


Article

Numerical Solutions of Variable Coefficient Higher-Order Partial Differential Equations Arising in Beam Models

Abdul Ghafoor ¹, Sirajul Haq ², Manzoor Hussain ³, Thabet Abdeljawad ^{4,5,*}  and Manar A. Alqudah ⁶

¹ Institute of Numerical Sciences, Kohat University of Science and Technology, Kohat 26000, KP, Pakistan; abdulghafoor@kust.edu.pk

² Faculty of Engineering Sciences, GIK Institute, Topi 23640, KP, Pakistan; siraj@giki.edu.pk

³ Department of Mathematics, Faculty of Sciences and Technology, Women University of Azad Jammu and Kashmir, Bagh 12500, AJK, Pakistan; manzoor366@gmail.com

⁴ Department of Mathematics and Sciences, Prince Sultan University, Riyadh 11586, Saudi Arabia

⁵ Department of Medical Research, China Medical University, Taichung 40402, Taiwan

⁶ Department of Mathematical Sciences, Faculty of Sciences, Princess Nourah Bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia; maalqudah@pnu.edu.sa

* Correspondence: tabdeljawad@psu.edu.sa

Abstract: In this work, an efficient and robust numerical scheme is proposed to solve the variable coefficients' fourth-order partial differential equations (FOPDEs) that arise in Euler–Bernoulli beam models. When partial differential equations (PDEs) are of higher order and invoke variable coefficients, then the numerical solution is quite a tedious and challenging problem, which is our main concern in this paper. The current scheme is hybrid in nature in which the second-order finite difference is used for temporal discretization, while spatial derivatives and solutions are approximated via the Haar wavelet. Next, the integration and Haar matrices are used to convert partial differential equations (PDEs) to the system of linear equations, which can be handled easily. Besides this, we derive the theoretical result for stability via the Lax–Richtmyer criterion and verify it computationally. Moreover, we address the computational convergence rate, which is near order two. Several test problems are given to measure the accuracy of the suggested scheme. Computations validate that the present scheme works well for such problems. The calculated results are also compared with the earlier work and the exact solutions. The comparison shows that the outcomes are in good agreement with both the exact solutions and the available results in the literature.

Keywords: higher-order PDEs; Euler–Bernoulli beam models; Haar wavelets; finite differences



Citation: Ghafoor, A.; Haq, S.; Hussain, M.; Abdeljawad, T.; Alqudah, M.A. Numerical Solutions of Variable Coefficient Higher-Order Partial Differential Equations Arising in Beam Models. *Entropy* **2022**, *24*, 567. <https://doi.org/10.3390/e24040567>

Academic Editor: António M. Lopes

Received: 7 February 2022

Accepted: 11 April 2022

Published: 18 April 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Small static and dynamic deflection problems can be observed properly in linear theory. For the determination of large (dynamic and static) deflection, linear theory is not beneficial and requires an accurate analysis. Linear theory admits inexact curvature in the study of beam deflections. The pioneering work in the field of thin beam theory was carried out by Bernoulli. Jacob Bernoulli then studied elastic theory and showed that the bending moment and curvature are both proportional. Later on, the Bernoulli theory was extended to loaded beams by Leonhard Euler.

The PDE of a thin and long beam is known as the Euler–Bernoulli model. The solution of this model shows the shortest distance (transverse vibration) from the beginning position in which stress and strain are linearly related. The mathematical form of the Euler–Bernoulli model [1] is described as follows:

$$\gamma(x)\partial_{tt}u(x,t) + \partial_{xx}[\mathcal{C}(x)\partial_{xx}u(x,t)] = \mathcal{E}(x,t), \quad 0 < x < 1, \quad 0 < t \leq T, \quad (1)$$

with appropriate initial and boundary conditions:

$$u(x,0) = Y_1(x), \quad \partial_t u(x,0) = Y_2(x), \quad 0 \leq x \leq 1, \quad (2)$$

$$\begin{cases} u(0,t) = \varepsilon_0(t), & \partial_x u(0,t) = \delta_0(t), & 0 < t \leq T, \\ u(1,t) = \varepsilon_1(t), & \partial_x u(1,t) = \delta_1(t), & 0 < t \leq T, \end{cases} \quad (3)$$

or

$$\begin{cases} u(0,t) = \zeta_0(t), & \partial_{xx} u(0,t) = \mu_0(t), & 0 < t \leq T \\ u(1,t) = \zeta_1(t), & \partial_{xx} u(1,t) = \mu_1(t), & 0 < t \leq T, \end{cases} \quad (4)$$

where the notations $\partial_{tt}u(x,t)$, $\partial_{xx}u(x,t)$ stand for the second derivative with respect to the time and space variables, respectively. Equation (1) is the variable coefficients' FOPDE in which $u(x,t)$ represents the beam displacement, $\gamma(x)$ is the mass per unit length, $\mathcal{C}(x)$ shows the bending stiffness of the beam, $\mathcal{E}(x,t)$ is the source term, and L is the total length. This kind of equation has widespread applications in robotics designs and large flexible space structures [2,3]. Several analytical methods have been applied to derive the closed-form solution of the governing equation. Wazwaz [4] used the Adomian decomposition method to solve the variable coefficients' FOPDEs. Lieu [5] applied He's variational iteration technique to explain free vibration in an Euler–Bernoulli beam. For more generic cases of initial conditions, analytical solutions are quite complicated; therefore, researchers are constantly trying to focus on the numerical solutions. A variety of numerical methods via finite difference schemes have been developed for the solution of different forms of Equation (1) such as Jain et al. [6], Evans [7], Conte [8], Richtmyer [9], and Cranial [10]. Evans et al. [11] established a stable computational method using the hopscotch algorithm. Aziz et al. [12] proposed a three-level scheme using a parametric quintic spline to solve the FOPDEs. Rashidinia [13] implemented a three-level implicit scheme coupled with a sextic spline to solve fourth-order equations. Recently, Imtiaz et al. [14] numerically studied fractional-order Korteweg–de Vries and Burger's equations via the meshless method. Senol and his co-authors [15] investigated the Coudrey–Dodd–Gibbon–Sawada–Kotera equation with three different methods. Akinyemi [16] solved $(1 + 3)$ -dimensional fractional reaction–diffusion trimolecular models. The same author [17] and his collaborators computed the numerical solutions of coupled nonlinear Schrödinger–Korteweg–de Vries and Maccari systems numerically. Jiwari [18] applied barycentric rational interpolation and local radial basis function algorithms for the multi-dimensional Sine–Gordon equation.

In the past few years, wavelet-based approximation techniques gained great importance for solving PDEs [19,20]. In all kinds of wavelets, the simplest family is the Haar wavelet (HW), which comprises rectangular box functions. The usage of these wavelet attracted more researchers because of its easy implementation and achievements with good results. The HW based on Haar functions introduced by Alfred Haar in 1910, which are quite simple mathematically, are discontinuous at breaking points of the interval and, hence, not differentiable. Due to this reason, the direct implementation of the HW for the solution of differential equations is not possible. For this purpose, Cattani [21,22] used interpolating splines to remove this ambiguity. Similarly, an alternative idea was given by Chen and Hasio [23]. They suggested approximating the maximum-order derivative with finite HW series. Later on, this approach was extended for the solution of different problems. Lepik [24,25] introduced a numerical method using the HW to obtain the solution of PDEs. Jiwari [26] used the HW coupled with quasi-linearization for the solution of Burgers' equation. Mittal et al. [27] studied numerically the system of viscous Burgers' equations with the HW. Oruç [28,29] applied the finite difference hybrid scheme combined with the HW for the solution of modified Burgers' and KdV equations. Kumar [30] solved the system of Burgers' equations through the finite difference HW technique. Somayeh et al. [31,32] used a semi-analytical approach for solving the Hunter–Saxton and foam drainage equations. The same author [33] implemented an HW-based scheme for the solution of the two-dimensional system of PDEs. Mittal and Pandit [34] solved unsteady squeezing nano-fluid problems via the HW. Jiwari [35] developed a hybrid numerical method consisting of the HW for the nonlinear Burger's equation. Pandit and Mitall [36] introduced the scale-3 Haar wavelet for the numerical solutions of the fractional advection dispersion equation. The same author

and his co-authors [37] defined an operational-matrix-based algorithm for computational modeling of hyperbolic-type wave equations. For more details about these wavelet, we refer to [25,38].

In the present work, a hybrid scheme consisting of the HW and finite difference is suggested to find the numerical solution of Equation (1) with the homogeneous and non-homogeneous forms with variable coefficients. The remainder of this paper is as follows. In Section 2 the main motive of the current work is given. The preliminaries of the HW and their integrals are given in Section 3. The description of the method and stability are presented in Sections 4 and 5, respectively. For the validation of the suggested scheme, some test problems are addressed in Section 6. In Section 7 we present the detail description of the initial disturbance and noisy data. Finally, the conclusion is reported in Section 8.

2. Motivation

To develop methods for solutions of higher-order PDEs either analytical, semi-analytical, or numerical is an essential need for the physical interpretation of the problem. In the literature, several numerical techniques have been adopted for FOPDEs. According to our knowledge, a hybrid numerical method based on the HW and finite difference along with stability for FOPDEs has not been reported yet. Therefore, the main motive of this work is to determine numerical solutions of FOPDEs using finite differences and the HW.

3. Haar Wavelets and Their Integrals

The HW based on Haar functions, which were defined in 1910, belongs to a well-known class of the wavelet family known as the Daubechies wavelet. The HW is a recent mathematical tool, which became popular in the numerical study of various differential and integral equations. Initially, it was introduced for the interval $[0, 1)$, but Lepik [24] extended this for any arbitrary interval $[\mathcal{A}, \mathcal{B})$.

To define the HW, assume $x \in [\mathcal{A}, \mathcal{B})$, then the HW family is defined for $i \geq 1$ as follows:

$$h_1(x) = \begin{cases} 1, & x \in [\mathcal{A}, \mathcal{B}) \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

$$h_i(x) = \begin{cases} 1, & x \in [\xi_1(i), \xi_2(i)) \\ -1, & x \in [\xi_2(i), \xi_3(i)) \\ 0, & \text{otherwise,} \end{cases} \quad (6)$$

where $\xi_{s+1}(i) = \mathcal{A} + (2k + s)v\delta x$, $s = 0, 1, 2$ and $v = \mathcal{M}/m$.

In the aforementioned equations, the interval is subdivided into $2\mathcal{M}$ intervals of equal length $\delta x = \frac{\mathcal{B}-\mathcal{A}}{2\mathcal{M}}$, where $\mathcal{M} = 2^\lambda$ and λ denote the maximal level of resolution. Next, two parameters $j = 0, \dots, \lambda$ and $k = 0, \dots, 2^j - 1$ are taken, which decompose the wavelet number $i = m + k + 1$, where $m = 2^j$. To solve nth-order PDEs, we need the following repeated integrals:

$$\mathcal{R}_{i,\beta}(x) = \int_{\mathcal{A}}^x \int_{\mathcal{A}}^x \cdots \int_{\mathcal{A}}^x h_i(z) dz^\beta = \frac{1}{(\beta-1)!} \int_{\mathcal{A}}^x (x-z)^{\beta-1} h_i(z) dz, \quad (7)$$

where

$$\beta = 1, 2, \dots, n, \quad i = 1, 2, \dots, 2\mathcal{M}.$$

Keeping in view Equations (5) and (6), the analytical expression of these integrals is given as [25]

$$\mathcal{R}_{1,\beta}(x) = \frac{(x - \mathcal{A})^\beta}{\beta!}, \tag{8}$$

$$\mathcal{R}_{i,\beta}(x) = \begin{cases} 0, & x < \xi_1(i) \\ \frac{1}{\beta!} [x - \xi_1(i)]^\beta & x \in [\xi_1(i), \xi_2(i)) \\ \frac{1}{\beta!} [(x - \xi_1(i))^\beta - 2((x - \xi_2(i))^\beta)] & x \in [\xi_2(i), \xi_3(i)) \\ \frac{1}{\beta!} [(x - \xi_1(i))^\beta - 2((x - \xi_2(i))^\beta + (x - \xi_3(i))^\beta)] & x \geq \xi_3(i). \end{cases} \tag{9}$$

Function Approximation

If $w(x) \in L^2[\mathcal{A}, \mathcal{B}]$ is a square integrable function, then it can be approximated via HW series as:

$$w(x) = \sum_{i=1}^{2\mathcal{M}} \alpha_i h_i(x), \tag{10}$$

where α_i are the unknown HW coefficients. At collocation points $x \rightarrow x_l$, Equation (10) takes the following discrete form:

$$\tilde{w} = \sum_{i=1}^{2\mathcal{M}} \alpha_i h_i(x_l). \tag{11}$$

In matrix form, Equation (11) can be written as:

$$\Theta = \mathcal{J} \mathcal{A}, \tag{12}$$

where Θ and \mathcal{A} are $2\mathcal{M} \times 1$ -dimensional matrices and \mathcal{J} is a $2\mathcal{M} \times 2\mathcal{M}$ -dimensional matrix. From Equation (12), one can calculate the unknown coefficients, and then, the approximation to $\mathcal{F}(x)$ can be computed using Equation (10) for different resolution levels.

4. Description of the Method

In this section of the manuscript, the proposed scheme is presented for Equation (1) with boundary conditions in the form of Equation (3). We rewrite Equation (1) as:

$$\gamma(x) \partial_t u + \mathcal{C}(x) \partial_{xxx} u + 2\mathcal{C}'(x) \partial_{xxx} u + \mathcal{C}''(x) \partial_{xx} u = \mathcal{E}, \tag{13}$$

where $\mathcal{C}'(x) = \frac{d\mathcal{C}}{dx}$. Applying the finite difference to the temporal part and (θ -weighted $\leq \theta \leq 1$ scheme, Equation (13) reduces to

$$\gamma(x) \frac{u^{j+1} - 2u^j + u^{j-1}}{\tau^2} + \theta [\mathcal{C}(x) \partial_{xxx} u + 2\mathcal{C}'(x) \partial_{xxx} u + \mathcal{C}''(x) \partial_{xx} u]^{j+1} + (1 - \theta) [\mathcal{C}(x) \partial_{xxx} u + 2\mathcal{C}'(x) \partial_{xxx} u + \mathcal{C}''(x) \partial_{xx} u]^j = \mathcal{E}^{j+1}, \tag{14}$$

where $u^j = u(x, t^j)$, $\mathcal{E}^j = \mathcal{E}(x, t^j)$, $t^{j+1} = \tau + t^j$, and τ is the time step size. The associated boundary conditions Equation (3) is transformed to

$$\begin{cases} u^{j+1}(0) = \varepsilon_0(t^{j+1}), & \partial_x u^{j+1}(0) = \delta_0(t^{j+1}), & 0 < t \leq T \\ u^{j+1}(1) = \varepsilon_1(t^{j+1}), & \partial_x u^{j+1}(1) = \delta_1(t^{j+1}), & 0 < t \leq T. \end{cases} \tag{15}$$

In more simplified form, Equation (14) can be written as

$$\gamma(x) u^{j+1} + \tau^2 \theta \mathcal{C}(x) \partial_{xxx} u^{j+1} + 2\theta \tau^2 \mathcal{C}'(x) \partial_{xxx} u^{j+1} + \theta \tau^2 \mathcal{C}''(x) \partial_{xx} u^{j+1} = \mathcal{N}^j + \tau^2 \mathcal{E}^{j+1}, \tag{16}$$

where

$$\mathcal{N}^j = (2u^j - u^{j-1}) \gamma(x) + (\theta - 1) \tau^2 [\mathcal{C}(x) \partial_{xxx} u + 2\mathcal{C}'(x) \partial_{xxx} u + \mathcal{C}''(x) \partial_{xx} u]^j.$$

Next, we assume the Haar wavelet approximation for the highest-order derivative as

$$\partial_{xxxx}u^{j+1}(x) = \sum_{i=1}^{2M} \alpha_i^{j+1} h_i(x), \tag{17}$$

where α_i^{j+1} are unknown constants to be determined. Integration of Equation (17) four times from 0 to x leads to:

$$\begin{aligned} \partial_{xxx}u^{j+1}(x) &= \sum_{i=1}^{2M} \alpha_i^{j+1} \mathcal{R}_{i,1}(x) + \partial_{xxx}u^{j+1}(0) \\ \partial_{xx}u^{j+1}(x) &= \sum_{i=1}^{2M} \alpha_i^{j+1} \mathcal{R}_{i,2}(x) + x\partial_{xxx}u^{j+1}(0) + \partial_{xx}u^{j+1}(0) \\ \partial_x u^{j+1}(x) &= \sum_{i=1}^{2M} \alpha_i^{j+1} \mathcal{R}_{i,3}(x) + \frac{x^2}{2!} \partial_{xxx}u^{j+1}(0) + x\partial_{xx}u^{j+1}(0) + \partial_x u^{j+1}(0), \\ u^{j+1}(x) &= \sum_{i=1}^{2M} \alpha_i^{j+1} \mathcal{R}_{i,4}(x) + \frac{x^3}{3!} \partial_{xxx}u^{j+1}(0) + \frac{x^2}{2!} \partial_{xx}u^{j+1}(0) + x\partial_x u^{j+1}(0) + u^{j+1}(0). \end{aligned} \tag{18}$$

Using the boundary conditions $u^{j+1}(1)$, $\partial_x u^{j+1}(1)$ in Equation (18), the unknown terms can be computed as

$$\begin{aligned} \partial_{xxx}u^{j+1}(0) &= \phi(t^{j+1}) + 12 \sum_{i=1}^{2M} \alpha_i^{j+1} \mathcal{R}_{i,4}(1) - 6 \sum_{i=1}^{2M} \alpha_i^{j+1} \mathcal{R}_{i,3}(1) \\ \partial_{xx}u^{j+1}(0) &= \psi(t^{j+1}) - 6 \sum_{i=1}^{2M} \alpha_i^{j+1} \mathcal{R}_{i,4}(1) + 2 \sum_{i=1}^{2M} \alpha_i^{j+1} \mathcal{R}_{i,3}(1), \end{aligned} \tag{19}$$

where

$$\begin{aligned} \phi(t^{j+1}) &= 6 \left[\delta_1(t^{j+1}) + \delta_o(t^{j+1}) - 2\varepsilon_1(t^{j+1}) + 2\varepsilon_o(t^{j+1}) \right] \\ \psi(t^{j+1}) &= -2 \left[\delta_1(t^{j+1}) + 2\delta_o(t^{j+1}) \right] - 6 \left[\varepsilon_o(t^{j+1}) - \varepsilon_1(t^{j+1}) \right]. \end{aligned}$$

Making use of Equation (19) in Equation (18), we obtain

$$\begin{aligned} \partial_{xxx}u^{j+1}(x) &= \sum_{i=1}^{2M} \alpha_i^{j+1} [\mathcal{R}_{i,1}(x) - 6\mathcal{R}_{i,3}(1) + 12\mathcal{R}_{i,4}(1)] + \phi(t^{j+1}) \\ \partial_{xx}u^{j+1}(x) &= \sum_{i=1}^{2M} \alpha_i^{j+1} [\mathcal{R}_{i,2}(x) + (2 - 6x)\mathcal{R}_{i,3}(1) + (12x - 6)\mathcal{R}_{i,4}(1)] + x\phi(t^{j+1}) + \psi(t^{j+1}) \\ \partial_x u^{j+1}(x) &= \sum_{i=1}^{2M} \alpha_i^{j+1} [\mathcal{R}_{i,3}(x) + (2x - 3x^2)\mathcal{R}_{i,3}(1) + (6x^2 - 6x)\mathcal{R}_{i,4}(1)] + \frac{x^2}{2} \phi(t^{j+1}) \\ &\quad + x\psi(t^{j+1}) + \delta_o(t^{j+1}) \\ u^{j+1}(x) &= \sum_{i=1}^{2M} \alpha_i^{j+1} [\mathcal{R}_{i,4}(x) + (x^2 - x^3)\mathcal{R}_{i,3}(1) + (2x^3 - 3x^2)\mathcal{R}_{i,4}(1)] + \frac{x^3}{3!} \phi(t^{j+1}) \\ &\quad + \frac{x^2}{2} \psi(t^{j+1}) + x\delta_o(t^{j+1}) + \varepsilon_o(t^{j+1}). \end{aligned} \tag{20}$$

The proposed technique is based on the collocation approach; therefore, the collocation points are

$$x_l = \frac{l - 0.5}{2M}, \quad l = 1, \dots, 2M.$$

Putting Equations (17) and (20) in Equation (16) and replacing x with x_l leads to the following system of linear equations

$$\sum_{i=1}^{2M} \alpha_i^{J+1} [\gamma(x_l)\mathbb{A}(i, l) + \tau^2\theta\mathcal{C}(x_l)h_i(x_l) + 2\theta\tau^2\mathcal{C}'(x_l)Y(i, l) + \tau^2\theta\mathcal{C}''(x_l)\Psi(i, l)] = \Lambda(i, J + 1) + \Omega(i, J + 1), \tag{21}$$

where

$$\begin{aligned} \mathbb{A}(i, l) &= [\mathcal{R}_{i,4}(x) + (x^2 - x^3)\mathcal{R}_{i,3}(1) + (2x^3 - 3x^2)\mathcal{R}_{i,4}(1)] \\ Y(i, l) &= [\mathcal{R}_{i,1}(x) - 6\mathcal{R}_{i,3}(1) + 12\mathcal{R}_{i,4}(1)] \\ \Psi(i, l) &= [\mathcal{R}_{i,2}(x) + (2 - 6x)\mathcal{R}_{i,3}(1) + (12x - 6)\mathcal{R}_{i,4}(1)] \\ \Lambda(l, t^{J+1}) &= \mathcal{N}(l, t^J) + \mathcal{E}(l, t^{J+1}) \\ \Omega(i, J + 1) &= -\gamma(x_l) \left\{ \frac{x^3}{3!} \phi(t^{J+1}) + \frac{x^2}{2} \psi(t^{J+1}) + x\delta_o(t^{J+1}) + \varepsilon_o(t^{J+1}) \right\} \\ &\quad - 2\tau^2\theta\mathcal{C}'(x) \left\{ \phi(t^{J+1}) \right\} - \tau^2\theta \left\{ x\phi(t^{J+1}) + \psi(t^{J+1}) \right\}. \end{aligned}$$

There are $2M$ equations in Equation (21), which can be solved for $2M$ unknowns iteratively. After the computation of the unknown constants, the required solution can be computed from Equation (20).

Note

To use the second kind of boundary conditions given in (Equation (4)), one may use first $u^{J+1}(1)$, $\partial_{xx}u^{J+1}(1)$ in Equation (18) to obtain the following system of equations:

$$\begin{aligned} \mu_1(t^{J+1}) &= \sum_{i=1}^{2M} \alpha_i^{J+1} \mathcal{R}_{i,2}(1) + \partial_{xxx}u^{J+1}(0) + \mu_0(t^{J+1}) \\ \zeta_1(t^{J+1}) &= \sum_{i=1}^{2M} \alpha_i^{J+1} \mathcal{R}_{i,4}(1) + \frac{1}{3!} \partial_{xxx}u^{J+1}(0) + \frac{1}{2!} \mu_0(t^{J+1}) + \partial_x u^{J+1}(0) + \zeta_0(t^{J+1}). \end{aligned} \tag{22}$$

After solving Equation (22), one can easily obtain $\partial_{xxx}u^{J+1}(0)$, $\partial_x u^{J+1}(0)$. Once these terms are calculated, then expressions for the derivatives and solutions can be extracted following the strategies given in Equations (19)–(21).

5. Stability

Here, we present the theoretical result related to the stability of the proposed technique. To derive the condition, one can deduce the following equations from Equation (20) as:

$$\partial_{xxxx}u^{J+1} = \mathcal{H}\alpha^{J+1}, \tag{23}$$

$$\partial_{xxx}u^{J+1} = Y\alpha^{J+1} + \tilde{Y}^{J+1}, \tag{24}$$

$$\partial_{xx}u^{J+1} = \Psi\alpha^{J+1} + \tilde{\Psi}^{J+1}, \tag{25}$$

$$u^{J+1} = \mathcal{F}\alpha^{J+1} + \tilde{\mathcal{F}}^{J+1}, \tag{26}$$

where \mathcal{H}, Y, Ψ are differentiation matrices, \mathcal{F} is the interpolation matrix, and $\tilde{Y}^{J+1}, \tilde{\Psi}^{J+1}, \tilde{\mathcal{F}}^{J+1}$ are boundary terms given in Equation (20). Using initial condition $\frac{u^J - u^{J-1}}{\tau} = Y_2(x)$ together with Equations (23)–(26) and $x \rightarrow x_l$ in Equation (16), we obtain

$$\begin{aligned} & \left[\gamma(x_l)\mathcal{F}(i,l) + \tau^2\theta C(x_l)\mathcal{H}(i,l) + 2\theta\tau^2C'(x_l)Y(i,l) + \theta\tau^2C''(x_l)\Psi(i,l) \right] \alpha^{j+1} = \left[\gamma(x_l)\mathcal{F}(i,l) \right. \\ & \left. + (\theta - 1)\tau^2 \left(C(x_l)\mathcal{H}(i,l) + 2C'(x_l)Y(i,l) + C''(x_l)\Psi(i,l) \right) \right] \alpha^j + \tau^2\mathcal{E}^{j+1} + \tau\gamma(x_l)Y_2(x_l). \end{aligned} \tag{27}$$

In alternative form, Equation (27) can be written as:

$$\mathbb{M}\alpha^{j+1} = \mathbb{N}\alpha^j + \mathbb{B}^{j+1}, \tag{28}$$

where

$$\begin{aligned} \mathbb{M} &= \gamma(x_l)\mathcal{F}(i,l) + \tau^2\theta C(x_l)\mathcal{H}(i,l) + 2\theta\tau^2C'(x_l)Y(i,l) + \theta\tau^2C''(x_l)\Psi(i,l), \\ \mathbb{N} &= \gamma(x_l)\mathcal{F}(i,l) + (\theta - 1)\tau^2 \left(C(x_l)\mathcal{H}(i,l) + 2C'(x_l)Y(i,l) + C''(x_l)\Psi(i,l) \right), \\ \mathbb{B}^{j+1} &= \tau^2\mathcal{E}^{j+1} + \tau\gamma(x_l)Y_2(x_l). \end{aligned}$$

It follows from Equation (28) that

$$\alpha^{j+1} = \mathbb{M}^{-1}\mathbb{N}\alpha^j + \mathbb{M}^{-1}\mathbb{B}^{j+1}. \tag{29}$$

Plugging Equation (29) in Equation (26), one obtains

$$\begin{aligned} u^{j+1} &= \mathcal{F} \left(\mathbb{M}^{-1}\mathbb{N}\alpha^j + \mathbb{M}^{-1}\mathbb{B}^{j+1} \right) + \mathcal{F}^{j+1} \\ &= \mathcal{F}\mathbb{M}^{-1}\mathbb{N}\alpha^j + \mathcal{F}\mathbb{M}^{-1}\mathbb{B}^{j+1} + \mathcal{F}^{j+1}. \end{aligned} \tag{30}$$

Using Equations (26) and (30), we have

$$\begin{aligned} u^{j+1} &= \mathcal{F}\mathbb{M}^{-1}\mathbb{N} \left(\mathcal{F}^{-1}u^j - \mathcal{F}^{-1}\tilde{\mathcal{F}}^{j+1} \right) + \mathcal{F}\mathbb{M}^{-1}\mathbb{B}^{j+1} + \mathcal{F}^{j+1} \\ &= \mathcal{F}\mathbb{M}^{-1}\mathbb{N}\mathcal{F}^{-1}u^j - \mathcal{F}\mathbb{M}^{-1}\mathbb{N}\mathcal{F}^{-1}\tilde{\mathcal{F}}^{j+1} + \mathcal{F}\mathbb{M}^{-1}\mathbb{B}^{j+1} + \mathcal{F}^{j+1}. \end{aligned} \tag{31}$$

Equation (31) shows an iterative formula between u^{j+1} and u^j . If \tilde{u}^j is the approximate solution, then

$$\tilde{u}^{j+1} = \mathcal{F}\mathbb{M}^{-1}\mathbb{N}\mathcal{F}^{-1}\tilde{u}^j - \mathcal{F}\mathbb{M}^{-1}\mathbb{N}\mathcal{F}^{-1}\tilde{\mathcal{F}}^{j+1} + \mathcal{F}\mathbb{M}^{-1}\mathbb{B}^{j+1} + \mathcal{F}^{j+1}. \tag{32}$$

Let $e^j = |u^j - \tilde{u}^j|$, then from Equations (31) and (32),

$$e^{j+1} = \Xi e^j, \tag{33}$$

where $\Xi = \mathcal{F}\mathbb{M}^{-1}\mathbb{N}\mathcal{F}^{-1}$ is the amplification matrix. According to the Lax–Richtmyer criterion, the stability condition will be fulfilled if $\|\Xi\| \leq 1$, which needs the spectral radius $\rho(\Xi) \leq 1$.

6. Illustrative Examples

In this section, we investigate the performance of the proposed method by solving various examples. To measure the efficiency, two error norms L_2 , L_∞ were addressed, which are defined by:

$$\begin{aligned} L_2 &= \left(\sum_i^{2\mathcal{M}} \left| u_i^{ext} - \tilde{u}_i^{app} \right|^2 \right)^{\frac{1}{2}} \\ L_\infty &= \max_{1 \leq i \leq 2\mathcal{M}} \left| u_i^{ext} - \tilde{u}_i^{app} \right|, \end{aligned}$$

where u and \tilde{u} denote the exact and numerical solutions, respectively. Furthermore, we compute the convergence rate by using the following formula

$$\text{Convergence rate} = \frac{\log(e_\infty^\lambda - e_\infty^{\lambda+1})}{\log(2)}.$$

6.1. Problem 5.1

Consider Equation (1) with $\gamma(x) = \mathcal{C}(x) = 1$, in the following form

$$\partial_{tt}u(x, t) + \partial_{xxxx}u(x, t) = (\pi^4 - 1) \sin \pi x \cos t, \quad 0 \leq x \leq 1, \quad 0 < t \leq T, \quad (34)$$

coupled with appropriate conditions

$$u(x, 0) = \sin \pi x, \quad \partial_t u(x, 0) = 0, \quad 0 \leq x \leq 1, \quad (35)$$

$$u(0, t) = u(1, t) = 0, \quad \partial_{xx}u(0, t) = \partial_{xx}u(1, t) = 0, \quad 0 < t \leq T. \quad (36)$$

The exact solution of this problem is $u(x, t) = \sin \pi x \cos t$. This problem was solved this problem with the help of the proposed method. In Table 1, pointwise errors are compared with existing results [12,39–41] for fixed $\tau = 0.005$ using different time and resolution levels. From the table, it is clear that the computed results are in good agreement with those available in the literature. Moreover, we calculated the convergence rate for this problem and addressed it in Table 2, which is approximately of order two. Graphical solutions in the form of 2D and 3D plots with the absolute error are shown in Figure 1 for time $t = 4$. It is clear from the figure that the proposed method gives accurate solutions for a small number of collocation points and matches well the exact solution.

Table 1. Absolute error at different points in Problem 5.1.

| Methods | Points | t | $x = 0.1$ | $x = 0.2$ | $x = 0.3$ | $x = 0.4$ | $x = 0.5$ |
|------------------|--------|------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| Present | 64 | 0.02 | 6.54×10^{-7} | 1.24×10^{-6} | 1.71×10^{-6} | 2.01×10^{-6} | 2.11×10^{-6} |
| | 64 | 0.05 | 5.22×10^{-6} | 9.94×10^{-6} | 1.36×10^{-5} | 1.60×10^{-5} | 1.69×10^{-5} |
| | 64 | 1 | 7.14×10^{-4} | 1.35×10^{-3} | 1.87×10^{-3} | 2.20×10^{-3} | 2.31×10^{-3} |
| | 128 | 0.02 | 2.60×10^{-7} | 4.94×10^{-7} | 6.81×10^{-7} | 8.00×10^{-7} | 8.42×10^{-7} |
| | 128 | 0.05 | 2.59×10^{-6} | 4.94×10^{-6} | 6.80×10^{-6} | 7.99×10^{-6} | 8.40×10^{-6} |
| | 128 | 1 | 6.83×10^{-4} | 1.29×10^{-3} | 1.78×10^{-3} | 2.10×10^{-3} | 2.21×10^{-3} |
| Mittal [39] | 91 | 0.02 | 3.20×10^{-5} | 6.08×10^{-5} | 8.37×10^{-5} | 9.84×10^{-5} | 1.04×10^{-4} |
| | 91 | 0.05 | 3.59×10^{-5} | 6.83×10^{-5} | 9.39×10^{-5} | 1.10×10^{-4} | 1.16×10^{-4} |
| | 91 | 1 | 6.32×10^{-5} | 1.20×10^{-4} | 1.65×10^{-4} | 1.94×10^{-4} | 2.04×10^{-4} |
| | 181 | 0.02 | 3.55×10^{-6} | 6.76×10^{-6} | 9.30×10^{-6} | 1.09×10^{-5} | 1.15×10^{-5} |
| | 181 | 0.05 | 3.99×10^{-6} | 7.58×10^{-6} | 1.04×10^{-5} | 1.23×10^{-5} | 1.29×10^{-5} |
| | 181 | 1 | 7.00×10^{-6} | 1.33×10^{-5} | 1.83×10^{-5} | 2.16×10^{-5} | 2.27×10^{-5} |
| Caglar [40] | 121 | 0.02 | 4.80×10^{-6} | 9.70×10^{-6} | 1.40×10^{-5} | 1.90×10^{-5} | 2.40×10^{-5} |
| | 191 | 0.02 | 5.20×10^{-6} | 2.10×10^{-6} | 3.10×10^{-6} | 4.20×10^{-6} | 5.20×10^{-6} |
| | 521 | 0.02 | 4.90×10^{-7} | 9.90×10^{-7} | 1.40×10^{-6} | 1.90×10^{-6} | 2.40×10^{-6} |
| Aziz et al. [12] | 20 | 0.05 | 9.30×10^{-6} | 8.00×10^{-6} | 2.80×10^{-6} | 1.00×10^{-6} | 2.70×10^{-6} |
| Rashidinia [13] | 20 | 0.05 | 2.91×10^{-6} | 1.73×10^{-6} | 1.60×10^{-6} | 2.23×10^{-6} | 2.60×10^{-7} |
| Mohammadi [41] | 40 | 0.05 | 2.96×10^{-6} | 1.77×10^{-6} | 1.64×10^{-6} | 2.28×10^{-6} | 2.65×10^{-7} |

Table 2. Convergence rate of Problem 5.1 at $t = 0.2$.

| Problem 4.1 | | | |
|-------------|--------|-------------------------|--------|
| λ | τ | L_∞ | Rate |
| 2 | 1/100 | $90,568 \times 10^{-3}$ | |
| 3 | 1/200 | 2.4428×10^{-3} | 1.8904 |
| 4 | 1/400 | 6.8223×10^{-4} | 1.8402 |
| 5 | 1/800 | 2.0398×10^{-4} | 1.7418 |

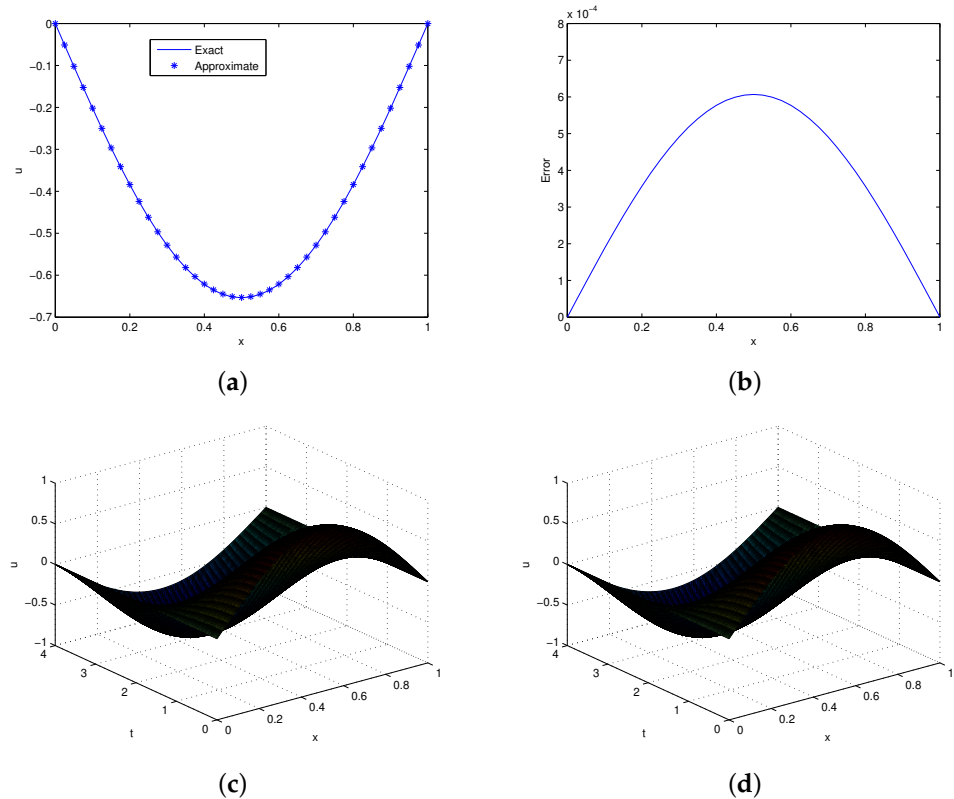


Figure 1. Solutions profile of Problem 5.1. (a) Exact and approximate solutions at $t = 4$, $\tau = 0.001$. (b) Absolute error in (a). (c) Exact 3D plot. (d) Approximate 3D plot at $t = 4$, $\tau = 0.01$, $\lambda = 4$.

6.2. Problem 5.2

Consider the non-homogeneous problem of the form [41]

$$\partial_{tt}u(x, t) + \partial_{xx}((1 + \sin \pi x)\partial_{xx}u(x, t)) = \mathcal{E}(x, t), \quad 0 \leq x \leq 1, \quad 0 < t \leq T, \quad (37)$$

coupled with the initial conditions

$$u(x, 0) = \partial_t u(x, 0) = 0, \quad 0 \leq x \leq 1, \quad (38)$$

and the boundary conditions

$$u(0, t) = u(1, t) = 0, \quad \partial_x u(0, t) = \partial_x u(1, t) = 0, \quad 0 < t \leq T. \quad (39)$$

The corresponding source term can be adjusted according to the exact solution:

$$u(x, t) = x(1 - x) \exp(-t)t^2 \sin 4\pi x.$$

We obtained the solution of this problem in the time domain $[0, 4]$. In Table 3, we record the absolute error at point $x = 0.5$, using different times. From table, the computations

show that our results are better than those described by Mohammadi [41]. In Table 4, different error norms are calculated, which identify that the proposed method produces good results at small resolution levels. The convergence rate of this problem was computed and is addressed in Table 5, which shows that the scheme is approximately of order two. Exact versus approximate solutions together with the absolute error are plotted at time $t = 4$ in Figure 2. It is clear from the figure that the exact and approximate solutions are in good agreement.

Table 3. Absolute error in displacement of Problem 5.2.

| Methods | Points | τ | $t = 0.2$ | $t = 0.4$ | $t = 0.8$ | $t = 1$ | $t = 2$ | $t = 4$ |
|---------|--------|--------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|
| Present | 32 | 0.001 | 1.76×10^{-13} | 5.72×10^{-13} | 1.80×10^{-12} | 2.14×10^{-12} | 2.87×10^{-12} | 2.39×10^{-12} |
| | 64 | 0.001 | 1.72×10^{-13} | 5.75×10^{-13} | 1.48×10^{-12} | 1.90×10^{-12} | 2.95×10^{-12} | 6.13×10^{-13} |
| [41] | 100 | 0.01 | 1.78×10^{-5} | 5.85×10^{-5} | 1.57×10^{-4} | 2.00×10^{-4} | 2.95×10^{-4} | 1.60×10^{-4} |
| | 200 | 0.005 | 2.38×10^{-6} | 7.80×10^{-6} | 2.09×10^{-5} | 2.67×10^{-5} | 3.94×10^{-5} | 2.57×10^{-5} |

Table 4. Maximum error norms of Problem 5.2 at different times with $\tau = 0.001$.

| λ | $t = 0.2$ | | $t = 0.5$ | | $t = 1$ | | $t = 4$ | |
|-----------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| | L_∞ | L_2 | L_∞ | L_2 | L_∞ | L_2 | L_∞ | L_2 |
| 4 | 6.10×10^{-5} | 3.38×10^{-4} | 3.86×10^{-4} | 2.11×10^{-3} | 1.02×10^{-3} | 5.58×10^{-3} | 8.64×10^{-4} | 4.72×10^{-3} |
| 5 | 1.11×10^{-5} | 7.36×10^{-5} | 5.85×10^{-5} | 3.29×10^{-4} | 2.24×10^{-4} | 1.23×10^{-3} | 2.29×10^{-4} | 1.25×10^{-3} |

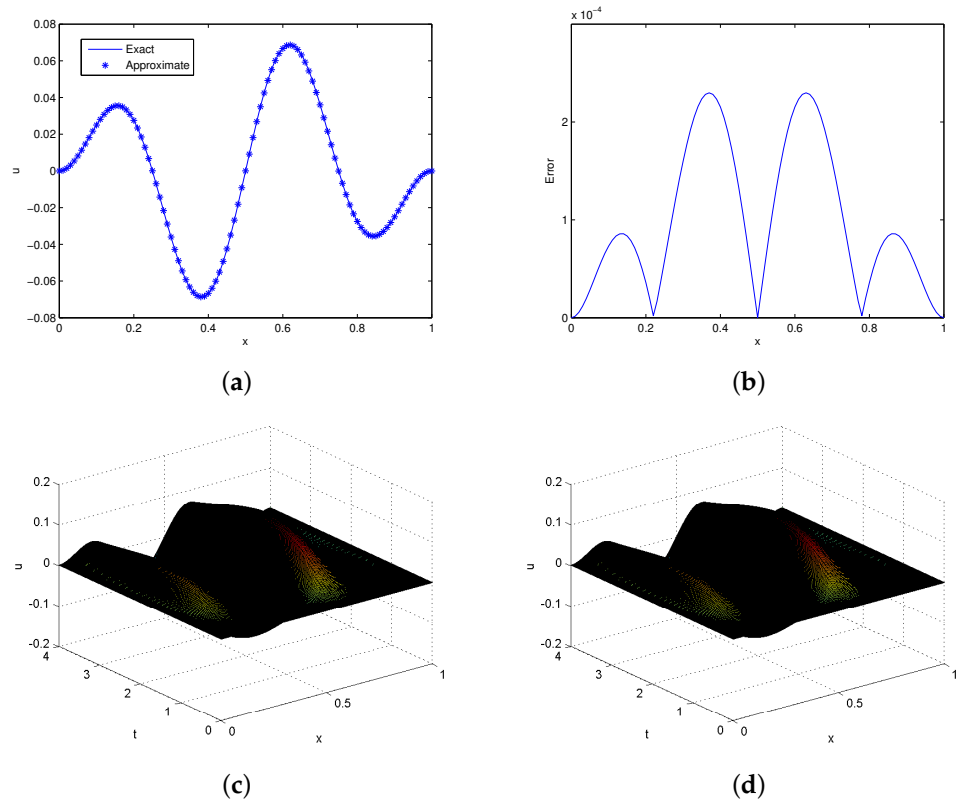


Figure 2. Solutions profile of Problem 5.2. (a) Exact and approximate solutions at $t = 4$, $\tau = 0.001$. (b) Absolute error in (a). (c) Exact 3D plot. (d) Approximate 3D plot at $t = 4$, $\tau = 0.01$, $\lambda = 5$.

Table 5. Convergence rate of Problem 5.2 at $t = 1$.

| λ | τ | L_∞ | Rate |
|-----------|--------|-------------------------|--------|
| 2 | 1/100 | 1.5254×10^{-2} | |
| 3 | 1/200 | 3.9824×10^{-3} | 1.9374 |
| 4 | 1/400 | 9.5864×10^{-4} | 2.0545 |
| 5 | 1/800 | 2.1397×10^{-4} | 2.1635 |

6.3. Problem 5.3

Now, we consider the following equation:

$$\partial_{tt}u(x,t) + \partial_{xx}((1 + \sin \pi x)\partial_{xx}u(x,t)) = \mathcal{E}(x,t), \quad 0 \leq x \leq 1, \quad 0 < t \leq T, \quad (40)$$

inscribed with the initial conditions

$$u(x,0) = \partial_t u(x,0) = 0, \quad 0 \leq x \leq 1, \quad (41)$$

and the boundary conditions

$$\begin{cases} u(0,t) = 0, & \partial_x u(0,t) = t^2 e^{-t}, & 0 < t \leq T, \\ u(1,t) = t^2 e^{-t}, & \partial_x u(1,t) = t^2 e^{-t}, & 0 < t \leq T. \end{cases} \quad (42)$$

The exact solution of this problem is given by

$$u(x,t) = (x + \sin^3 \pi x)t^2 e^{-t}.$$

The approximate solution of this problem was computed at different resolution levels. In Table 6, we present the error norms for different times ($t = 0.2, 0.5, 1, 4$) using $\lambda = 4, 5$. It is observed from the table that the error norms are small, which shows the efficiency of the suggested scheme. The absolute error in displacement at $x = 0.5$ matches those reported in [41] and presented in Table 7. It is clear from the table that the computed errors at different times are small. Moreover, the spectral radius for this problem and for previous two problems are addressed in Table 8 which shows that stability condition fulfilled. Graphical solutions and the absolute error are shown in Figure 3. From the figure, one can see that the exact and approximated solutions agree mutually.

Table 6. Maximum error norms of Problem 5.3 at different times with $\tau = 0.001$.

| λ | $t = 0.2$ | | $t = 0.5$ | | $t = 1$ | | $t = 4$ | |
|-----------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| | L_∞ | L_2 | L_∞ | L_2 | L_∞ | L_2 | L_∞ | L_2 |
| 4 | 2.73×10^{-3} | 1.93×10^{-2} | 1.29×10^{-2} | 9.08×10^{-2} | 3.14×10^{-2} | 2.20×10^{-1} | 2.50×10^{-2} | 1.75×10^{-1} |
| 5 | 2.72×10^{-3} | 1.93×10^{-3} | 1.27×10^{-2} | 9.07×10^{-2} | 3.08×10^{-2} | 2.20×10^{-1} | 2.46×10^{-2} | 1.75×10^{-1} |

Table 7. Absolute error in displacement at $x = 0.5$ of Problem 5.3.

| Methods | Points | τ | $t = 0.2$ | $t = 0.4$ | $t = 0.8$ | $t = 1$ | $t = 2$ | $t = 4$ |
|---------|--------|--------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| Present | 32 | 0.01 | 1.40×10^{-3} | 1.51×10^{-3} | 1.46×10^{-4} | 6.95×10^{-4} | 3.62×10^{-3} | 2.68×10^{-3} |
| [41] | 100 | 0.01 | 7.82×10^{-3} | 2.59×10^{-2} | 7.27×10^{-2} | 9.43×10^{-2} | 1.44×10^{-1} | 7.65×10^{-2} |

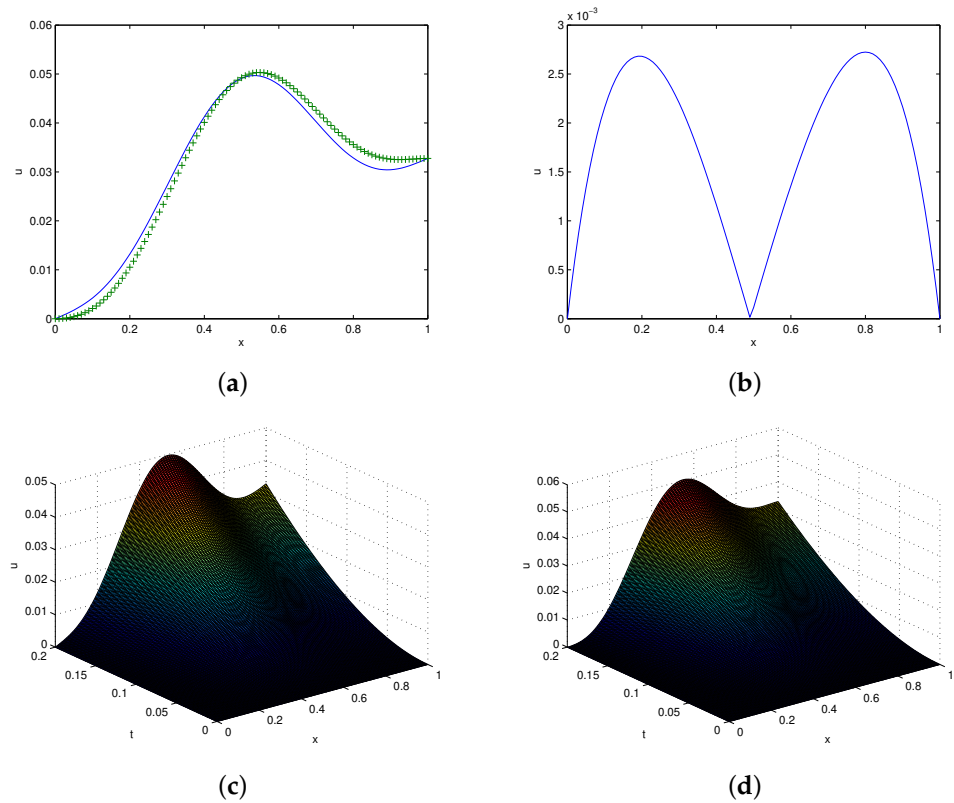


Figure 3. Solutions profile of problem 5.3. (a) Exact and approximate solutions at $t = 0.2$, $\tau = 0.001$. (b) Absolute error in (a). (c) Exact 3D plot. (d) = Approximate 3D plot at $t = 0.2$, $\tau = 0.01$, $\lambda = 5$.

Table 8. Spectral radius of Problems 5.1, 5.2, and 5.3 at $t = 0.2$.

| λ | Problem 5.1 | Problem 5.2 | Problem 5.3 |
|-----------|-------------|-------------|-------------|
| | $\rho(\Xi)$ | $\rho(\Xi)$ | $\rho(\Xi)$ |
| 1 | 0.99984 | 0.99937 | 0.99937 |
| 2 | 0.99984 | 0.99929 | 0.99929 |
| 3 | 0.99984 | 0.99927 | 0.99927 |
| 4 | 0.99984 | 0.99927 | 0.99927 |

7. Initial Disturbance and Noisy Data

Here, we discuss the effect of the small perturbation in the initial condition and the noisy data. In the initial perturbation, we take the initial condition $u_0 = Y_1(x) + \epsilon$ where ϵ is a small number. The idea of this perturbation was taken from [42]. Simulations were performed for different values of resolution levels λ and for different values of ϵ , which are given in Table 9. From the table, we concluded that the small perturbation in the initial data produces a small change in the solution, which shows that the method is stable. Similarly, we take $u_0(x_i) = Y_1(x_i) + (-1)^{i} \times \epsilon$, $i = 1, \dots, 2M$, for the noisy case [43]. In Table 10, we present the numerical results for the noisy case. From the table, it is pretty much clear that the variation in the resolution level for the noisy case still gives stable solutions.

In both cases, when we increase the resolution level, the accuracy increases, which is obvious from Tables 9 and 10. Besides this, the absolute error for the noisy case is plotted in Figure 4, which clearly indicates the accuracy of the techniques because this agrees with the previous error when no noise exists. The simulations and graphical error indicated that the method works for both types of data.

Table 9. Maximum error norms of all problems for $t = 1.0$, after initial disturbances.

| $\epsilon = 10^{-2}$ | | Problem 5.1 | | Problem 5.2 | | Problem 5.3 | |
|----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|--|
| λ | L_∞ | L_2 | L_∞ | L_2 | L_∞ | L_2 | |
| 1 | 4.53×10^{-2} | 2.02×10^{-1} | 4.82×10^{-1} | 3.16×10^0 | 1.30×10^{-1} | 2.27×10^{-1} | |
| 2 | 2.10×10^{-2} | 9.41×10^{-2} | 1.67×10^{-2} | 8.62×10^{-2} | 4.90×10^{-2} | 3.09×10^{-1} | |
| 3 | 1.46×10^{-2} | 6.55×10^{-2} | 4.14×10^{-3} | 2.18×10^{-2} | 3.40×10^{-2} | 2.26×10^{-1} | |
| 4 | 1.30×10^{-2} | 5.82×10^{-2} | 1.34×10^{-3} | 5.94×10^{-3} | 3.15×10^{-2} | 2.20×10^{-1} | |
| $\epsilon = 10^{-3}$ | | Problem 5.1 | | Problem 5.2 | | Problem 5.3 | |
| λ | L_∞ | L_2 | L_∞ | L_2 | L_∞ | L_2 | |
| 1 | 3.64×10^{-2} | 1.63×10^{-1} | 4.78×10^{-1} | 3.16×10^0 | 1.36×10^{-1} | 7.63×10^{-1} | |
| 2 | 1.18×10^{-2} | 5.30×10^{-2} | 1.55×10^{-2} | 8.56×10^{-2} | 5.02×10^{-2} | 3.16×10^{-1} | |
| 3 | 5.38×10^{-3} | 2.40×10^{-2} | 3.99×10^{-3} | 2.18×10^{-2} | 3.39×10^{-2} | 2.26×10^{-1} | |
| 4 | 3.75×10^{-3} | 1.67×10^{-2} | 9.02×10^{-4} | 4.70×10^{-3} | 3.13×10^{-2} | 2.20×10^{-1} | |

Table 10. Maximum error norms of all problems for $t = 1.0$ $\epsilon = 10^{-2}$, with noisy initial data.

| Noise = 1% | | Problem 5.1 | | Problem 5.2 | | Problem 5.3 | |
|------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|--|
| λ | L_∞ | L_2 | L_∞ | L_2 | L_∞ | L_2 | |
| 1 | 3.55×10^{-2} | 1.58×10^{-1} | 4.77×10^{-1} | 3.16×10^0 | 1.37×10^{-1} | 7.67×10^{-1} | |
| 2 | 1.08×10^{-2} | 4.84×10^{-2} | 1.54×10^{-2} | 8.56×10^{-2} | 5.03×10^{-2} | 3.16×10^{-1} | |
| 3 | 4.35×10^{-3} | 1.94×10^{-2} | 3.98×10^{-3} | 2.18×10^{-2} | 3.39×10^{-2} | 2.26×10^{-1} | |
| 4 | 2.72×10^{-3} | 1.21×10^{-2} | 8.47×10^{-4} | 4.69×10^{-3} | 3.13×10^{-2} | 2.20×10^{-1} | |

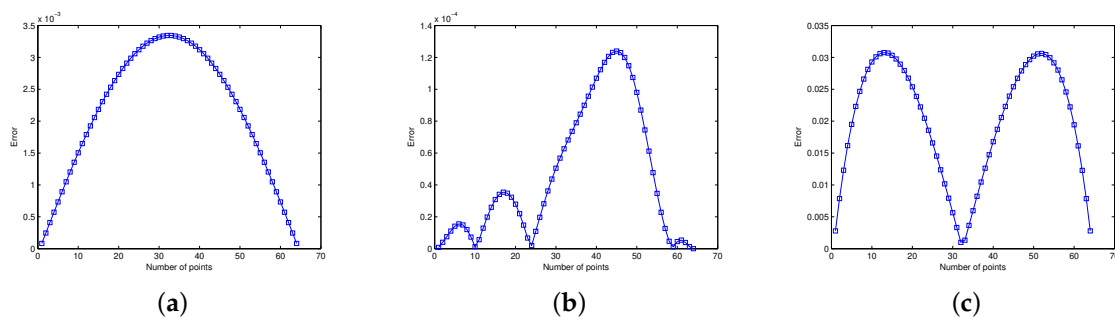


Figure 4. Error profile of all problems at $t = 1.0$. (a) Error in Problem 5.1. (b) Error in Problem 5.2. (c) Error in Problem 5.3.

8. Conclusions

In this work, a mixed numerical method based on the Haar wavelet coupled with the finite difference was proposed to solve the FOPDEs. First, the scheme was tested with initial data, and then, small perturbations with and with out noise were introduced. The obtained results matched with earlier work and exact solutions. Furthermore, the accuracy of the scheme was checked via computing the L_2 and L_∞ error norms. It was observed that the proposed scheme works well for smooth and noisy initial data, which indicates that the method can be applied for other such problems.

Author Contributions: Conceptualization, A.G. and S.H.; Methodology, A.G.; Software, A.G.; Validation, S.H., T.A. and M.A.A.; Formal Analysis, A.G.; Investigation, M.H.; Resources, T.A.; Writing–Original Draft Preparation, A.G.; Writing–Review and Editing, T.A.; Visualization, M.A.A.; Supervision, S.H. and T.A.; Funding Acquisition, M.A.A. and T.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research work is supported by Princess Nourah Bint Abdulrahman University researchers' program with Project Number PNURSP2022R14.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: If interested readers need the data, please contact the first author via email.

Acknowledgments: The authors are thankful to the anonymous reviewers for their valuable comments, which improved the quality of the manuscript. The author Thabet Abdeljawad would like to thank Prince Sultan University for the support through the TAS research lab, and the author Manar A. Alqudah acknowledges the financial support of Princess Nourah Bint Abdulrahman University researchers' program Project Number PNURSP2022R14.

Conflicts of Interest: The authors declare that they have no conflict of interest.

References

- Ahn, J.; Stewart, D.E. An Euler–Bernoulli beam with dynamic frictionless contact: penalty approximation and existence. *Numer. Funct. Anal. Optim.* **2007**, *28*, 1003–1026. [\[CrossRef\]](#)
- Kunisch, K.; Graif, E. Parameter estimation for the Euler–Bernoulli beam. *Mat. Applicada Comput.* **1985**, *4*, 95–124.
- Timoshenko, S.P.; Gere, J.M. *Theory of Elastic Stability*; McGraw-Hill: New York, NY, USA, 1961.
- Wazwaz, A.M. Analytic treatment for variable coefficient fourth-order parabolic partial differential equations. *Appl. Math. Comput.* **2001**, *123*, 219–227. [\[CrossRef\]](#)
- Liu, Y.; Gurram, C.S. The use of Hes variational iteration method for obtaining the free vibration of an EulerBernoulli beam. *Math. Comput. Model.* **2009**, *50*, 1545–1552. [\[CrossRef\]](#)
- Jain, M.K.; Iyengar, S.R.K.; Lone, A.G. Higher order difference formulas for a fourth order parabolic partial differential equation. *Int. J. Numer. Methods Eng.* **1976**, *10*, 1357–1367. [\[CrossRef\]](#)
- Evans, D.J. A stable explicit method for the finite difference solution of a fourth order parabolic partial differential equation. *Comput. J.* **1965**, *8*, 280–287. [\[CrossRef\]](#)
- Conte, S.D. A stable implicit finite difference approximation to a fourth order parabolic equation. *J. Assoc. Comput. Mech.* **1957**, *4*, 18–23. [\[CrossRef\]](#)
- Richtmyer, R.D.; Morton, K.W. *Difference Methods for Initial Value Problems*, 2nd ed.; Wiley-Interscience: New York, NY, USA, 1967.
- Crandall, S.H. Numerical treatment of a fourth order partial differential equations. *J. Assoc. Comput. Mech.* **1954**, *1*, 111–118. [\[CrossRef\]](#)
- Danaee, A.; Khan, A.; Khan, I.; Aziz, T.; Evans, D.J. Hopscotch procedure for a fourthorder parabolic partial differential equation. *Math. Comput. Simul.* **1982**, *24*, 326–329. [\[CrossRef\]](#)
- Aziz, T.; Khan, A.; Rashidinia, J. Spline methods for the solution of fourth-order parabolic partial differential equations. *Appl. Math. Comput.* **2005**, *167*, 153–166. [\[CrossRef\]](#)
- Rashidinia, J.; Mohammadi, R. Sextic spline solution of variable coefficient fourthorder parabolic equations. *Int. J. Comput. Math.* **2010**, *87*, 3443–3454. [\[CrossRef\]](#)
- Ahmad, I.; Hijaz, A.; Inc, M.; Rezapadeh, H.; Akbar, M.A.; Khater, M.M.A.; Akinyemi, L.; Jhangeer, A. Solution of fractional-order Korteweg-de Vries and Burgers' equations utilizing local meshless method. *J. O. Eng. Sci.* **2021**, *1*–12. [\[CrossRef\]](#)
- Senol, M.; Akinyemi, L.; Ata, A.; Iyiola, O.S. SApproximate and generalized solutions of conformable type Coudrey–Dodd–Gibbon–Sawada–Kotera equation. *Int. J. Mod. Phys. B* **2021**, *35*, 2150021. [\[CrossRef\]](#)
- Akinyemi, L.; Iyiola, O.S. Analytical Study of $(3 + 1)$ -Dimensional Fractional-Reaction Diffusion Trimolecular Models. *Int. J. Appl. Comp. Math.* **2021**, *7*, 1–24. [\[CrossRef\]](#)
- Akinyemi, L.; Veerasha, P.; Ajibola, S.O. Numerical simulation for coupled nonlinear Schrödinger–Korteweg–de Vries and Maccari systems of equations. *Mod. Phys. Lett. B* **2021**, *35*, 2150339. [\[CrossRef\]](#)
- Jiwari, R. Barycentric rational interpolation and local radial basis functions based numerical algorithms for multidimensional sine-Gordon equation. *Num. Meth. Part. Diff. Eqs.* **2021**, *37*, 1965–1992. [\[CrossRef\]](#)
- Bertoluzza, S. An adaptive collocation method based on interpolating wavelets. In *Multi-Scale Wavelet Methods for Partial Differential Equations*; Dahmen, W., Kurdila, A.J., Oswald, P., Eds.; Academic Press: San Diego, CA, USA, 1977; pp. 109–135.
- Beylkin, G.; Keiser, J.M. An adaptive pseudo-wavelet approach for solving nonlinear partial differential equations. In *Multi-Scale Wavelet Methods for Partial Differential Equations*; Dahmen, W., Kurdila, A.J., Oswald, P., Eds.; Academic Press: San Diego, CA, USA, 1977; pp. 137–197.
- Cattani, C. Haar wavelet splines. *J. Interdiscip. Math.* **2001**, *4*, 35–47. [\[CrossRef\]](#)
- Cattani, C. Haar wavelets based technique in evolution problems. *Proc. Estonian Acad. Sci. Phys. Math.* **2004**, *1*, 45–63. [\[CrossRef\]](#)
- Chen, C.F.; Hasio, C.H. Haar wavelet method for solving lumped and distributedparameter systems. *IEE Proc. Number Control Theory Appl.* **1997**, *144*, 87–94. [\[CrossRef\]](#)

24. Lepik, U. Numerical solution of evolution equations by the Haar wavelet method. *Appl. Math. Comput.* **2007**, *185*, 695–704. [[CrossRef](#)]
25. Lepik, U. Solving PDEs with the aid of two-dimensional Haar wavelets. *Comput. Math. Appl.* **2011**, *61*, 1873–1879. [[CrossRef](#)]
26. Jiwari, R. A Haar wavelet quasilinearization approach for numerical simulation of Burgers equation. *Comput. Phys. Comm.* **2012**, *183*, 2413–2423. [[CrossRef](#)]
27. Mittal, R.C.; Kaur, H.; Mishra, V. Haar Wavelet Based Numerical Investigation of Coupled Viscous Burgers equation. *Int. J. Comput. Appl.* **2015**, *92*, 1643–1659. [[CrossRef](#)]
28. Oruç, Ö.; Bulut, F.; Esen, A. A Haar wavelet-finite difference hybrid method for the numerical solution of the modified Burgers equation. *J. Math. Chem.* **2015**, *53*, 1592–1607. [[CrossRef](#)]
29. Oruç, Ö.; Bulut, F.; Esen, A. Numerical solution of the KdV equation by Haar wavelet method. *Pramana J. Phys.* **2016**, *87*, 94. [[CrossRef](#)]
30. Kumar, M.; Pandit, S. A composite numerical scheme for the numerical simulation of coupled Burgers equation. *Comput. Phys. Commun.* **2014**, *185*, 809–817. [[CrossRef](#)]
31. Arbabi, S.; Nazari, A.; Darvishi, M.T. A semi-analytical solution of Hunter-Saxton equation. *Optik* **2016**, *127*, 5255–5258. [[CrossRef](#)]
32. Arbabi, S.; Nazari, A.; Darvishi, M.T. A semi-analytical solution of foam drainage equation by Haar wavelets method. *Optik* **2016**, *127*, 5443–5447. [[CrossRef](#)]
33. Arbabi, S.; Nazari, A.; Darvishi, M.T. A two-dimensional Haar wavelets method for solving systems of PDEs. *Appl. Math. Comp.* **2017**, *292*, 33–46. [[CrossRef](#)]
34. Mittal, R.C.; Pandit, S. Numerical simulation of unsteady squeezing nanofluid and heat flow between two parallel plates using wavelets. *Int. J. Ther. Sci.* **2017**, *118*, 417–422. [[CrossRef](#)]
35. Jiwari, R. A hybrid numerical scheme for the numerical solution of the Burgers' equation. *Comput. Phys. Commun.* **2015**, *188*, 59–67. [[CrossRef](#)]
36. Pandit, S.; Mittal, R.C. A numerical algorithm based on scale-3 Haar wavelets for fractional advection dispersion equation. *Eng. Comput.* **2020**, *38*, 1706–1724 [[CrossRef](#)]
37. Pandit, S.; Jiwari, R.; Bedi, K.; Koxsal, M.E. Haar wavelets operational matrix based algorithm for computational modeling of hyperbolic type wave equations. *Eng. Comput.* **2017**, *34*, 2793–2814. [[CrossRef](#)]
38. Haq, S.; Ghafoor, A. An efficient numerical algorithm for multi-dimensional time dependent partial differential equations. *Comput. Math. Appl.* **2018**, *75*, 2723–2734. [[CrossRef](#)]
39. Mittal, R.C.; Jain, R.K. B-Splines methods with redefined basis functions for solving fourth order parabolic partial differential equations. *Appl. Math. Comput.* **2011**, *217*, 9741–9755. [[CrossRef](#)]
40. Caglar, H.; Caglar, N. Fifth-degree B-spline solution for a fourth-order parabolic partial differential equations. *Appl. Math. Comput.* **2008**, *201*, 597–603. [[CrossRef](#)]
41. Mohammadi, R. Sextic B-spline collocation method for solving Euler–Bernoulli Beam Models. *Appl. Math. Comput.* **2014**, *241*, 151–166. [[CrossRef](#)]
42. Shivanian, E.; Jafarabadi, A. Inverse Cauchy problem of annulus domains in the framework of spectral meshless radial point interpolation. *Eng. Comput.* **2017**, *33*, 431–442. [[CrossRef](#)]
43. Solodusha, S.V.; Mokry, I.V. A numerical solution of one class of Volterra integral equations of the first kind in terms of the machine arithmetic features. *Bull. SUSU MMCS* **2016**, *9*, 119–129. [[CrossRef](#)]