radial basis functions,” International Journal for Numerical Methods in Engineering, vol. 54, pp. 1623–1648, 2002.

[8] S. Wu, G. Liu, H. Zhang, and G. Zhang, “A node-based smoothed point interpolation method (ns-pim) for thermoelastic problems with solution bounds,” International Journal of Heat and Mass Transfer, vol. 52, pp. 1464–1471, 2009.

[9] S. Wu, G. Liu, X. Cui, T. Nguyen, and G. Zhang, “An edge-based smoothed point interpolation method (es-pim) for heat transfer analysis of rapid manufacturing system,” International Journal of Heat and Mass Transfer, vol. 53, pp. 1938–1950, 2010.

[10] G. Zhang and G.-R. Liu, “A meshfree cell-based smoothed point interpolation method for solid mechanics problems,” AIP Conference Proceedings, vol. 1233, no. 1, pp. 887–892, 2010.

[11] D. Meeker, “Force of an eight pole radial magnetic bearing,” http://www.femm.info/wiki/RadialMagneticBearing.

received 21 Oct 2012; for review 7 Dec 2012; accepted 15 July 2013

ISSN 2179-1074