Contents list available at JMCS

Journal of Mathematics and Computer Science

JMCS

Journal Homepage: www.tjmcs.com

Journal of Mathematics and Computer Science
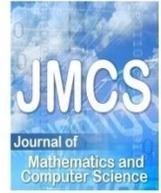
Numerical solution of Maxwell equations using local weak form meshless techniques

S. Sarabadan[1], M. Shahrezaee[1], J.A. Rad[2], K. Parand[2,*]

[1]Department of Mathematics, Imam Hossein University, P.O. Box 16895-198, Tehran, Iran
[2] Department of Computer Sciences, Faculty of Mathematical Sciences, Shahid Beheshti University, Evin, P.O. Box 198396-3113,Tehran,Iran

[*]k_parand@sbu.ac.ir

Abstract

   The aim of this work is to propose a numerical approach based on the local weak formulations and finite difference scheme to solve the Maxwell equation, especially in this paper we select and analysis local radial point interpolation (LRPI) based on multiquadrics radial basis functions (MQ-RBFs). LRPI scheme is the truly meshless method, because, a traditional non-overlapping, continuous mesh is not required, either for the construction of the shape functions, or for the integration of the local sub-domains. These shape functions which are constructed by point interpolation method using the radial basis functions have delta function property which allows one to easily impose essential boundary conditions. One numerical example is presented showing the behavior of the solution and the efficiency of the proposed method.

Keywords: Meshless weak form, Maxwell equation, Finite differences, Local radial point interpolation.

1. Introduction

Recently, a great attention has been paid to the development of various meshless formulations for solution of boundary value problems in many branches of science and engineering. Meshless methods have become viable alternatives to either finite element method (FEM) or boundary element method (BEM). Compared to the FEM and the BEM, the main feature of this type of method is the absence of an explicit mesh, and the approximate solutions are constructed entirely based on a cluster of scattered nodes. Meshless methods have been found to have special advantages on problems to which the conventional mesh-based methods are difficult to be applied. These include problems with complicated boundary, moving boundary and so on. Many of meshless methods are derived from a weak-form formulation on

global domain or a set of local sub-domains. In the global formulation background cells are required for the integration of the weak form. Strictly speaking, these meshless methods are not truly meshless methods. It should be noticed that integration is performed only those background cells with a nonzero shape function. In methods based on local weak-form formulation no cells are required and therefore they are often referred to as truly meshless methods. If a simple form is chosen for the geometry of the sub-domains, numerical integrations can be easily carried out over them. Recently, two family of meshless methods, based on the local weak form for arbitrary partial differential equations with moving least-square (MLS) and radial basis functions (RBFs) approximation have been developed. Local boundary integral equation method (LBIE) with moving least square approximation and local radial point interpolations (LRPI) with radial basis functions have been developed by Zhu et al. [1] and Liu et al. [2, 3], respectively. Both methods (LBIE and LRPI) are meshless, as no domain/boundary traditional non-overlapping meshes are required in these two approaches.

Particularly, the LRPI meshless method reduces the problem dimension by one, has shape functions with delta function properties, and expresses the derivatives of shape functions explicitly and readily. Thus it allows one to easily impose essential boundary and initial (or final) conditions. Though the LBIE method is an efficient meshless method, it is difficult to enforce the essential boundary conditions for that the shape function constructed by the moving least-squares (MLS) approximation lacks the delta function property.

Some special techniques have to be used to overcome the problem, for example, the Lagrange multiplier method and the penalty method [4]. For some works on the meshless LBIE method one can mention the papers of Zhu et al. [1, 5, 6] in linear and non-linear acoustic and potential problems and the works of Sladek brothers [7, 8] for heat conduction problems. The method has now been successfully extended to a wide range of problems in engineering. For some examples of these problems, see [9, 10] and other references therein. The interested reader of meshless methods can also see [11, 12].

In the current work, the LRPI method is employed to numerical analysis of the two-dimensional unsteady Maxwell equations.

The Maxwell equations are a set of PDEs published by Maxwell in 1865 [13], which describe how the electric and magnetic fields are related to their sources, charge density and current density. Later, O. Heaviside [14] and W. Gibbs transformed these equations into the today's vector notations.

To describe the electromagnetic wave propagation, under certain assumptions, in three dispersive medias; i.e. cold plasma, Debye medium and Lorentz medium; the unified model for governing equations is considered as [27,28]:

$$\mu \frac{\partial H}{\partial t} + \nabla \times E = 0, \qquad in \ \Omega \times (0, T], \qquad (1)$$

$$\epsilon \frac{\partial E}{\partial t} - \nabla \times H + E + J(x, t) = 0, \quad in \ \Omega \times (0, T], \qquad (2)$$

where E is electric field, H is magnetic field, ϵ is permittivity of free space, μ is the permeability of free space and the polarization current J is as follows [27,28]:

$$J(E, t) = \omega_p^2 \int_0^t e^{-\nu(t-s)} E(x, t) ds,$$

The boundary $\partial \Omega$ of Ω is considered to be perfect conduction boundary which means [27,28]:

$$n \times E = 0, \qquad on \quad \partial \Omega \times (0, T],$$

where ω_p is the plasma frequency, ν is the electron-neutral collision frequency and initial conditions are

$$E(x, 0) = E_0(x), \qquad H(x, 0) = H_0(x), \qquad in \quad \Omega$$

where E_0 and H_0 are given functions with H_0 satisfying [27,28]

$$\nabla.(\mu H_0) = 0, \quad in \ \Omega \qquad\qquad H.n = 0, \quad on \ \partial\Omega \qquad\qquad (3)$$

One can easily obtain, from 1 and 3, that

$$\nabla.(\mu H) = 0,$$

which is a solenoidal condition for magnetic flux density. Also, the second condition of 3, together with 1 and 2, leads to

$$H.n = 0, \quad on \ \partial\Omega$$

which is also a perfect conduct boundary condition. Several numerical methods have been proposed for Maxwell equations [15, 16, 17] but very few lectures have been studied these models using meshless methods. The interested reader can see [18, 27].

The objective of this paper is to extend the LRPI based on multiquadrics radial basis functions (MQRBFs) to evaluate Maxwell equations. To the best of our knowledge, the local weak form of meshless method has not yet been used in electromagnetic field. Therefore, it appears to be interesting to extend such a numerical technique also to Maxwell equations, which is done in the present manuscript.

The paper is organized as follows: Section 2 is devoted to presenting the LRPI approach and the application of such a numerical technique to the Maxwell problems considered is shown in this section. The numerical results obtained are presented and discussed in Section 3 and finally, in Section 4, some conclusions are drawn.

2. Methology

To evaluate the Maxwell equation, LRPI method is used in the present work. This method is based on local weak forms over intersecting subdomains At first we discuss a time-stepping method for the time derivative.

2.1 Time discretization

To obtain a fully discrete scheme, the time interval $(0, T)$ has been divided into the N uniform subintervals by employing nodes $0 = t_0 \leq t_1 \leq \cdots \leq t_N = T$, where $t_n = n \Delta t$, then to deal with the time derivatives, the following difference approximations have been considered

$$\frac{\partial H}{\partial t}(x, t) = \frac{H^{n+1}(x) - H^n(x)}{\Delta t},$$

$$\frac{\partial E}{\partial t}(x, t) = \frac{E^{n+1}(x) - E^n(x)}{\Delta t},$$

$$H^{n+1/2}(x, t) = \frac{H^{n+1}(x) + H^n(x)}{2},$$

$$E^{n+1/2}(x, t) = \frac{E^{n+1}(x) + E^n(x)}{2},$$

where $H^n(x) = H(x, n\,\Delta t)$ and $E^n(x) = E(x, n\,\Delta t)$. Therefore, Eqs. 1 and 2 become as follows

$$\mu \frac{H^{n+1}(x) - H^n(x)}{\Delta t} + \nabla \times E^{n+1/2} = 0,$$

$$\epsilon \frac{E^{n+1}(x) - E^n(x)}{\Delta t} - \nabla \times H^{n+1/2} + E^{n+1/2} + J^n(x,t) = 0,$$

thus we can obtain

$$\mu \frac{1}{\Delta t} H^{n+1} + \frac{1}{2} \nabla \times E^{n+1} = \mu \frac{1}{\Delta t} H^n - \frac{1}{2} \nabla \times E^n,$$

$$\epsilon \frac{1}{\Delta t} E^{n+1} - \frac{1}{2} \nabla \times H^{n+1} + \frac{1}{2} E^{n+1} = \epsilon \frac{1}{\Delta t} E^n + \frac{1}{2} \nabla \times H^n - \frac{1}{2} E^n - J^n,$$

$$\tag{4}$$

we assume the following recursive formula for J^n [27]

$$J^0 = 0,$$

$$J^n = e^{-\Delta t\,v} J^{n-1} + \Delta t\, \frac{\omega_p^2}{2} \left(e^{-\Delta t\,v} E^{n-1} + E^n \right), \qquad n \geq 1 \tag{5}$$

which is obtained by using trapezoidal rule to approximate the numerical integration from t_{n-1} to t_n as the expression of J^n is divided into integrations over two time intervals, $[0, t_{n-1}]$ and $[t_{n-1}, t_n]$, see [27].

2.2. The local weak form

Now, we use the local weak form instead of the global weak form. The LRPI method constructs the weak form over local sub-domains such as Ω_s, which is a small region taken for each node in the global domain Ω. The local sub-domains overlap each other and cover the whole global domain Ω. The local sub-domains could be of any geometric shape and size. For simplicity they are taken to be of circular shape. Therefore the local weak form of the approximate equation (4) for $x^i = (x^i, y^i) \in \Omega_s^i$ can be written as

$$\int_{\Omega_s^i} \{\mu \frac{1}{\Delta t} H^{n+1} + \frac{1}{2} \nabla \times E^{n+1}\}.\, v(x) d\Omega \quad = \int_{\Omega_s^i} \{\mu \frac{1}{\Delta t} H^n - \frac{1}{2} \nabla \times E^n\}.\, v(x) d\Omega, \tag{6}$$

$$\int_{\Omega_s^i} \{\epsilon \frac{1}{\Delta t} E^{n+1} - \frac{1}{2} \nabla \times H^{n+1} + \frac{1}{2} E^{n+1}\}.\, v(x) d\Omega$$

$$= \int_{\Omega_s^i} \{\epsilon \frac{1}{\Delta t} E^n + \frac{1}{2} \nabla \times H^n - \frac{1}{2} E^n - J^n\}.\, v(x) d\Omega, \tag{7}$$

which Ω_s^i is the local domain associated with the i. In LRPI, v is the Heaviside step function

$$v(x) = \begin{cases} 1, & x \in \Omega_s^i \\ 0, & o.\,w. \end{cases}$$

as the test function in each local domain.
Using the divergence theorem, we have

$$\mu \frac{1}{\Delta t} \int_{\Omega_s^i} H^{n+1} .\, v\, d\Omega + \frac{1}{2} \int_{\Omega_s^i} E^{n+1} .\, (\nabla \times v)\, d\Omega + \frac{1}{2} \int_{\partial \Omega_s^i} v.\, (n \times E^{n+1})\, d\Gamma$$

$$= \mu \frac{1}{\Delta t} \int_{\Omega_s^i} H^n . v \, d\Omega - \frac{1}{2} \int_{\Omega_s^i} E^n . (\nabla \times v) d\Omega$$

$$- \frac{1}{2} \int_{\partial\Omega_s^i} v . (n \times E^n) \, d\Gamma \tag{8}$$

$$\epsilon \frac{1}{\Delta t} \int_{\Omega_s^i} E^{n+1} . v \, d\Omega + \frac{1}{2} \int_{\Omega_s^i} E^{n+1} . v \, d\Omega - \frac{1}{2} \int_{\Omega_s^i} H^{n+1} . (\nabla \times v) \, d\Omega - \frac{1}{2} \int_{\partial\Omega_s^i} v . (n \times H^{n+1}) \, d\Gamma$$

$$= \epsilon \frac{1}{\Delta t} \int_{\Omega_s^i} E^n . v \, d\Omega - \frac{1}{2} \int_{\Omega_s^i} E^n . v \, d\Omega + \frac{1}{2} \int_{\Omega_s^i} H^n . (\nabla \times v) d\Omega + \frac{1}{2} \int_{\partial\Omega_s^i} v . (n \times H^n) \, d\Gamma$$

$$- \int_{\Omega_s^i} J^n . v \, d\Omega \tag{9}$$

In above relations, $\square\Omega_s^i$ is the boundary of Ω_s^i. Also, because the derivative of the Heaviside step function v is equal to zero, then for LRPI scheme the above local weak form system is transformed into the following simple local integral equation

$$\mu \frac{1}{\Delta t} \int_{\Omega_s^i} H^{n+1} \, d\Omega + \frac{1}{2} \int_{\partial\Omega_s^i} n \times E^{n+1} \, d\Gamma$$

$$= \mu \frac{1}{\Delta t} \int_{\Omega_s^i} H^n \, d\Omega - \frac{1}{2} \int_{\partial\Omega_s^i} n \times E^n \, d\Gamma \tag{10}$$

$$\epsilon \frac{1}{\Delta t} \int_{\Omega_s^i} E^{n+1} \, d\Omega + \frac{1}{2} \int_{\Omega_s^i} E^{n+1} \, d\Omega - \frac{1}{2} \int_{\partial\Omega_s^i} n \times H^{n+1} \, d\Gamma$$

$$= \epsilon \frac{1}{\Delta t} \int_{\Omega_s^i} E^n \, d\Omega - \frac{1}{2} \int_{\Omega_s^i} E^n \, d\Omega + \frac{1}{2} \int_{\partial\Omega_s^i} n \times H^n \, d\Gamma$$

$$- \int_{\Omega_s^i} J^n \, d\Omega \tag{11}$$

It is important to observe that in relations (10) and (11) exist unknown functions, we should approximate these functions. To this aim the local integral equations (10) and (11) are transformed in to a system of algebraic equations with real unknown quantities at nodes used for spatial approximation, as described in the next subsection.

2.3. Spatial approximation

Instead of using traditional non-overlapping, continuous meshes to form the interpolation scheme, the LRPIM uses a local interpolation or approximation to represent the trial or test functions with the values (or the fictitious values) of the unknown variable at some randomly located nodes. There are a number of local interpolation schemes for this purpose. The radial point interpolation is one of them. In this section, the fundamental idea of these approximations are reviewed.

2.3.1. Polynomial basis point interpolation (PPI) method

Polynomials have been used as famous basis functions in interpolation to creat shape functions in many numerical algorithm. The polynomial basis point interpolation scheme is investigated in meshfree methods by Liu and Gu [19] in the first time. In the point interpolation method (PIM), interpolation is based on a set of nodes in the vicinity of a point x named the local support domain. A support domain can have different shapes and its dimension and shape can be different from point to point. Most often used shapes are rectangular or circular.

Consider an influence domain Ω with a set of suitably located nodes $\{x_i\}^n_{i=1}$ is the number of nodes in the local support domain; in it. An interpolation of a function $u(x)$ in the neighborhood of a point x can be expressed in the form

$$\sum_{i=1}^{n} a_i P_i(x) = \mathbf{P}^{\mathrm{T}}(x)\mathbf{a}, \tag{12}$$

where $P_i(x)$ is a monomial in the spatial coordinate $x = [x, y]^T$, n is the number of nodes in the support domain of x, and a_i is the corresponding coefficient of the basis functions. In the vector form, Eq. (12), the vectors are defined as

$$\mathbf{P}(x) = (P_1(x), \dots, P_n(x))^T,$$

$$\mathbf{a} = (a_1, \dots, a_n)^T,$$

For example, in one-dimensional problems, two well known linear basis are

$$\mathbf{P}(x) = (1, x, y)^T,$$

$$\mathbf{P}(x) = (1, x, y, x^2, xy, y^2)^T,$$

the unknown coefficient a_i can be determined by enforcing $u(x)$ to be satisfied at the n nodes within the local support domain of the point of interest x. This leads to n linear equations, which can be expressed in the following matrix form as

$$\mathbf{U} = \mathbf{P_n}\mathbf{a}, \tag{13}$$

where

$$\boldsymbol{U} = (u_1, u_2, \dots, u_n)^T,$$

$$\mathbf{P_n} = \begin{pmatrix} 1 & x_1 & y_1 & x_1y_1 & \cdots \\ 1 & x_2 & y_2 & x_2y_2 & \cdots \\ \vdots & \vdots & & \vdots & \vdots & \ddots \\ 1 & x_n & y_n & x_ny_n & \cdots \end{pmatrix}.$$

In this interpolation scheme, let the variable u_i denote $u(x_i)$ for simplicity. Assuming the existence of P_n^{-1} , a unique solution of vector $\boldsymbol{a}$ can be obtained by solving Eq. (13) for $\boldsymbol{a}$:

$$\mathbf{a} = \mathbf{P_n^{-1}U}. \tag{14}$$

By substituting Eq. (13) into Eq. (14), we have

$$u(\boldsymbol{x}) = \boldsymbol{P}^T(\boldsymbol{x})\boldsymbol{P}_n^{-1}\boldsymbol{U} = \sum_{i=1}^{n} \varphi_i(\boldsymbol{x})u_i = \boldsymbol{\Phi}^T(\boldsymbol{x})\boldsymbol{U},$$

where the shape function $\boldsymbol{\Phi}$ $(\boldsymbol{x})$ is defined by

$$\boldsymbol{\Phi}^T(\boldsymbol{x}) = \boldsymbol{P}^T(\boldsymbol{x})\boldsymbol{P}_n^{-1} = (\varphi_1(\boldsymbol{x}), \varphi_2(\boldsymbol{x}), \dots, \varphi_n(\boldsymbol{x})).$$

The shape function $\boldsymbol{\Phi}$ $(\boldsymbol{x})$ obtained through the above scheme, possesses the delta function property, i.e. $\varphi_i(x_j) = \delta_{ij}$.

Remark 2: In order to avoid the singularity of $\mathbf{P_n}$, several strategies have been proposed. Liu and Gu [19] proposed a moving node method to change the coordinates of nodes randomly before computation. Changing basis function through the transformation of the local coordinate is the other effective method [20]. The matrix triangularization algorithm (MTA) is recently proposed [21]. Kansa [22, 23] has also solved this kind of singularity problem using radial basis functions. Since the RBFs are used, the matrix $\mathbf{P_n}$ is not singular in general. Kansa's method is an unsymmetric RBF collocation method based upon the GA and MQ interpolation functions. Although the above approach has been applied successfully in several cases, no existence of solution and convergence analysis is available in the literature and, for some cases, it has been reported that the resulting matrix was extremely ill-conditioned. The condition number of the above interpolation matrix for smooth RBFs like Gaussian or multiquadrics are extremely large. Several techniques have been proposed to improve the conditioning of the coefficient matrix and the solution accuracy. Fasshauer [24] suggested an alternative approach to the unsymmetric scheme based on the Hermite interpolation property of the radial basis functions. The advantage of Hermite-based approach is that the matrix resulting from the scheme is symmetric, as opposed to the completely unstructured matrix of the same size resulting from unsymmetric schemes. Also, Liu et al. [25] to overcome this deficiency used RBFs as the augmented terms in the point interpolation method.

2.3.2. Radial basis point interpolation (RBPI) method
The point interpolation form is rewritten as

$$u(\boldsymbol{x}) = \sum_{i=1}^{n} R_i(\boldsymbol{x})a_i + \sum_{j=1}^{m} P_j(\boldsymbol{x})b_j, \tag{15}$$

where $R_i(\boldsymbol{x})$ is the radial basis functions, n is the number of RBFs and is also identical to the number of nodes in the local support domain of the point of interest $\boldsymbol{x}$, and $P_j(\boldsymbol{x})$ is a monomial in the spatial coordinate $\boldsymbol{x} = [x, y]^T$, and m is the number of polynomial basis functions. If $m = 0$, pure RBFs are concluded. Coefficients a_i and b_j are interpolation constant yet to be determined. The polynomial term

in Eq. (15) is not always necessary. However, augment of polynomial in RPI shape functions has the following benefits [26]

1. Adding polynomial term up to the linear order can ensure the C^1 consistency that is needed to pass the standard patch test.
2. In general, adding polynomial can always improve the accuracy of the results.
3. Adding polynomial reduces the influence of the shape parameters on the accuracy of the results.
4. Adding polynomial can improve the interpolation stability for some RBFs.

In this work, linear polynomial terms $(1, x, y)^T$ are adopted to augment the RBFs.
Coefficients a_i and b_j in Eq. (15) can be determined by enforcing Eq. (15) to be satisfied at these n nodes, forming the support domain around point x. The interpolation at the kth point is:

$$u_k = u(x_k) = \sum_{i=1}^{n} R_i(x_k) a_i + \sum_{j=1}^{m} P_j(x_k) b_j, \quad k = 1, 2, \dots, n \tag{16}$$

The polynomial term is an extra-requirement that guarantees unique approximation. Following constraints are usually imposed

$$\sum_{i=1}^{n} P_j(x_i) a_i = 0, \quad j = 1, 2, \dots, m \tag{17}$$

The representation (16) and Eq. (17) consitute a $(n + m) \times (n + m)$ system of linear algebraic equations which can be expressed in matrix form as

$$\begin{bmatrix} U \\ 0 \end{bmatrix} = \begin{bmatrix} R_0 & P_m \\ P_m^T & 0 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix},$$

or

$$\begin{bmatrix} U \\ 0 \end{bmatrix} = G \begin{bmatrix} a \\ b \end{bmatrix},$$

where

$$U = (u_1, u_2, \dots, u_n)^T,$$

$$\{R_0\}_{ij} = R_i(r_j).$$

Unique solution is obtained if the inverse of matrix R_0 exists,

$$\begin{bmatrix} a \\ b \end{bmatrix} = G^{-1} \begin{bmatrix} U \\ 0 \end{bmatrix}.$$

Accordingly, Eq. (15) can be rewritten as

$$\hat{u}(x) = R^T(x)\, a + P^T(x)\, b = [R^T(x) \quad P^T(x)] \begin{bmatrix} a \\ b \end{bmatrix},$$

or

$$\hat{u}(x) = [R^T(x) \quad P^T(x)] G^{-1} \begin{bmatrix} U \\ 0 \end{bmatrix} = \Phi(x)\, U, \tag{18}$$

where the matrix of shape functions $\boldsymbol{\Phi}(x)$ is defined by

$$\boldsymbol{\Phi}(x) = [\varphi_1(x), \varphi_2(x), \dots, \varphi_n(x)],$$

in which

$$\varphi_k(x) = \sum_{i=1}^{n} R_i(x) \boldsymbol{G}_{i,k}^{-1} + \sum_{j=1}^{m} P_j(x) \boldsymbol{G}_{n+j,k}^{-1},$$

and $\boldsymbol{G}_{i,k}^{-1}$ is the (i,k) element of matrix $\boldsymbol{G}^{-1}$.
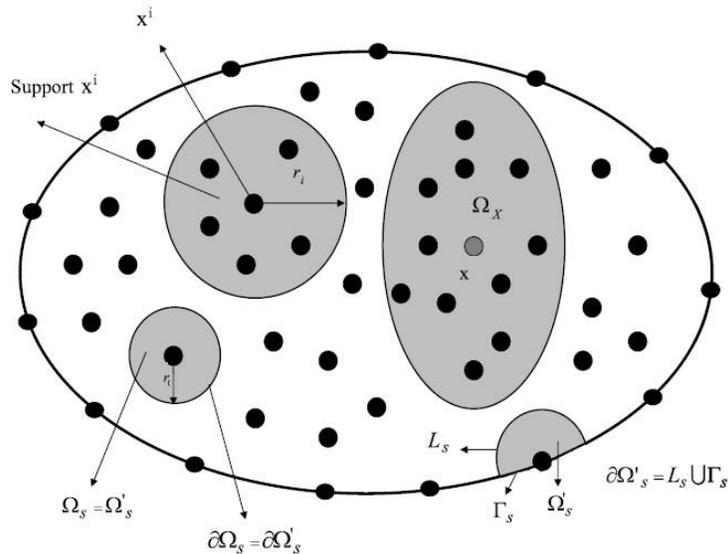
Figure 1: Sub-domains of different points in the problem domain.

2.4. Discretized equations
Before we show how to discretize model in the forms (10) and (11), we focus on how to select nodal points. Let $X = \{x_0, x_1, \dots, x_N\} \subset \Omega$ are scattered meshless points, where some points are located on the boundary Γ to enforce the boundary conditions.
To obtain the discrete equations from the locally weak forms (10) and (11), for the interior points, substituting approximation formula (18) into local integral equations (10) and (11) yields:

$$\mu \frac{1}{\Delta t} \sum_{j=1}^{n} \alpha_j^{n+1} \int_{\Omega_s^i} \varphi_j(x) \, d\Omega + \frac{1}{2} \sum_{j=1}^{n} \beta_j^{n+1} \int_{\partial \Omega_s^i} \boldsymbol{n} \times \varphi_j(x) \, d\Gamma$$

$$= \mu \frac{1}{\Delta t} \sum_{j=1}^{n} \alpha_j^{n} \int_{\Omega_s^i} \varphi_j(x) \, d\Omega - \frac{1}{2} \sum_{j=1}^{n} \beta_j^{n} \int_{\partial \Omega_s^i} \boldsymbol{n} \times \varphi_j(x) \, d\Gamma \qquad (19)$$

$$(\epsilon \frac{1}{\Delta t} + \frac{1}{2}) \sum_{j=1}^{n} \beta_j^{n+1} \int_{\Omega_s^i} \varphi_j(x) \, d\Omega - \frac{1}{2} \sum_{j=1}^{n} \alpha_j^{n+1} \int_{\partial \Omega_s^i} \boldsymbol{n} \times \varphi_j(x) \, d\Gamma$$

$$= \left(\epsilon\frac{1}{\Delta t} - \frac{1}{2}\right)\sum_{j=1}^{n} \beta_j^n \int_{\Omega_s^i} \varphi_j(x)\, d\Omega + \frac{1}{2}\sum_{j=1}^{n} \alpha_j^n \int_{\partial\Omega_s^i} \boldsymbol{n} \times \varphi_j(x)\, d\Gamma$$

$$- \int_{\Omega_s^i} J^n\, d\Omega \tag{20}$$

The matrix forms of the relations (19) and (20) are respectively as follows

$$\mu\frac{1}{\Delta t}\sum_{j=1}^{n} \alpha_j^{n+1}\, A_{ij} + \frac{1}{2}\sum_{j=1}^{n} \beta_j^{n+1} B_{ij}$$

$$= \mu\frac{1}{\Delta t}\sum_{j=1}^{n} \alpha_j^n\, A_{ij} - \frac{1}{2}\sum_{j=1}^{n} \beta_j^n\, B_{ij} \tag{21}$$
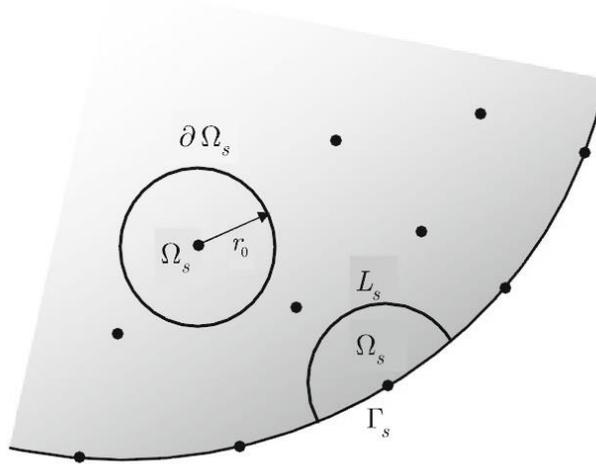
Figure 2: Sampling of meshless points and local subdomains

$$(\epsilon\frac{1}{\Delta t} + \frac{1}{2})\sum_{j=1}^{n} \beta_j^{n+1}\, A_{ij} - \frac{1}{2}\sum_{j=1}^{n} \alpha_j^{n+1}\, B_{ij}$$

$$= \left(\epsilon\frac{1}{\Delta t} - \frac{1}{2}\right)\sum_{j=1}^{n} \beta_j^n\, A_{ij} + \frac{1}{2}\sum_{j=1}^{n} \alpha_j^n\, B_{ij}$$

$$- \int_{\Omega_s^i} J^n\, d\Omega \tag{22}$$

where

$$A_{ij} = \int_{\Omega_s^i} \varphi_j(x)\, d\Omega,$$

$$B_{ij} = \int_{\partial\Omega_s^i} \boldsymbol{n} \times \varphi_j(\boldsymbol{x}) \ d\Gamma$$

Also for boundary nodes $x^i \in \partial\Omega_s^i$ where $\partial\Omega_s^i = L_s^i \ \cup \Gamma_s^i$, we have

$$\boldsymbol{\mu}\frac{1}{\Delta t}\sum_{j=1}^{n}\alpha_j^{n+1}\,A_{ij} + \frac{\boldsymbol{1}}{\boldsymbol{2}}\sum_{j=1}^{n}\beta_j^{n+1}\boldsymbol{B'_{ij}}$$
$$= \boldsymbol{\mu}\frac{1}{\Delta t}\sum_{j=1}^{n}\alpha_j^n\,A_{ij} - \frac{\boldsymbol{1}}{\boldsymbol{2}}\sum_{j=1}^{n}\beta_j^n\,\boldsymbol{B'_{ij}} \tag{23}$$

$$(\epsilon\frac{1}{\Delta t} + \frac{\boldsymbol{1}}{\boldsymbol{2}})\sum_{j=1}^{n}\beta_j^{n+1}\,A_{ij} - \frac{\boldsymbol{1}}{\boldsymbol{2}}\sum_{j=1}^{n}\alpha_j^{n+1}\,\boldsymbol{B'_{ij}}$$
$$= \left(\epsilon\frac{1}{\Delta t} - \frac{\boldsymbol{1}}{\boldsymbol{2}}\right)\sum_{j=1}^{n}\beta_j^n\,A_{ij} + \frac{\boldsymbol{1}}{\boldsymbol{2}}\sum_{j=1}^{n}\alpha_j^n\,\boldsymbol{B'_{ij}}$$
$$- \int_{\Omega_s^i} J^n\,d\Omega \tag{24}$$

where

$$B'_{ij} = \int_{\partial L_s^i} \boldsymbol{n} \times \varphi_j(\boldsymbol{x}) \ d\Gamma$$

3. Numerical results and discussions

The numerical simulations are run on a PC Laptop with an Intel(R) Core(TM)2 Duo CPU T9550 2.66 GHz 4 GB RAM and the software programs are written in Matlab.

Let H_x^M, H_y^M and E_z^M denote the exact solutions and also $H_x^{M,LRPI}, H_y^{M,LRPI}$ and $E_z^{M,LRPI}$ denote the approximations obtained using LRPI method developed in the Section 3. Note that in the following example, the exact values of H_x^M, H_y^M and E_z^M are available. Therefore, to show the rate of convergence

of the new schemes when $h \to 0$ and $\Delta t \to 0$, the values of ratio with the following formula have been reported in the tables

$$\boldsymbol{Ratio} = \boldsymbol{\log_2}\left(\frac{\text{MaxError in the previous row}}{\text{MaxError in the current row}}\right).$$

Table 1: Numerical results, efficiency of the LRPI scheme.

N	Δt	MaxError-H_x	Ratio-H_x	MaxError-H_y	Ratio-H_y	MaxError-E_z	Ratio-E_z
4×4	10^{-2}	1.03×10^{-2}	–	1.03×10^{-2}	–	1.01×10^{-2}	–
	10^{-3}	1.03×10^{-2}	–	1.03×10^{-2}	–	1.01×10^{-2}	–
	10^{-4}	1.03×10^{-2}	–	1.03×10^{-2}	–	1.01×10^{-2}	–
8×8	10^{-2}	3.55×10^{-3}	1.54	3.55×10^{-3}	1.54	2.56×10^{-3}	1.98
	10^{-3}	3.51×10^{-3}	1.55	3.51×10^{-3}	1.55	2.53×10^{-3}	1.99
	10^{-4}	3.51×10^{-3}	1.55	3.51×10^{-3}	1.55	2.53×10^{-3}	1.99
16×16	10^{-2}	8.77×10^{-4}	2.02	8.77×10^{-4}	2.02	6.02×10^{-4}	2.09
	10^{-3}	8.33×10^{-4}	2.08	8.33×10^{-4}	2.08	5.71×10^{-4}	2.15
	10^{-4}	8.32×10^{-4}	2.08	8.32×10^{-4}	2.08	5.71×10^{-4}	2.15
32×32	10^{-2}	2.41×10^{-4}	1.86	2.41×10^{-4}	1.86	1.66×10^{-4}	1.86
	10^{-3}	1.97×10^{-4}	2.08	1.97×10^{-4}	2.08	1.35×10^{-4}	2.08
	10^{-4}	1.97×10^{-4}	2.08	1.97×10^{-4}	2.08	1.35×10^{-4}	2.08

In the following analysis, the radius of the local sub-domain is selected $r_Q = 0.71h$, where h is the distance between the nodes. The size of r_Q is such that the union of these sub-domain must cover the whole global domain i.e. $\cup \, \Omega_s^i \subset \Omega$.

Now, in order to show that weak form meshless methods proposed in this paper, provide very accurate, stable and fast approximation, we present the following model

$$\mu_0 \frac{\partial H_x}{\partial t} + \frac{\partial E_z}{\partial y} = 0,$$

$$\mu_0 \frac{\partial H_y}{\partial t} - \frac{\partial E_z}{\partial x} = 0,$$

$$\epsilon_0 \frac{\partial E_z}{\partial t} - \frac{\partial H_y}{\partial x} + \frac{\partial H_x}{\partial y} = 0,$$

where $\epsilon_0 = \mu_0 = 1$. The exact solution of this model is

$$\begin{pmatrix} H_x \\ H_y \\ E_z \end{pmatrix} = \begin{pmatrix} -\dfrac{\pi}{\omega} \sin(\pi x) \cos(\pi y) \sin(\omega t) \\ \dfrac{\pi}{\omega} \cos(\pi x) \sin(\pi y) \sin(\omega t) \\ \sin(\pi x) \cos(\pi y) \cos(\omega t) \end{pmatrix},$$

where $\omega = \sqrt{2}\pi$. Of special interest to us is testifying the numerical convergence of the solution by this problem. Hence, applying the suggested schemes in this paper together with different choice of N and M, we get the consequences tabulated in Table 1. The number of time discretization steps is set equal to nodes distributed in the domain. As we have experimentally checked, this choice is such that in all the simulations performed the error due to the time discretization is negligible with respect to the error due to the LRPI discretization (note that in the present work we are mainly concerned with the LRPI spatial approximation).
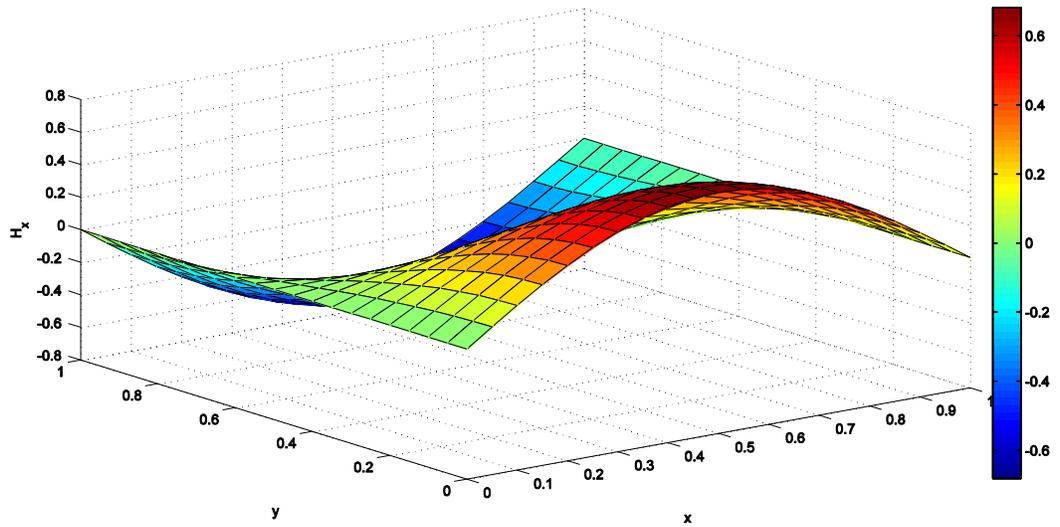
Figure 3: Surface plot of the numerical and analytical solutions H_x at time $t = 1$ by using $N = 32 \times 32$ and $\Delta t = 0.01$.
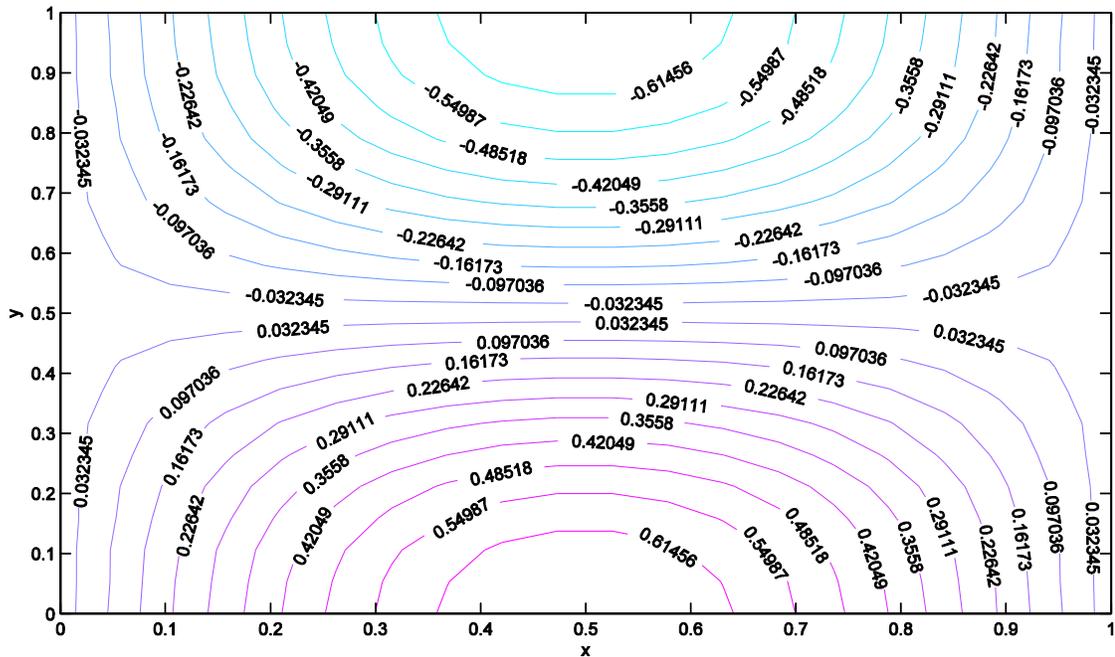
Figure 4: Contour lines corresponding to the H_x at time $t = 1$ by using $N = 32 \times 32$ and $\Delta t = 0.01$.

From Table 1, it can be seen that the LRPI method provide very accurate, stable and fast approximation for option pricing. We observe that the accuracy grows as the number of basis increases gradually, then the solution can be computed with a small error in a small computer time.
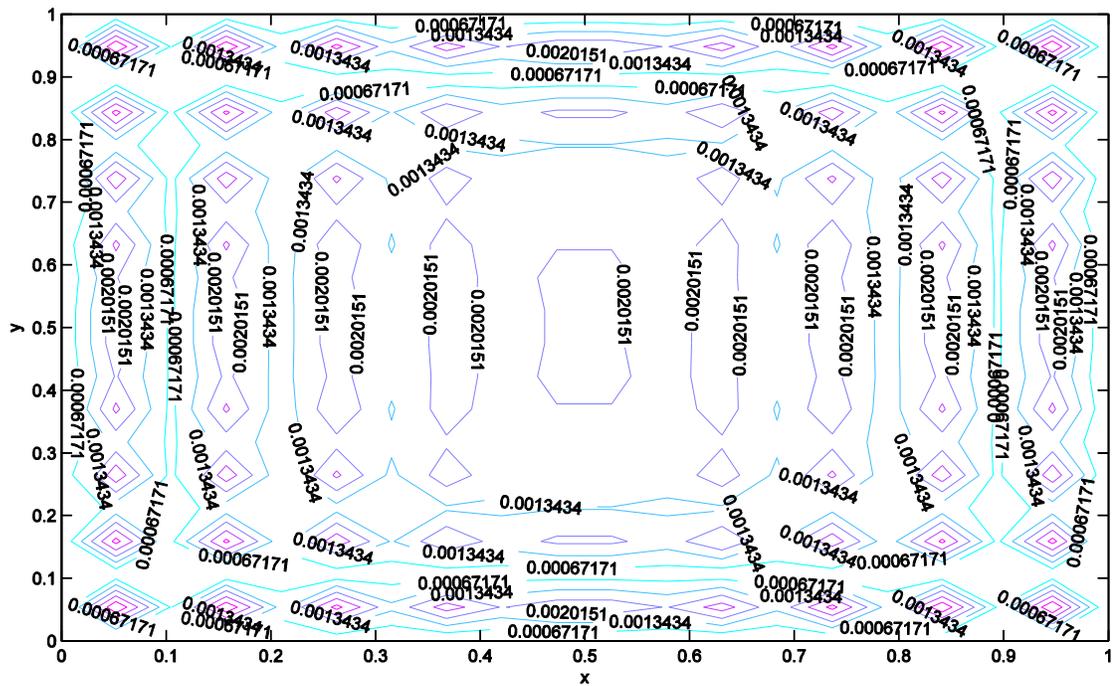
Figure 5: Contour lines corresponding to the absolute error between numerical and analytical solutions H_x at time $t = 1$ by using $N = 32 \times 32$ and $\Delta t = 0.01$.

Also, the results are shown in Figs. (3)-(11) as illustration. The results of our numerical experiments, confirm the validity of the new techniques.

Putting all these things together, we conclude that the numerical methods proposed in this paper are accurate, convergence and fast.
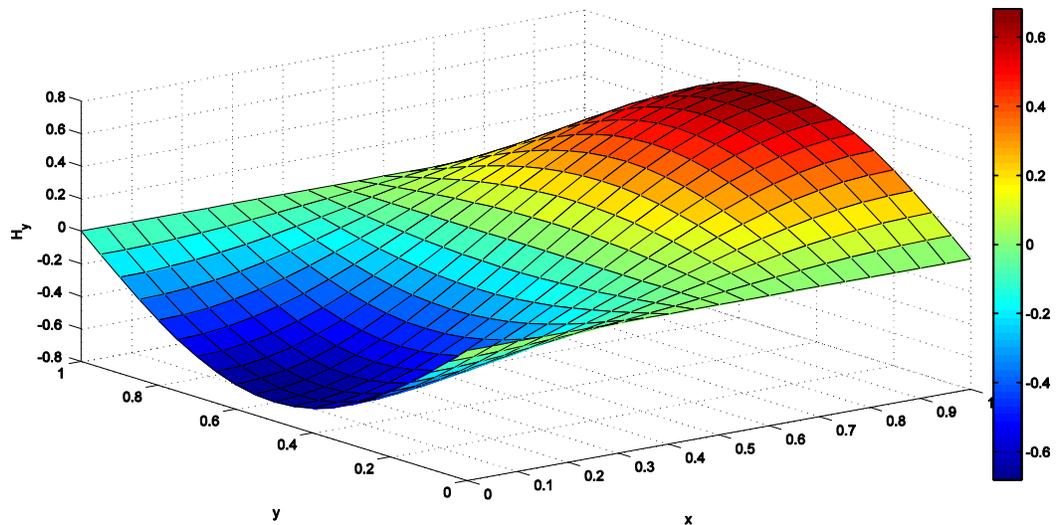
Figure 6: Surface plot of the numerical and analytical solutions H_y at time $t = 1$ by using $N = 32 \times 32$ and $\Delta t = 0.01$.
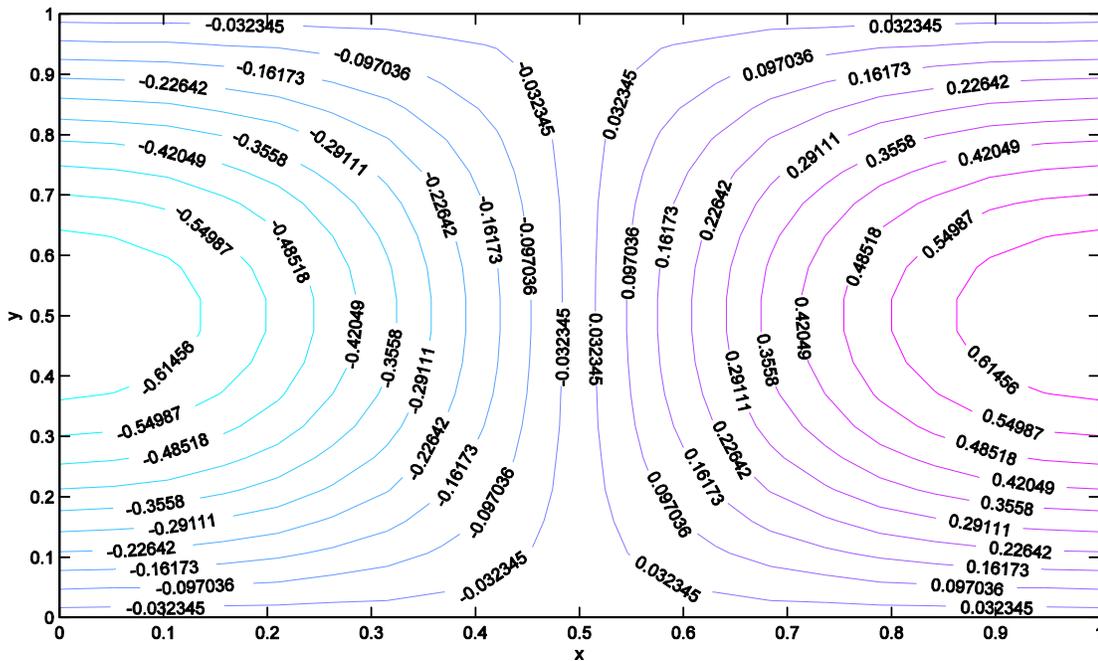
Figure 7: Contour lines corresponding to the H_y at time $t = 1$ by using $N = 32 \times 32$ and $\Delta t = 0.01$.
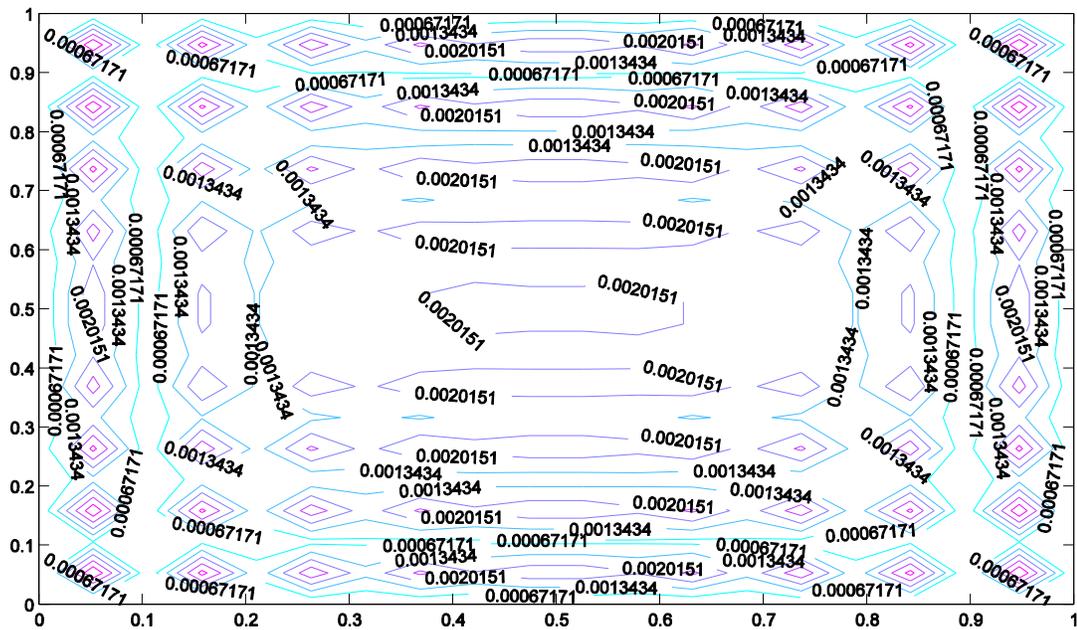
Figure 8: Contour lines corresponding to the absolute error between numerical and analytical solutions H_y at time $t = 1$ by using $N = 32 \times 32$ and $\Delta t = 0.01$.
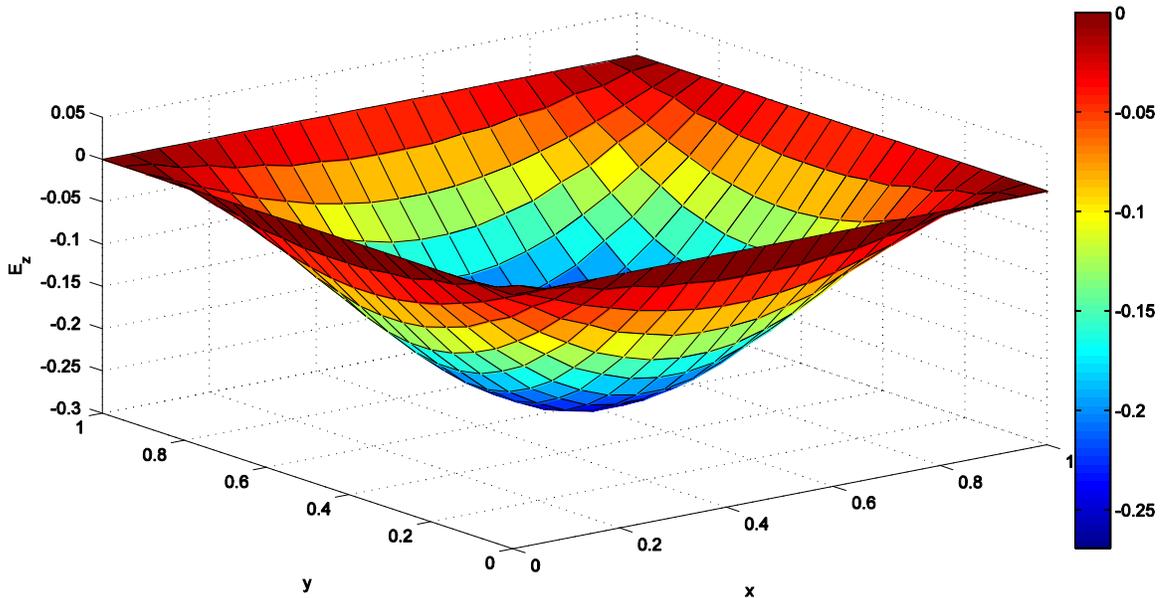
Figure 9: Surface plot of the numerical and analytical solutions E_z at time $t = 1$ by using $N = 32 \times 32$ and $\Delta t = 0.01$.

Figure 10: Contour lines corresponding to the E_z at time $t = 1$ by using $N = 32 \times 32$ and $\Delta t = 0.01$.
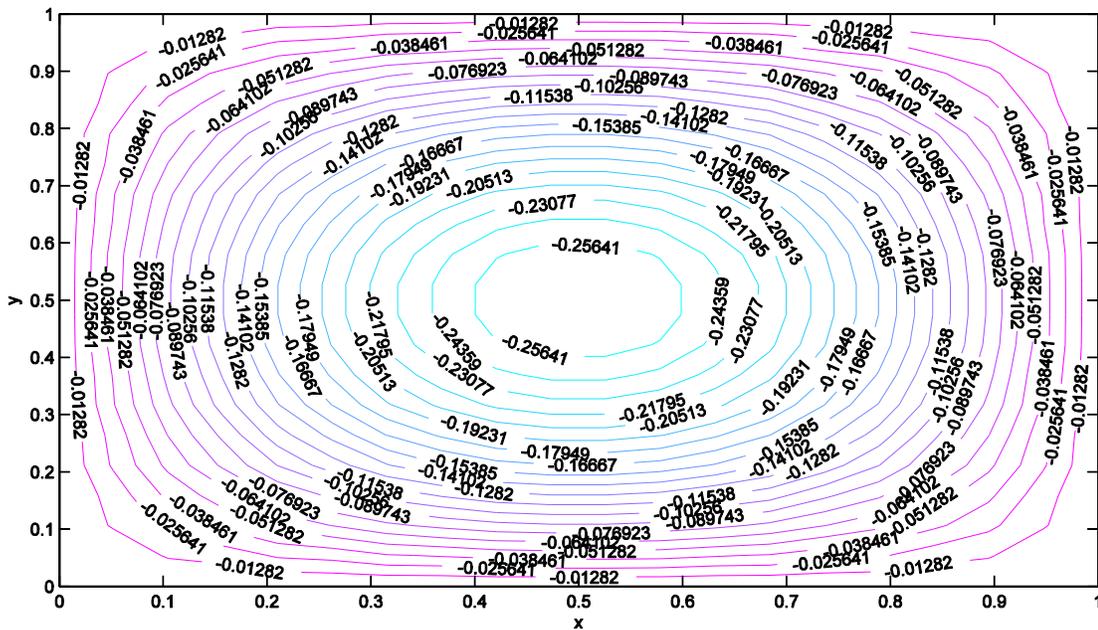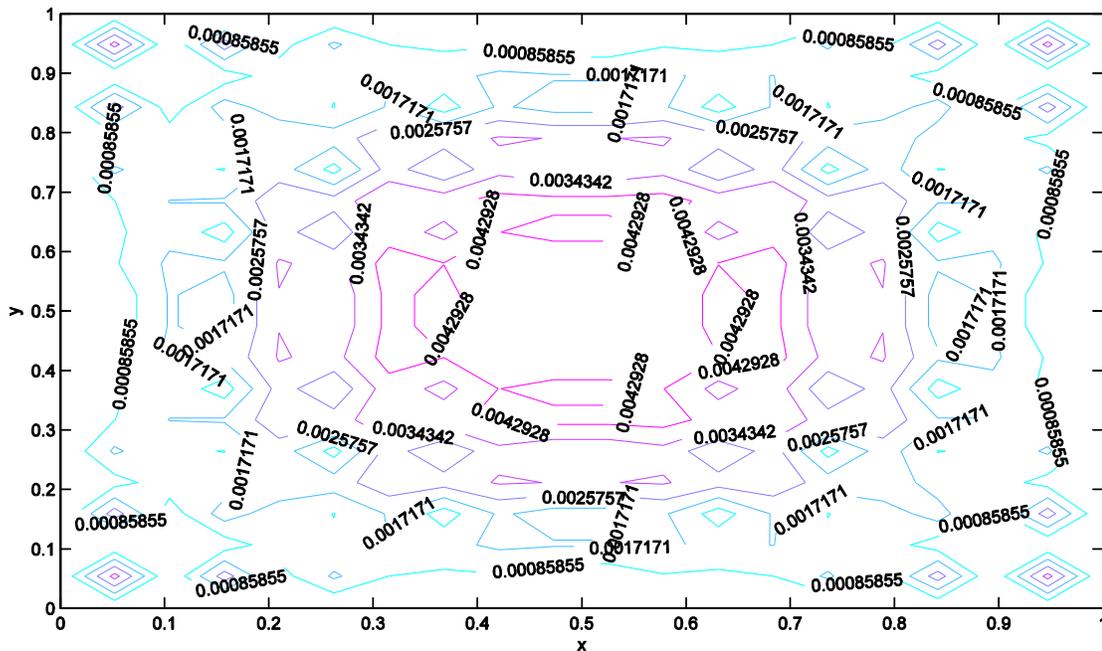
Figure 11: Contour lines corresponding to the absolute error between numerical and analytical solutions E_z at time $t = 1$ by using $N = 32 \times 32$ and $\Delta t = 0.01$.

5. **Conclusions**

In this paper, the weak form meshless methods namely local radial point interpolation (LRPI) based on multiquadrics radial basis functions (MQ-RBFs), were formulated and successfully applied to solve the timedependent Maxwell equations. The present methods are truly a meshless method, which do not need any element or mesh for both field interpolation and background integration. This is an alternative numerical tool to many existing computational methods. The main advantage is its simplicity. In this method, the shape functions have been constructed by the radial point interpolation method (RPIM). The shape functions so formulated posses the delta function property. The Heaviside step function was used as the test function in the local weak form method in the meshless local radial point interpolation method. Also to demonstrate the accuracy and usefulness of this method, one numerical example has been presented. The given example demonstrated that using the present approach leads to acceptable results in comparison with different approximate methods.

References

[1] T. Zhu, J. D. Zhang, S. N. Atluri, A local boundary integral equation (LBIE) method in computational mechanics, and a meshless discretization approach, Comput. Mech. 21 (1998) 223–235.

[2] J. Wang, G. Liu, A point interpolation meshless method based on radial basis functions, Int. J. Numer. Meth. Eng. 54 (2002) 1623–1648.

[3] G. Liu, Y. Gu, An Introduction to Meshfree Methods and Their Programing, Springer, Netherlands, 2005.

[4] S. N. Atluri, H. G. Kim, J. Y. Cho, A critical assessment of the truly meshless local Petrov-Galerkin (MLPG), and local boundary integral equation (LBIE) methods, Comput. Mech. 24 (1999) 348–372.

[5] T. Zhu, J. D. Zhang, S. N. Atluri, A meshless local boundary integral equation (LBIE) for solving nonlinear problems, Comput. Mech. 22 (1998) 174–186.

[6] T. Zhu, J. D. Zhang, S. N. Atluri, A meshless numerical method based on the local boundary integral equation (LBIE) to solve linear and non-linear boundary value problems, Eng. Anal. Bound. Elem. 23 (1999) 375–389.

[7] J. Sladek, V. Sladek, C. Zhang, Transient heat conduction analysis in functionally graded materials by the meshless local boundary integral equation method, Comput. Mat. Sci. 28 (2003) 494–504.

[8] J. Sladek, V. Sladek, J. Krivacek, C. Zhang, Local BIEM for transient heat conduction analysis in 3-D axisymmetric functionally graded solids, Comput. Mech. 32 (2003) 169–176.

[9] A. Shirzadi, V. Sladek, J. Sladek, A local integral equation formulation to solve coupled nonlinear reaction-diffusion equations by using moving least square approximation, Eng. Anal. Bound. Elem. 37 (2013) 8–14.

[10] S. M. Hosseini, V. Sladek, J. Sladek, Application of meshless local integral equations to two dimensional analysis of coupled non-fick diffusion-elasticity, Eng. Anal. Bound. Elem. 37 (2013) 603–615.

[11] V. Sladek, J. Sladek, Local integral equations implemented by MLS-approximation and analytical integrations, Eng. Anal. Bound. Elem. 34 (2010) 904–913.

[12] X. Li, Meshless galerkin algorithms for boundary integral equations with moving least square approximations, Appl. Numer. Math. 61 (2011) 1237–1256.

[13] J. C. Maxwell, A dynamical theory of the electromagnetic field, R. Soc. Trans. 155 (1865) 459–512.

[14] O. Heaviside, On the forces, stresses and fluxes of energy in the electromagnetic field, Phil. Trans. R. Soc. 183 (1892) 423–480.

[15] K. S. Yee, Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media, IEEE Trans. Antennas and Propagation 14 (1966) 302–307.

[16] F. B. Belgacem, A. Buffa, Y. Maday, The mortar finite element method for 3D Maxwell equations: first results, SIAM J. Numer. Anal. 39 (2002) 880–901.

[17] G. Rodrigue, D. White, A vector finite element time-domain method for solving Maxwell's equations on unstructured hexahedral grids, SIAM J. Sci. Comput. 23 (2001) 683–701.

[18] S. J. Lai, B. Z. Wang, Y. Dua, Meshless radial basis function method for transient electromagnetic computations, IEEE Trans. Magn. 44 (2008) 2288–2295.

[19] G. Liu, Y. Gu, A point interpolation method for two-dimensional solid, Int. J. Numer. Meth. Eng. 50 (2001) 937–951.

[20] G. Liu, Meshfree methods: moving beyond the finite element method, Boca Raton: Taylor and Francis/CRC Press, 2009.

[21] G. Liu, Y. Gu, A matrix triangularization algorithm for the polynomial point interpolation method, Comput. Meth. Appl. Mech. Eng. 192 (2003) 2269–2295.

[22] E. J. Kansa, Multiquadrics-A scattered data approximation scheme with applications to computational fluid-dynamics-I surface approximations and partial derivative estimates, Comput. Math. Appl. 19 (1990) 127–145.

[23] E. J. Kansa, Multiquadrics-A scattered data approximation scheme with applications to computational fluid-dynamics II. solutions to parabolic, hyperbolic and elliptic partial differential equations, Comput. Math. Appl. 19 (1990) 147–161.

[24] G. E. Fasshauer, Solving partial differential equations by collocation with radial basis functions, Vanderbilt University Press, Nashville, 1997.

[25] G. Liu, Y. Gu, A local radial point interpolation method (LRPIM) for free vibration analysis of 2-d solids, J. Sound. Vib. 246 (2001) 29–46.

[26] G. R. Liu, G. Y. Zhang, Y. T. Gu, Y. Y. Wang, A meshfree radial point interpolation method (RPIM) for three-dimensional solids, Comput. Mech. 36 (2005) 421–430.

[27] M. Dehghan, R. Salehi, A meshless local Petrov-Galerkin method for the time-dependent Maxwell equations, J. Comput. Appl. Math. 268 (2014) 93-110.

[28] J. Li, Unified Analysis of Leap-Frog Methods for Solving Time-Domain Maxwell's Equations in Dispersive Media, J. Sci. Comput. 47 (2011) 1-26.