CrossMark

ORIGINAL RESEARCH

An Efficient Modified Haar Wavelet Collocation Method for Numerical Solution of Two-Dimensional Elliptic PDEs

Imran Aziz[1] · **Siraj-ul-Islam**[2,3]

Abstract In this work, we present a new method for solving elliptic partial differential equations using Haar wavelet. This work improves the earlier work (Aziz et al. in Appl Math Model 37:676–694, 2013) in terms of efficiency and contains an extension to nonlinear elliptic partial differential equations as well. In this paper the earlier algorithm (Aziz et al. in Appl Math Model 37:676–694, 2013) has been modified by starting the approximation with a fourth order mixed derivative rather than approximation of the second order derivatives with respect to x and y separately which results in a more efficient algorithm than the earlier algorithm. The use of Kronecker tensor products makes the new algorithm robust and easier to implement in a programming language. A distinguishing feature of the new method is that it can be applied to a variety of boundary conditions with a little modification of the program. The method is tested on several benchmark linear as well as nonlinear models. The numerical results show convergence, simple applicability and efficiency of the method.

Introduction

In this paper, we propose a numerical method based on Haar wavelet to obtain solution of the following elliptic partial differential equations (EPDEs):

$$L(\mathbf{u}) = Q(x, y),$$
$$\Gamma(\mathbf{u}) = H(x, y),$$

(1)

✉ Imran Aziz
  imran_aziz@upesh.edu.pk

[1] Department of Mathematics, University of Peshawar, Peshawar, Pakistan

[2] Department of Basic Sciences, University of Engineering and Technology, Peshawar, Pakistan

[3] Department of Mathematics, South Asian University, Akbar Bhawan, New Delhi, India

where L is a differential operator (both linear and nonlinear) of the following form:

$$L = F(\nabla, \nabla^2), \tag{2}$$

Γ is the boundary operator and the functions $Q(x, y)$ and $H(x, y)$ are the source functions. The function **u** is the solution of the EPDE (1) defined on two-dimensional rectangular or square domain.

EPDEs have been used widely in applied mathematics, physics and engineering. The EPDEs are used to model real physical situations such as, the flow of air pollutants, temperature deflection, electrostatic potential, velocity potential, stream function, fluid flow [1,2], etc. Analytical solutions of the EPDEs can be found only for simple models. When the model equations are closely related to experimental and practical situations then analytical solutions rarely exist and alternative numerical methods are used for numerical simulations of such model equations. Several numerical methods have been developed by different researchers for numerical solution of EPDEs [3–5]. Mostly numerical methods convert the model equations containing partial differential equations into discretized model equations that appear in the form of a set of algebraic linear or nonlinear equations.

Keeping in view the size of the problem and the advantages of a direct solver, we prefer LU-factorization as it aims to calculate exact solution in a finite number of operations. However for very large scale problems, the main limitation of direct solvers is the large memory requirement needed for mathematical operations. In contrast to this, iterative solvers are memory efficient but they do not always work well in all situations. Performances of the iterative solvers are sometime hampered by convergence issues, the nature of the governing equation being solved and other physical dynamics, like dealing with highly oscillatory solutions [6].

Due to the excellent properties of the Haar wavelet, a recent surge has been witnessed in the application of Haar wavelet (see [7,8] and the references therein). Haar wavelet based approximation methods have attracted attention for various engineering and scientific problems [7,9–17]. In [7] Haar wavelet collocation method (HWCM) has been used for linear elliptic PDEs. The major stumbling block in the implementation of the algorithm [7] is the computational cost of the algorithm which is prohibitively large and practically renders the method ineffective when implemented either on a denser grid or on a system of PDEs in higher dimensions. In the current work, we propose a modification of HWCM. In modified HWCM the issue of the cost of the algorithm [7] has been addressed by Haar wavelet approximation of fourth order mixed derivative instead of second order derivatives. Another advantage of the current method is the use of Kronecker tensor products which results in a very easy and simple implementation of the algorithm in a computer programming language. The new algorithm is implemented on two-dimensional steady state linear and nonlinear PDEs. A distinguishing feature of the method is that it can be applied to a variety of boundary conditions (BCs) with a little modification.

Suppose **A** is an $m \times n$ matrix and **B** is a $p \times q$ matrix, then the Kronecker tensor product of **A** and **B** denoted by $\mathbf{A} \otimes \mathbf{B}$ is the matrix of order $mp \times nq$ and is defined as follows:

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & a_{12}\mathbf{B} & \cdots & a_{1n}\mathbf{B} \\ a_{21}\mathbf{B} & a_{22}\mathbf{B} & \cdots & a_{2n}\mathbf{B} \\ \cdots & \cdots & \cdots & \cdots \\ a_{m1}\mathbf{B} & a_{m2}\mathbf{B} & \cdots & a_{mn}\mathbf{B} \end{bmatrix}.$$

Note that in MATLAB the Kronecker tensor product can be implemented easily and efficiently using the built-in function *kron*().

The rest of the paper is distributed as follows. In Sect. 2 definition of Haar wavelet is discussed. In Sect. 3, we discuss the numerical method along with different kinds of BCs. The numerical results are compared with the results obtained from other existed methods in Sect. 4. Some conclusions are drawn in the Sect. 5.

Haar Wavelet

A wavelet family $(\psi_{j,i}(x))_{j\in\mathbb{N}, i\in\mathbb{Z}}$ is an orthonormal subfamily of the Hilbert space $L^2(\mathbb{R})$ with the property that all functions in the wavelet family are generated from a fixed function ψ called mother wavelet through dilations and translations. The wavelet family satisfies the following relation:

$$\psi_{j,i}(x) = 2^{j/2}\psi(2^j x - i).$$

The Haar wavelet family defined on the interval $[0, 1)$ consists of the following functions:

$$h_1(x) = \begin{cases} 1 & \text{for } x \in [0, 1) \\ 0 & \text{elsewhere} \end{cases}$$

and

$$h_i(x) = \begin{cases} 1 & \text{for } x \in [\alpha, \beta) \\ -1 & \text{for } x \in [\beta, \gamma) \\ 0 & \text{elsewhere}, \quad i = 2, 3, \ldots, \end{cases}$$

where

$$\alpha = \frac{k}{m}, \quad \beta = \frac{(k+0.5)}{m}, \quad \gamma = \frac{(k+1)}{m};$$
$$m = 2^j, \quad j = 0, 1, \ldots, \quad k = 0, 1, \ldots, m-1.$$

The integer j indicates the level of the wavelet and k is the translation parameter. The relation between i, m and k is given by $i = m + k + 1$. The function $h_1(x)$ is called scaling function whereas $h_2(x)$ is the mother wavelet for the Haar wavelet family.

Any square integrable function $f(x)$ defined on $[0, 1)$ can be expressed as a linear combination of members of Haar wavelet family and is given as follows:

$$f(x) = \sum_{i=1}^{\infty} a_i h_i(x),$$

where a_i are constants.

For approximation purpose we consider a maximum value J of the integer j, level of the Haar wavelet in the above definition. The integer J is then called maximum level of resolution. We also define the integer $M = 2^J$. With these notations any square integrable function $f(x)$ defined on $[0, 1)$ can be approximated as linear combination of finite members of Haar wavelet family and is given as follows:

$$f(x) \approx \sum_{i=1}^{2M} a_i h_i(x). \tag{3}$$

The following notations are introduced:

$$p_{i,1}(x) = \int_0^x h_i(x')\,dx',$$

and

$$p_{i,n+1}(x) = \int_0^x p_{i,n}(x')\,dx', \quad n = 1, 2, \dots$$

These integrals can be evaluated using the definition of Haar wavelet and are given as follows:

$$p_{i,n}(x) = \begin{cases} 0 & \text{for } x \in [0, \alpha) \\ \frac{1}{n!}(x - \alpha)^n & \text{for } x \in [\alpha, \beta) \\ \frac{1}{n!}\left[(x - \alpha)^n - 2(x - \beta)^n\right] & \text{for } x \in [\beta, \gamma) \\ \frac{1}{n!}\left[(x - \alpha)^n - 2(x - \beta)^n + (x - \gamma)^n\right] & \text{for } x \in [\gamma, 1), \quad n = 1, 2, \dots \end{cases}$$

where $i = 2, 3, \dots$ For $i = 1$, we have

$$p_{1,n}(x) = \frac{x^n}{n!}, \quad n = 1, 2, \dots$$

Numerical Method

In this section we will discuss a new collocation method based on Haar wavelet for EPDEs having Dirichlet, Neumann, mixed and periodic types of BCs. For Haar wavelet approximation the following collocation points are used:

$$x_k = \frac{k - 0.5}{n}, \quad k = 1, 2, \dots, N, \tag{4}$$

$$y_l = \frac{l - 0.5}{n}, \quad l = 1, 2, \dots, N, \tag{5}$$

where $N = 2M$ and M is the maximum level of resolution of the Haar wavelet. The computation domain for the given EPDEs (1) is $[0, 1] \times [0, 1]$ but can be extended to $[a, b] \times [a, b], \forall\, a, b \in \mathbb{R}$ through linear transformation.

Assume the following Haar wavelet approximation:

$$\frac{\partial^4 u}{\partial x^2 \partial y^2} = \sum_{i=1}^N \sum_{j=1}^N \lambda_{ij} h_i(x) h_j(y). \tag{6}$$

Integrating Eq. (6) with respect to y, we obtain

$$\frac{\partial^3 u}{\partial x^2 \partial y} = \frac{\partial^4 u}{\partial x^2 \partial y}(x, 0) + \sum_{i=1}^N \sum_{j=1}^N \lambda_{ij} h_i(x) p_{j,1}(y), \tag{7}$$

$$\frac{\partial^2 u}{\partial x^2} = \frac{\partial^4 u}{\partial x^2}(x, 0) + y\frac{\partial^4 u}{\partial x^2 \partial y}(x, 0) + \sum_{i=1}^N \sum_{j=1}^N \lambda_{ij} h_i(x) p_{j,2}(y). \tag{8}$$

Using Haar approximations for the single-valued functions $\frac{\partial^4 u}{\partial x^2}(x, 0)$ and $\frac{\partial^4 u}{\partial x^2 \partial y}(x, 0)$, we can write

$$\frac{\partial^2 u}{\partial x^2} = \sum_{i=1}^{N} \alpha_i h_i(x) + y \sum_{i=1}^{N} \beta_i h_i(x) + \sum_{i=1}^{N} \sum_{j=1}^{N} \lambda_{ij} h_i(x) p_{j,2}(y). \tag{9}$$

Substituting the collocation points defined in Eqs. (4) and (5), we have

$$\frac{\partial^2 u}{\partial x^2}(x_k, y_l) = \sum_{i=1}^{N} \alpha_i h_i(x_k) + y_l \sum_{i=1}^{N} \beta_i h_i(x_l)$$

$$+ \sum_{i=1}^{N} \sum_{j=1}^{N} \lambda_{ij} h_i(x_k) p_{j,2}(y_l), \quad k, l = 1, 2, \ldots, N. \tag{10}$$

The matrix $\mathbf{H}$ of Haar functions having order $N \times N$ is defined as follows:

$$\mathbf{H} = [h_{ji}] = [h_i(x_j)]. \tag{11}$$

In a similar manner, we define

$$\begin{aligned} \mathbf{P_1} &= [p_{i,1}(x_j)], \\ \mathbf{P_2} &= [p_{i,2}(x_j)]. \end{aligned} \tag{12}$$

Thus, in matrix notation Eq. (10) can be written as

$$\mathbf{u_{xx}} = (\mathbf{H} \otimes \mathbf{1}_N)\boldsymbol{\alpha} + (\mathbf{H} \otimes \mathbf{y})\boldsymbol{\beta} + (\mathbf{H} \otimes \mathbf{P_2})\boldsymbol{\lambda}, \tag{13}$$

where

$$\mathbf{u_{xx}} = \left[\frac{\partial^2 u}{\partial x^2}(x_1, y_1), \ldots, \frac{\partial^2 u}{\partial x^2}(x_1, y_N), \ldots, \frac{\partial^2 u}{\partial x^2}(x_N, y_1), \ldots, \frac{\partial^2 u}{\partial x^2}(x_N, y_N)\right]^T,$$

$$\mathbf{1}_N = [1, 1, \ldots, 1]^T, \quad \boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_N]^T, \quad \boldsymbol{\beta} = [\beta_1, \beta_2, \ldots, \beta_N]^T, \tag{14}$$

$$\mathbf{y} = [y_1, y_2, \ldots, y_N]^T,$$

$$\boldsymbol{\lambda} = [\lambda_{11}, \lambda_{12}, \ldots, \lambda_{1N}, \lambda_{21}, \lambda_{22}, \ldots, \lambda_{2N}, \ldots, \lambda_{N1}, \lambda_{N2}, \ldots, \lambda_{NN}]^T.$$

Similarly, integrating Eq. (6) with respect to x, we obtain

$$\mathbf{u_{yy}} = (\mathbf{1}_N \otimes \mathbf{H})\boldsymbol{\gamma} + (\mathbf{x} \otimes \mathbf{H})\boldsymbol{\delta} + (\mathbf{P_2} \otimes \mathbf{H})\boldsymbol{\lambda}, \tag{15}$$

where

$$\boldsymbol{\gamma} = [\gamma_1, \gamma_2, \ldots, \gamma_N]^T, \quad \boldsymbol{\delta} = [\delta_1, \delta_2, \ldots, \delta_N]^T, \quad \mathbf{x} = [x_1, x_2, \ldots, x_N]^T. \tag{16}$$

The numerical method for different types of BCs will be explained separately in the following subsections.

Dirichlet BCs

The Dirichlet BCs for EPDEs are defined as follows:

$$u(0, y) = g(0, y), \quad u(1, y) = g(1, y), \quad y \in [0, 1), \tag{17}$$

$$u(x, 0) = g(x, 0), \quad u(x, 1) = g(x, 1), \quad x \in [0, 1). \tag{18}$$

Integrating Eq. (10) and using the BCs, we obtain

$$
\begin{aligned}
\mathbf{u_x} &= \mathbf{1}_N \otimes (g(1, \mathbf{y}) - g(0, \mathbf{y})) + ((\mathbf{P_1} - \mathbf{1}_N \mathbf{p^T}) \otimes \mathbf{1}_N)\boldsymbol{\alpha} + ((\mathbf{P_1} - \mathbf{1}_N \mathbf{p^T}) \otimes \mathbf{y})\boldsymbol{\beta} \\
&\quad + ((\mathbf{P_1} - \mathbf{1}_N \mathbf{p^T}) \otimes \mathbf{P_2})\boldsymbol{\lambda},
\end{aligned}
\tag{19}
$$

$$
\begin{aligned}
\mathbf{u} &= \mathbf{1}_N \otimes g(0, \mathbf{y}) + \mathbf{x} \otimes (g(1, \mathbf{y}) - g(0, \mathbf{y})) + ((\mathbf{P_2} - \mathbf{x} \mathbf{p^T}) \otimes \mathbf{1}_N)\boldsymbol{\alpha} \\
&\quad + ((\mathbf{P_2} - \mathbf{x} \mathbf{p^T}) \otimes \mathbf{y})\boldsymbol{\beta} + ((\mathbf{P_2} - \mathbf{x} \mathbf{p^T}) \otimes \mathbf{P_2})\boldsymbol{\lambda},
\end{aligned}
\tag{20}
$$

where

$$
\mathbf{p} = [p_{1,2}(1), p_{2,2}(1), \ldots, p_{N,2}(1)]^T.
\tag{21}
$$

Similarly,

$$
\begin{aligned}
\mathbf{u_y} &= (g(\mathbf{x}, 1) - g(\mathbf{x}, 0)) \otimes \mathbf{1}_N + (\mathbf{1}_N \otimes (\mathbf{P_1} - \mathbf{1}_N \mathbf{p^T}))\boldsymbol{\gamma} + (\mathbf{x} \otimes (\mathbf{P_1} - \mathbf{1}_N \mathbf{p^T}))\boldsymbol{\delta} \\
&\quad + (\mathbf{P_2} \otimes (\mathbf{P_1} - \mathbf{1}_N \mathbf{p^T}))\boldsymbol{\lambda},
\end{aligned}
\tag{22}
$$

$$
\begin{aligned}
\mathbf{u} &= g(\mathbf{x}, 0) \otimes \mathbf{1}_N + (g(\mathbf{x}, 1) - g(\mathbf{x}, 0)) \otimes \mathbf{y} + (\mathbf{1}_N \otimes (\mathbf{P_2} - \mathbf{y} \mathbf{p^T}))\boldsymbol{\gamma} \\
&\quad + (\mathbf{x} \otimes (\mathbf{P_2} - \mathbf{y} \mathbf{p^T}))\boldsymbol{\delta} + (\mathbf{P_2} \otimes (\mathbf{P_2} - \mathbf{y} \mathbf{p^T}))\boldsymbol{\lambda}.
\end{aligned}
\tag{23}
$$

Discretizing Eq. (2) and substituting the expressions of $\mathbf{u}$ and its partial derivatives, a system of equations having N^2 equations and $N^2 + 4N$ unknowns is obtained. The additional $4N$ equations are obtained by comparing the two expressions of u given in Eqs. (20) and (23) respectively and substituting the values $x = 0$, $x = 1$, $y = 0$ and $y = 1$ one by one. These equations are given below.

$$
\begin{aligned}
(\mathbf{P_2} - \mathbf{y} \mathbf{p^T})\boldsymbol{\gamma} &= g(0, \mathbf{y}) - g(0, 0)\mathbf{1}_N - (g(0, 1) - g(0, 0))\mathbf{y}, \\
(\mathbf{P_2} - \mathbf{y} \mathbf{p^T})\boldsymbol{\gamma} &+ (\mathbf{P_2} - \mathbf{y} \mathbf{p^T})\boldsymbol{\delta} + (\mathbf{p^T} \otimes (\mathbf{P_2} - \mathbf{y} \mathbf{p^T}))\boldsymbol{\lambda} \\
&= g(1, \mathbf{y}) - g(1, 0)\mathbf{1}_N - (g(1, 1) - g(1, 0))\mathbf{y}, \\
(\mathbf{P_2} - \mathbf{x} \mathbf{p^T})\boldsymbol{\alpha} &= g(\mathbf{x}, 0) - g(0, 0)\mathbf{1}_N - (g(1, 0) - g(0, 0))\mathbf{x}, \\
(\mathbf{P_2} - \mathbf{x} \mathbf{p^T})\boldsymbol{\alpha} &+ (\mathbf{P_2} - \mathbf{x} \mathbf{p^T})\boldsymbol{\beta} + ((\mathbf{P_2} - \mathbf{x} \mathbf{p^T}) \otimes \mathbf{p^T})\boldsymbol{\lambda} \\
&= g(\mathbf{x}, 1) - g(0, 1)\mathbf{1}_N - (g(1, 1) - g(0, 1))\mathbf{x}.
\end{aligned}
$$

Combining all these equations results in a $(N^2 + 4N) \times (N^2 + 4N)$ linear or nonlinear system. Solution of this system yields the unknowns $\boldsymbol{\lambda}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}$ and $\boldsymbol{\delta}$. The approximate solution $u(x, y)$ and its derivatives at any point of the domain can be calculated using these values.

Neumann BCs

The Neumann BCs for EPDEs are given below.

$$
\frac{\partial u}{\partial x}(0, y) = g_x(0, y), \quad \frac{\partial u}{\partial x}(1, y) = g_x(1, y), \quad y \in [0, 1),
\tag{24}
$$

$$
\frac{\partial u}{\partial y}(x, 0) = g_y(x, 0), \quad \frac{\partial u}{\partial y}(x, 1) = g_y(x, 1), \quad x \in [0, 1).
\tag{25}
$$

For Neumann BCs, we integrate Eq. (10) and obtain the following expressions.

$$
\mathbf{u_x} = \mathbf{1}_N \otimes g_x(0, \mathbf{y}) + (\mathbf{P_1} \otimes \mathbf{1}_N)\boldsymbol{\alpha} + (\mathbf{P_1} \otimes \mathbf{y})\boldsymbol{\beta} + (\mathbf{P_1} \otimes \mathbf{P_2})\boldsymbol{\lambda},
\tag{26}
$$

$$
\mathbf{u} = (\mathbf{1}_N \otimes \mathbf{I}_N)u(0, \mathbf{y}) + \mathbf{x} \otimes g_x(0, \mathbf{y}) + (\mathbf{P_2} \otimes \mathbf{1}_N)\boldsymbol{\alpha} + (\mathbf{P_2} \otimes \mathbf{y})\boldsymbol{\beta} + (\mathbf{P_2} \otimes \mathbf{P_2})\boldsymbol{\lambda},
\tag{27}
$$

where $\mathbf{I}_N$ is the identity matrix of order $N \times N$. Similarly,

$$\mathbf{u_y} = g_y(\mathbf{x}, 0) \otimes \mathbf{1}_N + (\mathbf{1}_N \otimes \mathbf{P_1})\boldsymbol{\gamma} + (\mathbf{x} \otimes \mathbf{P_1})\boldsymbol{\delta} + (\mathbf{P_2} \otimes \mathbf{P_1})\boldsymbol{\lambda}, \tag{28}$$

$$\mathbf{u} = (\mathbf{I}_N \otimes \mathbf{1}_N)u(\mathbf{x}, 0) + g_y(\mathbf{x}, 0) \otimes \mathbf{y} + (\mathbf{1}_N \otimes \mathbf{P_2})\boldsymbol{\gamma} + (\mathbf{x} \otimes \mathbf{P_2})\boldsymbol{\delta} + (\mathbf{P_2} \otimes \mathbf{P_2})\boldsymbol{\lambda}. \tag{29}$$

These expressions when substituted in Eq. (1) give rise to a system of equations having N^2 equations but $N^2 + 6N$ unknowns $\boldsymbol{\lambda}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta}, u(0, \mathbf{y})$ and $u(\mathbf{x}, 0)$. Integrating Eq. (10) and using the BCs we get the following N equations:

$$\alpha_1 \mathbf{1}_N + \beta_1 \mathbf{y} + (\mathbf{e_1^T} \otimes \mathbf{P_2})\boldsymbol{\lambda} = g_x(1, \mathbf{y}) - g_x(0, \mathbf{y}), \tag{30}$$

where $\mathbf{e_1} = [1 \ 0 \cdots 0]^{\mathbf{T}}$. Similarly,

$$\gamma_1 \mathbf{1}_N + \delta_1 \mathbf{x} + (\mathbf{P_2} \otimes \mathbf{e_1^T})\boldsymbol{\lambda} = g_y(\mathbf{x}, 1) - g_y(\mathbf{x}, 0). \tag{31}$$

Another $4N$ equations are obtained by comparing two expressions of $\mathbf{u}$ given in Eqs. (27) and (29), and substituting $x = 0$, $x = 1$, $y = 0$ and $y = 1$ one by one. These equations are given below.

$$\mathbf{I}_N u(0, \mathbf{y}) - \mathbf{P_2}\boldsymbol{\gamma} - \mathbf{1}_N u(0, 0) = g_y(0, 0)\mathbf{y},$$
$$\mathbf{I}_N u(0, \mathbf{y}) + (\mathbf{p^T} \otimes \mathbf{1}_N)\boldsymbol{\alpha} + (\mathbf{p^T} \otimes \mathbf{y})\boldsymbol{\beta} - \mathbf{P_2}\boldsymbol{\gamma} - \mathbf{P_2}\boldsymbol{\delta} - \mathbf{1}_N u(1, 0) = g_y(1, 0)\mathbf{y} - g_x(0, \mathbf{y}),$$
$$\mathbf{I}_N u(\mathbf{x}, 0) - \mathbf{P_2}\boldsymbol{\alpha} - \mathbf{1}_N u(0, 0) = g_x(0, 0)\mathbf{x},$$
$$\mathbf{I}_N u(\mathbf{x}, 0) - \mathbf{P_2}\boldsymbol{\alpha} - \mathbf{P_2}\boldsymbol{\beta} + (\mathbf{1}_n \otimes \mathbf{p^T})\boldsymbol{\gamma} + (\mathbf{x} \otimes \mathbf{p^T})\boldsymbol{\delta} - \mathbf{1}_N u(0, 1) = g_x(0, 1)\mathbf{x} - g_y(\mathbf{x}, 0). \tag{32}$$

The above $4N$ equations involve three more unknowns $u(0, 0)$, $u(1, 0)$ and $u(0, 1)$. Substituting $y = 0$ and $y = 1$ in Eq. (30), and $x = 0$ in Eq. (31), we obtain the following three equations:

$$\alpha_1 = g_x(1, 0) - g_x(0, 0)$$
$$\alpha_1 + \beta_1 + (\mathbf{e_1^T} \otimes \mathbf{p^T})\boldsymbol{\lambda} = g_x(1, 1) - g_x(0, 1) \tag{33}$$
$$\gamma_1 = g_y(0, 1) - g_y(0, 0).$$

The numerical solution is obtained in a similar way as discussed in the case of Dirichlet BCs in the previous subsection.

Mixed BCs

In this section, two types of mixed BCs will be considered.

Type I

The first type of mixed BCs for EPDEs are given as follows:

$$u(0, y) = g(0, y), \quad u(1, y) = g(1, y), \qquad y \in [0, 1), \tag{34}$$

$$\frac{\partial u}{\partial y}(x, 0) = g_y(x, 0), \quad \frac{\partial u}{\partial y}(x, 1) = g_y(x, 1), \qquad x \in [0, 1). \tag{35}$$

Using BCs (34) the following expressions are obtained.

$$\mathbf{u_x} = \mathbf{1}_N \otimes (g(1, \mathbf{y}) - g(0, \mathbf{y})) + ((\mathbf{P_1} - \mathbf{1}_N \mathbf{p^T}) \otimes \mathbf{1}_N)\boldsymbol{\alpha} + ((\mathbf{P_1} - \mathbf{1}_N \mathbf{p^T}) \otimes \mathbf{y})\boldsymbol{\beta}$$
$$+((\mathbf{P_1} - \mathbf{1}_N \mathbf{p^T}) \otimes \mathbf{P_2})\boldsymbol{\lambda}, \tag{36}$$

$$\mathbf{u} = \mathbf{1}_N \otimes g(0, \mathbf{y}) + \mathbf{x} \otimes (g(1, \mathbf{y}) - g(0, \mathbf{y})) + ((\mathbf{P_2} - \mathbf{x}\mathbf{p^T}) \otimes \mathbf{1}_N)\boldsymbol{\alpha}$$
$$+((\mathbf{P_2} - \mathbf{x}\mathbf{p^T}) \otimes \mathbf{y})\boldsymbol{\beta} + ((\mathbf{P_2} - \mathbf{x}\mathbf{p^T}) \otimes \mathbf{P_2})\boldsymbol{\lambda}. \tag{37}$$

Similarly, using BCs (35), we obtain

$$\mathbf{u_y} = g_y(\mathbf{x}, 0) \otimes \mathbf{1}_N + (\mathbf{1}_N \otimes \mathbf{P_1})\boldsymbol{\gamma} + (\mathbf{x} \otimes \mathbf{P_1})\boldsymbol{\delta} + (\mathbf{P_2} \otimes \mathbf{P_1})\boldsymbol{\lambda}, \tag{38}$$

$$\mathbf{u} = (\mathbf{I}_N \otimes \mathbf{1}_N)u(\mathbf{x}, 0) + g_y(\mathbf{x}, 0) \otimes \mathbf{y} + (\mathbf{1}_N \otimes \mathbf{P_2})\boldsymbol{\gamma} + (\mathbf{x} \otimes \mathbf{P_2})\boldsymbol{\delta} + (\mathbf{P_2} \otimes \mathbf{P_2})\boldsymbol{\lambda}. \tag{39}$$

These expressions when substituted in Eq. (1) give rise to a system of equations having N^2 equations but $N^2 + 5N$ unknowns $\boldsymbol{\lambda}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta}$ and $u(\mathbf{x}, 0)$. Integrating Eq. (15) and using the BCs we get the following N equations.

$$\boldsymbol{\gamma_1}\mathbf{1}_N + \boldsymbol{\delta_1}\mathbf{x} + (\mathbf{P_2} \otimes \mathbf{e_1^T})\boldsymbol{\lambda} = g_y(\mathbf{x}, 1) - g_y(\mathbf{x}, 0). \tag{40}$$

Comparing the two expressions of $\mathbf{u}$ and substituting values of $x = 0, 1$ and $y = 0, 1$ we obtain the following $4N$ equations.

$$g(0, \mathbf{y}) = \mathbf{P_2}\boldsymbol{\gamma} + \mathbf{1}_n u(0, 0) + g_y(0, 0)\mathbf{y},$$
$$g(1, \mathbf{y}) = \mathbf{P_2}\boldsymbol{\gamma} + \mathbf{P_2}\boldsymbol{\delta} + \mathbf{1}_N u(1, 0) + g_y(1, 0)\mathbf{y} + (\mathbf{1}_n\mathbf{p^T} \otimes \mathbf{P_2})\boldsymbol{\lambda},$$
$$g(0, 0)\mathbf{1}_N + (g(1, 0) - g(0, 0))\mathbf{x} + (\mathbf{P_2} - \mathbf{x}\mathbf{p^T})\boldsymbol{\alpha} = u(\mathbf{x}, 0),$$
$$g(0, 1)\mathbf{1}_N + (g(1, 1) - g(0, 1))\mathbf{x} + (\mathbf{P_2} - \mathbf{x}\mathbf{p^T})\boldsymbol{\alpha} + (\mathbf{P_2} - \mathbf{x}\mathbf{p^T})\boldsymbol{\beta} \tag{41}$$
$$+((\mathbf{P_2} - \mathbf{x}\mathbf{p^T}) \otimes \mathbf{p^T})\boldsymbol{\lambda}$$
$$= \mathbf{I}_N u(\mathbf{x}, 0) + (\mathbf{1}_N \otimes \mathbf{p^T})\boldsymbol{\gamma} + (\mathbf{x} \otimes \mathbf{p^T})\boldsymbol{\delta} + g_y(\mathbf{x}, 0) + (\mathbf{P_2} \otimes \mathbf{1}_N \mathbf{p^T})\boldsymbol{\lambda}.$$

The solution procedure is similar to previous cases.

Type II

The second type of mixed BCs considered here are given below.

$$\frac{\partial u}{\partial x}(0, y) = g_x(0, y), \quad u(1, y) = g(1, y), \quad y \in [0, 1), \tag{42}$$

$$\frac{\partial u}{\partial y}(x, 0) = g_y(x, 0), \quad u(x, 1) = g(x, 1), \quad x \in [0, 1). \tag{43}$$

Different expressions of $\mathbf{u}$ and its derivatives are given as follows:

$$\mathbf{u_x} = \mathbf{1}_N \otimes g_x(0, \mathbf{y}) + (\mathbf{P_1} \otimes \mathbf{1}_N)\boldsymbol{\alpha} + (\mathbf{P_1} \otimes \mathbf{y})\boldsymbol{\beta} + (\mathbf{P_1} \otimes \mathbf{P_2})\boldsymbol{\lambda}, \tag{44}$$

$$\mathbf{u} = \mathbf{1}_N \otimes g(1, \mathbf{y}) - (\mathbf{1}_N - \mathbf{x}) \otimes g_x(0, \mathbf{y})$$
$$-((\mathbf{1}_N\mathbf{p^T} - \mathbf{P_2}) \otimes \mathbf{1}_N)\boldsymbol{\alpha} - ((\mathbf{1}_N\mathbf{p^T} - \mathbf{P_2}) \otimes \mathbf{y})\boldsymbol{\beta}$$
$$-((\mathbf{1}_N\mathbf{p^T} - \mathbf{P_2}) \otimes \mathbf{P_2})\boldsymbol{\lambda}, \tag{45}$$

$$\mathbf{u_y} = g_y(\mathbf{x}, 0) \otimes \mathbf{1}_N + (\mathbf{1}_N \otimes \mathbf{P_1})\boldsymbol{\gamma} + (\mathbf{x} \otimes \mathbf{P_1})\boldsymbol{\delta} + (\mathbf{P_2} \otimes \mathbf{P_1})\boldsymbol{\lambda}, \tag{46}$$

$$\mathbf{u} = g(\mathbf{x}, 1) \otimes \mathbf{1}_N - g_y(\mathbf{x}, 0) \otimes (\mathbf{1}_N - \mathbf{y}) - (\mathbf{1}_N \otimes (\mathbf{1}_N\mathbf{p^T} - \mathbf{P_2}))\boldsymbol{\gamma}$$
$$-(\mathbf{x} \otimes (\mathbf{1}_N\mathbf{p^T} - \mathbf{P_2}))\boldsymbol{\delta} - (\mathbf{P_2} \otimes (\mathbf{1}_N\mathbf{p^T} - \mathbf{P_2}))\boldsymbol{\lambda}, \tag{47}$$

Comparing the two different expressions of **u** and substituting $x = 0, 1$ and $y = 0, 1$ the following equations are obtained:

$$
\begin{aligned}
g(1, \mathbf{y}) &- g_x(0, \mathbf{y}) - \mathbf{1}_N \mathbf{p}^\mathbf{T} \boldsymbol{\alpha} - \mathbf{y}\mathbf{p}^\mathbf{T} \boldsymbol{\beta} - (\mathbf{p}^\mathbf{T} \otimes \mathbf{P_2})\boldsymbol{\lambda} \\
&= g(0, 1)\mathbf{1}_N - g_y(0, 0)(\mathbf{1}_N - \mathbf{y}) - (\mathbf{1}_N \mathbf{p}^\mathbf{T} - \mathbf{P_2})\boldsymbol{\gamma}, \\
g(1, \mathbf{y}) &= g(1, 1)\mathbf{1}_N - g_y(1, 0)(1 - \mathbf{y}) - (\mathbf{1}_N \mathbf{p}^\mathbf{T} - \mathbf{P_2})\boldsymbol{\gamma} - (\mathbf{1}_N \mathbf{p}^\mathbf{T} \\
&\quad - \mathbf{P_2})\boldsymbol{\delta} - (\mathbf{p}^\mathbf{T} \otimes (\mathbf{1}_N \mathbf{p}^\mathbf{T} - \mathbf{P_2}))\boldsymbol{\lambda}, \\
g(1, 0)\mathbf{1}_N &- g_x(0, 0)(\mathbf{1}_N - \mathbf{x}) - (\mathbf{1}_N \mathbf{p}^\mathbf{T} - \mathbf{P_2})\boldsymbol{\alpha} = g(\mathbf{x}, 1) - g_y(\mathbf{x}, 0) \\
&\quad - \mathbf{1}_N \mathbf{p}^\mathbf{T} \boldsymbol{\gamma} - \mathbf{x}\mathbf{p}^\mathbf{T} \boldsymbol{\delta} - (\mathbf{P_2} \otimes \mathbf{p}^\mathbf{T})\boldsymbol{\lambda}, \\
g(1, 1)\mathbf{1}_N &- g_x(0, 1)(\mathbf{1}_N - \mathbf{x}) - (\mathbf{1}_N \mathbf{p}^\mathbf{T} - \mathbf{P_2})\boldsymbol{\alpha} - (\mathbf{1}_N \mathbf{p}^\mathbf{T} - \mathbf{P_2})\boldsymbol{\beta} \\
&\quad - ((\mathbf{1}_N \mathbf{p}^\mathbf{T} - \mathbf{P_2}) \otimes \mathbf{p}^\mathbf{T})\boldsymbol{\lambda} = g(\mathbf{x}, 1).
\end{aligned}
\tag{48}
$$

The solution procedure is similar.

Periodic BCs

The periodic BCs for EPDEs are given as follows:

$$
\begin{aligned}
u(0, y) = u(1, y), &\quad \frac{\partial u}{\partial x}(0, y) = \frac{\partial u}{\partial x}(1, y), &\quad y \in [0, 1), \\
u(x, 0) = u(x, 1), &\quad \frac{\partial u}{\partial y}(x, 0) = \frac{\partial u}{\partial y}(x, 1), &\quad x \in [0, 1).
\end{aligned}
\tag{49}
$$

Integrating Eq. (10) and using BCs the following expressions are obtained:

$$
\begin{aligned}
\mathbf{u_x} &= ((\mathbf{P_1} - \mathbf{1}_N \mathbf{p}^\mathbf{T}) \otimes \mathbf{1}_N)\boldsymbol{\alpha} + ((\mathbf{P_1} - \mathbf{1}_N \mathbf{p}^\mathbf{T}) \otimes \mathbf{y})\boldsymbol{\beta} \\
&\quad + ((\mathbf{P_1} - \mathbf{1}_N \mathbf{p}^\mathbf{T}) \otimes \mathbf{P_2})\boldsymbol{\lambda},
\end{aligned}
\tag{50}
$$

$$
\begin{aligned}
\mathbf{u} &= (\mathbf{1}_N \otimes \mathbf{I}_N)u(0, \mathbf{y}) + ((\mathbf{P_2} - \mathbf{x}\mathbf{p}^\mathbf{T}) \otimes \mathbf{1}_N)\boldsymbol{\alpha} \\
&\quad + ((\mathbf{P_2} - \mathbf{x}\mathbf{p}^\mathbf{T}) \otimes \mathbf{y})\boldsymbol{\beta} + ((\mathbf{P_2} - \mathbf{x}\mathbf{p}^\mathbf{T}) \otimes \mathbf{P_2})\boldsymbol{\lambda}.
\end{aligned}
\tag{51}
$$

Similarly,

$$
\begin{aligned}
\mathbf{u_y} &= (\mathbf{1}_N \otimes (\mathbf{P_1} - \mathbf{1}_N \mathbf{p}^\mathbf{T}))\boldsymbol{\gamma} + (\mathbf{x} \otimes (\mathbf{P_1} - \mathbf{1}_N \mathbf{p}^\mathbf{T}))\boldsymbol{\delta} \\
&\quad + (\mathbf{P_2} \otimes (\mathbf{P_1} - \mathbf{1}_N \mathbf{p}^\mathbf{T}))\boldsymbol{\lambda},
\end{aligned}
\tag{52}
$$

$$
\begin{aligned}
\mathbf{u} &= (\mathbf{I}_N \otimes \mathbf{1}_N)u(\mathbf{x}, 0) + (\mathbf{1}_N \otimes (\mathbf{P_2} - \mathbf{y}\mathbf{p}^\mathbf{T}))\boldsymbol{\gamma} \\
&\quad + (\mathbf{x} \otimes (\mathbf{P_2} - \mathbf{y}\mathbf{p}^\mathbf{T}))\boldsymbol{\delta} + (\mathbf{P_2} \otimes (\mathbf{P_2} - \mathbf{y}\mathbf{p}^\mathbf{T}))\boldsymbol{\lambda}.
\end{aligned}
\tag{53}
$$

Additional equations obtained are given below.

$$
\begin{aligned}
&\alpha_1 \mathbf{1}_N + \beta_1 \mathbf{y} + (\mathbf{e_1^T} \otimes \mathbf{P_2})\boldsymbol{\lambda} = 0, \\
&\gamma_1 \mathbf{1}_N + \delta_1 \mathbf{x} + (\mathbf{P_2} \otimes \mathbf{e_1^T})\boldsymbol{\lambda} = 0, \\
&(\mathbf{P_2} - \mathbf{y}\mathbf{p}^\mathbf{T})\boldsymbol{\gamma} - \mathbf{I}_N u(0, \mathbf{y}) + u(0, 0) = 0, \\
&(\mathbf{p}^\mathbf{T} \otimes (\mathbf{P_2} - \mathbf{y}\mathbf{p}^\mathbf{T}))\boldsymbol{\lambda} + (\mathbf{P_2} - \mathbf{y}\mathbf{p}^\mathbf{T})\boldsymbol{\gamma} + (\mathbf{P_2} - \mathbf{y}\mathbf{p}^\mathbf{T})\boldsymbol{\delta} - \mathbf{I}_N u(0, \mathbf{y}) + u(1, 0) = 0, \\
&(\mathbf{P_2} - \mathbf{x}\mathbf{p}^\mathbf{T})\boldsymbol{\alpha} - \mathbf{I}_N u(\mathbf{x}, 0) + u(0, 0) = 0, \\
&((\mathbf{P_2} - \mathbf{x}\mathbf{p}^\mathbf{T}) \otimes \mathbf{p}^\mathbf{T})\boldsymbol{\lambda} + (\mathbf{P_2} - \mathbf{x}\mathbf{p}^\mathbf{T})\boldsymbol{\alpha} + (\mathbf{P_2} - \mathbf{x}\mathbf{p}^\mathbf{T})\boldsymbol{\beta} - \mathbf{I}_N u(\mathbf{x}, 0) + u(0, 1) = 0, \\
&\alpha_1 = 0, \\
&\alpha_1 + \beta_1 + (\mathbf{e_1^T} \otimes \mathbf{p}^\mathbf{T})\boldsymbol{\lambda} = 0,
\end{aligned}
\tag{54}
$$

$$\gamma_1 = 0;$$

The solution procedure is similar to the previous cases.

Numerical Experiments

In this section, numerical and graphical results obtained from modified HWCM is presented. The method is applied to four Test Problems with known exact solutions. Test Problem 1 is a Helmholtz equation which is a linear EPDE. This problem is considered in order to show the better efficiency of the modified HWCM as compared to HWCM [7]. Test Problems 2 and 3 are linear Poisson equations with constant and variable sources respectively. These steady-state problems have several applications in fluid dynamics, electrostatics, heat transfer and mass transfer. Finally Test Problem 4 is a generalized nonlinear Poisson equation. This problem is considered in order to show the applicability of modified HWCM to nonlinear problems as well.

The algorithm is implemented in MATLAB on Intel Core i7 processor with 32 GB RAM. The performance of the method is demonstrated using the L_∞ norm which is defined as

$$L_\infty = \max_{i,j} |u_{\text{app}}(x_i, y_j) - u_{\text{ex}}(x_i, y_j)|, \quad i, j = 1, 2, \ldots, N, \tag{55}$$

where $u_{\text{app}}(x_i, y_j)$ is the numerical and $u_{\text{ex}}(x_i, y_j)$ is the exact solution at the point (x_i, y_j).

Helmholtz Equation

Test Problem 1 Consider the Helmholtz equation [7]:

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + 900\, u(x, y) = f(x, y), \quad 0 \le x, y \le 1, \tag{56}$$

subject to Dirichlet BCs. The function $f(x, y)$ is taken such that the exact solution of the problem is $u(x, y) = e^{xy^2}$. Comparison of the CPU time of the present method with our earlier method [7] is given in Table 1. It is clear from the table that in addition to good accuracy, the new algorithm is almost 7 times faster than the previous algorithm [7].

Table 1 Comparison of CPU time of present method with method [7] for Test Problem 1

N	L_∞	CPU time (in s)	
		Modified HWCM	HWCM [7]
2×2	2.4×10^{-4}	0.003	0.004
4×4	2.3×10^{-4}	0.003	0.004
8×8	6.6×10^{-5}	0.004	0.01
16×16	4.1×10^{-5}	0.02	0.03
32×32	9.3×10^{-6}	0.1	0.4
64×64	2.4×10^{-6}	2	13
128×128	5.6×10^{-7}	91	608

Poisson Equation

Test Problem 2 (Constant source) Consider the Poisson equation:

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = Q, \quad -a \leq x \leq a, \ -b \leq y \leq b, \tag{57}$$

where Q is a constant, subject to the Dirichlet BCs:

$$\begin{aligned} u(-a, y) = 0, u(a, y) = 0, \quad -b \leq y \leq b, \\ u(x, -b) = 0, u(x, b) = 0, \quad -a \leq x \leq a. \end{aligned} \tag{58}$$

Equation (57) describes the steady-state two-dimensional diffusion (cross-wind integrated) of a pollutant in the absence of wind with constant source strength Q in a finite domain [1]. It also describes the steady-state transport of oxygen in a slab of tissue with a constant oxygen consumption rate [18]. In addition to this, it also represents the steady-state temperature distribution in a rectangular region with heat loss (production) at a constant rate Q [2].

The solution of Eq. (57) with BCs (58) is symmetric with respect to both x- and y-axis. Due to this reason Eq. (57) is solved in the region $(0 < x < a, \ 0 < y < b)$ subject to the following BCs [1]:

$$\begin{aligned} u(a, y) = 0, \frac{\partial u}{\partial x}(0, y) = 0, \quad 0 \leq y \leq b \\ u(x, b) = 0, \frac{\partial u}{\partial y}(x, 0) = 0, \quad 0 \leq x \leq a. \end{aligned} \tag{59}$$

The exact solution of the problem is given as follows [1]:

$$u(x, y) = \frac{2Q}{a} \sum_{k=0}^{\infty} \frac{(-1)^k}{(\lambda_k)^{3/2}} \left(\frac{\cosh \sqrt{\lambda_k} y}{\cosh \sqrt{\lambda_k} b} - 1 \right) \cos \left(\sqrt{\lambda_k} x \right), \tag{60}$$

where $\sqrt{\lambda_k} = \frac{(2k+1)\pi}{2a}$.

Maximum absolute errors for different number of collocation points are shown in Table 2. Like the other test problems an efficient and accurate solution has been obtained through the new algorithm.

Table 2 Maximum absolute errors and CPU time for Test Problem 2

N	L_∞	CPU time (in s)
2×2	5.9×10^{-3}	0.002
4×4	1.7×10^{-3}	0.002
8×8	4.5×10^{-4}	0.003
16×16	1.1×10^{-4}	0.01
32×32	2.8×10^{-5}	0.1
64×64	7.1×10^{-6}	2.2
128×128	1.8×10^{-6}	86

Test Problem 3 (Variable source) Consider Poisson equation with variable source:

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x, y), \quad -a \leq x \leq a, \ -b \leq y \leq b, \tag{61}$$

where source is a function of both x and y. The BCs are same as considered in Test Problem 2 which are given in Eq. (58).

Important applications of Eq. (61) occur in fluid dynamics (u is stream function and f is vorticity), electrostatics (u is electric potential and f is the ratio of charge density to dielectric constant), heat transfer (u is temperature and f is the loss/generation of heat) and mass transfer (u is concentration and f is source/sink term) [2].

The solution of Eq. (61) is expected to be symmetric in x and y [2] and therefore we consider the region $(0 < x < a, \ 0 < y < b)$ with the BCs (59).

We assume that $f(x, y)$ is the same function as the exact solution of Test Problem 2 which is given in Eq. (60). With this f, the exact solution of Eq. (61) is given below [1]:

$$\begin{aligned}
u(x, y) = \frac{Q}{a} \sum_{k=0}^{\infty} &\left[\frac{(-1)^k}{(\lambda_k)^{5/2}} \right] \left[\left\{ 2 + \sqrt{\lambda_k} y \frac{\sinh \sqrt{\lambda_k} y}{\cosh \sqrt{\lambda_k} b} \right\} \right. \\
&\left. - \left\{ 2 + \sqrt{\lambda_k} b \frac{\sinh \sqrt{\lambda_k} b}{\cosh \sqrt{\lambda_k} b} \right\} \left(\frac{\cosh \sqrt{\lambda_k} y}{\cosh \sqrt{\lambda_k} b} \right) \right] \cos\left(\sqrt{\lambda_k} x \right),
\end{aligned} \tag{62}$$

where $\sqrt{\lambda_k} = \frac{(2k+1)\pi}{2a}$.

Maximum absolute errors for different number of collocation points are shown in Table 3. It is conformed from the table that a converging solution has been reproduced by the new algorithm in a fairly less CPU time.

Generalized Nonlinear Poisson Equation

Test Problem 4 Consider a nonlinear Poisson problem [19]:

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \left(\frac{\partial u}{\partial y} \right)^2 = f(x, y), \quad 0 \leq x \leq 1, \ 0 \leq y \leq 1, \tag{63}$$

subject to the following Dirichlet BCs:

$$\begin{aligned}
u(0, y) &= 0, \quad u(1, y) = y + a, \\
u(x, 0) &= ax, \quad u(x, 1) = x(x + a),
\end{aligned} \tag{64}$$

where a is any constant. The analytical solution of the problem is $u(x, y) = x(xy + a)$. For simplicity, the value of the constant a is taken to be zero. The numerical results are shown

Table 3 Maximum absolute errors and CPU time for Test Problem 3

N	L_∞	CPU time (in s)
2×2	1.1×10^{-3}	0.002
4×4	3.3×10^{-4}	0.004
8×8	8.7×10^{-5}	0.01
16×16	2.2×10^{-5}	0.03
32×32	5.5×10^{-6}	0.17
64×64	1.4×10^{-6}	2.5
128×128	3.5×10^{-7}	88

Table 4 Maximum absolute errors and CPU time for Test Problem

N	L_∞	No. of iterations	CPU time (in s)
2×2	8.5×10^{-4}	7	0.01
4×4	3.4×10^{-4}	8	0.02
8×8	9.0×10^{-5}	10	0.035
16×16	2.3×10^{-5}	10	0.06
32×32	5.7×10^{-6}	11	1
64×64	1.4×10^{-6}	11	17
128×128	3.6×10^{-7}	11	455

in Table 4. The nonlinear system of equations obtained after substituting the Haar wavelet approximations which may be solved using either Newton's method or Broyden's method. For numerical experiments we have used Broyden's method as it is more economical in terms of CPU time. The initial guess for the Broyden's method was taken to be zero and the iterations were terminated when the convergence criterion 10^{-6} was satisfied. The number of iterations and the CPU time for different number of grid points are also shown in Table 4. It is evident from the table that the new method has obtained the desired accuracy in 11 iteration keeping the CPU time in a reasonable limit.

Conclusion

A modified Haar wavelet collocation method is presented for numerical solution of elliptic partial differential equations. The method can solve both linear as well as nonlinear EPDEs numerically. The method can be applied to different types of boundary conditions easily with a slight modification. The method is tested on several benchmark problems. It is observed that the proposed method is more efficient.

References

1. Modani, M., Sharan, M., Rao, S.: Numerical solution of elliptic partial differential equations on parallel systems. Appl. Math. Comput. **195**, 162–182 (2008)
2. Sharan, M., Kansa, E., Gupta, S.: Application of the multiquadratic method for numerical solution of elliptic partial differential equations. Appl. Math. Comput. **84**, 275–302 (1997)
3. Northrop, P.W., Ramachandran, P., Schiesser, W.E., Subramanian, V.R.: A robust false transient method of lines for elliptic partial differential equations. Chem. Eng. Sci. **90**, 32–39 (2013)
4. Saridakis, Y., Sifalakis, A., Papadopoulou, E.: Efficient numerical solution of the generalized Dirichlet-Neumann map for linear elliptic PDEs in regular polygon domains. J. Comput. Appl. Math. **236**, 2515–2528 (2012)
5. Huang, C.-S., Yen, H.-D., Cheng, A.H.-D.: On the increasingly flat radial basis function and optimal shape parameter for the solution of elliptic PDEs. Eng. Anal. Bound. Elem. **34**, 802–809 (2010)
6. Martinsson, P.: A direct solver for variable coefficient elliptic PDEs discretized via a composite spectral collocation method. J. Comput. Phy. **242**, 460–479 (2013)
7. Aziz, I., Siraj-ul-Islam, Sarler, B.: Wavelets collocation methods for the numerical solution of elliptic BV problems. Appl. Math. Model. **37**, 676–694 (2013)
8. Lepik, U.: Haar Wavelets with Applications. Springer, Berlin (2014)
9. Lepik, U.: Solving PDEs with the aid of two-dimensional Haar wavelets. Comput. Math. Appl. **61**, 1873–1879 (2011)

10. Siraj-ul-Islam, Aziz, I., Al-Fhaid, A., Shah, A.: A numerical assessment of parabolic partial differential equations using Haar and Legendre wavelets. Appl. Math. Model. **37**, 9455–9481 (2013)
11. Lepik, U.: Numerical solution of differential equations using Haar wavelets. Math. Comput. Simul. **68**, 127–143 (2005)
12. Siraj-ul-Islam, Aziz, I., Sarler, B.: The numerical solution of second-order boundary-value problems by collocation method with the Haar wavelets. Math. Comput. Model. **50**, 1577–1590 (2010)
13. Aziz, I., Siraj-ul-Islam: New algorithms for numerical solution of nonlinear Fredholm and Volterra integral equations using Haar wavelets. J. Comput. Appl. Math. **239**, 333–345 (2013)
14. Siraj-ul-Islam, Aziz, I., Fayyaz, M.: A new approach for numerical solution of integro-differential equations via Haar wavelets. Int. J. Comput. Math. **90**, 1971–1989 (2013)
15. Aziz, I., Siraj-ul-Islam, Khan, F.: A new method based on Haar wavelet for the numerical solution of two-dimensional nonlinear integral equations. J. Comput. Appl. Math. **272**, 70–80 (2014)
16. Siraj-ul-Islam, Aziz, I., Haq, F.: A comparative study of numerical integration based on Haar wavelets and hybrid functions. Comput. Math. Appl. **59**(6), 2026–2036 (2010)
17. Siraj-ul-Islam, Aziz, I., Al-Fhaid, A.: An improved method based on Haar wavelets for numerical solution of nonlinear integral and integro-differential equations of first and higher orders. J. Comput. Appl. Math. **260**, 449–469 (2014)
18. Sharan, M., Singh, M., Sud, I.: Mathematical modeling of oxygen transport in the systemic circulation in hyperbaric environment. Front. Med. Biol. Eng. **3**, 27–44 (1991)
19. Bourantas, G., Burganos, V.: An implicit meshless scheme for the solution of transient non-linear poisson-type equations. Eng. Anal. Bound. Elem. **37**, 1117–1126 (2013)