Research Paper
Computational Methods for Differential Equations
http://cmde.tabrizu.ac.ir
Vol. 10, No. 1, 2022, pp. 28-43
DOI:10.22034/cmde.2020.39472.1726

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

Swarn Singh[1,*], **Sandeep Bhatt**[2], **and Suruchi Singh**[3]
[1]Department of Mathematics, Sri Venkateswara College, University of Delhi, India.
[2]Department of Mathematics, University of Delhi, India.
[3]Department of Mathematics, Aditi Mahavidyalaya, University of Delhi, India.

Abstract

In this paper, an approximate solution of a 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 are 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:

$$v_t = F(x,t,v,v_x,v_{xx}), \quad a \le x \le b,\, 0 \le t \le T, \tag{1.1}$$

with the initial condition

$$v(x,0) = \psi(x), \quad a \le x \le b, \tag{1.2}$$

and boundary conditions (Neumann or Dirichlet or mixed).

The nonlinear parabolic partial differential equations can be seen in the most area of science and engineering. Some famous examples of such PDEs 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 Cherruault [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

Received: 25 April 2020 ; Accepted: 06 December 2020.
∗ Corresponding author. Email: ssingh@svc.ac.in.

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 a numerical technique for the solution of Fokker-Planck equation. This method uses the cubic B-spline scaling functions. The method consists of expanding the required approximate solution as the elements of cubic B-spline scaling function. Many authors presented numerical methods based on splines over uniform mesh [2, 11–13, 19, 23]. Not much work has been done on the cubic spline collocation method over the non-uniform mesh. Mittal [17] used the cubic B-spline collocation method on a uniform mesh to solve parabolic PDEs. 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 [7–9, 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 the collocation method. In the case of Dirichlet boundary conditions, we modify the cubic B-splines. The 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 $x_l = x_{l-1}+\delta_l; l = 1, 2, \dots, m-1$; mesh ratio $\sigma_l = \delta_{l+1}/\delta_l$, $l = 1, 2, \dots, m-1$ 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 $j = 0, 1, \dots, N$. For $\sigma_l = 1$; $l = 1, 2, \dots, m-1$, 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 outsides 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 $|x_p - x_{p-1}| = |x_{-p} - x_{-(p-1)}|$ 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.

$$\mathbf{S_l(x)} = \begin{cases} \dfrac{(x-x_{l-2})^3}{(x_{l-1}-x_{l-2})(x_l-x_{l-2})(x_{l+1}-x_{l-2})}, & x_{l-2} \le x < x_{l-1}, \\ \dfrac{(x-x_{l-2})^2(x_l-x)}{(x_l-x_{l-1})(x_l-x_{l-2})(x_{l+1}-x_{l-2})} + \dfrac{(x-x_{l-2})(x_{l+1}-x)(x-x_{l-1})}{(x_l-x_{l-1})(x_{l+1}-x_{l-1})(x_{l+1}-x_{l-2})} & \\ \quad + \dfrac{(x_{l+2}-x)(x-x_{l-1})^2}{(x_l-x_{l-1})(x_{l+1}-x_{l-1})(x_{l+2}-x_{l-1})}, & x_{l-1} \le x < x_l, \\ \dfrac{(x-x_{l-2})(x_{l+1}-x)^2}{(x_{l+1}-x_{l-1})(x_{l+1}-x_{l-2})(x_{l+1}-x_l)} + \dfrac{(x-x_{l-1})(x_{l+1}-x)(x_{l+2}-x)}{(x_{l+1}-x_{l-1})(x_{l+2}-x_{l-1})(x_{l+1}-x_l)} & \\ \quad + \dfrac{(x-x_l)(x_{l+2}-x)^2}{(x_{l+1}-x_l)(x_{l+2}-x_l)(x_{l+2}-x_{l-1})}, & x_l \le x < x_{l+1}, \\ \dfrac{(x_{l+2}-x)^3}{(x_{l+2}-x_{l-1})(x_{l+2}-x_l)(x_{l+2}-x_{l+1})}, & x_{l+1} \le x < x_{l+2}, \\ 0 & \text{otherwise.} \end{cases} \tag{2.1}$$

for $l = -1, 0, 1, \dots, m-1, m, m+1$. 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:

$$S_{l-1}(x_l) = \frac{\delta_{l+1}^2}{(\delta_{l+1}+\delta_l)(\delta_{l+1}+\delta_l+\delta_{l-1})}, \tag{2.2}$$

$$S_l(x_l) = \frac{\delta_{l+1}(\delta_l+\delta_{l-1})}{(\delta_{l+1}+\delta_l)(\delta_{l+1}+\delta_l+\delta_{l-1})} + \frac{\delta_l(\delta_{l+2}+\delta_{l+1})}{(\delta_{l+1}+\delta_l)(\delta_{l+2}+\delta_{l+1}+\delta_l)}, \tag{2.3}$$

$$S_{l+1}(x_l) = \frac{\delta_l^2}{(\delta_{l+1}+\delta_l)(\delta_{l+2}+\delta_{l+1}+\delta_l)}, \tag{2.4}$$

$$S'_{l-1}(x_l) = \frac{-3\delta_{l+1}}{(\delta_{l+1}+\delta_l)(\delta_{l+1}+\delta_l+\delta_{l-1})}, \tag{2.5}$$

$$S'_l(x_l) = \frac{3\delta_l}{(\delta_{l+1}+\delta_l)(\delta_{l+2}+\delta_{l+1}+\delta_l)}, \tag{2.6}$$

$$S''_{l-1}(x_l) = \frac{6}{(\delta_{l+1}+\delta_l)(\delta_{l+1}+\delta_l+\delta_{i-1})}, \tag{2.7}$$

$$S''_l(x_l) = \frac{2(\delta_l+\delta_{l-1}-2\delta_{l+1})}{\delta_{l+1}(\delta_{l+1}+\delta_l)(\delta_{l+1}+\delta_l+\delta_{l-1})} - \frac{2(\delta_l+2\delta_{l+1}+\delta_{l+2})}{\delta_{l+1}(\delta_{l+1}+\delta_l)(\delta_{l+2}+\delta_{l+1}+\delta_l)}, \tag{2.8}$$

$$S''_{l+1}(x_l) = \frac{6}{(\delta_{l+1}+\delta_l)(\delta_{l+2}+\delta_{l+1}+\delta_l)}. \tag{2.9}$$

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

$$V(x,t) = \sum_{l=-1}^{m+1} \gamma(t) S_l(x), \tag{2.10}$$

where, $\gamma_l(t)$ are unknown quantities depending on time. We will obtain 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

$$V_l = S_{l-1}(x_l)\gamma_{l-1} + S_l(x_l)\gamma_l + S_{l+1}(x_l)\gamma_{l+1}, \tag{2.11}$$

$$(V_x)_l = S'_{l-1}(x_l)\gamma_{l-1} + S'_l(x_l)\gamma_l + S'_{l+1}(x_l)\gamma_{l+1}, \tag{2.12}$$

$$(V_{xx})_l = S''_{l-1}(x_l)\gamma_{l-1} + S''_l(x_l)\gamma_l + S''_{l+1}(x_l)\gamma_{l+1}. \tag{2.13}$$

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

$$v_t = F(x,t,v,v_x,v_{xx}), \quad a \le x \le b,\ 0 \le t \le T, \tag{3.1}$$

with the initial condition

$$v(x,0) = \psi(x) \qquad a \le x \le b, \tag{3.2}$$

and boundary conditions

$$v_x(a,t) = g_1(t), \quad v_x(b,t) = g_2(t), \qquad t \ge 0. \tag{3.3}$$

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

$$\begin{aligned}\gamma_{-1} &= \frac{1}{S'_{-1}(x_0)}\Big(g_1(t) - S'_0(x_0)\gamma_0 - S'_1(x_0)\gamma_1\Big),\\ \gamma_{m+1} &= \frac{1}{S'_{m+1}(x_m)}\Big(g_2(t) - S'_{m-1}(x_m)\gamma_{m-1} - S'_m(x_m)\gamma_m\Big).\end{aligned} \tag{3.4}$$

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

$$(V_t)_l = \sum_{l=-1}^{m+1} \dot{\gamma}_l(t)S_l(x_l) = S_{l-1}(x_l)\dot{\gamma}_{l-1} + S_l(x_l)\dot{\gamma}_l + S_{l+1}(x_l)\dot{\gamma}_{l+1},$$

where $\dot{\gamma}_l = \dfrac{d\gamma}{dt}$. Now, applying the Crank-Nicolson scheme on equation (3.1), we get

$$\frac{V^{n+1}-V^n}{\Delta t} = \frac{1}{2}\Big(F\Big(t^{n+1},x,V^{n+1},V_x^{n+1},V_{xx}^{n+1}\Big) + F\Big(t^n,x,V^n,V_x^n,V_{xx}^n\Big)\Big),$$
$$\Rightarrow \quad 0 = V^{n+1} - \frac{\Delta t}{2}F\Big(t^{n+1},x,V^{n+1},V_x^{n+1},V_{xx}^{n+1}\Big) - V^n - \frac{\Delta t}{2}F\Big(t^n,x,V^n,V_x^n,V_{xx}^n\Big). \tag{3.5}$$

Using equations (2.11)-(3.5), we get

$$\begin{aligned}0 = &\Big(S_{l-1}(x_l)\gamma_{l-1}^{n+1} + S_l(x_l)\gamma_l^{n+1} + S_{l+1}(x_l)\gamma_{l+1}^{n+1}\Big) - \frac{\Delta t}{2}F\Bigg(t^{n+1},x_l,\Big(S_{l-1}(x_l)\gamma_{l-1}^{n+1} + S_l(x_l)\gamma_l^{n+1} + S_{l+1}(x_l)\gamma_{l+1}^{n+1}\Big),\\ &\Big(S'_{l-1}(x_l)\gamma_{l-1}^{n+1} + S'_l(x_l)\gamma_l^{n+1} + S'_{l+1}(x_l)\gamma_{l+1}^{n+1}\Big),\Big(S''_{l-1}(x_l)\gamma_{l-1}^{n+1} + S''_l(x_l)\gamma_l^{n+1} + S''_{l+1}(x_l)\gamma_{l+1}^{n+1}\Big)\Bigg)\\ &-\Big(S_{l-1}(x_l)\gamma_{l-1}^{n} + S_l(x_l)\gamma_l^{n} + S_{l+1}(x_l)\gamma_{l+1}^{n}\Big) - \frac{\Delta t}{2}F\Bigg(t^{n},x_l,\Big(S_{l-1}(x_l)\gamma_{l-1}^{n} + S_l(x_l)\gamma_l^{n} + S_{l+1}(x_l)\gamma_{l+1}^{n}\Big),\\ &\Big(S'_{l-1}(x_l)\gamma_{l-1}^{n} + S'_l(x_l)\gamma_l^{n} + S'_{l+1}(x_l)\gamma_{l+1}^{n}\Big),\Big(S''_{l-1}(x_l)\gamma_{l-1}^{n} + S''_l(x_l)\gamma_l^{n} + S''_{l+1}(x_l)\gamma_{l+1}^{n}\Big)\Bigg).\end{aligned} \tag{3.6}$$

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

$$P\gamma^{n+1} = Q\gamma^n + R, \tag{3.7}$$

where P and Q are tridiagonal matrices, $\gamma^n = [\gamma_0^n, \gamma_1^n, \ldots, \gamma_{m-1}^n, \gamma_m^n]^T$ 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 the case of a nonlinear parabolic partial differential equation, we will find the value of γ^{n+1} using the Newton-Raphson method. We can write (3.6) as

$$F^*(\gamma_l^{n+1}) = 0, \tag{3.8}$$

where

$$
\begin{aligned}
F^*(\gamma_l^{n+1}) \;=\;& \Big(S_{l-1}(x_l)\gamma_{l-1}^{n+1} + S_l(x_l)\gamma_l^{n+1} + S_{l+1}(x_l)\gamma_{l+1}^{n+1}\Big) \\
&-\frac{\Delta t}{2} F\Bigg(t^{n+1}, x_l, \Big(S_{l-1}(x_l)\gamma_{l-1}^{n+1} + S_l(x_l)\gamma_l^{n+1} + S_{l+1}(x_l)\gamma_{l+1}^{n+1}\Big), \\
&\qquad \Big(S'_{l-1}(x_l)\gamma_{l-1}^{n+1} + S'_l(x_l)\gamma_l^{n+1} + S'_{l+1}(x_l)\gamma_{l+1}^{n+1}\Big), \\
&\qquad \Big(S''_{l-1}(x_l)\gamma_{l-1}^{n+1} + S''_l(x_l)\gamma_l^{n+1} + S''_{l+1}(x_l)\gamma_{l+1}^{n+1}\Big)\Bigg) \\
&-V_i^n - \frac{\Delta t}{2} F\left(t^n, x_l, V_l^n, (V_x)_l^n, (V_{xx})_l^n\right). \qquad (3.9)
\end{aligned}
$$

By Newton-Raphson method, γ^{n+1} can be calculated from

$$
\left(\gamma_l^{n+1}\right)^{k+1} = \left(\gamma_l^{n+1}\right)^{k+1} - \frac{F^*\left(\left(\gamma_l^{n+1}\right)^k\right)}{F^{*'}\left(\left(\gamma_l^{n+1}\right)^k\right)}, \qquad \text{for} \quad k = 1, 2, 3, \dots \qquad (3.10)
$$

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

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

$$
\begin{aligned}
V_x(x_0, 0) &= g_1(0), \\
V(x_l, 0) &= \psi(x_l), \quad \text{for } l = 0, 1, 2, \cdots, \\
V_x(x_m, 0) &= g_2(0).
\end{aligned} \qquad (3.11)
$$

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

$$
P\gamma^0 = Q, \qquad (3.12)
$$

where, P is a tri-diagonal matrix

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

$$
S_0^*(x_0) = S_0(x_0) - \frac{S_0'(x_0)}{S_{-1}'(x_0)} S_0(x_0),
$$

$$
S_1^*(x_0) = S_1(x_0) - \frac{S_1'(x_0)}{S_{-1}'(x_0)} S_1(x_0),
$$

$$
S_{m-1}^*(x_m) = S_{m-1}(x_m) - \frac{S_{m-1}'(x_m)}{S_{m+1}'(x_m)} S_{m-1}(x_m),
$$

$$
S_m^*(x_m) = S_m(x_m) - \frac{S_m'(x_m)}{S_{m+1}'(x_m)} S_m(x_m),
$$

$$\gamma^0 = \begin{pmatrix} \gamma_0^0 \\ \gamma_1^0 \\ \gamma_2^0 \\ . \\ . \\ \gamma_{m-2}^0 \\ \gamma_{m-1}^0 \\ \gamma_m^0 \end{pmatrix}, Q = \begin{pmatrix} \psi(x_0) + \dfrac{S_{-1}(x_0)}{S'_{-1}(x_0)} g_1(0) \\ \psi(x_1) \\ \psi(x_2) \\ . \\ . \\ \psi(x_{m-2}) \\ \psi(x_{m-1}) \\ \psi(x_m) - \dfrac{S_{m+1}(x_m)}{S'_{m+1}(x_m)} g_2(0) \end{pmatrix}.$$

Now, the initial vector γ^0 can be found from equation (3.12) using the Thomas algorithm.

3.2. **Modified cubic B-splines for Dirichlet boundary conditions.** Now we consider the PDE subject to Dirichlet boundary conditions

$$v_t = F(x, t, v, v_x, v_{xx}), \quad a \le x \le b,\ 0 \le t \le T, \tag{3.13}$$

with the initial condition

$$v(x, 0) = \psi(x), \quad a \le x \le b, \tag{3.14}$$

and Dirichlet boundary conditions

$$v(a, t) = g_1^*(t), \quad v(b, t) = g_2^*(t), \quad t \ge 0. \tag{3.15}$$

In the collocation method, when dealing with Dirichlet boundary conditions, the basis functions should vanish on the boundary, but in the set of cubic B-splines the basis functions $S_{-1}, S_0, S_1, ..., S_{m-1}, S_m, S_{m+1}$ 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:

$$\begin{aligned} &\tilde{S}_0(x) = S_0(x) + 2S_{-1}(x), \\ &\tilde{S}_1(x) = S_1(x) - \frac{S_1(x_0)}{S_{-1}(x_0)} S_{-1}(x), \\ &\tilde{S}_l(x) = S_l(x) \qquad \text{for } l = 2, 3, \dots, m-3, m-2, \\ &\tilde{S}_{m-1}(x) = S_{m-1}(x) - \frac{S_{m-1}(x_m)}{S_{m+1}(x_m)} S_{m+1}(x), \\ &\tilde{S}_m(x) = S_m(x) + 2S_{m+1}(x). \end{aligned} \tag{3.16}$$

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

$$V(x, t) = \sum_{l=0}^{m} \gamma_l(t) \tilde{S}_l(x). \tag{3.17}$$

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:

$$V_l = \tilde{S}_{l-1}(x_l)\gamma_{l-1} + \tilde{S}_l(x_l)\gamma_l + \tilde{S}_{l+1}(x_l)\gamma_{l+1}, \tag{3.18}$$

$$(V_x)_l = \tilde{S}'_{l-1}(x_l)\gamma_{l-1} + \tilde{S}'_l(x_l)\gamma_l + \tilde{S}'_{l+1}(x_l)\gamma_{l+1}, \tag{3.19}$$

$$(V_{xx})_l = \tilde{S}''_{l-1}(x_l)\gamma_{l-1} + \tilde{S}''_l(x_l)\gamma_l + \tilde{S}''_{l+1}(x_l)\gamma_{l+1}. \tag{3.20}$$

Now, using the collocation method in space direction and Crank-Nicolson in time direction on PDE (3.13), we can find the value of γ^{n+1} in the same way as in section 3.1.

3.2.1. *The Initial vector* γ^0. The initial vector γ^0 can be obtained from the initial condition (3.14) and boundary condition (3.15) as:

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

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

$$\mathbf{P}\gamma^0 = Q, \tag{3.22}$$

where P is the tridiagonal matrix given by

$$P = \begin{pmatrix} \tilde{S}_0(x_0) & 0 & & & \\ \tilde{S}_0(x_1) & \tilde{S}_1(x_1) & \tilde{S}_2(x_1) & & \\ & \cdots & \cdots & \cdots & \\ & & \tilde{S}_{m-2}(x_{m-1}) & \tilde{S}_{m-1}(x_{m-1}) & \tilde{S}_m(x_{m-1}) \\ & & & 0 & \tilde{S}_m(x_m) \end{pmatrix},$$

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

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

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

$$v_t = F(x,t,v,v_x,v_{xx}), \quad a \le x \le b,\ 0 \le t \le T, \tag{3.23}$$

with the initial condition

$$v(x,0) = \psi(x), \quad a \le x \le b, \tag{3.24}$$

and boundary conditions

$$\begin{gathered} v(a,t) = g(t), \quad v_x(b,t) = g^*(t), \quad t \ge 0. \\ \text{or} \\ v_x(a,t) = g^*(t), \quad v(b,t) = h(t), \quad t \ge 0. \end{gathered} \tag{3.25}$$

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 the same.

4. Stability Analysis

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

$$v_t = \lambda v_{xx} - \varepsilon v, \tag{4.1}$$

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

$$\begin{aligned} &\frac{V^{n+1}-V^{n}}{\Delta t}=\lambda\frac{V_{xx}^{n+1}+V_{xx}^{n}}{2}-\varepsilon\frac{V^{n+1}+V^{n}}{2},\\ \Rightarrow\quad &\left(1+\frac{\varepsilon\Delta t}{2}\right)V^{n+1}-\frac{\lambda\Delta t}{2}V_{xx}^{n+1}=\left(1-\frac{\varepsilon\Delta t}{2}\right)V^{n}+\frac{\lambda\Delta t}{2}V_{xx}^{n}. \end{aligned} \tag{4.2}$$

Substituting values from equation (2.10), we get

$$\begin{aligned} &\sum_{l=-1}^{m+1}\left(\gamma_l^{n+1}(t)S_l(x)+\frac{\varepsilon\Delta t}{2}\gamma_l^{n+1}(t)S_l(x)-\frac{\lambda\Delta t}{2}\gamma_l^{n+1}(t)S_l^{''}(x)\right)=\\ &\qquad\sum_{l=-1}^{m+1}\left(\gamma_l^{n}(t)S_l(x)-\frac{\varepsilon\Delta t}{2}\gamma_l^{n}(t)S_l(x)+\frac{\lambda\Delta t}{2}\gamma_l^{n}(t)S_l^{''}(x)\right). \end{aligned} \tag{4.3}$$

Using the properties of B-splines, we get:

$$p\gamma_{l-1}^{n+1}+q\gamma_l^{n+1}+r\gamma_{l+1}^{n+1}=x\gamma_{l-1}^{n}+y\gamma_l^{n}+z\gamma_{l+1}^{n}, \tag{4.4}$$

where

$$\begin{aligned} p&=S_{l-1}(x_l)+\frac{\varepsilon\Delta t}{2}S_{l-1}(x_l)-\frac{\lambda\Delta t}{2}S_{l-1}^{''}(x_l),\\ q&=S_{l}(x_l)+\frac{\varepsilon\Delta t}{2}S_{l}(x_l)-\frac{\lambda\Delta t}{2}S_{l}^{''}(x_l),\\ r&=S_{l+1}(x_l)+\frac{\varepsilon\Delta t}{2}S_{l+1}(x_l)-\frac{\lambda\Delta t}{2}B_{l+1}^{''}(x_l),\\ x&=S_{l-1}(x_l)-\frac{\varepsilon\Delta t}{2}S_{l-1}(x_l)+\frac{\lambda\Delta t}{2}S_{l-1}^{''}(x_l),\\ y&=S_{l}(x_l)-\frac{\varepsilon\Delta t}{2}S_{l}(x_l)+\frac{\lambda\Delta t}{2}S_{l}^{''}(x_l),\\ z&=S_{l+1}(x_l-\frac{\varepsilon\Delta t}{2}S_{l+1}(x_l)+\frac{\lambda\Delta t}{2}S_{l+1}^{''}(x_l). \end{aligned} \tag{4.5}$$

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

$$\gamma_l^n=\xi^n e^{\mathbf{i}\beta x_l}, \tag{4.6}$$

where $\mathbf{i}=\sqrt{-1}$. Substituting (4.6) in (4.4) and simplifying, we get

$$\xi=\frac{(x\cos\beta\delta_l+y+z\cos\beta\delta_{l+1})+\mathbf{i}(z\sin\beta\delta_{l+1}-x\sin\beta\delta_l)}{(p\cos\beta\delta_l+q+r\cos\beta\delta_{l+1})+\mathbf{i}(r\sin\beta\delta_{l+1}-p\sin\beta\delta_l)}. \tag{4.7}$$

For stability we need

$$\begin{aligned} &|\xi|\le 1\\ \Leftrightarrow\quad &|(x\cos\beta\delta_l+y+z\cos\beta\delta_{l+1})+\mathbf{i}(z\sin\beta\delta_{l+1}-x\sin\beta\delta_l)|\\ &\qquad\le|(p\cos\beta\delta_l+q+r\cos\beta\delta_{l+1})+\mathbf{i}(r\sin\beta\delta_{l+1}-p\sin\beta\delta_l)|\\ \Leftrightarrow\quad &(x\cos\beta\delta_l+y+z\cos\beta\delta_{l+1})^2+(z\sin\beta\delta_{l+1}-x\sin\beta\delta_l)^2\\ &\qquad\le(p\cos\beta\delta_l+q+r\cos\beta\delta_{l+1})^2+(r\sin\beta\delta_{l+1}-p\sin\beta\delta_l)^2\\ \Leftrightarrow\quad &(\sigma^3+\sigma)(\cos\beta(\delta_l+\delta_{l+1})-1)+(\cos\beta\delta_l-1)(\sigma^3-\sigma^4+2\sigma^2)+(\cos\beta(\delta_l\\ &+\delta_{l+1})-1)(2\sigma^2+\sigma-1)-\frac{\varepsilon\delta_l^2}{\gamma}\left(\left(\sigma^3\cos\beta\delta_l+2\sigma^2+2\sigma+\cos\beta\delta_{l+1}\right)^2\right.\\ &\left.+\left(\sin\beta\delta_{l+1}-\sigma^3\sin\beta\delta_l\right)^2\right)\le 0, \end{aligned}$$

which is true for $\frac{1}{2} \le \sigma \le 2$. Therefore, the method is stable for $\frac{1}{2} \le \sigma \le 2$.

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

$$\begin{aligned} b - a &= x_m - x_0 \\ &= (x_m - x_{m-1}) + (x_{m-1} - x_{m-2}) + \cdots + (x_2 - x_1) + (x_1 - x_0) \\ &= \delta_m + \delta_{m-1} + \cdots + \delta_2 + \delta_1 \\ &= (\sigma_{m-1}\sigma_{m-2}\cdots\sigma_2\sigma_1 + \cdots + \sigma_1 + 1)\delta_1. \end{aligned} \tag{5.1}$$

In our numerical experiments, we consider $\sigma_i = \sigma(\text{constant})$. So from (5.1) we have

$$\delta_1 = \frac{b - a}{1 + \sigma + \sigma^2 + \cdots + \sigma^{m-1}} = \frac{b - a}{1 - \sigma^m}. \tag{5.2}$$

So, with total $m + 1$ grid points, we can evaluate δ_1 using (5.2). δ_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_∞ 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]):

$$v_t = [(1 - v)v_x]_x - 2v^2 + 2v, \quad -\frac{\pi}{2} \le x \le \frac{\pi}{2}, \ 0 \le t \le T,$$

with initial condition

$$v(x, 0) = \frac{2 - \sin x}{3}, \quad -\frac{\pi}{2} \le x \le \frac{\pi}{2},$$

and Neumann boundary conditions

$$(v_x)_{(-\frac{\pi}{2},t)} = 0 \quad \text{and} \quad (v_x)_{(\frac{\pi}{2},t)} = 0, \qquad t \ge 0.$$

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 1. L_∞ 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 σ. In Table 2, L_∞ and L_2 errors have been tabulated at various time levels for $\sigma = 0.76$ and $\sigma = 1.2$.

Example 5.2. Consider the Burger's equation (Asaithambi [4]):

$$v_t = \nu v_{xx} - v v_x, \quad 0 \le x \le 1,\, t \ge 0,$$

with initial condition

$$v(x,0) = \frac{2\pi\nu \sin \pi x}{\cos \pi x + \gamma}, \quad 0 \le x \le 1,$$

and Dirichlet boundary conditions

$$v(0,t) = 0 \quad \text{and} \quad v(1,t) = 0, \quad t \ge 0.$$

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 3. We have computed L_∞ and L_2 errors and displayed them in the table at time $T = 1.5$. L_∞ and L_2 errors have been evaluated at different time levels and displayed in Table 4. We have illustrated the behavior of the maximum absolute error as time progress in Figure 4 for different value of σ. The approximate solution is presented graphically for $\sigma = 0.9$ and $\sigma = 1.14$ is shown in Figure 5. The exact and approximate solution is illustrated in Figure 6.

Example 5.3. Consider the Burger's equation (Raslan [22]):

$$v_t = \nu v_{xx} - \gamma v v_x, \qquad 0 \le x \le 1,\, 0 \le t \le T,$$

with initial condition is

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

and mixed boundary conditions are

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

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 σ is given in Table 5 at different time levels. The physical behavior of the solution is illustrated in Figure 7.

Example 5.4. We consider the following convection-diffusion equation

$$v_t = \gamma v_{xx} - \varepsilon v_x, \quad 0 \le x \le 1,\, 0 \le t \le T,$$

with initial condition

$$v(x,0) = e^{\nu x}, \quad 0 \le x \le 1,$$

and Neumann boundary conditions

$$(v_x)_{(0,t)} = \nu e^{\eta t}, \qquad (v_x)_{(1,t)} = \nu e^{\nu + \eta t}, \quad t \ge 0.$$

The analytic solution is given by

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

In our first computation, first we take $\varepsilon = 0.1, \gamma = 0.02, \eta = -0.09, \sigma = 0.8, \nu = 1.17712434446770$. The maximum absolute error is shown in Table 6 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 7 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 1. Comparison of the analytic and approximate solution for Example 5.1 at $T = 2$ for $\Delta t = 0.001, M = 21$.

	$\sigma = 0.76$			$\sigma = 1.2$	
x	Present Method	Exact	x	Present Method	Exact
-1.5708	0.9999999	1.0000000	-1.5708	0.9999999	1.0000000
-0.2383	0.9908248	0.9908394	-1.5338	0.9999919	0.9999918
0.5314	0.9818371	0.9819332	-1.4805	0.9999511	0.9999511
0.9759	0.9779057	0.9780781	-1.4037	0.9998330	0.9998330
1.2327	0.9764903	0.9766972	-1.2931	0.9995407	0.9995408
1.3810	0.9760143	0.9762336	-1.1340	0.9988736	0.9988740
1.4667	0.9758598	0.9760833	-0.9047	0.9974364	0.9974371
1.5162	0.9758115	0.9760363	-0.5746	0.9945286	0.9945266
1.5448	0.9757973	0.9760225	-0.0993	0.9892184	0.9891981
1.5613	0.9757937	0.9760189	0.5852	0.9814672	0.9813863
1.5708	0.9757931	0.9760184	1.5708	0.9761683	0.9760184
L_∞ Error	2.2526242e-04	-	L_∞ Error	1.4987125e-04	-
L_2 Error	8.6018027e-04	-	L_2 Error	2.1715947e-04	-

Table 2. Error norms of Example 5.1 at different time levels for $M = 21, \Delta t = 0.001$.

t	$\sigma = 0.76$		$\sigma = 1.2$	
	L_∞ Error	L_2 Error	L_∞ Error	L_2 Error
0.1	4.1076e-03	1.1908e-02	4.7682e-04	6.0761e-04
0.4	4.2711e-03	1.4337e-02	1.1249e-03	1.5221e-03
0.7	2.5044e-03	9.1215e-03	1.0870e-03	1.5184e-03
1.0	1.4628e-03	5.4896e-03	8.0494e-04	1.1444e-03
2	2.2526e-04	8.6018e-04	1.4987e-04	2.1716e-04
3	3.1142e-05	1.1915e-04	2.1196e-05	3.0794e-05
4	4.2269e-06	1.6176e-05	2.8861e-06	4.1944e-06
5	5.7223e-07	2.1900e-06	3.9094e-07	5.6820e-07
7	1.0480e-08	4.0107e-08	7.1627e-09	1.0411e-08
10	2.5966e-11	9.9304e-11	1.7729e-11	2.5755e-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. The Crank Nicolson scheme is used to discritize 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 has 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.

TABLE 3. Comparison for Approximate and Exact Solution of Example 5.2 for $\sigma = 0.9, \Delta t = 0.001, T = 1.5, M = 21, \nu = 0.1$.

x	Approximate	Exact
0.1138	0.0226561	0.0226137
0.3085	0.0554362	0.0553681
0.4662	0.0702687	0.0702329
0.5939	0.0707326	0.0707345
0.6974	0.0622736	0.0622952
0.7812	0.0497116	0.0497370
0.8490	0.0363003	0.0363214
0.9040	0.0238028	0.0238174
0.9485	0.0129511	0.0129591
0.9846	0.0038923	0.0038947
L_∞ Error	6.8098989e-05	-
L_2 Error	1.3805602e-04	-

TABLE 4. Error computation for Example 5.2 at different time levels for $\sigma = 1.14, M = 21, \Delta t = 0.001$.

t	L_∞ Error	L_2 Error
0.1	3.4443e-03	3.9777e-03
0.5	3.0277e-03	4.7924e-03
1.0	1.9320e-03	3.6925e-03
1.5	1.2569e-03	2.8626e-03
2.0	8.5449e-04	2.1400e-03
4.0	1.8093e-04	4.8875e-04
6.0	3.3455e-05	9.1227e-05
8.0	5.7895e-06	1.5804e-05
10	9.6081e-07	2.6243e-06
12	1.5499e-07	4.2350e-07
14	2.4489e-08	6.6935e-08
16	3.8086e-09	1.0413e-08

TABLE 5. Maximum absolute error of Example 5.3 for different values of σ.

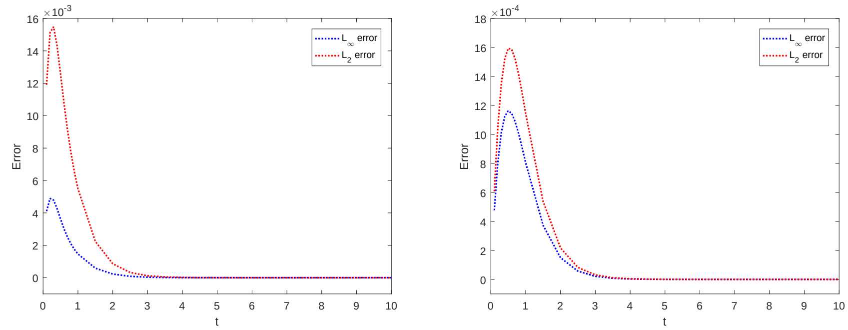
t	σ			
	0.87	1	1.12	1.25
0.1	4.6404e-03	1.8486e-03	4.4251e-03	1.0242e-02
0.5	4.0684e-03	9.7491e-04	2.3347e-03	5.4330e-03
1	2.7587e-03	4.1361e-04	9.7682e-04	2.2511e-03
2	1.6182e-03	1.3446e-04	3.1541e-04	7.1841e-04
2.5	1.3387e-03	9.0467e-05	2.1249e-04	4.83e-04
3	1.1385e-03	6.5058e-05	1.5207e-04	3.4741e-04
5	7.1458e-04	2.5096e-05	5.8819e-05	1.3442e-04
6	6.0309e-04	1.7810e-05	4.1490e-05	9.4644e-05
7	5.2138e-04	1.3242e-05	3.0874e-05	7.0542e-05

TABLE 6. L_∞ error for Example 5.1.

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

TABLE 7. L_∞ error for Example 5.4.

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



(A) (B)

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

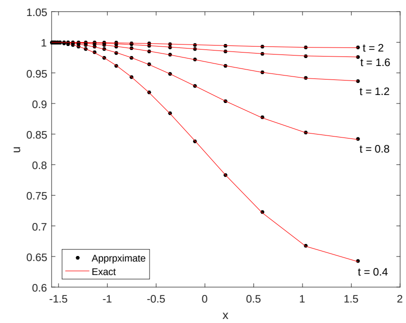
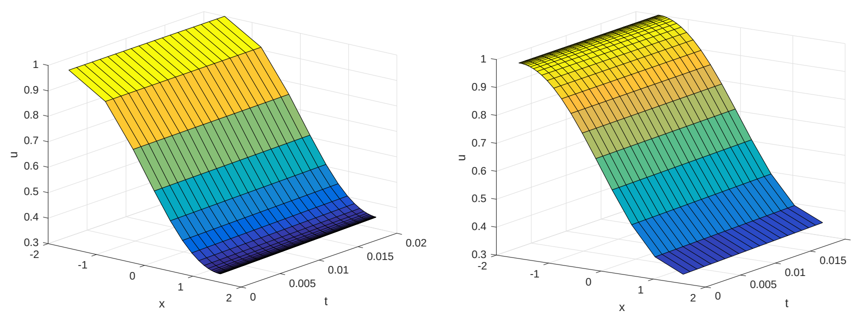


FIGURE 2. Graphical representation of the exact and the approximate solution of Example 5.1 for $t \leq 2, \Delta t = 0.001, \sigma = 1.2, M = 21$.



(A) (B)

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

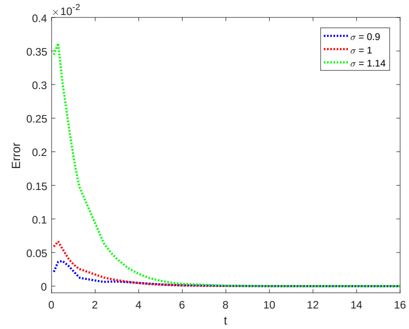
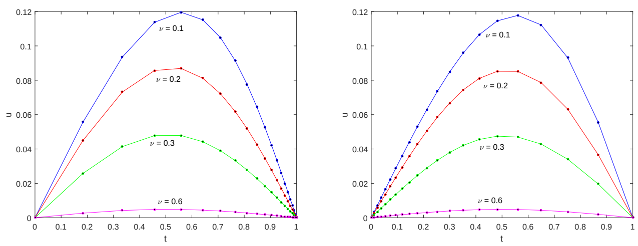
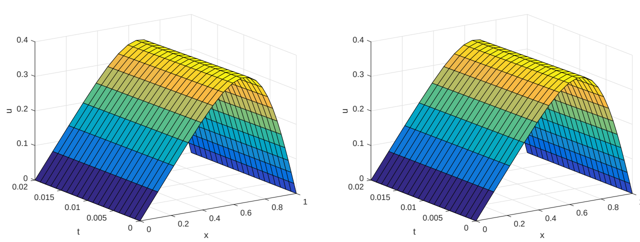


FIGURE 4. Maximum absolute error plot of Example 5.2 along time for different value of $\sigma, \Delta t = 0.001$.



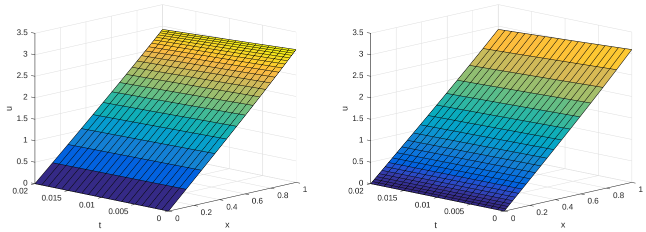
(A) (B)

FIGURE 5. Numerical Solution of Example 5.2 at time $T = 1$ for (A) $\sigma = 0.82$ and (B) $\sigma = 1.14$.



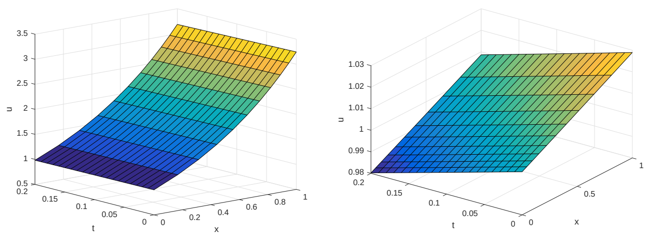
(A) (B)

FIGURE 6. Exact and approximate solutions of Example 5.2 for $\sigma = 0.9, M = 21$.



(A) (B)

FIGURE 7. Physical interpretation of solution of Example 5.3 for (A) $\sigma = 0.87, M = 21$ and (B) $\sigma = 1.12, M = 21$.



(A) (B)

FIGURE 8. Approximate solution of Example 5.4 for (A) $\sigma = 0.88, M = 11$ and (B) $\sigma = 1.18, M = 11$.

References

[1] K. Al-Khaled, *Numerical study of Fisher's reactiondiffusion equation by the Sinc collocation method*, J. Comput. Appl. Math., *137(2)* (2001), 245–255.

[2] A. H. A. Ali, G. A. Gardner, and L. R. T. Gardner, *A collocation solution for Burgers' equation using cubic B-spline finite elements*, Comput. Methods Appl. Mech. Eng., *100* (1992), 325–337.

[3] K. K. Ali, K. R. Raslan, and T. S. El-Danaf, *Non-polynomial Spline Method for Solving Coupled Burgers Equations*, Comput. Methods Differ. Equ., *3* (2015), 218–230.

[4] A. Asaithambi, *Numerical solution of the Burgers equation by automatic differentiation*, Appl. Math. Comput., *216* (2010), 2700–2708.

[5] M. M. Cecchi, R. Nociforo, and P. P. Grego, *Space-time finite elements numerical solutions of Burgers Problems*, Le Matematiche, *51* (1996), 43–57.

[6] R. Cherniha, *Exact and numerical solutions of the generalized Fisher equation*, Rep. Math. Phys., *47* (2001), 393–411.

[7] C. Clavero, J. C. Jorge, and F. Lisbona, *A uniformly convergent scheme on a nonuniform mesh for convection-diffusion parabolic problems*, J. Comput. Appl. Math., *154* (2003), 415–429.

[8] M. G. Cox, *The numerical evaluation of B-splines*, IMA J. Appl. Math., *10* (1972), 134–149.

[9] C. De Boor, *A practical guide to splines*, New York: springer-verlag, *27* (1978), 325.

[10] A. Dogan, *A Galerkin finite element approach to Burgers' equation*, Appl. Math. Comput., *157(2)* (2004), 331–346.

[11] X. Han, *Quadratic trigonometric polynomial curves with a shape parameter*, Comput. Aided Geom. Des., *19* (2002), 503–512.

[12] M. K. Jain, R. K. Jain, and R. K. Mohanty, *High order difference methods for system of 1d nonlinear parabolic partial differential equations*, Int. J. Comput. Math., *37* (1990), 105–112, Doi: 10.1080/00207169008803938.

[13] A. Korkmaz and İ. Dağ, *Cubic Bspline differential quadrature methods and stability for Burgers' equation*, Eng. Comput., *30* (2013), 320–344, Doi: 10.1108/02644401311314312.

[14] M. Lakestani and M. Dehghan, *Numerical solution of FokkerPlanck equation using the cubic Bspline scaling functions*, Numer. Methods Partial Differ. Equ., *25* (2009), 418–429.

[15] M. Lakestani and M. Dehghan, *Numerical solutions of the generalized KuramotoSivashinsky equation using B-spline functions*, Appl. Math. Model., *36* (2012), 605–617.

[16] T. Mavoungou and Y. Cherruault, *Numerical study of Fisher's equation by Adomian's method*, Math. Comput. Model., *19* (1994), 89–95.

[17] R. C. Mittal and R. K. Jain, *Numerical solution of convection-diffusion equation using cubic B-splines collocation methods with Neumanns boundary conditions*, Int. J. Appl. Math. Comput., *4* (2012), 115–127.

[18] R. K. Mohanty, D. J. Evans, and N. Khosla, *An non-uniform mesh cubic spline TAGE method for nonlinear singular two-point boundary value problems*, Int. J. Comput. Math., *82* (2005), 1125–1139.

[19] R. K. Mohanty and M. K. Jain, *High-accuracy cubic spline alternating group explicit methods for 1D quasi-linear parabolic equations*. Int. J. Comput. Math., *86* (2009), 1556–1571.

[20] R. Pourgholi and A. Saeedi, *Applications of cubic Bsplines collocation method for solving nonlinear inverse parabolic partial differential equations*,Numer. Methods. Partial. Differ. Equ., *33* (2017), 88–104.

[21] L. B. Rall, *Automatic differentiation: Techniques and applications*,Lect. Notes Comput. Sci., *120* (1981).

[22] K. R. Raslan, *A collocation solution for Burgers equation using quadratic B-spline finite elements*, Int. J. Comput. Math., *80* (2003), 931–938.

[23] S. Singh, S. Singh, and R. Arora, *Numerical solution of second-order one-dimensional hyperbolic equation by exponential B-spline collocation method*, Numer. Anal. Appl., *10*(2017), 164–176, Doi: 10.1134/S1995423917020070.

[24] X. Y. Wang, *Exact and explicit solitary wave solutions for the generalized Fisher equation*, Physics letters A, *131* (1988), 277–279.

[25] H. Zadvan and J. Rashidinia, *Development of non-polynomial spline and new B-spline with application to solution of Klein-Gordon equation*, Computational Methods for Differential Equations, (2020), Doi: 10.22034/cmde.2020.27847.1377.