Palestine Journal of Mathematics

Vol. 12(Special Issue IV)(2023) , 119–131

Approximate Solution of Fourth Order Parabolic Equation using Splines

Sandeep Bhatt, Swarn Singh and Suruchi Singh

MSC 2010 Classifications: Primary 13F10, 20M99; Secondary 13M05, 13A15.

Keywords and phrases: Fourth Order Method, Spline, Cubic B-Spline, Fourth order parabolic PDE.

Abstract In the present article, we compute the fourth order accurate solution of the fourth order parabolic PDE that describes the behavior of a vibrating beam. We derive a new compact method by using cubic B-splines. We have used the Crank-Nicolson method for discretization along the time direction. We have exhibited the unconditional stability of the method using the stability analysis. A few numerical experiments have been done to validate the accuracy of the method. The advantages of the current method over the methods available are its easy implementation and lack of computational efforts.

1 Introduction

In this article, we are interested in deriving a stable and high accurate method to find the approximate solution of a fourth order parabolic PDE

$$u_{tt} + u_{xxxx} = f, \tag{1.1}$$

where $(x,t) \in [a,b] \times [0,T]$. The IC's and BC's are given by

$$\begin{aligned} &u(x,0) = \alpha_1(x),\, u_t(x,0) = \alpha_2(x), \quad x \in [a,b],\\ &u(a,t) = \beta_1(t),\, u(b,t) = \beta_2(t),\\ &u_{xx}(a,t) = \beta_3(t),\, u_{xx}(b,t) = \beta_4(t), \forall\, t \in [0,T]. \end{aligned} \tag{1.2}$$

Various numerical methods have been discussed in the literature to solve the fourth order parabolic PDE. Mohanty et al. [5, 6] proposed three level implicit stable finite difference methods to compute the numerical solution of the quasi linear fourth order parabolic PDE. Fairweather and Gourlay [7] formulated explicit and implicit finite difference methods, Evans and Yousif [8] developed an alternating group explicit (AGE) method, Conte [22] proposed an 11 point based stable implicit finite difference scheme to solve the fourth order parabolic PDE. An implicit compact difference scheme with three levels was presented in [10]to find the generalized form of the fourth order parabolic PDE. Many authors used spline techniques to solve the fourth order partial PDEs. In [12], authors used fifth degree B-splines, Aziz et al. [13] solved equation (1.1) using a three level method based on finite difference discretization in time and parametric quintic spline in space. Warwaz [14] solved the variable coefficient fourth order parabolic by applying the adomian decomposition method. This method computed the solution in a series form. Mittal and Jain [15] used cubic B-splines and quintic B-splines to derive two unconditionally stable methods for solving equation (1.1). Rashidinia and Mohammadi [16] proposed a sextic spline collocation method for the approximate solution of a fourth order non homogeneous parabolic PDE with variable coefficients. Sinc-Galerkin method to compute the numerical solution of a variable coefficient fourth-order PDE is presented in El-Gamel [17]. By using finite difference discretization in time and nonpolynomial cubic tension spline in space, Sultana and Khandelwal [20] computed the solution of fourth order parabolic PDE. Khan and Sultana [21] developed a three level implicit method based on finite difference discretization in time and parametric septic spline in space to solve fourth order nonhomogeneous parabolic PDE. A two-level implicit cubic spline numerical method was proposed by Mohanty and Sharma [22] for the approximate solution of 1D time-dependent quasilinear biharmonic equation. Kaur and

Mohanty [23] proposed a compact difference scheme based on half-step discretization to solve fourth order time dependent PDE. To solve a special type of fourth order parabolic PDE, Mohanty et al. [24] discussed two-level implicit methods by transforming the original problem to a coupled system of two second order parabolic PDEs.

In this article, we present a new compact method to approximate the solution of fourth order parabolic PDE using the collocation of cubic B-splines. We have shown that the method is of order 2 in time and of order 4 in space. Using a tri-diagonal system of equations to find the solution, we computed the solution with less computation effort and greater efficacy. The method is also analyzed to discuss the fourth order accuracy and stability. We have compared the method's accuracy and efficiency with those available in the literature using a few numerical experiments.

We have compiled this article in the following manner: In section 2, we have discussed the collocation method using cubic B-splines to solve the given problem. We present how to implement the method and the stability in section 3. Section 4 describes the computation of initial approximation using IC's and BC's. Using the proposed method, we have done some numerical experiments in section 5. The computed solutions are compared with that available in the literature to show the accuracy and reliability of the proposed method. Conclusions about the proposed method are presented in section 6.

2 Description of the Method

We introduce new variables v and w as

$$v = u_{xx} \quad \text{and} \quad w = u_t. \tag{2.1}$$

Using equations (2.1), equation (1.1) can be written in the form of two simultaneous partial differential equations as

$$\begin{aligned} &w_t + v_{xx} = f, \\ &v_t = w_{xx}. \end{aligned} \tag{2.2}$$

Equation (2.2) can be written as

$$\mathbf{P}_t = \mathbf{C}\mathbf{P}_{xx} + \mathbf{F}, \tag{2.3}$$

where

$$\mathbf{P} = \begin{pmatrix} v \\ w \end{pmatrix}, \quad \mathbf{C} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \quad \text{and} \quad \mathbf{F} = \begin{pmatrix} 0 \\ f \end{pmatrix}. \tag{2.4}$$

Let us consider a uniformly spaced partition $\Pi = \{x_0 < x_1 < x_2 < \ldots < x_{J-2} < x_{J-1} < x_J\}$ of interval $[a, b]$ where $x_j = a + jh, j = 0, 1, \ldots, J$; where $h = \dfrac{b-a}{J}$. Let V and W be approximations of v and w, respectively, obtained using the cubic B-spline collocation method. Therefore we have

$$\begin{aligned} \overline{V}(x,t) &= \sum_{j=-1}^{J+1} \overline{v}_j(t)\Psi_j^3(x), \\ \overline{W}(x,t) &= \sum_{j=-1}^{J+1} \overline{w}_j(t)\Psi_j^3(x), \end{aligned} \tag{2.5}$$

where $\overline{v}_j(t)$ and $\overline{w}_j(t)$ are unknowns that depend on time and will be evaluated later. In the domain $[a, b]$, The set of piecewise cubic polynomials $\{\Psi_{-1}^3, \Psi_0^3, \Psi_1^3, \ldots, \Psi_j^3, \Psi_{J+1}^3\}$ serves as a basis for the space of all the cubic splines over the partition Π. The cubic B-spline functions $\Psi_j^3(x)$ at the nodes are as follows (Boor [21]):

$$\Psi_j^3(x) = \frac{1}{h^3}\begin{cases} (x-x_j)^3, & x \in [x_{j-2}, x_{j-1}), \\ 3(x_{j+1}-x)^3 + 3h(x_{j+1}-x)^2 + 3h^2(x-x_{j+1}) + h^3, & x \in [x_{j-1}, x_j), \\ 3(x-x_{j+1})^3 + 3h(x-x_{j+1})^2 + 3h^2(x_{j+1}-x) + h^3, & x \in [x_j, x_{j+1}), \\ (x_{j+2}-x)^3, & x \in [x_{j+1}, x_{j+2}), \\ 0, & \text{otherwise.} \end{cases} \tag{2.6}$$

With the help of equation (2.6), values of $\Psi_j^3(x), \Psi_j^{3'}(x)$ and $\Psi_j^{3''}(x)$ at grid points are displayed in the following manner:

$$\Psi_j^3(x) = \begin{cases} 1, & x = x_{j+1}, x_{j+3}, \\ 4, & x = x_{j+2}, \\ 0, & \text{otherwise.} \end{cases}, \quad \Psi_j^{3'}(x) = \begin{cases} \frac{3}{h}, & x = x_{j+1}, \\ \frac{-3}{h}, & x = x_{j+3}, \\ 0, & \text{otherwise.} \end{cases},$$
$$\Psi_j^{3''}(x) = \begin{cases} \frac{6}{h^2}, & x = x_{j+1}, x_{j+3}, \\ \frac{-12}{h^2}, & x = x_{j+2}, \\ 0, & \text{otherwise.} \end{cases}. \tag{2.7}$$

Let $\overline{\mathbf{P}}(x,t)$ be a cubic spline. Therefore $\overline{\mathbf{P}}(x,t)$ is given by

$$\overline{\mathbf{P}}(x,t) = \sum_{j=-1}^{J+1} \overline{\mathbf{p}}_j(t)\Psi_j^3(x), \tag{2.8}$$

where $\overline{\mathbf{p}}_j(t) = [\overline{v}_j(t), \overline{w}_j(t)]^T$, for $j = -1, 0, 1, ..., J+1$. After calculating the values $\overline{\mathbf{p}}_j(t)$ at time t, one can easily compute the approximate solution $\overline{\mathbf{P}}(x,t)$ at that particular time level. The approximate value of the solution $\overline{\mathbf{P}}$, the derivatives $\overline{\mathbf{P}}_x \equiv D_x\overline{\mathbf{P}}$ and $\overline{\mathbf{P}}_{xx} \equiv D_x^2\overline{\mathbf{P}}$, at the node x_j in terms of parameters $\overline{\mathbf{p}}_j \equiv \overline{\mathbf{p}}_j(t)$ using (2.7) and (2.8) are as follows:

$$\begin{aligned} \overline{\mathbf{P}}_j &= \overline{\mathbf{p}}_{j-1} + 4\overline{\mathbf{p}}_j + \overline{\mathbf{p}}_{j+1}, \\ (D_x\overline{\mathbf{P}})_j &= \frac{3}{h}\overline{\mathbf{p}}_{j+1} - \frac{3}{h}\overline{\mathbf{p}}_{j-1}, \\ (D_x^2\overline{\mathbf{P}})_j &= \frac{6}{h^2}\overline{\mathbf{p}}_{j-1} - \frac{12}{h^2}\overline{\mathbf{p}}_j + \frac{6}{h^2}\overline{\mathbf{p}}_{j+1}, \end{aligned} \tag{2.9}$$

for $j = 0, 1, \ldots, J$. We will use the following lemma to derive the method.

Lemma 2.1. *Let* $\mathbf{P} \in C^6[a,b]$ *be the exact solution of equation* (2.3). *If* $\mathbf{S}$ *represents a cubic spline interpolation of* $\mathbf{P}$, *defined as*

$$\mathbf{S}_j = \mathbf{P}_j, \quad j = 0, 1, 2, \ldots, J,$$
$$D_x^2\mathbf{S}_j = D_x^2\mathbf{P}_j - \frac{h^2}{12}D_x^4\mathbf{P}_j + O(h^4), \quad j = 0, J.$$

Then from Lucas [2]:

(i) $\mathbf{P}_j = \mathbf{S}_j + O(h^4)$,

(ii) $D_x\mathbf{P}_j = D_x\mathbf{S}_j + O(h^3)$,

(iii) $D_x^2\mathbf{P}_j = D_x^2\mathbf{S}_j + O(h^2)$,

(iv) $D_x^2\mathbf{S}_j = D_x^2\mathbf{P}_j - \frac{h^2}{12}D_x^4\mathbf{P}_j + O(h^4)$,

for $0 \le j \le J$.

Consider,

$$\begin{aligned}\mathbf{F}_j - \frac{h^2}{12}\mathbf{F}_{xx_j} &= \mathbf{P}_{tj} - \mathbf{C}\mathbf{P}_{xxj} - \frac{h^2}{12}\left(D_x^2\mathbf{P}_{tj} - D_x^4\mathbf{C}\mathbf{P}_j\right)\\ &= \mathbf{P}_{tj} - \frac{h^2}{12}D_x^2\mathbf{P}_{tj} - \mathbf{C}\left(D_x^2\mathbf{P}_j - \frac{h^2}{12}D_x^4\mathbf{P}_j\right)\end{aligned}$$

Therefore, using Lemma 2.1, the fourth order accurate method for the solution of the PDE (2.3) is as follows:

$$\mathcal{L}_1\overline{\mathbf{P}}_j + \mathcal{L}_2\overline{\mathbf{P}}_{t_j} = \mathbf{F}_j - \frac{h^2}{12}\mathbf{F}_{xx_j} \quad \text{for} \quad j = 0, 1, \ldots, J, \tag{2.10}$$

where

$$\begin{aligned}\mathcal{L}_1\overline{\mathbf{P}}_j &= -\mathbf{C}D_x^2\overline{\mathbf{P}}_j\\ &= \begin{pmatrix} 0 & -D_x^2 \\ D_x^2 & 0 \end{pmatrix}\begin{pmatrix} \overline{\mathbf{V}}_j \\ \overline{\mathbf{W}}_j \end{pmatrix},\end{aligned} \tag{2.11}$$

and

$$\begin{aligned}\mathcal{L}_2\overline{\mathbf{P}}_{t_j} &= \left(\mathbf{I} - \frac{h^2}{12}\mathbf{I}D_x^2\right)\overline{\mathbf{P}}_{t_j}\\ &= \begin{pmatrix} 1 - \frac{h^2}{12}D_x^2 & 0 \\ 0 & 1 - \frac{h^2}{12}D_x^2 \end{pmatrix}\begin{pmatrix} \overline{\mathbf{V}}_{t_j} \\ \overline{\mathbf{W}}_{t_j} \end{pmatrix},\end{aligned} \tag{2.12}$$

with $\overline{\mathbf{P}} = \mathbf{P} = \begin{pmatrix} u_{xx} \\ u_t \end{pmatrix}$ on the boundary of the domain and $\mathbf{I} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$.

3 Implementation of the Method and Stability Analysis

Let us consider $t_n = nk, \forall n = 0, 1, 2 \ldots$, where k represents the step size in the time direction. Using the Crank-Nicolson method in time direction for the equation (2.10), we get

$$\frac{\mathcal{L}_1\overline{\mathbf{P}}_j^{n+1} + \mathcal{L}_1\overline{\mathbf{P}}_j^{n}}{2} + \frac{\mathcal{L}_2\overline{\mathbf{P}}_j^{n+1} - \mathcal{L}_2\overline{\mathbf{P}}_j^{n}}{k} = \frac{\mathbf{F}_j^{n+1} + \mathbf{F}_j^{n}}{2} - \frac{h^2}{12}\frac{\mathbf{F}_{xx_j}^{n+1} + \mathbf{F}_{xx_j}^{n}}{2} \tag{3.1}$$

for $j = 0, 1, \ldots, J$. Using the equations (2.8),(2.10) and (2.12), the equation (3.1) can be written as

$$\begin{aligned}\left(\frac{1}{k}\left(\mathbf{I} - \frac{h^2}{12}\mathbf{I}D_x^2\right) - \frac{1}{2}\mathbf{C}D_x^2\right)\overline{\mathbf{P}}_j^{n+1} = \left(\frac{1}{k}\left(\mathbf{I} - \frac{h^2}{12}\mathbf{I}D_x^2\right) + \frac{1}{2}\mathbf{C}D_x^2\right)\overline{\mathbf{P}}_j^{n} + \frac{\mathbf{F}_j^{n+1} + \mathbf{F}_j^{n}}{2}\\ - \frac{h^2}{12}\frac{\mathbf{F}_{xx_j}^{n+1} + \mathbf{F}_{xx_j}^{n}}{2}.\end{aligned} \tag{3.2}$$

By using (2.4), it can be further written as

$$\begin{aligned}\begin{pmatrix} \frac{1}{k} - \frac{h^2}{12k}D_x^2 & -\frac{1}{2}D_x^2 \\ \frac{1}{2}D_x^2 & \frac{1}{k} - \frac{h^2}{12k}D_x^2 \end{pmatrix}\begin{pmatrix} \overline{v}_j^{n+1} \\ \overline{w}_j^{n+1} \end{pmatrix} = \begin{pmatrix} \frac{1}{k} - \frac{h^2}{12k}D_x^2 & \frac{1}{2}D_x^2 \\ -\frac{1}{2}D_x^2 & \frac{1}{k} - \frac{h^2}{12k}D_x^2 \end{pmatrix}\begin{pmatrix} \overline{v}_j^{n} \\ \overline{w}_j^{n} \end{pmatrix}\\ + \begin{pmatrix} 0 \\ \frac{f_j^{n+1} + f_j^{n}}{2} - \frac{h^2}{12}\frac{f_{xx_j}^{n+1} + f_{xx_j}^{n}}{2} \end{pmatrix},\end{aligned} \tag{3.3}$$

for all $j = 0, 1, \ldots, J$.

Let us consider the matrices

$$\mathbf{B_0} = \begin{pmatrix} 4 & 1 & & & & \\ 1 & 4 & 1 & & & \\ & \cdots & \cdots & \cdots & & \\ & & \cdots & \cdots & \cdots & \\ & & & 1 & 4 & 1 \\ & & & & 1 & 4 \end{pmatrix}, \quad \mathbf{B_2} = \frac{6}{h^2} \begin{pmatrix} -2 & 1 & & & & \\ 1 & -2 & 1 & & & \\ & \cdots & \cdots & \cdots & & \\ & & \cdots & \cdots & \cdots & \\ & & & 1 & -2 & 1 \\ & & & & 1 & -2 \end{pmatrix},$$

and

$$\overline{\mathbf{F}}^n = \frac{1}{2}\left(f_0^n - \frac{h^2}{12} f_{xx_0}^n, f_1^n - \frac{h^2}{12} f_{xx_1}^n, \ldots, f_{J-1}^n - \frac{h^2}{12} f_{xx_{J-1}}^n, f_J^n - \frac{h^2}{12} f_{xx_J}^n \right)^T.$$

Therefore,by using equations (2.7) and (3.3) we have

$$\begin{pmatrix} \frac{1}{k}\mathbf{B_0} - \frac{h^2}{12k}\mathbf{B_2} & -\frac{1}{2}\mathbf{B_2} \\ \frac{1}{2}\mathbf{B_2} & \frac{1}{k}\mathbf{B_0} - \frac{h^2}{12k}\mathbf{B_2} \end{pmatrix} \begin{pmatrix} \overline{\overline{\mathbf{V}}}^{n+1} \\ \overline{\overline{\mathbf{W}}}^{n+1} \end{pmatrix} = \begin{pmatrix} \frac{1}{k}\mathbf{B_0} - \frac{h^2}{12k}\mathbf{B_2} & \frac{1}{2}\mathbf{B_2} \\ -\frac{1}{2}\mathbf{B_2} & \frac{1}{k}\mathbf{B_0} - \frac{h^2}{12k}\mathbf{B_2} \end{pmatrix} \begin{pmatrix} \overline{\overline{\mathbf{V}}}^{n} \\ \overline{\overline{\mathbf{W}}}^{n} \end{pmatrix} + \begin{pmatrix} 0 \\ \overline{\mathbf{F}}^n + \overline{\mathbf{F}}^{n+1} \end{pmatrix},$$

$$\Rightarrow \begin{pmatrix} \overline{\overline{\mathbf{V}}}^{n+1} \\ \overline{\overline{\mathbf{W}}}^{n+1} \end{pmatrix} = \begin{pmatrix} \frac{1}{k}\mathbf{B_0} - \frac{h^2}{12k}\mathbf{B_2} & -\frac{1}{2}\mathbf{B_2} \\ \frac{1}{2}\mathbf{B_2} & \frac{1}{k}\mathbf{B_0} - \frac{h^2}{12k}\mathbf{B_2} \end{pmatrix}^{-1} \begin{pmatrix} \frac{1}{k}\mathbf{B_0} - \frac{h^2}{12k}\mathbf{B_2} & \frac{1}{2}\mathbf{B_2} \\ -\frac{1}{2}\mathbf{B_2} & \frac{1}{k}\mathbf{B_0} - \frac{h^2}{12k}\mathbf{B_2} \end{pmatrix} \begin{pmatrix} \overline{\overline{\mathbf{V}}}^{n} \\ \overline{\overline{\mathbf{W}}}^{n} \end{pmatrix}$$
$$+ \begin{pmatrix} \frac{1}{k}\mathbf{B_0} - \frac{h^2}{12k}\mathbf{B_2} & -\frac{1}{2}\mathbf{B_2} \\ \frac{1}{2}\mathbf{B_2} & \frac{1}{k}\mathbf{B_0} - \frac{h^2}{12k}\mathbf{B_2} \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ \overline{\mathbf{F}}^n + \overline{\mathbf{F}}^{n+1} \end{pmatrix}. \tag{3.4}$$

where $\overline{\overline{\mathbf{V}}}^n = [\overline{v}_0^n, \overline{v}_1^n, \overline{v}_2^n, \ldots, \overline{v}_{J-2}^n, \overline{v}_{J-1}^n, \overline{v}_J^n]^T$, and $\overline{\overline{\mathbf{W}}}^n = [\overline{w}_0^n, \overline{w}_1^n, \overline{w}_2^n, \ldots, \overline{w}_{J-2}^n, \overline{w}_{J-1}^n, \overline{w}_J^n]^T$. We can write equation (3.4) as

$$\mathbf{R}^{n+1} = \mathcal{M}\mathbf{R}^n + \mathcal{F}^n, \tag{3.5}$$

where

$$\mathcal{M} = \begin{pmatrix} \frac{1}{k}\mathbf{B_0} - \frac{h^2}{12k}\mathbf{B_2} & -\frac{1}{2}\mathbf{B_2} \\ \frac{1}{2}\mathbf{B_2} & \frac{1}{k}\mathbf{B_0} - \frac{h^2}{12k}\mathbf{B_2} \end{pmatrix}^{-1} \begin{pmatrix} \frac{1}{k}\mathbf{B_0} - \frac{h^2}{12k}\mathbf{B_2} & \frac{1}{2}\mathbf{B_2} \\ -\frac{1}{2}\mathbf{B_2} & \frac{1}{k}\mathbf{B_0} - \frac{h^2}{12k}\mathbf{B_2} \end{pmatrix},$$
$$= \begin{pmatrix} \frac{1}{k}\mathbf{I_n} - \frac{h^2}{12k}\mathbf{B_0}^{-1}\mathbf{B_2} & -\frac{1}{2}\mathbf{B_0}^{-1}\mathbf{B_2} \\ \frac{1}{2}\mathbf{B_0}^{-1}\mathbf{B_2} & \frac{1}{k}\mathbf{I_n} - \frac{h^2}{12k}\mathbf{B_0}^{-1}\mathbf{B_2} \end{pmatrix}^{-1} .$$
$$. \begin{pmatrix} \frac{1}{k}\mathbf{I_n} - \frac{h^2}{12k}\mathbf{B_0}^{-1}\mathbf{B_2} & \frac{1}{2}\mathbf{B_0}^{-1}\mathbf{B_2} \\ -\frac{1}{2}\mathbf{B_0}^{-1}\mathbf{B_2} & \frac{1}{k}\mathbf{I_n} - \frac{h^2}{12k}\mathbf{B_0}^{-1}\mathbf{B_2} \end{pmatrix}, \tag{3.6}$$

$$\mathcal{F}^n = \begin{pmatrix} \frac{1}{k}\mathbf{B_0} - \frac{h^2}{12k}\mathbf{B_2} & -\frac{1}{2}\mathbf{B_2} \\ \frac{1}{2}\mathbf{B_2} & \frac{1}{k}\mathbf{B_0} - \frac{h^2}{12k}\mathbf{B_2} \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ \overline{\mathbf{F}}^n + \overline{\mathbf{F}}^{n+1} \end{pmatrix}, \tag{3.7}$$

$$\mathbf{R}^n = \begin{pmatrix} \overline{\overline{\mathbf{V}}}^{n} \\ \overline{\overline{\mathbf{W}}}^{n} \end{pmatrix} \text{ and } \quad \mathbf{I_n} \text{ is an identity matrix of order } n. \tag{3.8}$$

Once the value of $\mathbf{R}^0$ is evaluated, we can use equation (3.5) to calculate $\mathbf{R}^n$ for any value of n. Now, the approximate solution $U(x,t)$ of the exact solution $u(x,t)$ of equation (1.1) can be obtained from the values of $\overline{\mathbf{V}}^n$ by means of the cubic B-spline collocation method of the differential equation

$$v = u_{xx}. \tag{3.9}$$

Therefore, the approximate solution $U(x,t)$ using the cubic B-spline collocation method is given in the following form

$$U(x,t) = \sum_{j=-1}^{J+1} c_j(t)\Psi_j^3(x), \tag{3.10}$$

where $c_j{}'s$ need to be determined. The method to solve the equation (3.9) is given as

$$D_x{}^2 U_j^n = \left(1 - \frac{h^2}{12} D_x{}^2\right) \overline{\mathbf{V}}_j^n, \quad \text{for} \quad j = 0, 1, \ldots, J.$$

Using equations (2.7), we have the system of equations

$$\begin{aligned} \mathbf{B_2 c}^n &= \left(\mathbf{B_0} - \frac{h^2}{12}\mathbf{B_2}\right)\overline{\mathbf{V}}^n \\ \mathbf{c}^n &= \mathbf{B_2}^{-1}\left(\mathbf{B_0} - \frac{h^2}{12}\mathbf{B_2}\right)\overline{\mathbf{V}}^n. \end{aligned} \tag{3.11}$$

where $\mathbf{c}^n = (c_{-1}^n, c_0^n, \ldots, c_J^n, c_{J+1}^n)^T$ and $\overline{\mathbf{V}}^n$ is the vector obtained using (2.5) . This equation generates a $(J+1) \times (J+3)$ system of equations with variables namely $c_{-1}^n, c_0^n, \ldots, c_J^n, c_{J+1}^n$. After eliminating the variables c_{-1}^n and c_{J+1}^n using the BC's, one can easily solve the system and therefore compute the approximate solution using equation (3.10).

Theorem 3.1. *The method presented in equation* (2.10)*, is unconditionally stable.*

Proof. Let us assume that the eigenvalue of the matrix $\mathbf{B}_0^{-1}\mathbf{B}_2$ is λ. The corresponding eigenvalues of the coefficient matrix $\mathcal{M}$ are given by

$$\begin{aligned} &\begin{pmatrix} \frac{1}{k} - \frac{h^2}{12k}\lambda & -\frac{1}{2}\lambda \\ \frac{1}{2}\lambda & \frac{1}{k} - \frac{h^2}{12k}\lambda \end{pmatrix}^{-1} \begin{pmatrix} \frac{1}{k} - \frac{h^2}{12k}\lambda & \frac{1}{2}\lambda \\ -\frac{1}{2}\lambda & \frac{1}{k} - \frac{h^2}{12k}\lambda \end{pmatrix} \\ &\qquad = \frac{1}{\left(\frac{1}{k} - \frac{h^2}{12k}\lambda\right)^2 + \frac{1}{4}\lambda^2} \begin{pmatrix} \frac{1}{k} - \frac{h^2}{12k}\lambda & \frac{1}{2}\lambda \\ -\frac{1}{2}\lambda & \frac{1}{k} - \frac{h^2}{12k}\lambda \end{pmatrix}^2 \end{aligned}$$

which is a unitary matrix. Therefore, all eigenvalues of $\mathcal{M}$ have modulus 1. Therefore, the suggested method is unconditionally stable. □

4 Approximations at Initial Step and Boundary

Using the BC's (1.2) and equation (2.1) , we have

$$\begin{aligned} v(a,t) &= \beta_3(t), \\ v(b,t) &= \beta_4(t), \\ w(a,t) = u_t(a,t) &= \beta_1{}'(t), \\ w(b,t) = u_t(b,t) &= \beta_2{}'(t), \end{aligned} \tag{4.1}$$

for $t \in [0, T]$. By equations (2.5) and (2.7), the equations (4.1) can be written as

$$\begin{aligned}
&\overline{v}_{-1}(t) + 4\overline{v}_0(t) + \overline{v}_1(t) = \beta_3(t),\\
&\overline{v}_{J-1}(t) + 4\overline{v}_J(t) + \overline{v}_{J+1}(t) = \beta_4(t),\\
&\overline{w}_{-1}(t) + 4\overline{w}_0(t) + \overline{w}_1(t) = {\beta_1}'(t),\\
&\overline{w}_{J-1}(t) + 4\overline{w}_J(t) + \overline{w}_{J+1}(t) = {\beta_2}'(t),
\end{aligned} \tag{4.2}$$

$\forall\, t \in [0, T]$. Since $\overline{\mathbf{p}}_j(t) = [\overline{v}_j(t), \overline{w}_j(t)]^T$. Therefore, we have

$$\begin{aligned}
&\overline{\mathbf{p}}_{-1}(t) + 4\overline{\mathbf{p}}_0(t) + \overline{\mathbf{p}}_1(t) = \begin{pmatrix}\beta_3(t)\\ \beta_1'(t)\end{pmatrix},\\
&\overline{\mathbf{p}}_{J-1}(t) + 4\overline{\mathbf{p}}_J(t) + \overline{\mathbf{p}}_{J+1}(t) = \begin{pmatrix}\beta_4(t)\\ \beta_2'(t)\end{pmatrix},
\end{aligned} \tag{4.3}$$

$\forall\, t \in [0, T]$. Now, using the IC's (1.2) and equation (2.1), we can write

$$\begin{aligned}
&v(x, 0) = u_{xx}(x, 0) = {\alpha_1}''(x),\\
&w(x, 0) = \alpha_2(x),
\end{aligned} \tag{4.4}$$

$\forall\, x \in [a, b]$. Therefore, from the equations (4.4) we obtain,

$$\overline{\mathbf{p}}^0_{j-1} + 4\overline{\mathbf{p}}^0_j + \overline{\mathbf{p}}^0_{j+1} = \begin{pmatrix}\alpha_1''(x_j)\\ \alpha_2(x_j)\end{pmatrix}, \quad \text{for} \quad j = 0, 1, \ldots, J. \tag{4.5}$$

Equations (4.3), enable us to eliminate $\overline{\mathbf{p}}_{-1}$ and $\overline{\mathbf{p}}_{J+1}$ from the system of equations obtained by the equations (4.5). So, using equations (4.3) and (4.5), we get a $(J + 1) \times (J + 1)$ system of equations of the form

$$\mathbf{A}\mathbf{R^0} = \mathbf{E} \tag{4.6}$$

where $\mathbf{A}$ is a block tridiagonal matrix given by

$$\mathbf{A} = \begin{pmatrix}
6\mathbf{I} & 0 & & & & & & \\
\mathbf{I} & 4\mathbf{I} & \mathbf{I} & & & & & \\
0 & \mathbf{I} & 4\mathbf{I} & \mathbf{I} & & & & \\
 & & & \cdots & \cdots & \cdots & & \\
 & & & & \cdots & \cdots & \cdots & \\
 & & & & & \mathbf{I} & 4\mathbf{I} & \mathbf{I} & 0\\
 & & & & & & \mathbf{I} & 4\mathbf{I} & \mathbf{I}\\
 & & & & & & & 0 & 6\mathbf{I}
\end{pmatrix}, \mathbf{R^0} = \begin{pmatrix}
\overline{\mathbf{p}}^0_0\\ \overline{\mathbf{p}}^0_1\\ \overline{\mathbf{p}}^0_2\\ \cdots\\ \overline{\mathbf{p}}^0_{J-2}\\ \overline{\mathbf{p}}^0_{J-1}\\ \overline{\mathbf{p}}^0_J
\end{pmatrix}$$

and $\mathbf{E}$ is the vector obtained with the right hand side values of equations (4.3) and (4.5). We can find the initial vector $\mathbf{R^0}$ by solving equation (4.6).

5 Numerical Experiments

In this section, we compute the approximate solution of some test problems using the proposed method. To validate the order of convergence and accuracy of the proposed method, we will use the following formula

$$\begin{aligned}
&\text{Maximum Absolute Error}\,(L_\infty) = \max_j |u_j - U_j|,\\
&\text{Order} = \frac{\log\left(L_\infty(J_1)/L_\infty(J_2)\right)}{\log\left(J_1/J_2\right)},
\end{aligned}$$

where $L_\infty(J_1)$ and $L_\infty(J_2)$ are the errors corresponding to the number of grid points J_1, J_2 respectively.

Example 1: Let us consider the equation (Mittal and Jain [15]):

$$\frac{\partial^2 u}{\partial t^2} + \frac{\partial^4 u}{\partial x^4} = \left(\pi^4 - 1\right) \sin \pi x \cos t, \quad x \in [0,1],\, t > 0,$$

with IC's

$$u(x,0) = \sin \pi x, u_t(x,0) = 0, \quad x \in [0,1],$$

and the BC's

$$u(0,t) = u(1,t) = u_{xx}(0,t) = u_{xx}(1,t) = 0, \quad t \geq 0.$$

$u(x,t) = \sin \pi x \cos t$ is the exact solution of the PDE.

For our first computation, we fix $k = 0.005$. Now choose $J = 90, J = 180, J = 270$ and compute solution at time levels $t = 0.02, t = 0.05$ and $t = 1.0$. In Table 1, a comparison between our results with the results obtained by Mittal and Jain [15] is displayed. We can see from the table that our method produces a better accurate solution. For our next computation, we take $k = 0.005, h = 0.05$ and compute the solution at time level $t = 0.05$. The comparison between our results and the results obtained by Rashindini and Mohammadi [16] is displayed in Table 2. In[16], the authors used the sextic spline method, which uses a much larger stencil.
we have also displayed the CPU time and calculated the order in the Table 3. Here we can observe that method is of order four. For $h = 0.05, k = 0.01$, numerical solution of the given problem is illustrated in Figure 1.

Table 1: Absolute error of Example 1, for $k = 0.005$.

Method	Time	J	$x=\frac{1}{10}$	$x=\frac{2}{10}$	$x=\frac{3}{10}$	$x=\frac{4}{10}$	$x=\frac{5}{10}$
Present Method	0.02	90	2.1442e-09	4.0785e-09	5.6135e-09	6.5991e-09	6.9387e-09
		180	3.7454e-10	7.1241e-10	9.8055e-10	1.1527e-09	1.2120e-09
		270	2.7986e-10	5.3232e-10	7.3268e-10	8.6131e-10	9.0564e-10
	0.05	90	3.3397e-09	6.3526e-09	8.7436e-09	1.0279e-08	1.0808e-08
		180	1.6863e-09	3.2076e-09	4.4148e-09	5.1900e-09	5.4571e-09
		270	1.5979e-09	3.0393e-09	4.1833e-09	4.9177e-09	5.1708e-09
	1	90	1.8534e-08	3.5253e-08	4.8522e-08	5.741e-08	5.9976e-08
		180	1.9236e-08	3.6589e-08	5.0361e-08	5.9203e-08	6.2250e-08
		270	1.9274e-08	3.6661e-08	5.0459e-08	5.9319e-08	6.2371e-08
[15]	0.02	90	6.1000e-07	1.1500e-06	1.5900e-06	1.8700e-06	1.9600e-06
		180	1.5000e-07	2.9000e-07	3.9000e-07	4.6000e-07	4.9000e-07
		270	7.0000e-08	1.3000e-07	1.7000e-07	2.0000e-07	2.1000e-07
	0.05	90	4.4700e-06	8.4900e-06	1.1700e-05	1.3700e-05	1.4500e-05
		180	1.1000e-06	2.0900e-06	2.8800e-06	3.3800e-06	3.5600e-06
		270	4.8000e-07	9.1000e-07	1.2500e-06	1.4600e-06	1.5400e-06
	1	90	3.5600e-05	6.7700e-05	9.3200e-05	1.1000e-04	1.1500e-04
		180	3.6200e-06	6.8900e-06	9.4800e-06	1.1100e-05	1.1700e-05
		270	2.3000e-06	4.3800e-06	6.0300e-06	7.0900e-06	7.4500e-06

Example 2: Consider the PDE

$$\frac{\partial^2 u}{\partial t^2} + \frac{\partial^4 u}{\partial x^4} = \left(24 - x^2 + 2x^3 - x^4\right) \cos t,\, x \in [0,1],\, t > 0,$$

with IC's

$$u(x,0) = x^2 - 2x^3 + x^4, \quad u_t(x,0) = 0, x \in [0,1],$$

Table 2: Absolute error of Example 1, for $h = 0.05$ at time $t = 0.05$.

x	Present Method	Rashidinia and Mohammadi [16]
1/10	7.2577e-07	2.9100e-06
2/10	1.3800e-06	1.7300e-06
3/10	1.9000e-06	1.6000e-06
4/10	2.2310e-06	2.2300e-06
5/10	2.3483e-06	2.6000e-07

Table 3: CPU time and order table of Example 1 at time $t = 1$ for $k = h^2$.

h	Error	Order	CPU time (in seconds)
1/4	4.9208e-04	-	0.01
1/8	3.5674e-05	3.79	0.02
1/16	2.3425e-06	3.93	0.03
1/32	1.4857e-07	3.98	0.13
1/64	9.3169e-09	4.00	1.11
1/128	5.8252e-10	4.00	10.04

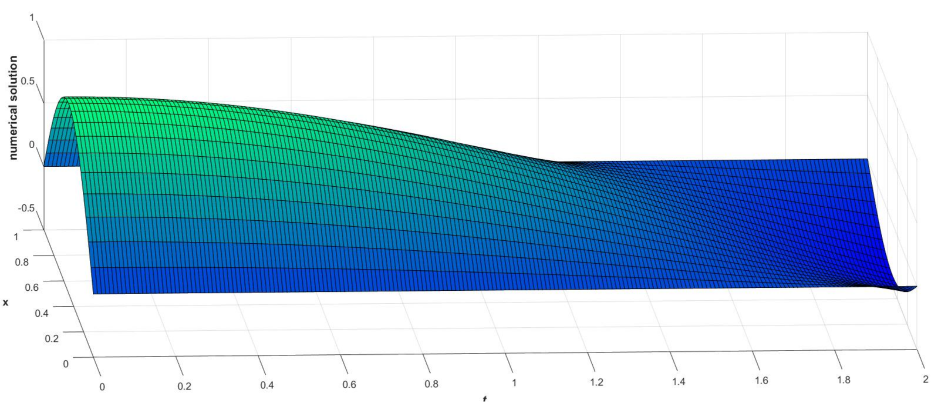

Figure 1: Numerical solution of Example 1 for $h = 0.05, k = 0.01$ at time $t = 2$.

and BC's

$$u(0,t) = 0 = u(1,t), u_{xx}(0,t) = 2\cos t = u_{xx}(1,t), \quad t \geq 0.$$

$u(x,t) = x^2(1-x)^2 \cos t$ is the exact solution of the PDE. For $k = h^2$ and time $t = 5$, we have displayed the CPU time and calculated the order in Table 4, and we can see that our method is of order four. For $h = 0.05, k = 0.01$ and time $t = 1$, the numerical solution is illustrated in Figure 2.

Example 3: Let us consider the PDE

$$\frac{\partial^2 u}{\partial t^2} + \frac{\partial^4 u}{\partial x^4} = 0, x \in [0,1], t > 0,$$

with IC's

$$u(x,0) = \frac{x}{12}(2x^2 - x^3 - 1), u_t(x,0) = 0, x \in [0,1],$$

and BC's

$$u(0,t) = u(1,t) = u_{xx}(0,t) = u_{xx}(1,t) = 0, \quad \forall\, t \geq 0.$$

Table 4: CPU time and order table of Example 2 at time $t = 5$ for $k = h^2$.

h	Error	Order	CPU time (in seconds)
1/4	9.6006e-06	-	0.02
1/8	1.3677e-07	6.13	0.02
1/16	1.1085e-08	3.63	0.08
1/32	7.1685e-10	3.95	0.52
1/64	4.4744e-11	4.00	4.28

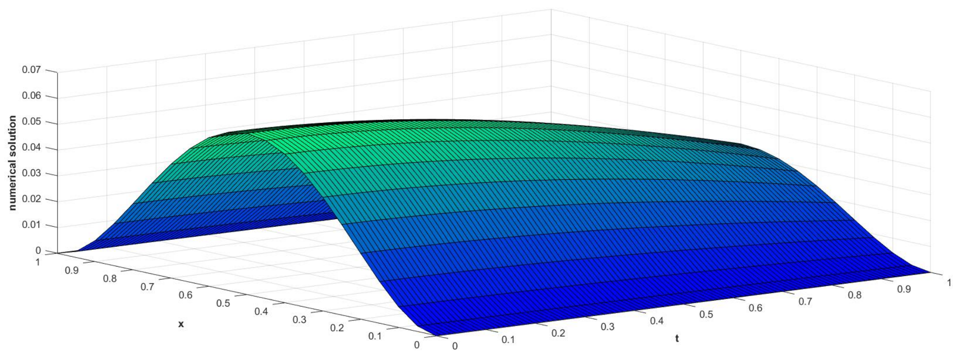

Figure 2: Numerical solution of Example 2 for $h = 0.05, k = 0.01$ at time $t = 1$.

The exact solution of the PDE is given by

$$u(x,t) = -\sum_{r=1}^{\infty} \frac{8}{(2r+1)^5\pi^5} \sin(2r+1)\pi x \cos(2r+1)^2\pi^2 t.$$

For our first computation, we fix $h = 0.05$ and compute the solution at $t = 0.02$ and $t = 1$ with respective time step lengths $k = 0.0025$ and $k = 0.005$. Errors between the approximate and exact solutions are displayed in Table 5. We have also calculated the same results for bending moment $\frac{\partial^2 u}{\partial x^2}$ and illustrated it in Table 6. The results are compared with the errors obtained from the hopscotch procedure by Danaee and Evans [19]. The order of the method is alos displayed in Table 7. We can see that the method is of order 4 in space. For the next computation, we choose $h = 0.1, k = 0.02$. Errors between the exact and approximate values of u and u_{xx} at time $t = 1$ are evaluated and compared with Fairweather and Gourlay [7] in Table 8. In [7], the authors have used semi explicit finite difference method. In Figure 3, a three dimensional plot of the numerical solution is displayed for the values $h = 0.1, k = h^2$ at time $t = 2$.

6 Conclusion

In this article, we have derived a new compact high order method to solve the fourth order PDE governing the behavior of a vibrating beam. The present method is based on the cubic B-spline collocation technique and requires only three grid points in the space direction. We have shown the method to be unconditionally stable and discussed the accuracy of the method by performing some numerical experiments. The results obtained by the present method have been compared with those available in the literature. The advantage of the present method over the existing techniques are its compactness, easy implementation, and highly accurate results with less computational effort.

Table 5: Errors obtained for the solution of Example 3 for $h = 0.05$ at various time levels .

x	Present Method		Danaee and Evans [19]	
	$k = 0.00125, t = 0.02$	$k = 0.005, t = 1$	$k = 0.00125, t = 0.02$	$k = 0.005, t = 1$
1/10	1.4646e-07	1.5011e-05	2.5000e-06	3.1880e-03
2/10	2.3670e-07	6.5572e-06	3.9000e-06	2.7270e-03
3/10	5.2983e-07	1.8211e-05	1.3700e-05	9.8030e-03
4/10	4.0198e-07	3.4812e-05	2.6000e-06	1.2459e-02
5/10	2.4761e-07	3.5029e-05	9.8000e-06	1.4032e-02

Table 6: Error obtained at computing $\frac{\partial^2 u}{\partial x^2}$ in Example 3 for $h = 0.05$ at various time levels.

x	Present Method		Danaee and Evans [19]	
	$k = 0.00125, t = 0.02$	$k = 0.005, t = 1$	$k = 0.00125, t = 0.02$	$k = 0.005, t = 1$
1/10	7.8080e-05	2.7000e-03	4.9570e-04	1.7410e-03
2/10	4.9414e-04	1.7000e-03	4.7700e-05	3.2820e-03
3/10	2.96273-04	1.0000e-03	2.3793e-03	4.5540e-03
4/10	1.6233e-04	2.8000e-03	4.3360e-04	5.3690e-03
5/10	9.0149e-04	1.5081e-04	3.1216e-03	5.6440e-03

Table 7: CPU time and order table of Example 3 at time $t = 1$ for $k = h^2$.

h	Error	Order	CPU time (in seconds)
1/20	3.1310e-05	-	0.11
1/40	2.3094e-06	3.76	0.64
1/80	1.4557e-07	3.99	5.18
1/160	9.1029e-09	3.99	55.50

Table 8: Errors of Example 3 for $k = 0.02, h = 0.1$ at time $t = 1$.

x	Present Method		Fairweather and Gourlay [7]	
	u	u_{xx}	u	u_{xx}
1/10	2.5303e-04	1.6400e-02	1.3760e-03	2.9855e-02
2/10	3.7029-04	1.7100e-02	2.4560e-03	5.3930e-02
3/10	3.3783e-04	9.5000e-03	3.0140e-03	6.0127e-02
4/10	2.3195-04	3.4000e-03	2.9870e-03	4.8259e-02
5/10	1.7319-04	1.2200e-02	2.4840e-03	2.7202e-02

References

[1] S. H. Crandall, Numerical treatment of a fourth order parabolic partial differential equation, *Journal of the ACM (JACM)*, **1(3)**, 111–118 (1954).

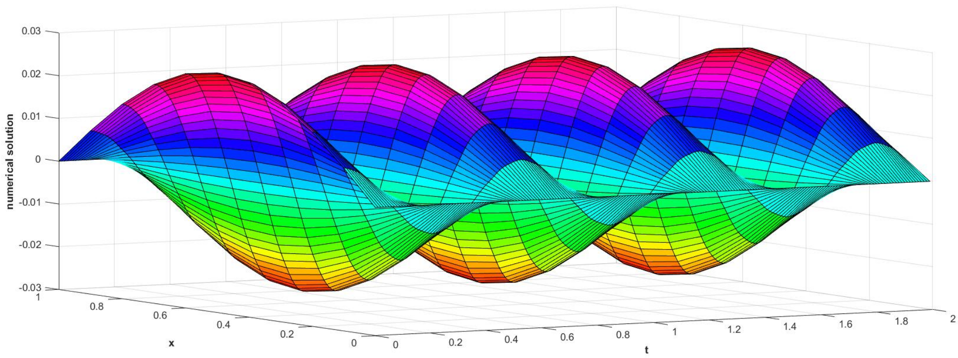

Figure 3: Numerical solution of Example 3 for $h = 0.1, k = h^2$ at time $t = 2$.

[2] Thomas R Lucas, Error bounds for interpolating cubic splines under various end conditions, *SIAM Journal on Numerical Analysis*, **11(3)**, 569–584 , (1974).

[3] M. K. Jain, S. R. K. Iyengar and A. G. Lone, Higher order difference formulas for a fourth order parabolic partial differential equation, *International Journal for Numerical Methods in Engineering*, **10(6)**, 1357–1367 (1976).

[4] M. Dehghan and J. Manafian, The solution of the variable coefficients fourth-order parabolic partial differential equations by the homotopy perturbation method, *Zeitschrift fur Naturforschung A*, **64(7-8)**, 420–430 (2009).

[5] R. K. Mohanty and D. J. Evans, The numerical solution of fourth order mildly quasi-linear parabolic initial boundary value problem of second kind, *International journal of computer mathematics*, **80(9)**, 1147–1159 (2003).

[6] R. K. Mohanty, D. J. Evans and D. Kumar, High accuracy difference formulae for a fourth order quasi-linear parabolic initial boundary value problem of first kind, *International journal of computer mathematics*, **80(3)**, 381–398 (2003).

[7] G. Fairweather and A. R. Gourlay, Some stable difference approximations to a fourth-order parabolic partial differential equation, *Mathematics of Computation*, **21(97)**, 1–11 (1967).

[8] D. J. Evans and W. S. Yousif, A note on solving the fourth order parabolic pde by the AGE method, *International journal of computer mathematics*, **40(1-2)**, 93–97 (1991).

[9] S. D. Conte, A Stable Implicit Finite Difference Approximation to a Fourth Order Parabolic Equation, *J. ACM*, **4**, 18–23 (1957).

[10] D. Kaur and R. K. Mohanty, Highly accurate compact difference scheme for fourth order parabolic pde with Dirichlet and Neumann BC's: Application to good Boussinesq equation, *Applied Mathematics and Computation*, 378, 125202 (2020).

[11] M. Nadeem, F. Li and H. Ahmad, Modified Laplace variational iteration method for solving fourth-order parabolic partial differential equation with variable coefficients, *Computers and Mathematics with Applications*, **78(6)**, 2052–2062 (2019).

[12] H. Caglar and N. Caglar, Fifth-degree B-spline solution for a fourth-order parabolic partial differential equations, *Applied Mathematics and Computation*, **201(1-2)**, 597–603 (2008).

[13] T. Aziz, A. Khan and J. Rashidinia, Spline methods for the solution of fourth-order parabolic partial differential equations, *Applied mathematics and computation*, **167(1)**, 153–166 (2005).

[14] A. M. Wazwaz, Analytic treatment for variable coefficient fourth-order parabolic partial differential equations, *Applied Mathematics and Computation*, **123(2)**, 219–227 (2001).

[15] R. C. Mittal and R. K. Jain, B-splines methods with redefined basis functions for solving fourth order parabolic partial differential equations, *Applied mathematics and computation*, **217(23)**, 9741–9755 (2011).

[16] J. Rashidinia and R. Mohammadi, Sextic spline solution of variable coefficient fourth-order parabolic pdes, *International Journal of Computer Mathematics*, **87(15)**, 3443–3454 (2010).

[17] M. El-Gamel, A note on solving the fourth-order parabolic pde by the sinc-Galerkin method, *Calcolo*, **52(3)**, 327–342 (2015).

[18] Carl de Boor, *A Practical Guide to Splines*, Revised edition, Applied Mathematical Sciences, Springer-Verlag, New York, ISBN: 0-387-95366-3 (2001).

[19] A. Danaee and D. J. Evans, Hopscotch procedures for a fourth-order parabolic partial differential equation differential equation, *Mathematics and Computers in Simulation*, **24(4)**, 326–329 (1982).

[20] T. Sultana and P. Khandelwal, Tension spline technique for the solution of fourth-order parabolic partial differential equation, *Malaya Journal of Matematik (MJM)*, **6(3)**, 514–520 (2018).

[21] A. Khan and T. Sultana, Numerical solution of fourth order parabolic partial differential equation using parametric septic spline, *Hacettepe J. Math. Stat*, **45(4)**, 1067–1082 (2016).

[22] R. K. Mohanty and S. Sharma, A new two-level implicit scheme based on cubic spline approximations for the 1D time-dependent quasilinear biharmonic problems, *Engineering with Computers*, **36(4)**, 1485–1498 (2020).

[23] D. Kaur and R. K. Mohanty, Two-level implicit high order method based on half-step discretization for 1D unsteady biharmonic problems of first kind, *Applied Numerical Mathematics*, **139**, 1–14 (2019).

[24] R. K. Mohanty, S. McKee and D. Kaur, A class of two-level implicit unconditionally stable methods for a fourth order parabolic pde, *Applied Mathematics and Computation*, **309**, 272–280 (2017).

Author information

Sandeep Bhatt, Swarn Singh and Suruchi Singh, Department of Mathematics, Ram Lal Anand College, University of Delhi, Delhi-110021; Department of Mathematics, Sri Venkateswara College, University of Delhi, Delhi-110021 and Department of Mathematics, Aditi Mahavidyalaya, University of Delhi, Delhi-110039, India.
E-mail: bhattsandeep026@gmail.com, ssingh@svc.ac.in and suruchi@aditi.du.ac.in