Available online at http://scik.org
J. Math. Comput. Sci. 11 (2021), No. 4, 4351-4368
https://doi.org/10.28919/jmcs/5659
ISSN: 1927-5307

OPTIMAL CUBIC SPLINE METHOD FOR CONVECTION DIFFUSION EQUATION

SWARN SINGH[1], SURUCHI SINGH[2,*], SANDEEP BHATT[3]

[1]Department of Mathematics, Sri Venkateswara College, University of Delhi, Delhi, 110021, India

[2]Department of Mathematics, Aditi Mahavidyalaya, University of Delhi, Delhi, 110039, India

[2]Department of Mathematics, University of Delhi, Delhi, 110007, India

Abstract. In the proposed work, we present a new collocation technique based on cubic splines to solve initial-boundary value parabolic partial differential equation. To attain fourth order accuracy, the proposed method requires only three spatial grid points as compare to the requirement of five grid points in the literature, using the collocation methods based on splines. We have used two stage Gauss Legendre method in time direction. An analysis has been done to prove the unconditional stability of the technique. To show the better accuracy of our method, numerical experiments are done by taking some examples from the literature. The results obtained, show the efficiency and the order of accuracy of the technique.

Keywords: cubic B-spline; collocation; fourth order method; two stage Gauss Legendre method; stability.

2010 AMS Subject Classification: Primary 65N12, Secondary 35K05.

1. INTRODUCTION

In this paper, we are concerned with finding a stable and high accurate method for the approximate solution of a second order linear parabolic partial differential equation:

$$u_t = \mathscr{P}u + f, \tag{1}$$

*Corresponding author
E-mail address: ssuruchi2005@yahoo.co.in
Received March 08, 2021

in a domain $[a,b] \times [0,T]$, with the initial condition

$$u(x,0) = \varphi(x), \quad x \in [a,b], \tag{2}$$

and Dirichlet boundary conditions,

$$u(a,t) = \chi_1(t) \quad \text{and} \quad u(b,t) = \chi_2(t), \quad \forall t \in [0,T], \tag{3}$$

where $\mathscr{P}u(x,t) = \kappa u_{xx}(x,t) + \varepsilon u_x(x,t) + \eta u(x,t)$ is a linear second order operator and κ, ε and η are constants and $\kappa > 0$.

Collocation method is one of the popular methods used for finding the numerical solution of ordinary differential equations (ODEs) and partial differential equations (PDEs). In the collocation method, the approximate solution for differential equation lies in an approximation space of piecewise polynomials, with respect to the partition of the domain of the problem. Orthogonal piecewise polynomial collocation and spline collocation methods are few of them. The choice of spline functions as the piecewise polynomials results in a solution that is smooth at the nodes of the partition up to a certain degree. In literature, there are two typical ways for introducing collocation methods based on splines for the discretizing parabolic differential equations in spatial direction. One of these is standard second order formulation which requires the solution of a tridiagonal system of equations at each time step. Second way is to use a fourth order accurate deferred-correction or extrapolated spline collocation methods, which requires the solution of either a pentadiagonal system or two tridiagonal systems of equations at each time step. Archer [1] proposed cubic spline collocation method for quasi linear parabolic equations, which gives rise to fourth order pentadiagonal matrix structure. Mittal and Jain [6] used a combination of cubic spline collocation (CSC) and Crank-Nicolson (CN) to solve a convection diffusion equation which resulted in second order accurate solution. Various methods based on splines have been discussed to solve (1) such as cubic trigonometric B-splines [9], combination of spline and finite element [10], cubic spline [11], quartic and quintic B-spline [12] and exponential B-splines [13]. High order finite difference method for non linear parabolic partial differential equations are discussed in [14]. Non-polynomial cubic spline methods for the solution of parabolic equations are presented in [15].

In [2, 7], a combination of quadratic spline collocation (QSC) and finite difference method (FDM) is used to solve general linear parabolic PDEs. This algorithm gives fourth order accurate solution in space variable and second order accurate solution in time variable. In [4], Liu and Wang used QSC-TG method based on QSC and two-stage Gauss method for solving linear parabolic PDEs. The solutions obtained by this algorithm are fourth order accurate in each time and space variable. Even though QSC-TG method gives forth order accuracy in both time and space direction, it uses a pentadiagonal system which results in a higher computational time and effort.

In this work, we propose a CSC-GL4 algorithm for linear parabolic PDEs which is based on CSC method and fourth order two stage Gauss Legendre method. This algorithm gives fourth order accuracy in both the space and time variables with minimal computational efforts. Stability analysis is done in an analytical manner in comparison to Christara et. al. [2] and [4], which proved stability in numerical manner. Since the proposed CSC-GL4 method uses triadiagonal system to find the solution of parabolic equation with high order accuracy in both time and space direction. So, it is better than QSC method in terms of computational time and efforts.

The paper is organized in the following manner. In Section 2, fourth order the cubic spline collocation method is proposed which gives rise to a tridiagonal structure. In Section 3, two stage Gauss-Legendre Method (GL4 Method) of order four is presented. In Section 4, stability of the CSC-GL4 algorithm is shown. In Section 5, numerical experiments has been done to illustrate the effectiveness of the algorithm.

2. Fourth Order Cubic B-Spline Collocation Method

Let us consider $\Omega = \{a = x_0 < x_1 < x_2 < \ldots < x_{J-1} < x_J = b\}$ to be an equally spaced partition of interval $[a,b]$ with mesh spacing h and nodes $x_j = a + jh, j = 0, 1, \ldots, J$. A cubic spline function S over partition Ω of interval $[a,b]$ is a function such that $S \in C^2[a,b]$ and $S_{[x_{j-1},x_j]}$ is a cubic polynomial for $1 \le j \le J$.

In this paper, we use the collocation form of cubic B-spline to approximate the exact solution $u(x,t)$ of the differential equation. Any cubic spline function $U(x,t)$ defined on the partition Ω

can be written as a linear combination

$$U(x,t) = \sum_{j=-1}^{J-1} s_j(t)\Psi_{3,j}(x), \tag{4}$$

where $s_j(t)$ are time dependent parameters and $\{\Psi_{3,-1}, \Psi_{3,0}, \Psi_{3,1}, \ldots, \Psi_{3,j}, \Psi_{3,J+1}\}$ are cubic polynomial that forms a basis for the space of all cubic spline over the partition Ω. The cubic spline basis functions $\Psi_{3,j}(x)$ at the nodes are defined as (Boor[3]):

$$\Psi_{3,j}(x) = \frac{1}{h^3}\begin{cases} (x-x_j)^3 & x_{j-2} \le x \le x_{j-1} \\ h^3 + 3h^2(x-x_{j+1}) + 3h(x-x_{j+1})^2 - 3(x-x_{j+1})^3 & x_{j-1} \le x \le x_j \\ h^3 + 3h^2(x_{j+1}-x) + 3h(x_{j+1}-x)^2 - 3(x_{j+1}-x)^3 & x_j \le x \le x_{j+1} \\ (x_{j+2}-x)^3 & x_{j+1} \le x \le x_{j+2} \\ 0 & \text{elsewhere} \end{cases} \tag{5}$$

The values of $\Psi_{3,j}(x)$ and its derivatives at different grid points obtained from (5) are given in the table 1.

The approximate value of the solution U and its two derivatives D_xU and D_x^2U at the node x_j in terms of parameters $s_j \equiv s_j(t)$ are as follows:

$$U_j = s_{j-1} + 4s_j + s_{j+1}, \tag{6}$$

$$D_xU_j = \frac{3}{h}(s_{j+1} - s_{j-1}), \tag{7}$$

$$D_x^2U_j = \frac{6}{h^2}(s_{j-1} - 2s_j + s_{j+1}), \tag{8}$$

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

Fourth order accurate approximate solution of the parabolic partial differential equation (1), is obtained by following cubic spline collocation method

$$\mathscr{P}_1U_j + \mathscr{P}_2U_{t_j} = f_j - \frac{\varepsilon h^2}{12\kappa} f_{x_j} - \frac{h^2}{12} f_{xx_j} \quad \text{for} \quad j = 0, 1, \ldots, J, \tag{9}$$

where

$$\mathscr{P}_1U_j = -\kappa\left(1 + \frac{\varepsilon^2h^2}{12\kappa^2} - \frac{\eta h^2}{12\kappa}\right)D_x^2U_j - \varepsilon\left(1 + \frac{\eta h^2}{12\kappa}\right)D_xU_j - \eta U_j, \tag{10}$$

$$\mathscr{P}_2U_j = U_j - \frac{h^2}{12}D_x^2U_j + \frac{\varepsilon h^2}{12\kappa}D_xU_j, \tag{11}$$

with $U = u$ on the boundary of the domain.

TABLE 1. Values of the spline basis function and derivatives at knots.

x	$\Psi_{3,j}(x)$	$\Psi'_{3,j}(x)$	$\Psi''_{3,j+1}(x)$
x_j	0	0	0
x_{j+1}	1	$3/h$	$6/h^2$
x_{j+2}	4	0	$-12/h^2$
x_{j+3}	1	$-3/h$	$6/h^2$
x_{j+4}	0	0	0

To obtain the approximate solution $U(x,t)$ at a particular grid point x_j, we need to evaluate the parameters $s_j(t)$. This can be done by using the given boundary conditions and the collocation form (9) of the differential equation. The method (9), using equations (10)-(11) can be written as an equation with six unknown time dependent parameters $s_{j-1}, s_j, s_{j+1}, \dot{s}_{j-1}, \dot{s}_j$ and $\dot{s}_{j+1}$ where $\dot{s}_j$ represents the derivative of s_j at the knot x_j with respect to time. To get rid two extra parameters we make use of boundary conditions. Using equations (3) and (6), we get

$$\begin{aligned} s_{-1} + 4s_0 + s_1 &= \chi_1(t), \\ s_{J-1} + 4s_J + s_{J+1} &= \chi_2(t). \end{aligned} \tag{12}$$

These equations allows us to eliminate $s_{-1}, s_{J+1}, \dot{s}_{-1}$ and $\dot{s}_{J+1}$ from the system of equations (9). So, we get a system of $J+1$ equations in $J+1$ unknowns.

Next, we will show that the numerical solution obtained by the method (9) is of order four. For this, let us consider a cubic spline interpolant $\mathscr{V}$ of u defined by

$$\mathscr{V}_j = u_j, j = 0, 1, 2, \ldots, J, \tag{13}$$

$$D_x^2 \mathscr{V}_j = u_{xxj} - \frac{h^2}{12} u^4_{x_j} + O(h^4), j = 0, J. \tag{14}$$

Theorem 1. *Let $\mathscr{V}$ be a cubic spline interpolation of the exact solution $u \in C^6[a,b]$ which satisfies* (13)-(14). *Let U be as defined by equations* (9)-(11). *Then for $0 \leq j \leq N$,*

(i) $\mathscr{V}_j = u_j + O(h^4)$,

(ii) $D_x\mathscr{V}_j = u_{x\,j} + O(h^4)$,

(iii) $D_x^2\mathscr{V}_j = u_{xx\,j} + O(h^2)$,

(iv) $D_x^2\mathscr{V}_j = u_{xx\,j} - \frac{h^2}{12}u_{x^4\,j} + O(h^4)$,

(v) $\mathscr{P}_1(\mathscr{V}_j - U_j) + \mathscr{P}_2(\mathscr{V}_{t_j} - U_{t_j}) = O(h^4)$.

Proof. The proofs *(i)-(iv)* follows from [5].

From the equation (9), we obtain

$$\mathscr{P}_1U_j + \mathscr{P}_2U_{t_j} = f_j - \frac{h^2}{12}f_{xx_j} + \frac{\varepsilon h^2}{12\kappa}f_{x_j}, \tag{15}$$

for $j = 0, 1, \ldots, J$. From the differential equation (1) and the equation (15), we get

$$\begin{aligned}\mathscr{P}_1U_j + \mathscr{P}_2U_{t_j} =& u_{t_j} - \frac{h^2}{12}u_{xxt_j} + \frac{\varepsilon h^2}{12\kappa}u_{xt_j} - \varepsilon\left(1 + \frac{\eta h^2}{12}u_{x_j}\right)\\ &- \kappa\left(u_{xx\,j} - \frac{h^2}{12}u_{x^4\,j}\right) - \eta\left(u_j - \frac{h^2}{12}u_{xx\,j}\right) - \frac{\varepsilon^2h^2}{12\kappa}u_{xx\,j},\end{aligned} \tag{16}$$

Using equations (6)-(8) and (10)-(11), we have

$$\begin{aligned}\mathscr{P}_1\mathscr{V}_j + \mathscr{P}_2\mathscr{V}_{t_j} =& -\eta(1 - \frac{h^2}{12}D_x^2)\mathscr{V}_j - \left(\kappa + \frac{\varepsilon^2h^2}{12\kappa}\right)D_x^2\mathscr{V}_j\\ &- \varepsilon\left(1 + \frac{\eta h^2}{12\kappa}\right)D_x\mathscr{V}_j + \left(1 - \frac{h^2}{12}D_x^2 + \frac{\varepsilon h^2}{12\kappa}D_x\right)\mathscr{V}_{t_j},\end{aligned} \tag{17}$$

The equations (16), (17) using *(i)-(iv)* provides us

$$\mathscr{P}_1(\mathscr{V}_j - U_j) + \mathscr{P}_2(\mathscr{V}_{t_j} - U_{t_j}) = O(h^4), \tag{18}$$

for all $j = 0, 1, \ldots, J$. □

From Theorem 1, it can be observed that the CSC method (9) for the partial differential equation (1) is fourth order accurate.

3. Two Stage Gauss-Legendre Method(GL4 Method)

The two-stage Gauss-Legendre method of order four (GL4) is a particular class of implicit Runge Kutta methods. The Butcher tableau for the GL4 method is given by table 2.

TABLE 2. Butcher tableau of GL4 Method.

$$
\begin{array}{c|cc}
0.5\left(1-\frac{1}{\sqrt{3}}\right) & \frac{1}{4} & 0.5\left(0.5-\frac{1}{\sqrt{3}}\right) \\
0.5\left(1+\frac{1}{\sqrt{3}}\right) & 0.5\left(0.5+\frac{1}{\sqrt{3}}\right) & \frac{1}{4} \\
\hline
 & \frac{1}{2} & \frac{1}{2}
\end{array}
$$

Let Δt be the step size in time direction so that $t_n = n\Delta t$ for $n = 0,1,\ldots$. In this method, solution of the initial value problem

$$\dot{c} = g(t,c), c(0) = c^0, \tag{19}$$

is obtained by the formula

$$c^{n+1} = c^n + \frac{\Delta t}{2}(w_1 + w_2), \tag{20}$$

where, the weights w_1 and w_2 are calculated by

$$
\begin{aligned}
w_1 &= g\left(t_n + 0.5\left(1-\tfrac{1}{\sqrt{3}}\right), c^n + \tfrac{\Delta t}{4} w_1 + 0.5\left(0.5-\tfrac{1}{\sqrt{3}}\right)\Delta t w_2\right), \\
w_2 &= g\left(t_n + 0.5\left(1+\tfrac{1}{\sqrt{3}}\right), c^n + 0.5\left(0.5+\tfrac{1}{\sqrt{3}}\right)\Delta t w_1 + \tfrac{\Delta t}{4} w_2\right).
\end{aligned} \tag{21}
$$

After the initial vector $\mathbf{C}^0$ has been found from the initial conditions, the approximate solution of the differential equation (1) at any time t can be obtained using the recurrence relation (20). The CSC method together with the GL4 method produces an error of order $O(h^4 + \Delta t^4)$.

4. STABILITY OF CSC-GL4 METHOD

In this section, we will discuss the stability of the CSC-GL4 method (Cubic spline collocation in space direction and two stage Gauss Legendre method in time direction). The cubic spline collocation method given by equations (9) for the differential equation (1), in matrix form can be written as

$$\mathbf{D}_2\mathbf{C}_t = \mathbf{D}_1\mathbf{C} + \mathbf{F}, \tag{22}$$

where $\mathbf{D}_1$ and $\mathbf{D}_2$ are square matrices defined as follows

$$\mathbf{D}_1 = \kappa\left(1+\frac{\varepsilon^2 h^2}{12\kappa^2}-\frac{\eta h^2}{12\kappa}\right)\mathbf{B}_2+\varepsilon\left(1+\frac{\eta h^2}{12\kappa}\right)\mathbf{B}_1+\eta\mathbf{B}_0, \tag{23}$$

$$\mathbf{D}_2 = \mathbf{B}_0-\frac{h^2}{12}\mathbf{B}_2+\frac{\varepsilon h^2}{12\kappa}\mathbf{B}_1. \tag{24}$$

Here, $\mathbf{B}_0, \mathbf{B}_1$ and $\mathbf{B}_2$ are tri-diagonal square matrices of order $J+1$, given by $\mathbf{B}_0 = tridiag[1\,4\,1], \mathbf{B}_1 = tridiag[\frac{-3}{h}\,0\,\frac{3}{h}]$ and $\mathbf{B}_2 = tridiag[\frac{6}{h^2}\,\frac{-12}{h^2}\,\frac{6}{h^2}]$. $\mathbf{F}$ is a column vector of order $J+1$ corresponding to the given forcing function f and the boundary values. $\mathbf{C}$ is a column vector given by

$$\mathbf{C} = \begin{pmatrix} s_0(t) & s_1(t) & \cdots & s_J(t) \end{pmatrix}^t.$$

In homogeneous case, the two-stage cubic spline Gauss-Legendre method can be written as

$$\mathbf{C}^{n+1} = \mathbf{C}^n + \Delta t\frac{w_1+w_2}{2}. \tag{25}$$

Using equations(21), the weights w_1 and w_2 are obtained from the equations

$$\begin{aligned} \mathbf{D}_2 w_1 &= \mathbf{D}_1\left(\mathbf{C}^n+\tfrac{\Delta t}{4}w_1+0.5\left(0.5-\tfrac{1}{\sqrt{3}}\right)\Delta t w_2\right),\\ \mathbf{D}_2 w_2 &= \mathbf{D}_1\left(C^n+0.5\left(0.5+\tfrac{1}{\sqrt{3}}\right)\Delta t w_1+\tfrac{\Delta t}{4}w_2\right). \end{aligned} \tag{26}$$

The system of equations (26) can be written as

$$\begin{pmatrix} \mathbf{D}_2-\frac{\Delta t}{4}\mathbf{D}_1 & -\Delta t\upsilon_1\mathbf{D}_1 \\ -\Delta t\upsilon_2\mathbf{D}_1 & \mathbf{D}_2-\frac{\Delta t}{4}\mathbf{D}_1 \end{pmatrix}\begin{pmatrix} w_1 \\ w_2 \end{pmatrix} = \begin{pmatrix} \mathbf{D}_1\mathbf{C}^n \\ \mathbf{D}_1\mathbf{C}^n \end{pmatrix}, \tag{27}$$

where $\upsilon_1 = 0.5\left(0.5-\frac{1}{\sqrt{3}}\right)$ and $\upsilon_2 = 0.5\left(0.5+\frac{1}{\sqrt{3}}\right)$.

From the equations (25) and (27), we get

$$\mathbf{C}^{n+1} = \mathbf{C}^n+\frac{\Delta t}{2}\begin{pmatrix} \mathbf{I} & \mathbf{I} \end{pmatrix}\begin{pmatrix} \mathbf{D}_2-\frac{\Delta t}{4}\mathbf{D}_1 & -\Delta t\upsilon_1\mathbf{D}_1 \\ -\Delta t\upsilon_2\mathbf{D}_1 & \mathbf{D}_2-\frac{\Delta t}{4}\mathbf{D}_1 \end{pmatrix}^{-1}\begin{pmatrix} \mathbf{D}_1\mathbf{C}^n & 0 \\ 0 & \mathbf{D}_1\mathbf{C}^n \end{pmatrix}. \tag{28}$$

Here, $\mathbf{I}$ is an identity matrix of order $J+1$. We can further write the equation (28) as

$$\mathbf{C}^{n+1} = \frac{1}{2}\begin{pmatrix} \mathbf{I} & \mathbf{I} \end{pmatrix}\begin{pmatrix} \mathbf{D}_2-\frac{\Delta t}{4}\mathbf{D}_1 & -\Delta t\upsilon_1\mathbf{D}_1 \\ -\Delta t\upsilon_2\mathbf{D}_1 & \mathbf{D}_2-\frac{\Delta t}{4}\mathbf{D}_1 \end{pmatrix}^{-1}\begin{pmatrix} \mathbf{D}_2+\frac{3\Delta t}{4}\mathbf{D}_1 & -\Delta t\upsilon_1\mathbf{D}_1 \\ -\Delta t\upsilon_2\mathbf{D}_1 & \mathbf{D}_2+\frac{3\Delta t}{4}\mathbf{D}_1 \end{pmatrix}\begin{pmatrix} \mathbf{C}^n \\ \mathbf{C}^n \end{pmatrix}. \tag{29}$$

Since the matrix $\mathbf{D}_2$ is diagonally dominant, it is invertible. So, from the equation (29) we obtain

(30)
$$\mathbf{C}^{n+1} = \frac{1}{2}\begin{pmatrix}\mathbf{I} & \mathbf{I}\end{pmatrix}\begin{pmatrix}\mathbf{I}-\frac{\Delta t}{4}\mathbf{D}_2^{-1}\mathbf{D}_1 & -\Delta t\upsilon_1\mathbf{D}_2^{-1}\mathbf{D}_1\\ -\Delta t\upsilon_2\mathbf{D}_2^{-1}\mathbf{D}_1 & \mathbf{I}-\frac{\Delta t}{4}\mathbf{D}_2^{-1}\mathbf{D}_1\end{pmatrix}^{-1}\begin{pmatrix}\mathbf{I}+\frac{3\Delta t}{4}\mathbf{D}_2^{-1}\mathbf{D}_1 & -\Delta t\upsilon_1\mathbf{D}_2^{-1}\mathbf{D}_1\\ -\Delta t\upsilon_2\mathbf{D}_2^{-1}\mathbf{D}_1 & \mathbf{I}+\frac{3\Delta t}{4}\mathbf{D}_2^{-1}\mathbf{D}_1\end{pmatrix}\begin{pmatrix}\mathbf{C}^n\\ \mathbf{C}^n\end{pmatrix}.$$

To prove that the CSC-GL4 method is stable, it is sufficient to prove that the eigen values of the matrix $\mathbf{X}$ are less than the eigen values of the matrix $\mathbf{Y}$, where the matrices $\mathbf{X}$ and $\mathbf{Y}$ are given by

$$\mathbf{X} = \begin{pmatrix}1+\frac{3\Delta t}{4}\mathbf{D}_2^{-1}\mathbf{D}_1 & -\Delta t\upsilon_1\mathbf{D}_2^{-1}\mathbf{D}_1\\ -\Delta t\upsilon_2\mathbf{D}_2^{-1}\mathbf{D}_1 & 1+\frac{3\Delta t}{4}\mathbf{D}_2^{-1}\mathbf{D}_1\end{pmatrix}, \tag{31}$$

and

$$\mathbf{Y} = \begin{pmatrix}1-\frac{\Delta t}{4}\mathbf{D}_2^{-1}\mathbf{D}_1 & -\Delta t\upsilon_1\mathbf{D}_2^{-1}\mathbf{D}_1\\ -\Delta t\upsilon_2\mathbf{D}_2^{-1}\mathbf{D}_1 & 1-\frac{\Delta t}{4}\mathbf{D}_2^{-1}\mathbf{D}_1\end{pmatrix}. \tag{32}$$

When the mesh size increases, i.e. $h \to 0$ the matrix $\mathbf{D}_2^{-1}\mathbf{D}_1 \to \kappa\mathbf{B}_0^{-1}\mathbf{B}_2$. It can be easily seen that the eigen values of the matrix $\mathbf{B}_0^{-1}\mathbf{B}_2$ are negative. Let λ be an eigen value of the matrix $\mathbf{D}_2^{-1}\mathbf{D}_1$. Then the eigen values of the matrix $\mathbf{X}$ corresponding to the same eigen vector are given by the characteristic equation

$$\Lambda^2 - \left(2+\frac{3\Delta t}{2}\lambda\right)\Lambda + \left(1+\frac{3\Delta t}{4}\lambda\right)^2 - \upsilon_1\upsilon_2\Delta t^2\lambda^2 = 0, \tag{33}$$

and the eigen values of the matrix $\mathbf{Y}$ are given by the equation

$$\Lambda^2 - \left(2-\frac{\Delta t}{2}\lambda\right)\Lambda + \left(1-\frac{\Delta t}{4}\lambda\right)^2 - \upsilon_1\upsilon_2\Delta t^2\lambda^2 = 0. \tag{34}$$

Since $Min(eig(\mathbf{Y})) = 1-\frac{\Delta t}{4}\lambda - \Delta t\lambda\sqrt{\upsilon_1\upsilon_2}$ and the $Max(eig(\mathbf{X})) = 1+\frac{3\Delta t}{4}\lambda + \Delta t\lambda\sqrt{\upsilon_1\upsilon_2}$. It can be easily seen that $Max(eig(X)) < Min(eig(Y))$. Thus the proposed CSC-GL4 method is unconditionally stable.

5. Numerical Examples

In this section, we implement the proposed CSC-GL4 method to compute the approximate solution of some test problems. To show the accuracy of the method, we have calculated the maximum absolute error between the exact and approximate solution by the formula, $L_\infty = \max_j |u_j - U_j|$. Order of convergence of the proposed method is calculated using the following formula:

$$\text{Order} = \frac{\log(L_\infty(J_1)/L_\infty(J_2))}{\log(J_2/J_1)},$$

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

Example 1. Consider the test problem from Liu and Wang[4],

$$u_t = \kappa u_{xx} + (2\kappa\pi^2 - 1)e^{-t/2}\sin\pi x, \quad (x,t) \in [0,1]\times[0,5],$$

with initial condition

$$u(x,0) = 2\sin\pi x, \quad x \in [0,1],$$

and Dirichlet boundary conditions

$$u(0,t) = u(1,t) = 0, \forall t \in [0,5].$$

Here $\kappa > 0$ is a constant. The exact solution of this problem is

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

We have calculated the approximate solution for different values of κ. At first, we choose $\kappa = 1$ and time step $\Delta t = 1/1024$. The errors for various values of space step sizes and the order of the method obtained is shown in table 3. It can be seen from the tabulated values that the errors obtained are fourth order accurate in space direction. We have also shown fourth order accuracy in time direction for different values of time step by choosing $h = 1/1024$ in table 4. A comparisons of the errors with the results obtained in [4] are also done in the tables 3 and 4. In table 5, errors are calculated for $\kappa = 0.5$ at time $T = 5$. Fourth order accuracy can be seen for $h = k$.

Example 2: Let us consider the differential equation considered in [2]

$$u_t = \kappa u_{xx} + (\kappa\pi^2 - 1)e^{-t}\sin\pi x, \quad (x,t) \in [0,1]\times[0,T],$$

TABLE 3. Comparisons of the errors for Example 1 for $\Delta t = 1/1024$ and different values of h.

h	CSC-GL4		Liu and Wang[4]	
	Error	Order	Error	Order
1/8	9.6565e-06	-	8.6200e-05	-
1/16	6.1834e-07	4.32	2.2500e-06	5.26
1/32	3.8877e-08	4.17	1.1400e-07	4.30
1/64	2.4334e-09	4.09	6.9100e-09	4.05
1/128	1.5214e-10	4.05	4.3000e-10	3.96

TABLE 4. Comparisons of the errors of Example 1 for $h = 1/1024$ and different values of Δt.

Δt	CSC-GL4		Liu and Wang[4]	
	Error	Order	Error	Order
1/4	1.7029e-06	-	1.84e-05	-
1/8	1.1113e-07	4.64	1.16e-06	4.70
1/16	7.1319e-09	4.32	7.26e-08	4.36
1/32	-	-	4.58e-09	4.17

TABLE 5. errors and order of accuracy of Example 1 for $\kappa = 0.5$ and $h = k$.

h	1/8	1/16	1/32	1/64	1/128
CSC-GL4	8.6851e-06	5.5791e-07	3.5104e-08	2.1976e-09	1.3741e-10
Order	-	4.32	4.20	4.09	4.04

with the exact solution

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

The initial and boundary conditions are obtained from the exact solution. For $\kappa = 1$, we compute the solution for different grid sizes and taking $h = k$ at time $T = 1$. Fourth order accuracy and errors are displayed in table 6. Comparisons with methods presented by Christara et. al.[2] are listed in the table. In [2], the authors discussed one step (1QSC-CN) and two step (2QSC-CN) quadratic spline collocation methods. We can also observe from the table that the time step chosen for our method is larger in compared to the QSC-CN, which reduces the computational time to obtain the solution at a required time level.

TABLE 6. Error comparisons with the QSC-CN methods [2] of Example 2.

h	CSC-GL4 ($\Delta t = h$)		QSC-CN[2] $\left(\Delta t = \frac{5.06}{5.5}h^2\right)$		1QSC-CN[2] $\left(\Delta t = \frac{5.06}{5.5}h^2\right)$		2QSC-CN[2] $\left(\Delta t = \frac{5.06}{5.5}h^2\right)$	
	Error	Order	Error	Order	Error	Order	Error	Order
1/16	4.3777e-06	-	1.2e-05	-	2.2e-05	-	2.2e-05	-
1/32	2.7959e-07	4.15	7.3e-07	4.22	1.5e-06	3.82	1.5e-06	3.85
1/64	1.7638e-08	4.67	4.5e-08	4.11	9.6e-08	3.97	9.7e-08	3.98
1/128	1.1071e-09	4.04	2.8e-09	4.05	6.1e-09	3.99	6.1e-09	3.99

Example 3. Consider the problem

$$u_t = \kappa u_{xx} + \varepsilon u_x + \eta u, \quad (x,t) \in [0,1] \times [0,1],$$

subject to the initial condition

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

The exact solution of the problem is given by

$$u(x,t) = e^t (x - x^2).$$

We compute the solution for various values of κ, ε and η. First, we choose $\kappa = 0.1, \varepsilon = 100$ and $\eta = 1$. For our second computation, we choose $\kappa = 0.1, \varepsilon = \eta = 0$. Errors for different values of step size, $h = k$ are computed and fourth order of accuracy of the method is displayed in table 7. Graphical representation of numerical solution obtained for $\kappa = 0.1, \varepsilon = 100$, $\eta = 1$, $J = 16$ is shown in figure 1.

Example 4. Consider the convection diffusion equation

$$u_t + \varepsilon u_x = \kappa u_{xx}, \quad (x,t) \in [0,1] \times [0,T].$$

The initial condition and Dirichlet boundary conditions are considered in accordance with the exact solution

$$u(x,t) = exp(\alpha x + \beta t).$$

For our first computation, we choose $\alpha = 1.1771, \beta = -0.09, \kappa = 0.01, \varepsilon = 0.1, h = k = 0.01$. We have evaluated the errors at various time levels, namely $T = 1, T = 3$ and $T = 5$ and displayed in the table 8. The errors obtained from proposed method are compared with those of Mittal and Jain [6] and Ismail at el.[8]. From the comparisons done in the table, we can see that our method works better compared to Mittal and Jain [6] in every possible way but sometimes it is less accurate than Ismail at el.[8]. However, our method is unconditionally stable compared to conditionally stable method of Ismail at el.[8]. To show the fourth order accuracy of the method we have calculated the errors for various and listed in table 9. Graphical representation of numerical solution obtained is shown in figure 3.

For our second computation, we choose $\alpha = 0.02855, \beta = -0.0999, \kappa = 0.022, \varepsilon = 3.5$. Errors have been computed at various time levels for $h = k = 0.01$ and comparisons with the results listed in Mittal and Jain[6] and Ismail at el.[8] have been done in table 10. We can see that our method works better.

TABLE 7. Error computation and order of accuracy of Example 3 at time $T = 1$ and various mesh size h.

h	$\kappa = 0.1, \varepsilon = 100, \eta = 1$		$\kappa = 0.1, \varepsilon = \eta = 0$	
	Error	Order	Error	Order
1/8	1.6771e-04	-	3.2238e-07	-
1/16	1.0182e-05	4.41	2.0366e-08	4.34
1/32	7.0246e-07	4.03	1.2873e-09	4.16
1/64	5.0242e-08	3.89	8.1027e-11	4.08
1/128	3.0903e-09	4.07	5.0841e-12	4.04
1/256	1.8116e-10	4.12	3.1845e-13	4.02

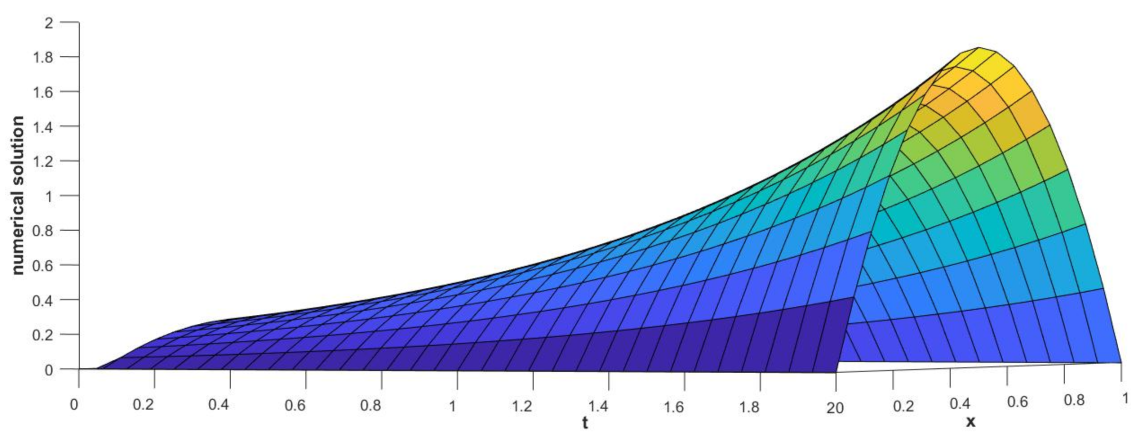

FIGURE 1. Numerical solution for $\kappa = 0.1, \varepsilon = 100, \eta = 1$.

TABLE 8. Error comparison of Example 4 at various time levels for $\alpha = 1.1771, \beta = -0.09, \kappa = 0.01, \varepsilon = 0.1, h = k = 0.01$.

x	Proposed Method (CSC-GL4)			Mittal and Jain [6]			Ismail et al. [8]		
	$T = 1$	$T = 2$	$T = 5$	$T = 1$	$T = 2$	$T = 5$	$T = 1$	$T = 2$	$T = 5$
0.1	2.1958e-11	2.9077e-11	3.2798e-11	1.73e-07	2.29e-07	2.58e-07	2.22e-16	2.22e-16	3.33e-16
0.5	6.6558e-11	1.1591e-10	1.7224e-10	5.24e-07	9.13e-07	1.36e-06	8.88e-16	1.33e-15	2.44e-15
0.9	6.8137e-11	1.0267e-10	1.4288e-10	5.37e-07	8.09e-07	1.12e-06	0.00e+00	4.44e-16	8.88e-16

TABLE 9. Error comparison of Example 4 at various value of step size h with $\alpha = 1.1771, \beta = -0.09, \kappa = 0.01, \varepsilon = 0.1$.

h	Error	Order
1/8	2.0750e-06	-
1/16	1.2902e-07	4.37
1/32	8.0872e-09	4.18
1/64	5.0528e-10	4.09
1/128	3.1581e-11	4.05
1/256	1.9735e-12	4.03

TABLE 10. Error comparison of Example 4 for $\alpha = 0.02855, \beta = -0.0999, \kappa = 0.022, \varepsilon = -3.5, h = \Delta t = 0.01$.

x	Proposed Method (CSC-GL4)			Mittal and Jain [6]			Ismail et al.[8]		
	$T = 1$	$T = 2$	$T = 5$	$T = 1$	$T = 2$	$T = 5$	$T = 1$	$T = 2$	$T = 5$
0.1	5.5511e-16	3.6637e-15	7.2164e-15	2.16e-12	1.95e-12	1.45e-12	2.56e-10	2.38e-10	5.65e-10
0.5	5.5511e-16	3.5527e-15	6.5503e-15	1.09e-11	9.88e-12	7.32e-12	8.39e-10	1.38e-409	1.91e-09
0.9	4.4409e-16	2.5535e-15	4.7740e-15	1.99e-11	1.80e-11	1.33e-11	1.33e-09	2.83e-09	3.97e-09

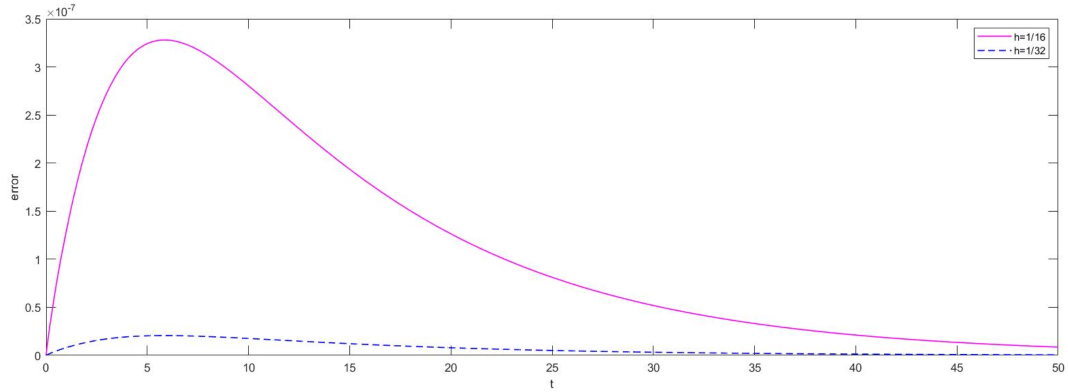

FIGURE 2. Error comparison of Example 4 for $h = k = 1/16$ and $h = k = 1/32$.

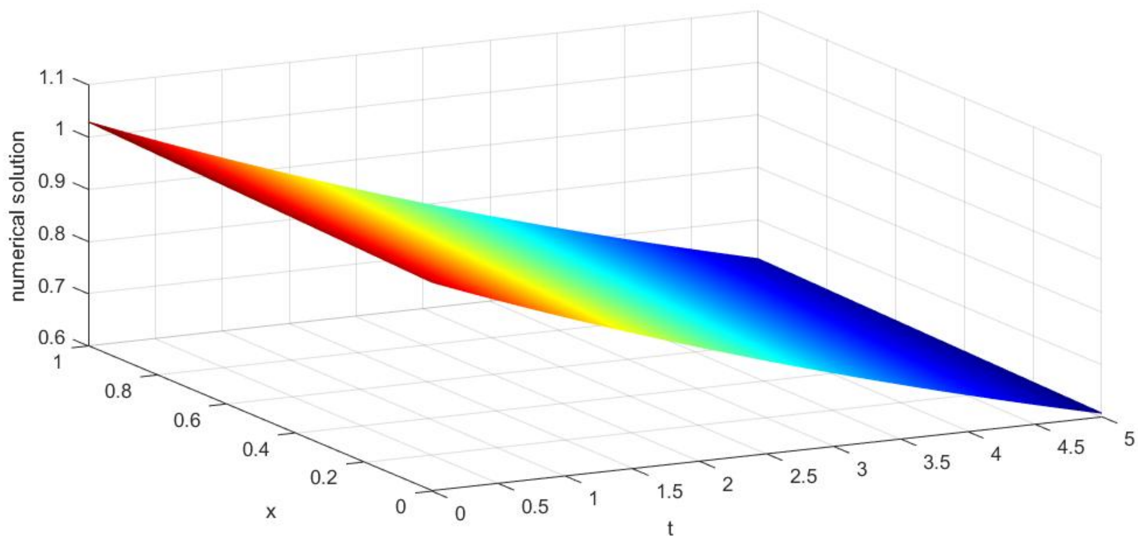

FIGURE 3. Numerical solution of Example 4.

6. Conclusion

In this work, we proposed a new CSC-GL4 method to solve initial-boundary value parabolic partial differential equation. A compact collocation method based on cubic splines is used in space direction. Discretization in time direction is handled by using two stage Gauss Legendre method. To attain high order accuracy the proposed method requires only three spatial grid points at each time step as compare to the requirement of five grid points at each time step in the literature, using the collocation methods based on splines. The proposed method is of order

$O(\Delta x^4 + \Delta t^4)$. The CSC-GL4 method is unconditionally stable. The numerical experiments performed, exhibits the efficiency and accuracy of the method. Comparison analysis with the literature shows the better accuracy with less computational efforts.

CONFLICT OF INTERESTS

The author(s) declare that there is no conflict of interests.

REFERENCES

[1] D. Archer, An $O(h^4)$ cubic spline collocation method for quasilinear parabolic equations. SIAM J. Numer. Anal. 14(4) (1977), 620-637.

[2] C.C. Christara, T. Chen, D.M. Dang, Quadratic spline collocation for one-dimensional linear parabolic partial differential equations. Numer. Algorithms, 53(4) (2010), 511-553.

[3] C. de Boor, A practical guide to splines, Springer, New York, (2001).

[4] J. Liu, Y. Wang, A Numerical Method of High Accuracy for Linear Parabolic Partial Differential Equations, Math. Probl. Eng. 2012 (2012), 497365.

[5] T.R. Lucas, Error bounds for interpolating cubic splines under various end conditions. SIAM J. Numer. Anal. 11(3) (1974), 569-584.

[6] R.C. Mittal, R. Jain, Redefined cubic B-splines collocation method for solving convection–diffusion equations. Appl. Math. Model. 36(11) (2012), 5555-5573.

[7] T. Chen, An efficient algorithm based on quadratic spline collocation and finite difference methods for parabolic partial differential equations. Ontario, University of Toronto, Canada, 2005.

[8] H.N. Ismail, E.M. Elbarbary, G.S. Salem, Restrictive Taylor's approximation for solving convection–diffusion equation. Appl. Math. Comput. 147(2) (2004), 355-363.

[9] T. Nazir, M. Abbas, A.I.M. Ismail, A.A. Majid, A. Rashid, The numerical solution of advection–diffusion problems using new cubic trigonometric B-splines approach. Appl. Math. Model. 40(7-8) (2016), 4586-4611.

[10] R. Szymkiewicz, Solution of the advection–diffusion equation using the spline function and finite elements. Commun. Numer. Meth. Eng. 9(3) (1993), 197-206.

[11] J. Goh, A. Abd. Majid, A.I.Md. Ismail, Cubic B-Spline Collocation Method for One-Dimensional Heat and Advection-Diffusion Equations, J. Appl. Math. 2012 (2012), 458701.

[12] A. Korkmaz, I. Dağ, Quartic and quintic B-spline methods for advection–diffusion equation. Appl. Math. Comput. 274 (2016), 208-219.

[13] R. Mohammadi, Exponential B-Spline Solution of Convection-Diffusion Equations, Appl. Math. 4(6) (2013), 933-944.

[14] M. K.Jain, R.K. Jain, R.K. Mohanty, High order difference methods for system of id nonlinear parabolic partial differential equations. Int. J. Computer Math. 37(1-2) (1990), 105-112.

[15] J. Rashidinia, R. Mohammadi, Non-polynomial cubic spline methods for the solution of parabolic equations. Int. J. Computer Math. 85(5) (2008), 843-850.