Cubic B-spline collocation method on a non-uniform mesh for solving nonlinear parabolic partial differential equation

Swarn Singh
Department of Mathematics,
Sri Venkateswara College, University of Delhi, India.
E-mail: ssingh@svc.ac.in

Sandeep Bhatt
Department of Mathematics,
University of Delhi, India.
E-mail: bhattsandeep026@gmail.com

Suruchi Singh
Department of Mathematics,
Aditi Mahavidyalaya, University of Delhi, India.
E-mail: ssuruchi2005@yahoo.co.in

Abstract
In this paper, an approximate solution of nonlinear parabolic partial differential equation is obtained for a non-uniform mesh. The scheme for partial differential equation subject to Neumann boundary conditions is based on cubic B-spline collocation method. Modified cubic B-splines are proposed over non-uniform mesh to deal with the Dirichlet boundary conditions. This scheme produces a system of first order ordinary differential equations. This system is solved by Crank Nicholson method. The stability is also discussed using Von Neumann stability analysis. The accuracy and efficiency of the scheme is shown by numerical experiments. We have compared the approximate solutions with that in the literature.

Keywords. Nonlinear parabolic partial differential equation, Collocation method, Cubic B-spline, Non-uniform mesh, Crank-Nicolson method, Burger’s equation, Fisher equation.

2010 Mathematics Subject Classification. 35K55, 65N12.

1. INTRODUCTION

Consider the nonlinear partial differential equation on a bounded domain \( \mathbb{D} = \{(x,t)|x \in [a,b], t \in [0,T]\} \) defined as:
\[
v_t = F(x,t,v,v_x,v_{xx}), \quad a \leq x \leq b, \quad 0 \leq t \leq T,
\]
with the initial condition
\[
v(x,0) = \psi(x), \quad a \leq x \leq b,
\]
and boundary conditions (Neumann or Dirichlet or mixed).

Received: 25 April 2020 ; Accepted: 06 December 2020.
* corresponding.
The nonlinear parabolic partial differential equations can be seen in the most area of science and engineering. Some famous examples of such PDE’s are Burger’s equation, Fisher’s reaction-diffusion equations, generalized Burger-Fisher equation and etc. These equations have application in areas of gas dynamics, heat conduction, traffic flow, fluid mechanics, population dynamics etc. In recent past, several numerical schemes have been formulated to solve nonlinear parabolic partial differential equations. Jain et al. [12] described a high order finite difference method in order to solve a system of one dimensional nonlinear parabolic partial differential equations with the help of three spatial grid points. By introducing nonlinear transformations, analytical and explicit solitary wave solutions were derived for the generalized Fisher’s equation by Wang [24]. Mavoungou and Cherrouault [16] solved Fisher’s equation using Adomian’s method. Cecchi et al. [5] presented a numerical method to solve a weak formulation of quasilinear parabolic problems on space time-domain governed by Burger’s equation. Dogan [10] solved the Burger’ equation by taking a Galerkin Finite element approach. Fisher’s reaction-diffusion equation was solved by Al-Khaled [1] with the help of Sinc collocation method. Automatic differentiation techniques are given by Asaithambi [4] and Rall [21]. Mohanty et al. [19] proposed a cubic spline alternating group method for one dimensional quasilinear PDE. K Ali et al. [3] presented non-polynomial spline method to solve coupled Burgers equations. Pourgholi and Saeedi [20] proposed a numerical method based on cubic B-splines for solving some nonlinear inverse parabolic partial differential equations with Dirichlet boundary conditions. Lakestani and Dehghan [15] presented a numerical technique based on the finite difference and collocation methods for the solution of generalized Kuramoto-Sivashinsky (GKS) equation. Zadvan and Rashidinia [25] developed a non polynomial cubic spline functions called “TS splin” and collocation method based on this B-spline for the numerical solution of the nonlinear Klein-Gordon equation. Lakestani and Dehghan [14] presented numerical technique for the solution of FokkerPlanck equation. This method uses the cubic Bspline scaling functions. The method consists of expanding the required approximate solution as the elements of cubic Bsplines scaling function. Many authors presented numerical methods based on splines over uniform mesh [2, 11, 12, 3, 19, 23]. Not much work has been done on cubic spline collocation method over the non-uniform mesh. Mittal [17] used cubic b-spline collocation method on uniform mesh to solve parabolic PDE’s. The applicability of the schemes to non-uniform mesh is important for the problems with rough solution behaviors, layers etc and for multidimensional problems in non rectangular domains. We were motivated by the authors [7, 8, 9, 18] and developed a cubic B-spline collocation method for non-uniform mesh.

In this article, we suggest a procedure to obtain the approximate solution of a nonlinear parabolic partial differential equation using cubic B-splines over non-uniform mesh. This procedure is based on collocation method. In case of Dirichlet boundary conditions we modify the cubic B-splines. Crank Nicholson method is used to solve the system of first order ordinary differential equations. This provides us with an efficient explicit solution with minimal computational effort. The benefits of the present scheme is simple computation and low storage cost for the PDE with nonlinear terms.
This article is systematized as follows: In section 2, cubic B-splines over non-uniform mesh are described. In section 3, we have explained the method for different types of boundary conditions. For Dirichlet boundary conditions, we modified the cubic B-splines. In section 4, we have done stability analysis using Von Neumann stability analysis. In section 5, we have presented four numerical examples to show the efficiency and accuracy of the suggested method. Section 6, includes conclusions that briefly summarize the proposed technique.

2. Cubic B-splines over non-uniform mesh

In this section, we will depict cubic B-splines and its derivatives over a non-uniform mesh. Consider a non-uniform mesh in x-direction as \( x_0 = a, x_m = b \) and \( x_l = x_{l-1} + \delta_l ; l = 1, 2, \ldots, m-1 \); mesh ratio \( \sigma_l = \delta_{l+1}/\delta_l \), \( l = 1, 2, \ldots, m-1 \) and uniform mesh \( 0 = t_0 < t_1 < \cdots < t_N = T \) in time direction with time step \( k = t_j - t_{j-1} \) for \( j = 0, 1, \ldots, N \). For \( \sigma_l = 1 ; l = 1, 2, \ldots, m-1 \), the mesh in space direction will be uniform. The cubic B-splines are defined on an increasing set of \( m+1 \) nodes over problem domain plus six additional nodes outside the problem domain. The six additional points are \( x_{-3}, x_{-2}, x_{-1}, x_{m+1}, x_{m+2} \) and \( x_{m+3} \) with \( |x_{-p} - x_{-(p-1)}| = |x_{m+3} - x_{m+2}| \) where \( p = 1, 2, 3, m+1, m+2, m+3 \). Cubic B-splines are defined by iteratively convoluting lower-order B-splines \([8, 9]\). Let \( S_l(x) \) denote the cubic B-spline defined at node \( x_l \).

\[
S_l(x) = \begin{cases} 
\frac{(x - x_{l-2})^3}{(x_l - x_{l-1})(x_l - x_{l-2})(x_{l+1} - x_{l-2})} + \frac{(x_l - x_{l-1})(x_l - x_{l-2})(x_{l+1} - x_{l-2})}{(x_l - x_{l-2})^3} & \text{if } x_{l-2} \leq x < x_{l-1}, \\
\frac{(x_l - x_{l-2})^2}{(x_l - x_{l-1})(x_l - x_{l-2})} + \frac{(x_l - x_{l-2})^2}{(x_l - x_{l-1})^2} & \text{if } x_{l-1} \leq x < x_l, \\
\frac{(x_l - x_{l-1})(x_{l+1} - x_{l-1})(x_{l+2} - x_{l-1})}{(x_{l+1} - x_{l-1})^2(x_{l+1} - x_l)} + \frac{(x_{l+1} - x_{l-1})(x_{l+1} - x_l)(x_{l+2} - x_l)}{(x_{l+1} - x_{l-1})^2} & \text{if } x_l \leq x < x_{l+1}, \\
\frac{(x_{l+1} - x_l)(x_{l+2} - x_l)(x_{l+2} - x_{l+1})}{(x_{l+2} - x_{l-1})^3} & \text{if } x_{l+1} \leq x < x_{l+2}, \\
0 & \text{otherwise.}
\end{cases}
\] (2.1)

for \( l = -1, 0, 1, \ldots, m-1, m, m+1 \).

Only three cubic B-splines can contribute to a particular node \( x_l \) which are \( S_{l-1}, S_l \) and \( S_{l+1} \). The values of \( S_l \) and its derivatives at the node \( x_l \) are given by:
\[ S_{l-1}(x_l) = \frac{\delta_{l+1}^2}{(\delta_{l+1} + \delta_l)(\delta_{l+1} + \delta_l + \delta_{l-1})}, \quad \text{(2.2)} \]

\[ S_l(x_l) = \frac{\delta_{l+1}}{\delta_{l+1} + \delta_l + \delta_{l-1}} \]

\[ + \frac{\delta_l(\delta_{l+2} + \delta_{l+1})}{(\delta_{l+1} + \delta_l)(\delta_{l+2} + \delta_{l+1} + \delta_l)}, \quad \text{(2.3)} \]

\[ S_{l+1}(x_l) = \frac{\delta_l^2}{(\delta_{l+1} + \delta_l)(\delta_{l+2} + \delta_{l+1} + \delta_l)}, \quad \text{(2.4)} \]

\[ S_{l-1}'(x_l) = \frac{-3\delta_{l+1}}{(\delta_{l+1} + \delta_l)(\delta_{l+1} + \delta_l + \delta_{l-1})}, \quad \text{(2.5)} \]

\[ S_l'(x_l) = \frac{3\delta_l}{(\delta_{l+1} + \delta_l)(\delta_{l+2} + \delta_{l+1} + \delta_l)}, \quad \text{(2.6)} \]

\[ S_{l-1}''(x_l) = \frac{6}{\delta_{l+1}(\delta_{l+1} + \delta_l)(\delta_{l+1} + \delta_l + \delta_{l-1})}, \quad \text{(2.7)} \]

\[ S_l''(x_l) = \frac{2(\delta_l + \delta_{l-1} - 2\delta_{l+1})}{\delta_{l+1}(\delta_{l+1} + \delta_l)(\delta_{l+1} + \delta_l + \delta_{l-1})} \]

\[ - \frac{2(\delta_l + 2\delta_{l+1} + \delta_{l+2})}{\delta_{l+1}(\delta_{l+1} + \delta_l)(\delta_{l+2} + \delta_{l+1} + \delta_l)}, \quad \text{(2.8)} \]

\[ S_{l+1}''(x_l) = \frac{6}{(\delta_{l+1} + \delta_l)(\delta_{l+2} + \delta_{l+1} + \delta_l)}, \quad \text{(2.9)} \]

for \(l = -1, 0, 1, \ldots, m - 1, m, m + 1\). The set of functions \(\{S_{-1}, S_0, S_1, \ldots, S_{m-1}, S_m, S_{m+1}\}\) creates a basis for the functions defined on the interval \(a \leq x \leq b\). The approximate solution \(V(x, t)\) for the analytic solution \(v(x, t)\) of the given problem can be written as

\[ V(x, t) = \sum_{l=-1}^{m+1} \gamma_l(t)S_l(x), \quad \text{(2.10)} \]

where, \(\gamma_l(t)\) are unknown quantities depending on time. We will obtained these from the collocation form of the differential equation and available boundary conditions. With the help of cubic B-splines (2.2)-(2.9) and the approximate solution (2.10), the approximate value of the solution \(V(x, t), V'(x, t)\) and \(V''(x, t)\) at node \(x_l\) are given by

\[ V_l = S_{l-1}(x_l)\gamma_{l-1} + S_l(x_l)\gamma_l + S_{l+1}(x_l)\gamma_{l+1}, \quad \text{(2.11)} \]

\[ (V_x)_l = S'_{l-1}(x_l)\gamma_{l-1} + S'_l(x_l)\gamma_l + S'_{l+1}(x_l)\gamma_{l+1}, \quad \text{(2.12)} \]

\[ (V_{xx})_l = S''_{l-1}(x_l)\gamma_{l-1} + S''_l(x_l)\gamma_l + S''_{l+1}(x_l)\gamma_{l+1}. \quad \text{(2.13)} \]

3. Implementation of the Method

3.1. Neumann Boundary Conditions. We first consider the problem (1.1) with Neumann boundary conditions at the end points. So we have the partial differential
equation
\[ v_t = F(x, t, v, v_x, v_{xx}), \quad a \leq x \leq b, \quad 0 \leq t \leq T, \quad (3.1) \]

with the initial condition
\[ v(x, 0) = \psi(x) \quad a \leq x \leq b, \quad (3.2) \]

and boundary conditions
\[ v_x(a, t) = g_1(t), \quad v_x(b, t) = g_2(t), \quad t \geq 0. \quad (3.3) \]

Using equations (2.12) and (3.3), the approximate solution at the boundary points is given by:
\[ V_x(x_0, t) = S'_{-1}(x_0)\gamma_{-1} + S'_0(x_0)\gamma_0 + S'_1(x_0)\gamma_1 = g_1(t), \]
\[ V_x(x_m, t) = S'_{m-1}(x_m)\gamma_{m-1} + S'_m(x_m)\gamma_m + S'_{m+1}(x_m)\gamma_{m+1} = g_2(t). \]

where \( S'_{-1}(x_0) \) and \( S'_{m+1}(x_m) \) can be evaluated from (2.5) and (2.6) respectively. So we have
\[ \gamma_{-1} = \frac{1}{S'_{-1}(x_0)} \left( g_1(t) - S'_0(x_0)\gamma_0 - S'_1(x_0)\gamma_1 \right), \]
\[ \gamma_{m+1} = \frac{1}{S'_{m+1}(x_m)} \left( g_2(t) - S'_{m-1}(x_m)\gamma_{m-1} - S'_m(x_m)\gamma_m \right). \quad (3.4) \]

Using collocation method on PDE (3.1), we get
\[ V_t = F(x, t, V, V_x, V_{xx}), \]
where \( V, V_x, V_{xx} \) are as in (2.11)-(2.13). Also
\[ (V_j) = \sum_{i=1}^{m+1} \gamma_i(t)S_i(x_i) = S_{i-1}(x_i)^\gamma_{i-1} + S_i(x_i)^\gamma_i + S_{i+1}(x_i)^\gamma_{i+1}, \]

where \( \gamma_i = d\gamma_i/dt \). Now, applying the Crank-Nicolson scheme on equation (3.1) we get,
\[ \frac{V^{n+1} - V^n}{\Delta t} = \frac{1}{2} \left( F\left(t^{n+1}, x, V^{n+1}, V_x^{n+1}, V_{xx}^{n+1}\right) + F\left(t^n, x, V^n, V_x^n, V_{xx}^n\right) \right), \]
\[ \Rightarrow \quad 0 = V^{n+1} - \frac{\Delta t}{2} F\left(t^{n+1}, x, V^{n+1}, V_x^{n+1}, V_{xx}^{n+1}\right) - V^n - \frac{\Delta t}{2} F\left(t^n, x, V^n, V_x^n, V_{xx}^n\right). \quad (3.5) \]
Using equations (2.11)-(3.5), we get

\[ 0 = \left( S_{l-1}(x_l)\gamma_{l-1}^{n+1} + S_l(x_l)\gamma_l^{n+1} + S_{l+1}(x_l)\gamma_{l+1}^{n+1} \right) \]

\[ -\frac{\Delta t}{2} F\left( t^{n+1}, x_l, \left( S_{l-1}(x_l)\gamma_{l-1}^{n+1} + S_l(x_l)\gamma_l^{n+1} + S_{l+1}(x_l)\gamma_{l+1}^{n+1} \right), \right. \]

\[ \left. \left( S'_{l-1}(x_l)\gamma_{l-1}^{n+1} + S'_l(x_l)\gamma_l^{n+1} + S'_{l+1}(x_l)\gamma_{l+1}^{n+1} \right), \right. \]

\[ \left. \left( S''_{l-1}(x_l)\gamma_{l-1}^{n+1} + S''_l(x_l)\gamma_l^{n+1} + S''_{l+1}(x_l)\gamma_{l+1}^{n+1} \right) \right) \]

\[-\left( S_{l-1}(x_l)\gamma_{l-1}^{n} + S_l(x_l)\gamma_l^{n} + S_{l+1}(x_l)\gamma_{l+1}^{n} \right) \]

\[-\frac{\Delta t}{2} F\left( t^n, x_l, \left( S_{l-1}(x_l)\gamma_{l-1}^{n} + S_l(x_l)\gamma_l^{n} + S_{l+1}(x_l)\gamma_{l+1}^{n} \right), \right. \]

\[ \left. \left( S'_{l-1}(x_l)\gamma_{l-1}^{n} + S'_l(x_l)\gamma_l^{n} + S'_{l+1}(x_l)\gamma_{l+1}^{n} \right), \right. \]

\[ \left. \left( S''_{l-1}(x_l)\gamma_{l-1}^{n} + S''_l(x_l)\gamma_l^{n} + S''_{l+1}(x_l)\gamma_{l+1}^{n} \right) \right). \]  

(3.6)

In case of linear parabolic PDEs with respect to Neumann boundary conditions, we obtain a system of equations in matrix form as

\[ P\gamma^{n+1} = Q\gamma^n + R, \]  

(3.7)

where P and Q are tridiagonal matrices, \( \gamma^n = [\gamma_0^n, \cdots, \gamma_{m-1}^n, \gamma_m^n]^T \) is the unknown time dependent quantity at time level \( n \) and R is a column vector obtained from the forcing function and boundary conditions. The system (3.7) can be solved using Thomas algorithm. In case of nonlinear parabolic partial differential equation, we will find the value of \( \gamma^{n+1} \) using the Newton-Raphson method. We can write (3.6) as

\[ F^*(\gamma_l^{n+1}) = 0, \]  

(3.8)

where

\[ F^*(\gamma_l^{n+1}) = \left( S_{l-1}(x_l)\gamma_{l-1}^{n+1} + S_l(x_l)\gamma_l^{n+1} + S_{l+1}(x_l)\gamma_{l+1}^{n+1} \right) \]

\[-\frac{\Delta t}{2} F\left( t^{n+1}, x_l, \left( S_{l-1}(x_l)\gamma_{l-1}^{n+1} + S_l(x_l)\gamma_l^{n+1} + S_{l+1}(x_l)\gamma_{l+1}^{n+1} \right), \right. \]

\[ \left. \left( S'_{l-1}(x_l)\gamma_{l-1}^{n+1} + S'_l(x_l)\gamma_l^{n+1} + S'_{l+1}(x_l)\gamma_{l+1}^{n+1} \right), \right. \]

\[ \left. \left( S''_{l-1}(x_l)\gamma_{l-1}^{n+1} + S''_l(x_l)\gamma_l^{n+1} + S''_{l+1}(x_l)\gamma_{l+1}^{n+1} \right) \right) \]

\[-V_l^n = \frac{\Delta t}{2} F\left( t^n, x_l, V_l^n, (V_x)_l^n, (V_{xx})_l^n \right), \]  

(3.9)
By Newton-Raphson method, $\gamma^{n+1}$ can be calculated from

$$
(\gamma_{i}^{n+1})^{k+1} = (\gamma_{i}^{n+1})^{k+1} - \frac{F'(\gamma_{i}^{n+1})^{k}}{F''(\gamma_{i}^{n+1})^{k}}, \quad \text{for } k = 1, 2, 3, \ldots
$$

where $l = 0, 1, 2, \ldots, m - 1, m$. After finding the parameter $\gamma$ at a specified time level, we can find the solution at the required grid points.

3.1.1. Initial Vector $\gamma^0$. The initial vector $\gamma^0$ can be found from the boundary conditions (3.3) and initial condition (3.2) as:

$$
V_x(x_0, 0) = g_1(0),
V(x_l, 0) = \psi(x_l), \quad \text{for } l = 0, 1, 2, \ldots,
V_x(x_m, 0) = g_2(0).
$$

Using equations (2.11), (2.12) and (3.4), we get an $(m+1) \times (m+1)$ system of equations in matrix form as:

$$
P\gamma^0 = Q,
$$

where, $P$ is a tri-diagonal matrix

$$
P = \begin{pmatrix}
S_0'(x_0) & S_0'(x_0) & 0 & \cdots & 0 \\
S_0(x_1) & S_1(x_1) & S_2(x_1) & \cdots & 0 \\
\cdots & \cdots & \cdots & \cdots & \cdots \\
S_{m-2}(x_{m-1}) & S_{m-1}(x_{m-1}) & S_m(x_{m-1}) & S^*_m & 0 \\
S_{m-1}(x_m) & S_m(x_m) & \cdots & \cdots & \cdots
\end{pmatrix},
$$

$$
\gamma^0 = \begin{pmatrix}
\gamma_0^0 \\
\gamma_1^0 \\
\gamma_2^0 \\
\vdots \\
\gamma_{m-2}^0 \\
\gamma_{m-1}^0 \\
\gamma_m^0
\end{pmatrix},
Q = \begin{pmatrix}
\psi(x_0) + \frac{S_{-1}(x_0)}{S_{-1}'(x_0)} g_1(0) \\
\psi(x_1) \\
\psi(x_2) \\
\vdots \\
\psi(x_{m-2}) \\
\psi(x_{m-1}) \\
\psi(x_m) - \frac{S_{m+1}(x_m)}{S_{m+1}'(x_m)} g_2(0)
\end{pmatrix}.
$$
Now, the initial vector $\gamma^0$ can be found from equation (3.12) using Thomas algorithm.

### 3.2. Modified cubic B-splines for Dirichlet boundary conditions.

Now we consider the PDE subject to Dirichlet boundary conditions

$$v_t = F(x, t, v, v_x, v_{xx}), \quad a \leq x \leq b, \quad 0 \leq t \leq T,$$

(3.13)

with the initial condition

$$v(x, 0) = \psi(x), \quad a \leq x \leq b,$$

(3.14)

and Dirichlet boundary conditions

$$v(a, t) = g_1(t), \quad v(b, t) = g_2(t), \quad t \geq 0.$$

(3.15)

In collocation method, when dealing with Dirichlet boundary conditions, the basis functions should vanish on the boundary. However, in the set of cubic B-splines, the basis functions $S_1, S_0, S_1, \ldots, S_m, S_{m+1}$ are not vanishing at the boundary points. So, there is a necessity of redefining the basis functions into a new set of basis functions. We modify cubic B-splines as:

$$\tilde{S}_0(x) = S_0(x) + 2S_{-1}(x),$$

$$\tilde{S}_1(x) = S_1(x) - \frac{S_1(x_0)}{S_{-1}(x_0)} S_{-1}(x),$$

$$\tilde{S}_i(x) = S_i(x) \quad \text{for} \quad i = 2, 3, \ldots, m - 3, m - 2,$$

(3.16)

$$\tilde{S}_{m-1}(x) = S_{m-1}(x) - \frac{S_{m-1}(x_m)}{S_{m+1}(x_m)} S_{m+1}(x),$$

$$\tilde{S}_m(x) = S_m(x) + 2S_{m+1}(x).$$

So $\{\tilde{S}_0, \tilde{S}_1, \ldots, \tilde{S}_{m-1}, \tilde{S}_m\}$ is a modified set of basis functions. Now the approximate solution $V(x, t)$ is given as:

$$V(x, t) = \sum_{l=0}^{m} \gamma_l(t) \tilde{S}_l(x).$$

(3.17)

Here the value of $V$ at node $x_l$ depends upon $\tilde{S}_{l-1}, \tilde{S}_l$ and $\tilde{S}_{l+1}$ only. The approximate value of the solution $V(x, t), V'(x, t)$ and $V''(x, t)$ at node $x_l$ are given by:

$$V_l = \tilde{S}_{l-1}(x_l)\gamma_{l-1} + \tilde{S}_l(x_l)\gamma_l + \tilde{S}_{l+1}(x_l)\gamma_{l+1},$$

(3.18)

$$\langle V_x \rangle_l = \tilde{S}'_{l-1}(x_l)\gamma_{l-1} + \tilde{S}'_l(x_l)\gamma_l + \tilde{S}'_{l+1}(x_l)\gamma_{l+1},$$

(3.19)

$$\langle V_{xx} \rangle_l = \tilde{S}''_{l-1}(x_l)\gamma_{l-1} + \tilde{S}''_l(x_l)\gamma_l + \tilde{S}''_{l+1}(x_l)\gamma_{l+1}.$$ 

(3.20)

Now, using the collocation method in space direction and Crank-Nicolson in time direction on PDE (3.13), we can find the value of $\gamma^{n+1}$ in the same way as in section 3.1.
3.2.1. The Initial vector $\gamma^0$. The initial vector $\gamma^0$ can be obtained from the initial condition (3.14) and boundary condition (3.15) as:

\[
\begin{align*}
V(a, 0) &= V(x_0, 0) = \delta_1(0), \\
V(x_l, 0) &= \psi(x_l), \quad \text{for } l = 1, 2, \cdots, m - 1, \\
V(b, 0) &= V(x_m, 0) = \delta_2(0).
\end{align*}
\] (3.21)

Using (3.16) and (3.18), we get a $(m + 1) \times (m + 1)$ system of equation of the form

\[
P\gamma^0 = Q,
\] (3.22)

where $P$ is the tridiagonal matrix given by

\[
P = \begin{pmatrix}
\tilde{S}_0(x_0) & 0 & & \\
\tilde{S}_0(x_1) & \tilde{S}_1(x_1) & \cdots & \\
& \ddots & \ddots & \ddots \\
& & \tilde{S}_{m-2}(x_{m-1}) & \tilde{S}_{m-1}(x_{m-1}) & \tilde{S}_m(x_{m-1}) \\
\end{pmatrix},
\]

\[
\gamma^0 = \begin{pmatrix}
\gamma_0 \\
\gamma_1 \\
\gamma_2 \\
\vdots \\
\gamma_{m-2} \\
\gamma_{m-1} \\
\gamma_m
\end{pmatrix}, \quad Q = \begin{pmatrix}
g_1'(0) \\
\psi(x_1) \\
\psi(x_2) \\
\vdots \\
\psi(x_{m-2}) \\
\psi(x_{m-1}) \\
g_2'(0)
\end{pmatrix}.
\]

Now, the initial vector $\gamma^0$ can be found from (3.22) using Thomas algorithm.

3.3. Mixed Boundary Conditions. In case of mixed boundary, we have the PDE

\[
v_t = F(x, t, v, v_x, v_{xx}), \quad a \leq x \leq b, \quad 0 \leq t \leq T,
\] (3.23)

with the initial condition

\[
v(x, 0) = \psi(x), \quad a \leq x \leq b,
\] (3.24)

and boundary conditions

\[
v(a, t) = g(t), \quad v_x(b, t) = g^*(t), \quad t \geq 0.
\] or

\[
v_x(a, t) = g^*(t), \quad v(b, t) = h(t), \quad t \geq 0.
\] (3.25)

To solve this type of equation, at Neumann boundary point we proceed like in section 3.1 and at Dirichlet boundary point we modify the Cubic B-splines as done in section 3.2. Rest of the procedure is same.
4. Stability Analysis

We will prove the stability of the method for following linear 1-D problem:

\[ v_t = \lambda v_{xx} - \varepsilon v, \]  

(4.1)

where \( \varepsilon > 0 \) and \( \lambda > 0 \). In order to make things easier, we consider mesh ratio \( \sigma_l = \sigma \) (a constant), for \( l = 1, 2, \ldots, m - 1 \). Using collocation method in space direction and Crank-Nicolson in time direction on PDE (4.1), we get

\[
\frac{V^{n+1} - V^n}{\Delta t} = \frac{\lambda V^{n+1}_{xx} + V^n_{xx}}{2} - \frac{\varepsilon}{2} V^{n+1} + V^n,
\]

(4.2)

\[
\Rightarrow \left( 1 + \frac{\varepsilon \Delta t}{2} \right) V^{n+1} - \frac{\lambda \Delta t}{2} V^{n+1}_{xx} = \left( 1 - \frac{\varepsilon \Delta t}{2} \right) V^n + \frac{\lambda \Delta t}{2} V^n_{xx}.
\]

Substituting values from equation (2.10), we get

\[
\sum_{l=-1}^{m+1} \left( \gamma^n_{l+1}(t) S_l(x) + \frac{\varepsilon \Delta t}{2} \gamma^n_{l+1}(t) S_l(x) - \frac{\lambda \Delta t}{2} \gamma^n_{l+1}(t) S''_l(x) \right) = \sum_{l=-1}^{m+1} \left( \gamma^n_{l}(t) S_l(x) - \frac{\varepsilon \Delta t}{2} \gamma^n_{l}(t) S_l(x) + \frac{\lambda \Delta t}{2} \gamma^n_{l}(t) S''_l(x) \right).
\]

(4.3)

Using the properties of B-splines, we get:

\[
p^n_{l-1} + q^n_{l-1} + r^n_{l+1} = x^n_{l-1} + y^n_{l} + z^n_{l+1},
\]

(4.4)

where

\[
p = S_{l-1}(x_l) + \frac{\varepsilon \Delta t}{2} S_{l-1}(x_l) - \frac{\lambda \Delta t}{2} S''_{l-1}(x_l),
\]

\[
q = S_l(x_l) + \frac{\varepsilon \Delta t}{2} S_l(x_l) - \frac{\lambda \Delta t}{2} S''_l(x_l),
\]

\[
r = S_{l+1}(x_l) + \frac{\varepsilon \Delta t}{2} S_{l+1}(x_l) - \frac{\lambda \Delta t}{2} B''_{l+1}(x_l),
\]

\[
x = S_{l-1}(x_l) - \frac{\varepsilon \Delta t}{2} S_{l-1}(x_l) + \frac{\lambda \Delta t}{2} S''_{l-1}(x_l),
\]

\[
y = S_l(x_l) - \frac{\varepsilon \Delta t}{2} S_l(x_l) + \frac{\lambda \Delta t}{2} S''_l(x_l),
\]

\[
z = S_{l+1}(x_l) - \frac{\varepsilon \Delta t}{2} S_{l+1}(x_l) + \frac{\lambda \Delta t}{2} S''_{l+1}(x_l).
\]

(4.5)

Using Von-Neumann stability analysis we will investigate the stability of the method. Let

\[
\gamma^n_l = \xi^n e^{i \beta x_l},
\]

(4.6)

where \( i = \sqrt{-1} \) and \( \delta_l \) is the step length. Substituting (4.6) in (4.4) and simplifying, we get

\[
\xi = \frac{(x \cos \beta \delta_l + y + z \cos \beta \delta_{l+1}) + i(z \sin \beta \delta_{l+1} - x \sin \beta \delta_l)}{(p \cos \beta \delta_l + q + r \cos \beta \delta_{l+1}) + i(r \sin \beta \delta_{l+1} - p \sin \beta \delta_l)}.
\]

(4.7)
For stability we need
\[
|\xi| \leq 1
\]
\[
\Leftrightarrow |(x \cos \beta \delta_l + y + z \cos \beta \delta_{l+1} + i(z \sin \beta \delta_{l+1} - x \sin \beta \delta_l)|
\leq |(p \cos \beta \delta_l + q + r \cos \beta \delta_{l+1}) + i(r \sin \beta \delta_{l+1} - p \sin \beta \delta_l)|
\]
\[
\Leftrightarrow (x \cos \beta \delta_l + y + z \cos \beta \delta_{l+1})^2 + (z \sin \beta \delta_{l+1} - x \sin \beta \delta_l)^2
\leq (p \cos \beta \delta_l + q + r \cos \beta \delta_{l+1})^2 + (r \sin \beta \delta_{l+1} - p \sin \beta \delta_l)^2
\]
\[
\Leftrightarrow (\sigma^3 + \sigma)(\cos \beta (\delta_l + \delta_{l+1}) - 1) + (\cos \beta (\delta_l - 1)(\sigma^3 - \sigma^4 + 2\sigma^2) + (\cos \beta (\delta_l + \delta_{l+1}) - 1)(2\sigma^2 + \sigma - 1) - \frac{\varepsilon \delta_l^2}{\gamma} \left( (\sigma^3 \cos \beta \delta_l + 2\sigma^2 + 2\sigma + \cos \beta \delta_{l+1})^2 + (\sin \beta \delta_{l+1} - \sigma^3 \sin \beta \delta_l)^2 \right) \leq 0.
\]
which is true for \( \frac{1}{2} \leq \sigma \leq 2 \). Therefore, method is stable for \( \frac{1}{2} \leq \sigma \leq 2 \).

5. NUMERICAL ILLUSTRATION

We examine a few test problems in this section using the discussed method to check the efficiency and accuracy. We divide the interval \([a, b]\) into \(m + 1\) points with \(x_0 = a, x_m = b\) and \(x_l = x_{l-1} + \delta_l; l = 1, 2, \ldots, m + 1\); mesh ratio \(\sigma_l = \delta_{l+1}/\delta_l\), \(l = 1, 2, \ldots, m - 1\).

We can write
\[
b - a = x_m - x_0
\]
\[
= (x_m - x_{m-1}) + (x_{m-1} - x_{m-2}) + \cdots + (x_2 - x_1) + (x_1 - x_0)
\]
\[
= \delta_m + \delta_{m-1} + \cdots + \delta_2 + \delta_1
\]
\[
= (\sigma_m - 1)\sigma_{m-2} \cdots \sigma_2\sigma_1 + \cdots + \sigma_1 + 1)\delta_1.
\]
In our numerical experiments, we consider \(\sigma_i = \sigma(\text{constant})\). So from (5.1) we have
\[
\delta_1 = \frac{b - a}{1 + \sigma + \sigma^2 + \cdots + \sigma^{m-1}} = \frac{b - a}{1 - \sigma^m}.
\]
So, with total \(m + 1\) grid points, we can evaluate value of \(\delta_1\) using (5.2). \(\delta_1\) is the spacing between left boundary point and first grid point. The spacing between the remaining grid points are determined by \(\delta_{l+1} = \sigma \delta_l, l = 1, 2, \ldots, m - 1\). Exact solution of each problem is given. We will determine the efficiency and accuracy by measuring the \(L_\infty\) and \(L_2\) norms of the difference scheme with respect to the approximate solution and analytic solutions of the PDE:
\[
L_\infty = \|v - V\|_\infty = \max_l |v_l - V_l|,
\]
\[
L_2 = \|v - V\|_2 = \sqrt{\sum_{l=0}^{m} |v_l - V_l|^2}.
\]
Example 5.1. Consider a nonlinear reaction-diffusion equation (Cherniha [6]):
\[
v_t = \left[ (1 - v)v_x \right]_x - 2v^2 + 2v, \quad \frac{-\pi}{2} \leq x \leq \frac{\pi}{2}, \quad 0 \leq t \leq T,
\]
with initial condition
\[
v(x, 0) = \frac{2 - \sin x}{3}, \quad \frac{-\pi}{2} \leq x \leq \frac{\pi}{2},
\]
and Neumann boundary conditions
\[
(v_x)(-\frac{\pi}{2}, t) = 0 \quad \text{and} \quad (v_x)(\frac{\pi}{2}, t) = 0, \quad t \geq 0.
\]
The analytic solution is
\[
v(x, t) = \frac{2 - (1 - \tanh t) \sin x + \tanh t}{3}.
\]
In our computation we take \(M = 21, \Delta t = 0.001\). We have compared the approximate solution and the analytic solution for \(\sigma = 0.76\) and \(\sigma = 1.2\) and displayed in Table I. \(L_\infty\) and \(L_2\) errors are also shown in the table at time \(T = 2\). We have demonstrated the behavior of error in figure 1 with respect to time. The graphical depiction of the analytic solution and the approximate solution for \(\sigma = 1.2\) is displayed in figure 2.

In figure 3, we have shown the physical behaviour of the approximate solution for different values of \(\sigma\). In table II, \(L_\infty\) and \(L_2\) errors has been tabulated at various time levels for \(\sigma = 0.76\) and \(\sigma = 1.2\).

Example 5.2. Consider the Burger’s equation (Asaithambi [4]):
\[
v_t = \nu v_{xx} - v v_x, \quad 0 \leq x \leq 1, \quad t \geq 0,
\]
with initial condition
\[
v(x, 0) = \frac{2\pi \nu \sin \pi x}{\cos \pi x + \gamma}, \quad 0 \leq x \leq 1,
\]
and Dirichlet boundary conditions
\[
v(0, t) = 0 \quad \text{and} \quad v(1, t) = 0, \quad t \geq 0.
\]
The analytic solution is
\[
v(x, t) = \frac{2\pi \nu e^{-\nu \pi^2 t} \sin \pi x}{e^{-\nu \pi^2 t} \cos \pi x + \gamma}.
\]
In our computation we take \(\gamma = 2, \nu = 0.1, M = 21\) and \(\Delta t = 0.001\). The comparison between the approximate solution and the analytic solution for \(\sigma = 0.9\) is given in Table III. We have computed \(L_\infty\) and \(L_2\) errors and displayed them in the table at time \(T = 1.5\). \(L_\infty\) and \(L_2\) errors has been evaluated at different time levels and displayed in Table IV. We have illustrated the behavior of maximum absolute error as time progress in Figure 4 for different value of \(\sigma\). The approximate solution is presented graphically for \(\sigma = 0.9\) and \(\sigma = 1.14\) is shown in Figure 5. Exact and approximate solution is illustrated in Figure 6.
Example 5.3. Consider the Burger’s equation (Raslan [22]):

\[ v_t = \nu v_{xx} - \gamma vv_x, \quad 0 \leq x \leq 1, \quad 0 \leq t \leq T, \]

with initial condition is

\[ v(x, 0) = \nu \left[ \tan \frac{x}{2} + x \right], \quad 0 \leq x \leq 1, \]

and mixed boundary conditions are

\[ v(0, t) = 0, \quad (v_x(1, t)) = \frac{\nu}{1 + \nu t} \left[ \frac{1}{2 + 2\nu t} \sec \left[ \frac{1}{2 + 2\nu t} \right] + 1 \right], \quad t \geq 0. \]

The analytic solution is given by

\[ v(x, t) = \frac{\nu}{1 + \nu t} \left[ \tan \left[ \frac{x}{2 + 2\nu t} \right] + x \right]. \]

In our computation we take \( M = 21, \Delta t = 0.001, \nu = 2 \). Maximum absolute error for different values of \( \sigma \) is given in Table V at different time levels. Physical behavior of solution is illustrated in Figure 7.

Example 5.4. We consider the following convection-diffusion equation

\[ v_t = \gamma v_{xx} - \varepsilon v_x, \quad 0 \leq x \leq 1, \quad 0 \leq t \leq T, \]

with initial condition

\[ v(x, 0) = e^{\nu x}, \quad 0 \leq x \leq 1, \]

with Neumann boundary conditions

\[ (v_x(0, t)) = \nu e^{\eta t}, \quad (v_x(1, t)) = \nu e^{\nu x + \eta t}, \quad t \geq 0. \]

The analytic solution is given by

\[ v(x, t) = e^{\nu x + \eta t}. \]

In our first computation, first we take \( \varepsilon = 0.1, \gamma = 0.02, \eta = -0.09, \sigma = 0.8, \nu = 1.17712434464770. \) The maximum absolute error is shown in Table VI for different time levels. For \( M = 11 \) and time step \( \Delta t = 0.01 \), the errors are compared with the same obtained by Mittal and Jain [17]. In our second computation, we take \( \varepsilon = 3.5, \gamma = 0.022, \eta = -0.0999, \sigma = 1.24, \nu = 0.02854797991928. \) The maximum absolute error is shown in Table VII for different time levels. For \( M = 11 \) and time step \( \Delta t = 0.01 \), the errors are compared with the same obtained by Mittal and Jain [17]. Numerical solutions has been illustrated in Figure 8 for \( \sigma = 0.88 \) and \( \sigma = 1.18. \)
Table I. Comparison of the analytic and approximate solution for Example 5.1 at $T = 2$ for $\Delta t = 0.001$, $M = 21$.

<table>
<thead>
<tr>
<th>$\sigma = 0.76$</th>
<th>$x$</th>
<th>Present Method</th>
<th>Exact</th>
<th>$\sigma = 1.2$</th>
<th>$x$</th>
<th>Present Method</th>
<th>Exact</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1.5708</td>
<td>0.9999999</td>
<td>1.0000000</td>
<td></td>
<td>-1.5708</td>
<td>0.9999999</td>
<td>1.0000000</td>
<td></td>
</tr>
<tr>
<td>-0.2383</td>
<td>0.9908248</td>
<td>0.9908394</td>
<td></td>
<td>-1.5338</td>
<td>0.9999919</td>
<td>0.9999919</td>
<td></td>
</tr>
<tr>
<td>0.5314</td>
<td>0.9818371</td>
<td>0.9819332</td>
<td></td>
<td>-1.4805</td>
<td>0.9999511</td>
<td>0.9999511</td>
<td></td>
</tr>
<tr>
<td>0.9759</td>
<td>0.9779057</td>
<td>0.9780781</td>
<td></td>
<td>-1.4037</td>
<td>0.9998330</td>
<td>0.9998330</td>
<td></td>
</tr>
<tr>
<td>1.2327</td>
<td>0.9764903</td>
<td>0.9766972</td>
<td></td>
<td>-1.2931</td>
<td>0.9995407</td>
<td>0.9995408</td>
<td></td>
</tr>
<tr>
<td>1.3810</td>
<td>0.9760143</td>
<td>0.9762336</td>
<td></td>
<td>-1.1340</td>
<td>0.9988736</td>
<td>0.9988740</td>
<td></td>
</tr>
<tr>
<td>1.4667</td>
<td>0.9758598</td>
<td>0.9760833</td>
<td></td>
<td>-0.9047</td>
<td>0.9974364</td>
<td>0.9974371</td>
<td></td>
</tr>
<tr>
<td>1.5162</td>
<td>0.9758115</td>
<td>0.9760363</td>
<td></td>
<td>-0.5746</td>
<td>0.9945286</td>
<td>0.9945266</td>
<td></td>
</tr>
<tr>
<td>1.5448</td>
<td>0.9757973</td>
<td>0.9760225</td>
<td></td>
<td>-0.0993</td>
<td>0.9892184</td>
<td>0.9891981</td>
<td></td>
</tr>
<tr>
<td>1.5613</td>
<td>0.9757937</td>
<td>0.9760189</td>
<td></td>
<td>0.5852</td>
<td>0.9814672</td>
<td>0.9813803</td>
<td></td>
</tr>
<tr>
<td>1.5708</td>
<td>0.9757931</td>
<td>0.9760184</td>
<td></td>
<td>1.5708</td>
<td>0.9761683</td>
<td>0.9760184</td>
<td></td>
</tr>
</tbody>
</table>

$L_1$ Error: 2.2526242e-04, $L_2$ Error: 8.6018072e-04

Table II. Error norms of Example 5.1 at different time levels for $M = 21$, $\Delta t = 0.001$.

<table>
<thead>
<tr>
<th>$t$</th>
<th>$\sigma = 0.76$</th>
<th>$L_\infty$ Error</th>
<th>$L_2$ Error</th>
<th>$\sigma = 1.2$</th>
<th>$L_\infty$ Error</th>
<th>$L_2$ Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>4.1076e-03</td>
<td>1.1908e-02</td>
<td>6.674e-04</td>
<td>4.7682e-04</td>
<td>1.9278e-03</td>
<td>6.0761e-04</td>
</tr>
<tr>
<td>0.4</td>
<td>4.2711e-03</td>
<td>1.4337e-02</td>
<td>1.1249e-03</td>
<td>1.5221e-03</td>
<td>1.0870e-03</td>
<td>1.5184e-03</td>
</tr>
<tr>
<td>0.7</td>
<td>2.5044e-03</td>
<td>9.1215e-03</td>
<td>8.0494e-04</td>
<td>1.1444e-03</td>
<td>1.0729e-03</td>
<td>1.1584e-03</td>
</tr>
<tr>
<td>1.0</td>
<td>1.4628e-03</td>
<td>5.4966e-03</td>
<td>8.0494e-04</td>
<td>1.1444e-03</td>
<td>1.0729e-03</td>
<td>1.1584e-03</td>
</tr>
<tr>
<td>3</td>
<td>3.1142e-04</td>
<td>1.1915e-04</td>
<td>2.1196e-05</td>
<td>3.0794e-05</td>
<td>1.7729e-05</td>
<td>2.5755e-05</td>
</tr>
<tr>
<td>4</td>
<td>4.2269e-05</td>
<td>1.6176e-05</td>
<td>2.8861e-06</td>
<td>4.1944e-06</td>
<td>2.0411e-06</td>
<td>3.011e-06</td>
</tr>
<tr>
<td>5</td>
<td>5.7223e-06</td>
<td>2.1900e-06</td>
<td>3.9094e-07</td>
<td>5.6820e-07</td>
<td>2.5755e-07</td>
<td>3.5682e-07</td>
</tr>
<tr>
<td>7</td>
<td>1.0480e-08</td>
<td>4.0107e-08</td>
<td>7.1627e-09</td>
<td>1.0411e-08</td>
<td>4.1944e-09</td>
<td>5.5682e-09</td>
</tr>
</tbody>
</table>
### Table III. Comparison for Approximate and Exact Solution of Example 5.2 for $\sigma = 0.9, \Delta t = 0.001, T = 1.5, M = 21, \nu = 0.1.$

<table>
<thead>
<tr>
<th>$x$</th>
<th>Approximate</th>
<th>Exact</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1138</td>
<td>0.0226561</td>
<td>0.0226137</td>
</tr>
<tr>
<td>0.3085</td>
<td>0.0554362</td>
<td>0.0553681</td>
</tr>
<tr>
<td>0.4662</td>
<td>0.0702687</td>
<td>0.0702329</td>
</tr>
<tr>
<td>0.5939</td>
<td>0.0707326</td>
<td>0.0707345</td>
</tr>
<tr>
<td>0.6974</td>
<td>0.0622736</td>
<td>0.0622952</td>
</tr>
<tr>
<td>0.7812</td>
<td>0.0497116</td>
<td>0.0497370</td>
</tr>
<tr>
<td>0.8490</td>
<td>0.0363003</td>
<td>0.0363214</td>
</tr>
<tr>
<td>0.9040</td>
<td>0.0238028</td>
<td>0.0238174</td>
</tr>
<tr>
<td>0.9485</td>
<td>0.0129511</td>
<td>0.0129591</td>
</tr>
<tr>
<td>0.9846</td>
<td>0.0038923</td>
<td>0.0038947</td>
</tr>
</tbody>
</table>

$L_\infty$ Error: 6.8098989e-05 
$L_2$ Error: 1.3805602e-04

### Table IV. Error computation for Example 5.2 at different time levels for $\sigma = 1.14, M = 21, \Delta t = 0.001.$

<table>
<thead>
<tr>
<th>$t$</th>
<th>$L_\infty$ Error</th>
<th>$L_2$ Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>3.443e-03</td>
<td>3.977e-03</td>
</tr>
<tr>
<td>0.5</td>
<td>3.027e-03</td>
<td>4.792e-03</td>
</tr>
<tr>
<td>1.0</td>
<td>1.932e-03</td>
<td>3.692e-03</td>
</tr>
<tr>
<td>1.5</td>
<td>1.259e-03</td>
<td>2.862e-03</td>
</tr>
<tr>
<td>2.0</td>
<td>8.549e-04</td>
<td>2.140e-03</td>
</tr>
<tr>
<td>4.0</td>
<td>1.809e-04</td>
<td>4.887e-04</td>
</tr>
<tr>
<td>6.0</td>
<td>3.345e-05</td>
<td>9.122e-05</td>
</tr>
<tr>
<td>8.0</td>
<td>5.789e-06</td>
<td>1.580e-05</td>
</tr>
<tr>
<td>10</td>
<td>9.608e-07</td>
<td>2.024e-06</td>
</tr>
<tr>
<td>12</td>
<td>1.549e-07</td>
<td>4.235e-07</td>
</tr>
<tr>
<td>14</td>
<td>2.489e-08</td>
<td>6.693e-08</td>
</tr>
<tr>
<td>16</td>
<td>3.808e-09</td>
<td>1.041e-08</td>
</tr>
</tbody>
</table>

### Table V. Maximum absolute error of Example 5.3 for different values of $\sigma.$

<table>
<thead>
<tr>
<th>$t$</th>
<th>$0.87$</th>
<th>$1$</th>
<th>$1.12$</th>
<th>$1.25$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>4.640e-03</td>
<td>1.848e-03</td>
<td>4.425e-03</td>
<td>1.024e-02</td>
</tr>
<tr>
<td>0.5</td>
<td>4.068e-03</td>
<td>9.749e-04</td>
<td>2.334e-03</td>
<td>5.433e-03</td>
</tr>
<tr>
<td>1</td>
<td>2.758e-03</td>
<td>4.136e-04</td>
<td>9.768e-04</td>
<td>2.251e-03</td>
</tr>
<tr>
<td>2</td>
<td>1.618e-03</td>
<td>1.344e-04</td>
<td>3.154e-04</td>
<td>7.184e-04</td>
</tr>
<tr>
<td>2.5</td>
<td>1.338e-03</td>
<td>9.046e-05</td>
<td>2.124e-04</td>
<td>4.83e-04</td>
</tr>
<tr>
<td>3</td>
<td>1.138e-03</td>
<td>6.505e-05</td>
<td>1.520e-04</td>
<td>3.474e-04</td>
</tr>
<tr>
<td>5</td>
<td>7.145e-04</td>
<td>2.509e-05</td>
<td>5.881e-05</td>
<td>1.344e-04</td>
</tr>
<tr>
<td>6</td>
<td>6.030e-04</td>
<td>1.781e-05</td>
<td>4.149e-05</td>
<td>9.464e-05</td>
</tr>
<tr>
<td>7</td>
<td>5.213e-04</td>
<td>1.324e-05</td>
<td>3.087e-05</td>
<td>7.054e-05</td>
</tr>
</tbody>
</table>
Table VI. $L_\infty$ error for Example 5.1.

<table>
<thead>
<tr>
<th>T</th>
<th>Present Method $\sigma = 0.88$</th>
<th>Mittal and Jain $\sigma = 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2</td>
<td>1.9264e-05</td>
<td>1.8106e-05</td>
</tr>
<tr>
<td>0.4</td>
<td>3.8441e-05</td>
<td>3.5221e-05</td>
</tr>
<tr>
<td>0.6</td>
<td>5.6907e-05</td>
<td>5.1278e-05</td>
</tr>
<tr>
<td>0.8</td>
<td>7.5132e-05</td>
<td>6.6846e-05</td>
</tr>
<tr>
<td>1</td>
<td>9.2782e-05</td>
<td>8.1503e-05</td>
</tr>
<tr>
<td>5</td>
<td>3.7951e-04</td>
<td>2.6820e-04</td>
</tr>
<tr>
<td>10</td>
<td>6.1422e-04</td>
<td>3.5797e-04</td>
</tr>
<tr>
<td>20</td>
<td>8.5381e-04</td>
<td>4.2114e-04</td>
</tr>
</tbody>
</table>

Table VII. $L_\infty$ error for Example 5.4.

<table>
<thead>
<tr>
<th>T</th>
<th>Present Method $\sigma = 1.24$</th>
<th>Mittal and Jain $\sigma = 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2</td>
<td>1.7012e-09</td>
<td>1.6741e-09</td>
</tr>
<tr>
<td>0.4</td>
<td>3.3421e-09</td>
<td>3.2861e-09</td>
</tr>
<tr>
<td>0.6</td>
<td>4.9265e-09</td>
<td>4.8744e-09</td>
</tr>
<tr>
<td>0.8</td>
<td>6.4774e-09</td>
<td>6.4205e-09</td>
</tr>
<tr>
<td>1</td>
<td>7.9965e-09</td>
<td>7.9274e-09</td>
</tr>
<tr>
<td>5</td>
<td>3.2711e-08</td>
<td>3.2734e-08</td>
</tr>
<tr>
<td>10</td>
<td>5.2493e-08</td>
<td>5.2549e-08</td>
</tr>
<tr>
<td>20</td>
<td>7.1829e-08</td>
<td>7.1814e-08</td>
</tr>
</tbody>
</table>

Figure 1. Comparison of error of Example 5.1 along time for (A) $\sigma = 0.76$ and (B) $\sigma = 1.2$. 

![Comparison of error for Example 5.1](image-url)
Figure 2. Graphical representation of the exact and the approximate solution of Example 5.1 for $t \leq 2$, $\Delta t = 0.001$, $\sigma = 1.2$, $M = 21$.

Figure 3. Physical interpretation of solution of Example 5.1 for (A) $\sigma = 0.76$, $M = 21$ and (B) $\sigma = 1.2$, $M = 21$. 

(a) \hspace{5cm} (b)
Figure 4. Maximum absolute error plot of Example 5.2 along time for different value of $\sigma$, $\Delta t = 0.001$.

Figure 5. Numerical Solution of Example 5.2 at time $T = 1$ for (A) $\sigma = 0.82$ and (B) $\sigma = 1.14$. 
Figure 6. Exact and approximate solution of Example 5.2 for $\sigma = 0.9, M = 21$.

Figure 7. Physical interpretation of solution of Example 5.3 for (A) $\sigma = 0.87, M = 21$ and (B) $\sigma = 1.12, M = 21$. 
Figure 8. Approximate solution of Example 5.4 for (A) \( \sigma = 0.88, M = 11 \) and (B) \( \sigma = 1.18, M = 11 \).

6. Conclusion

In this article, we developed a collocation method based on cubic B-splines basis functions for a non-uniform mesh to solve the nonlinear parabolic PDE. Modification of cubic B-splines over the non-uniform mesh has been done to solve the Dirichlet boundary conditions. Crank Nicolson scheme is used to discretize the time derivative. We have discussed the Stability of this method using Von Neumann stability analysis. The numerical approximation of solutions have been obtained without linearization. We tested the method on some examples and the results obtained are satisfactory. Error analysis have been done to show the numerical validity. It was also found that the accuracy of numerical results with the proposed method is comparable to that obtained with the uniform mesh. Easy implementation is the strength of this method.

References