Journal of Mathematical Modeling
Vol. -, No. -, 2026, pp. -. Research Article

JMM

Parameter-uniform spline in compression method for singularly perturbed parabolic boundary turning point problem

Ding Jian[1], **Wondwosen Gebeyaw Melesse**[1,2], **Fasika Wondimu Gelu**[2*], **Jemal Abdule Bedaso** [2]

[1] *Department of Mathematics, Nanjing University of Information Science and Technology, Nanjing, 210044 Jingsu, China,*
[2]*Department of Mathematics, Dilla University, Dilla, 419 Southern State, Ethiopia*

Email(s): df2001101@126.com, wondoseng@du.edu.et, Fasikag@du.edu.et, jemalabdule1557@gmail.com

Abstract. This study investigates a parameter-uniform numerical method for solving singularly perturbed parabolic boundary turning point problem. The continuous problem is discretized using a spline in compression method for the space derivative on a Shishkin mesh and the Crank-Nicolson method for the time derivative on an equidistant mesh. The error analysis shows that the method is parameter-uniformly convergent with an error bound of order $O(N_x^{-2}\ln^2 N_x + N_t^2)$, where N_x is the numebr of space mesh intervals and N_t is the number of time mesh intervals. Three test examples are used to perform several numerical experiments, and comparative analyses are presented to validate the applicability and efficiency of the proposed method.

Keywords: Spline in compression method, boundary turning point, Crank-Nicolson method, Shishkin mesh.
AMS Subject Classification 2010: 34B08, 65L11, 65L20.

1 Introduction

A differential equation is called 'singularly perturbed' when a small positive parameter $\varepsilon(0 < \varepsilon \ll 1)$ multiplies the highest-order derivative. It is generally known that the numerical solution to a singularly perturbed differential equation contains boundary and/or interior layers. Due to their importance in simulating various real-life occurrences in both science and engineering, singularly perturbed problems with a boundary turning point have attracted the interest of many scholars. When the convection coefficient vanishes at one of the boundaries of the domain, the problem is referred to as a boundary turning point.

*Corresponding author
Received: 18 February 2026/ Revised: 07 July 2026/ Accepted: 07 July 2026
DOI: 10.22124/jmm.2026.33001.3005

Interior turning points are zeros inside the domain of differential equations, while boundary turning points are zeros that coincide with the boundary.

The author of [1] claimed that a linear velocity distribution causes a single boundary turning point problem when modelling heat and mass transport flow close to an oceanic rise. Modeling the thermal boundary layers in laminar flow [2] provides multiple boundary turning point problems. When convective transport processes of diffusing substances (heat, matter) are mathematically described, a singularly perturbed problem with a boundary turning point occurs because the rate of flow from one of the boundaries is proportional to the distance from the boundary.

The rest of this article is organized as follows: Section 2 presents a statement of the continuous problems. Section 3 discusses the fully discrete problem. In Section 4, the convergence analysis of the discrete problem is described through error analysis. Section 5 provides numerical examples and discussions. Finally, Section 6 offers a summary.

2 Statement of the problem

In this study, we consider the singularly perturbed parabolic boundary turning point problem of the form

$$\begin{cases} P_\varepsilon \phi(x,t) \equiv -\phi_t + \varepsilon \phi_{xx} + a(x)\phi_x - b(x)\phi = f(x,t), & (x,t) \in \Omega, \\ \begin{cases} \phi(x,0) = \vartheta(x), & x \in \Theta_x, \\ \phi(0,t) = \psi_L(t), \quad \phi(1,t) = \psi_R(t), & t \in \Theta_t, \end{cases} \end{cases} \tag{1}$$

where $\varepsilon (0 < \varepsilon \ll 1)$ is the diffusion parameter which makes Eq. (1) singularly perturbed differential equation, $\Omega = \Theta_x \times \Theta_t = (0,1) \times (0,T]$ and $\partial\Omega = \bar{\Omega} \backslash \Omega$. Assume $a(x) = a_0(x)x^p$, $b(x)$, $f(x,t)$, $\vartheta(x)$, $\psi_L(t)$ and $\psi_R(t)$ are smooth and bounded satisfying conditions $a_0(x) \geq \alpha > 0$, $b(x) \geq \beta > 0$, $(x,t) \in \overline{\Omega}$, p denotes the multiplicity of the turning point. In Eq. (1), the convection coefficient $a(x)$ vanishes at $x = 0$. Due to this, $x = 0$ is called the turning point of Eq. (1). The problem in Eq. (1) is known as boundary turning point because this turning point and the left boundary point coincide. A simple turning point problem is one in which the convective coefficient $a(x)$ vanishes at $x = 0$ in Eq. (1) when $p = 1$. The problem is referred to as a multiple turning point problem when $p > 1$. The convective coefficient $a(x)$ and its derivatives vanish at $x = 0$ for multiple turning point problems. Earlier numerical methods established in [3, 5–9, 11–15, 21–23] were fitted finite difference methods (fitted mesh and fitted operator methods) to solve Eq. (1). A cubic B-spline collocation method in [4] and a quadratic B-spline method in [10] used to solve Eq. (1) were quite different than the cubic spline in compression method. The latter method produces a better numerical solution than the former methods in terms of smaller maximum point-wise errors and boundary resolving property. In this study, we use the Crank-Nicolson method for the time variable discretization and the cubic spline in compression method on a Shishkin mesh for the space variable discretization. The proposed method provides an effective and uniformly convergent numerical solutions for the considered class of problems and demonstrates favorable performance when compared with existing methods reported in the literature. The cubic spline in compression method has not previously been applied to singularly perturbed parabolic boundary turning point problems. Singularly perturbed parabolic problem with time delay and interior and boundary turning points was studied in [18–20]. The following compatibility conditions at the domain's corners are assumed to be satisfied:

$$\begin{aligned} &\vartheta(0) = \psi_L(0), \qquad \vartheta(1) = \psi_R(1), \\ &-\psi_{L_t}(0,0) + \varepsilon\vartheta_{xx}(0,0) + a(0)\vartheta_x(0,0) - b(0)\vartheta(0,0) = f(0,0), \\ &-\psi_{R_t}(1,0) + \varepsilon\vartheta_{xx}(1,0) + a(1)\vartheta_x(1,0) - b(1)\vartheta(1,0) = f(1,0). \end{aligned}$$

Setting $\varepsilon = 0$ in Eq. (1) yields the reduced problem given by

$$\begin{cases} -(\phi_0)_t + a(x)(\phi_0)_x - b(x)\phi_0 = f(x,t), & \\ \phi_0(x,0) = \vartheta(x), & x \in \Theta_x, \\ \phi_0(1,t) = \psi_R(t), & t \in \Theta_t, \end{cases} \tag{2}$$

which is first order hyperbolic PDE along with initial and right boundary conditions. The reduced problem in Eq. (2) does not satisfy left boundary $x = 0$. Therefore, the solution of Eq. (1) has a boundary layer near $x = 0$. The differential operator in Eq. (1) fulfills the minimum principle.

Lemma 1. *(Continuous minimum principle) Let $\Psi(x,t) \in C^{2,1}(\bar{\Omega})$ and $\Psi(x,t) \geq 0$, for all $(x,t) \in \partial\Omega$. If $P_\varepsilon\Psi(x,t) \leq 0$, for all $(x,t) \in \Omega$, then $\Psi(x,t) \geq 0$, for all $(x,t) \in \bar{\Omega}$.*

Proof. See [6]. □

Lemma 1 provides the stability bound for the solution to Eq. (1), which is stated in the next lemma.

Lemma 2. *For all $\varepsilon > 0$, the solution $\phi(x,t)$ of Eq. (1) satisfies the bound*

$$\|\phi(x,t)\|_{\bar{\Omega}} \leq \|\phi(x,t)\|_{\partial\Omega} + \frac{T}{\beta}\|f\|_{\bar{\Omega}},$$

where $\|f\| = \max\limits_{(x,t)\in\bar{\Omega}} |f(x,t)|$ and T is the final time.

Proof. See [4, 12, 14]. □

The theorem below establishes bounds on the solution's mixed derivatives.

Theorem 1. *For all $j,k > 0$ satisfying $0 \leq j + 3k \leq 4$, the solution $\phi(x,t)$ of Eq. (1) satisfies*

$$\left\| \frac{\partial^{j+k}\phi}{\partial x^j \partial t^k} \right\|_{\bar{\Omega}} \leq C\left(1 + \varepsilon^{-j/2} e^{-x\sqrt{\frac{\beta}{\varepsilon}}}\right),$$

where C is a generic constant independent of ε.

Proof. The proof can be found in [4, 23]. □

The classical bound in Theorem 1 is not strong enough to prove parameter-uniform convergence. To establish stronger error bounds on $\phi(x,t)$ and its derivatives, we split the solution into two:

$$\phi(x,t) = v(x,t) + w(x,t), \qquad \forall (x,t) \in \bar{\Omega},$$

where $v(x,t)$ and $w(x,t)$ represent the smooth and layer components, respectively. The following theorem gives bounds for the mixed derivatives of $\phi(x,t)$.

Theorem 2. *For all $j,k \geq 0$ satisfying $0 \leq j+2k \leq 6$, the smooth and layer components satisfy the bound*

$$\left\| \frac{\partial^{j+k} v}{\partial x^j \partial t^k} \right\|_{\bar{\Omega}} \leq C\left(1+\varepsilon^{1-j/2}\right), \qquad \left\| \frac{\partial^{j+k} w}{\partial x^j \partial t^k} \right\|_{\bar{\Omega}} \leq C\varepsilon^{-j/2} e^{-x\sqrt{\frac{\beta}{\varepsilon}}},$$

where C is a generic constant independent of ε.

Proof. See [17, 23]. □

3 The discrete problem

3.1 Time variable discretization

The time domain $[0,T]$ is divided into uniform mesh points $\Theta_t^{N_t} = \{t_0 = 0, t_n = t_0 + n\Delta t, n = 1,2,\ldots,N_t - 1, t_{N_t} = T\}$ with step length $\Delta t = \frac{T}{N_t}$. Let $\Phi^{n+1} = \Phi^{n+1}(x)$ be the approximation of $\phi(x,t_{n+1})$ at $(n+1)^{th}$ time level. Now, the time variable discrete form of Eq. (1) using Crank-Nicolson method is given by

$$P_\varepsilon \Phi^{n+1} \equiv -\frac{\Phi^{n+1}-\Phi^n}{\Delta t} + \frac{\varepsilon}{2}\left(\Phi_{xx}^{n+1} + \Phi_{xx}^n\right) + \frac{a(x)}{2}\left(\Phi_x^{n+1} + \Phi_x^n\right) - \frac{b(x)}{2}\left(\Phi^{n+1} + \Phi^n\right) = f^{n+\frac{1}{2}} \tag{3}$$

with the following semi-discrete conditions:

$$\begin{cases} \Phi^0(x) = \vartheta(x), & x \in [0,1], \\ \Phi^{n+1}(0) = \psi_L(t_{n+1}), \quad \Phi^{n+1}(1) = \psi_R(t_{n+1}), & n = 0,\cdots,N_t - 1. \end{cases} \tag{4}$$

The semi-discrete differential operator $P_\varepsilon \Phi^{n+1}$ fulfills the semi-discrete minimum principle, which may also be verified as a continuous minimum principle, see [8]. The local truncation error of the time direction is represented by e_{n+1} such that $e_{n+1} = \phi^{n+1} - \Phi^{n+1}$. The following Lemma determines the local truncation error bound.

Lemma 3. *The local truncation error e_{n+1} of the time direction is bounded by*

$$\|e_{n+1}\| \leq C(\Delta t)^3.$$

Proof. Interested readers are referred to [8] for the proof. □

Further, the global error is the sum of all of the local truncation errors at every time level, i.e., $E_n = \sum_{k=0}^{n} e_k$. Next, we establish the global error bound for the discretization of time variable.

Lemma 4. *Assuming Lemma 3, the global error E_n estimate at t_n can be calculated by*

$$\|E_n\| \leq C(\Delta t)^2, \qquad n\Delta t \leq T.$$

Proof. According to Lemma 3, we use the local error up to the n^{th} time step to get

$$\begin{aligned}\|E_n\| = \left\|\sum_{k=0}^{n} e_n\right\| &\leq \|e_0\| + \|e_2\| + \ldots + \|e_n\| \\ &\leq C_1 n(\Delta t)^3 \\ &\leq C_1 (n\Delta t)(\Delta t)^2 \\ &\leq C_1 T(\Delta t)^2 \quad \text{since } n\Delta t \leq T \\ &\leq C(\Delta t)^2, \quad C_1 T = C,\end{aligned}$$

where C is a generic constant independent of ε and Δt. □

The bound in Theorem 2 can be derived using similar arguments.

Lemma 5. *Let $\Phi^{n+1}(x)$ is decomposed as $\Phi^{n+1}(x) = v^{n+1}(x) + w^{n+1}(x)$. Then, for all $j,k \geq 0$ satisfies $0 \leq j+2k \leq 6$, the regular and singular components satisfy*

$$\left|\frac{\partial^j \Phi^{n+1}(x)}{\partial x^j}\right| \leq C\left(1+\varepsilon^{-j/2} e^{-x\sqrt{\frac{\beta}{\varepsilon}}}\right), \quad \left|\frac{\partial^j v^{n+1}(x)}{\partial x^j}\right| \leq C\left(1+\varepsilon^{1-j/2}\right),$$
$$\left|\frac{\partial^j w^{n+1}(x)}{\partial x^j}\right| \leq C\varepsilon^{-j/2} e^{-x\sqrt{\frac{\beta}{\varepsilon}}}.$$

Proof. See [7]. □

3.2 Space variable discretization

We introduce a Shishkin mesh for the space variable discretization that uses cubic spline in compression method to solve the semi-discrete problem in Eqs. (3)-(4). The proposed method incorporates layer-resolving capability directly into the approximation via the compression parameter, allowing accurate, stable, and nearly uniform solutions of boundary turning point problems without relying heavily on special meshes or suffering from parameter sensitivity. At a turning point, the highest derivative term vanishes, causing standard methods to lose accuracy or even stability. For a given positive even integer N_x and ε, the space domain $\Theta_x^{N_x} = [0,1]$ is divided into two sub-domains $[0,\sigma]$ and $(\sigma,1]$ where the transition parameter σ is given by

$$\sigma = \min\left\{0.5, \frac{2\sqrt{\varepsilon}}{\sqrt{\beta}} \ln N_x\right\},$$

where $N_x \geq 4$ be positive even integer. Each sub-domain $[0,\sigma]$ and $(\sigma,1]$ has $N_x/2$ mesh points placed in them. Thus, the Shishkin mesh points given by

$$x_i = \begin{cases} \frac{2i\sigma}{N_x}, & \text{if } i = 0,1,2,\cdots,0.5\times N_x, \\ \sigma + (i-0.5\times N_x)\frac{2(1-\sigma)}{N_x}, & \text{if } i = 0.5\times N_x+1,\cdots,N_x. \end{cases}$$

We divide the space domain $[0,1]$ into elements $[x_i, x_{i+1}]$, $i = 0,1,2,\cdots,N_x-1$, we fully discretize Eq. (3) using spline in compression in space direction. Let $\Phi_i^{n+1} = \Phi(x_i, t_{n+1})$ as the numerical solution of

$\phi^{n+1}(x)$. Let $h_i = x_{i+1} - x_i$, for $i = 1, \cdots, N$ be the non-uniform mesh spacing with $\hat{h}_i = h_{i+1} + h_i$ be the mesh diameter. The spline function $S(x,\tau) = S(x)$ satisfying the differential equation

$$S''(x) + \tau S(x) = \left[S''(x_i) + \tau S(x_i)\right] \frac{(x_{i+1} - x)}{h_{i+1}} + \left[S''(x_{i+1}) + \tau S(x_{i+1})\right] \frac{(x - x_i)}{h_i} \tag{5}$$

in $[x_i, x_{i+1}]$, where $S(x_i) = \Phi_i$, is known as spline in compression function and the parameter $\tau > 0$ is the compression parameter [16]. A function $S(x,\tau) \in C^2[0,1]$ that depends on a parameter τ and interpolates $\phi(x)$ at the mesh points x_i reduces to a cubic spline in $[0,1]$ as $\tau \to 0$. Solving the linear second-order differential equation in Eq. (5) and using the interpolatory conditions $S(x_{i+1}) = \Phi(x_{i+1})$ and $S(x_i) = \Phi(x_i)$, we get

$$\begin{aligned} S(x) = {} & \frac{-h_{i+1}^2}{\mu^2 \sin\mu} \left[M_{i+1} \sin\frac{\mu(x - x_i)}{h_{i+1}} + M_i \sin\frac{\mu(x_{i+1} - x)}{h_{i+1}}\right] \\ & + \frac{h_{i+1}^2}{\mu^2} \left[\frac{(x - x_i)}{h_{i+1}} (M_{i+1} + \frac{\mu^2}{h_{i+1}^2}\Phi_{i+1}) + \frac{(x_{i+1} - x)}{h_{i+1}} (M_i + \frac{\mu^2}{h_{i+1}^2}\Phi_i)\right], \end{aligned} \tag{6}$$

where $\mu = h_{i+1}\sqrt{\tau}$. Differentiating Eq. (6) on the interval $[x_i, x_{i+1}]$ and as $x \to x_i$ $[x_{i+1} - x_i \to h_{i+1}]$ gives

$$S'(x_i^+) = \frac{\Phi_{i+1} - \Phi_i}{h_{i+1}} + \frac{h_{i+1}}{\mu^2} \left[\left(1 - \frac{\mu}{\sin\mu}\right) M_{i+1} + (\mu\cot\mu - 1) M_i\right]. \tag{7}$$

Similarly, on the interval $[x_{i-1}, x_i]$, solving Eq. (5) yields spline function of the form

$$\begin{aligned} S(x) = {} & \frac{-h_i^2}{\mu^2 \sin\mu} \left[M_i \sin\frac{\mu(x - x_{i-1})}{h_i} + M_{i-1} \sin\frac{\mu(x_i - x)}{h_i}\right] \\ & + \frac{h_i^2}{\mu^2} \left[\frac{(x - x_{i-1})}{h_i} \left(M_i + \frac{\mu^2}{h_i^2}\Phi_i\right) + \frac{(x_i - x)}{h_i} \left(M_{i-1} + \frac{\mu^2}{h_i^2}\Phi_{i-1}\right)\right], \end{aligned} \tag{8}$$

where $\mu = h_i\sqrt{\tau}$. Differentiating Eq. (8) on the interval $[x_{i-1}, x_i]$ and as $x \to x_i$ $[x_i - x_{i-1} \to h_i]$ results in

$$S'(x_i^-) = \frac{\Phi_i - \Phi_{i-1}}{h_i} + \frac{h_i}{\mu^2} \left[(1 - \mu\cot\mu) M_i + \frac{h_i}{\mu^2} \left(\frac{\mu}{\sin\mu} - 1\right) M_{i-1}\right]. \tag{9}$$

Equating Eqs. (7) and (9) with the continuity requirement of the first derivatives yields the tridiagonal system of linear equations for a non-uniform grid.

$$\lambda_1 h_i M_{i-1} + \lambda_2 \hat{h}_i M_i + \lambda_1 h_{i+1} M_{i+1} = \frac{\Phi_{i+1} - \Phi_i}{h_{i+1}} - \frac{\Phi_i - \Phi_{i-1}}{h_i}, \tag{10}$$

for $i = 1, 2, \cdots, N_x - 1$ and where $\lambda_1 = \frac{1}{\mu^2}\left(\frac{\mu}{\sin\mu} - 1\right)$ and $\lambda_2 = \frac{1}{\mu^2}(1 - \mu\cot\mu)$. We obtain the condition $\lambda_1 + \lambda_2 = 0.5$ by simply equating the coefficients of $M_i's$ in Eq. (10). After replacing the second-order derivatives $(\Phi_{xx})_i^{n+1}$ in Eq. (4) with the spline in compression function, we have

$$\varepsilon\left(\frac{M_r^n + M_r^{n+1}}{2}\right) = \frac{\Phi_r^{n+1} - \Phi_r^n}{\Delta t} - a_r\left(\frac{(\Phi_x)_r^n + (\Phi_x)_r^{n+1}}{2}\right) + b_r\left(\frac{\Phi_r^n + \Phi_r^{n+1}}{2}\right) + \frac{f_r^{n+1} + f_r^n}{2}, \tag{11}$$

where $(\Phi_{xx})_r^{n+1} = M_r^{n+1}$ and $r = i, i \pm 1$. At $(n+1)^{th}$ time level, Eq. (10) becomes

$$\lambda_1 h_i M_{i-1}^{n+1} + \lambda_2 \hat{h}_i M_i^{n+1} + \lambda_1 h_{i+1} M_{i+1}^{n+1} = \frac{\Phi_{i+1}^{n+1} - \Phi_i^{n+1}}{h_{i+1}} - \frac{\Phi_i^{n+1} - \Phi_{i-1}^{n+1}}{h_i}, \tag{12}$$

for $1 \le i \le N_x - 1,\ \ 0 \le n \le N_t - 1$. Similarly, at n^{th} time level, we have

$$\lambda_1 h_i M_{i-1}^{n} + \lambda_2 \hat{h}_i M_i^{n} + \lambda_1 h_{i+1} M_{i+1}^{n} = \frac{\Phi_{i+1}^{n} - \Phi_i^{n}}{h_{i+1}} - \frac{\Phi_i^{n} - \Phi_{i-1}^{n}}{h_i}, \tag{13}$$

for $1 \le i \le N_x - 1,\ \ 0 \le n \le N_t$. Adding Eqs. (12) and (13) and multiplying the result by $\frac{\varepsilon}{2}$, we obtain

$$\begin{aligned}
&\lambda_1 h_i \varepsilon \left(\frac{M_{i-1}^n + M_{i-1}^{n+1}}{2} \right) + \lambda_2 \hat{h}_i \varepsilon \left(\frac{M_i^n + M_i^{n+1}}{2} \right) + \lambda_1 h_{i+1} \varepsilon \left(\frac{M_{i+1}^n + M_{i+1}^{n+1}}{2} \right) \\
&= \frac{\varepsilon}{2h_{i+1}} \left(\Phi_{i+1}^n - \Phi_i^n + \Phi_{i+1}^{n+1} - \Phi_i^{n+1} \right) - \frac{\varepsilon}{2h_i} \left(\Phi_i^n - \Phi_{i-1}^n + \Phi_i^{n+1} - \Phi_{i-1}^{n+1} \right).
\end{aligned} \tag{14}$$

From Eq. (11), we have

$$\begin{aligned}
\varepsilon \left(\frac{M_{i-1}^n + M_{i-1}^{n+1}}{2} \right) &= \frac{\Phi_{i-1}^{n+1} - \Phi_{i-1}^n}{\Delta t} - a_{i-1} \left(\frac{(\Phi_x)_{i-1}^n + (\Phi_x)_{i-1}^{n+1}}{2} \right) + b_{i-1} \left(\frac{\Phi_{i-1}^n + \Phi_{i-1}^{n+1}}{2} \right) + f_{i-1}^{n+\frac{1}{2}}, \\
\varepsilon \left(\frac{M_{i}^n + M_{i}^{n+1}}{2} \right) &= \frac{\Phi_{i}^{n+1} - \Phi_{i}^n}{\Delta t} - a_{i} \left(\frac{(\Phi_x)_{i}^n + (\Phi_x)_{i}^{n+1}}{2} \right) + b_{i} \left(\frac{\Phi_{i}^n + \Phi_{i}^{n+1}}{2} \right) + f_{i}^{n+\frac{1}{2}}, \\
\varepsilon \left(\frac{M_{i+1}^n + M_{i+1}^{n+1}}{2} \right) &= \frac{\Phi_{i+1}^{n+1} - \Phi_{i+1}^n}{\Delta t} - a_{i+1} \left(\frac{(\Phi_x)_{i+1}^n + (\Phi_x)_{i+1}^{n+1}}{2} \right) + b_{i+1} \left(\frac{\Phi_{i+1}^n + \Phi_{i+1}^{n+1}}{2} \right) + f_{i+1}^{n+\frac{1}{2}}.
\end{aligned} \tag{15}$$

Plugging Eq. (15) in Eq. (14) and some rearrangement yields

$$\begin{aligned}
&\frac{-\varepsilon}{2h_i} \Phi_{i-1}^{n+1} - \frac{\lambda_1 h_i a_{i-1}}{2} (\Phi_x)_{i-1}^{n+1} + \lambda_1 h_i c_{i-1} \Phi_{i-1}^{n+1} + \frac{\varepsilon \hat{h}_i}{2h_i h_{i+1}} \Phi_i^{n+1} - \frac{\lambda_2 \hat{h}_i a_i}{2} (\Phi_x)_i^{n+1} + \lambda_2 \hat{h}_i c_i \Phi_i^{n+1} \\
&- \frac{\varepsilon}{2h_{i+1}} \Phi_{i+1}^{n+1} - \frac{\lambda_1 h_{i+1} a_{i+1}}{2} (\Phi_x)_{i+1}^{n+1} + \lambda_1 h_{i+1} c_{i+1} \Phi_{i+1}^{n+1} \\
&= \frac{\varepsilon}{2h_i} \Phi_{i-1}^{n} + \frac{\lambda_1 h_i a_{i-1}}{2} (\Phi_x)_{i-1}^{n} - \lambda_1 h_i d_{i-1} \Phi_{i-1}^{n} - \frac{\varepsilon \hat{h}_i}{2h_i h_{i+1}} \Phi_i^{n} + \frac{\lambda_2 \hat{h}_i a_i}{2} (\Phi_x)_i^{n} - \lambda_2 \hat{h}_i d_i \Phi_i^{n} \\
&\quad + \frac{\varepsilon}{2h_{i+1}} \Phi_{i+1}^{n} + \frac{\lambda_1 h_{i+1} a_{i+1}}{2} (\Phi_x)_{i+1}^{n} - \lambda_1 h_{i+1} d_{i+1} \Phi_{i+1}^{n} - \frac{\lambda_1 h_i}{2} f_{i-1}^{n+\frac{1}{2}} - \lambda_2 \frac{\hat{h}_i}{2} f_i^{n+\frac{1}{2}} - \frac{\lambda_1 h_{i+1}}{2} f_{i+1}^{n+\frac{1}{2}},
\end{aligned} \tag{16}$$

where $c_{i-1} = \frac{b_{i-1}}{2} + \frac{1}{\Delta t}$, $c_i = \frac{b_i}{2} + \frac{1}{\Delta t}$, $c_{i+1} = \frac{b_{i+1}}{2} + \frac{1}{\Delta t}$, $d_{i-1} = \frac{b_{i-1}}{2} - \frac{1}{\Delta t}$, $d_i = \frac{b_i}{2} - \frac{1}{\Delta t}$ and $d_{i+1} = \frac{b_{i+1}}{2} - \frac{1}{\Delta t}$. The first derivatives are approximated by the following in order to achieve the overall second-order approximation in the space direction.

$$\begin{aligned}
(\Phi_x)_i^{n+1} &\cong \frac{-h_{i+1}}{h_i \hat{h}_i} \Phi_{i-1}^{n+1} + \frac{h_{i+1} - h_i}{h_i h_{i+1}} \Phi_i^{n+1} + \frac{h_i}{h_{i+1} \hat{h}_i} \Phi_{i+1}^{n+1} \\
(\Phi_x)_{i+1}^{n+1} &\cong \frac{h_{i+1}}{h_i \hat{h}_i} \Phi_{i-1}^{n+1} - \frac{\hat{h}_i}{h_i h_{i-1}} \Phi_i^{n+1} + \frac{h_{i-1} + 2h_i}{h_i \hat{h}_i} \Phi_{i+1}^{n+1} \\
(\Phi_x)_{i-1}^{n+1} &\cong \frac{-(h_{i+1} + 2h_i)}{h_i \hat{h}_i} \Phi_{i-1}^{n+1} + \frac{\hat{h}_i}{h_i h_{i+1}} \Phi_i^{n+1} - \frac{h_i}{h_{i+1} \hat{h}_i} \Phi_{i+1}^{n+1}.
\end{aligned} \tag{17}$$

Inserting Eq. (17) in Eq. (16) and rearranging the terms, we obtain the following discrete numerical scheme

$$P_\varepsilon \Phi^{N_x,N_t} \equiv \mathscr{L}_i^- \Phi_{i-1}^{n+1} + \mathscr{L}_i^c \Phi_i^{n+1} + \mathscr{L}_i^+ \Phi_{i+1}^{n+1} = \mathscr{R}_i^- \Phi_{i-1}^n + \mathscr{R}_i^c \Phi_i^n + \mathscr{R}_i^+ \Phi_{i+1}^n + \mathscr{F}_i, \tag{18}$$

with discrete conditions

$$\begin{cases} \Phi_i^0 = \vartheta_i, & 1 \le i \le N_x - 1, \\ \Phi_0^{n+1} = \psi_L^{n+1}, \quad \Phi_N^{n+1} = \psi_R^{n+1}, & 0 \le n \le N_t - 1, \end{cases} \tag{19}$$

where

$$\begin{aligned} \mathscr{L}_i^- &= \frac{\varepsilon}{h_i \hat{h}_i} - \frac{\lambda_1 (h_{i+1} + 2h_i)}{\hat{h}_i^2} a_{i-1} - \frac{\lambda_2 h_{i+1}}{h_i \hat{h}_i} a_i + \frac{\lambda_1 h_{i+1}^2}{h_i \hat{h}_i^2} a_{i+1} - \frac{\lambda_1 h_i}{\hat{h}_i} c_{i-1}, \\ \mathscr{L}_i^c &= \frac{-\varepsilon}{h_i h_{i+1}} + \frac{\lambda_1}{h_{i+1}} a_{i-1} + \frac{\lambda_2 (h_{i+1} - h_i)}{h_i h_{i+1}} a_i - \frac{\lambda_1}{h_i} a_{i+1} - \lambda_2 c_i, \\ \mathscr{L}_i^+ &= \frac{\varepsilon}{h_{i+1} \hat{h}_i} - \frac{\lambda_1 h_i^2}{h_{i+1} \hat{h}_i^2} a_{i-1} + \frac{\lambda_2 h_i}{h_{i+1} \hat{h}_i} a_i + \frac{\lambda_1 (h_i + 2h_{i+1})}{\hat{h}_i^2} a_{i+1} - \frac{\lambda_1 h_{i+1}}{\hat{h}_i} c_{i+1}, \end{aligned} \tag{20}$$

and

$$\begin{aligned} \mathscr{R}_i^- &= \frac{-\varepsilon}{h_i \hat{h}_i} + \frac{\lambda_1 (h_{i+1} + 2h_i)}{\hat{h}_i^2} a_{i-1} + \frac{\lambda_2 h_{i+1}}{h_i \hat{h}_i} a_i - \frac{\lambda_1 h_{i+1}^2}{h_i \hat{h}_i^2} a_{i+1} + \frac{\lambda_1 h_i}{\hat{h}_i} d_{i-1}, \\ \mathscr{R}_i^c &= \frac{\varepsilon}{h_i h_{i+1}} - \frac{\lambda_1}{h_{i+1}} a_{i-1} - \frac{\lambda_2 (h_{i+1} - h_i)}{h_{i+1} h_i} a_i + \frac{\lambda_1}{h_i} a_{i+1} + \lambda_2 d_i, \\ \mathscr{R}_i^+ &= \frac{-\varepsilon}{h_{i+1} \hat{h}_i} + \frac{\lambda_1 h_i^2}{h_{i+1} \hat{h}_i^2} a_{i-1} - \frac{\lambda_2 h_i}{h_{i+1} \hat{h}_i} a_i - \frac{\lambda_1 (h_i + 2h_{i+1})}{\hat{h}_i^2} a_{i+1} + \frac{\lambda_1 h_{i+1}}{\hat{h}_i} d_{i+1}, \\ \mathscr{F}_i &= \frac{\lambda_1 h_i}{\hat{h}_i} f_{i-1}^{n+\frac{1}{2}} + \lambda_2 f_i^{n+\frac{1}{2}} + \frac{\lambda_1 h_{i+1}}{\hat{h}_i} f_{i+1}^{n+\frac{1}{2}}. \end{aligned} \tag{21}$$

The resulting tridiagonal linear system is efficiently solved using the Thomas algorithm.

Remark 1. *The proposed cubic spline based compression method provides a second-order uniformly convergent numerical solution for* $\lambda_1 = \frac{1}{12}, \lambda_2 = \frac{5}{12}$.

4 Error analysis of discrete problem

The present section first establishes stability, then consistency, and lastly convergence analysis using truncation error for the current method. Thomas algorithm for the solution of Eq. (18) is numerically stable [24] if $|\mathscr{L}_i^-| > 0, \;\; |\mathscr{L}_i^+| > 0, \;\; |\mathscr{L}_i^c| \ge |\mathscr{L}_i^-| + |\mathscr{L}_i^+|, \;\; |\mathscr{L}_i^-| \le |\mathscr{L}_i^+|$. For $i \in (0,1]$, we assume

$a_{i-1}, a_i, a_{i+1} \geq \gamma > 0$ and $c_{i-1}, c_i, c_{i+1} \geq \eta > 0$. We have

$$
\begin{aligned}
|\mathscr{L}_i^-| &= \left| \frac{\varepsilon}{h_i \hat{h}_i} - \frac{\lambda_1 (h_{i+1} + 2h_i)}{\hat{h}_i^2} a_{i-1} - \frac{\lambda_2 h_{i+1}}{h_i \hat{h}_i} a_i + \frac{\lambda_1 h_{i+1}^2}{h_i \hat{h}_i^2} a_{i+1} - \frac{\lambda_1 h_i}{\hat{h}_i} c_{i-1} \right|, \\
&= \left| \frac{\varepsilon}{h_i \hat{h}_i} - \frac{(\lambda_1 + \lambda_2) h_i h_{i+1} \gamma + 2\lambda_1 h_i^2 \gamma + (\lambda_2 - \lambda_1) h_{i+1}^2 \gamma}{h_i \hat{h}_i^2} + \frac{\lambda_1 h_i \eta}{\hat{h}_i} \right|, \qquad \text{since } \lambda_1 + \lambda_2 = \frac{1}{2}, \\
&= \left| \frac{\varepsilon}{h_i \hat{h}_i} + \frac{(4\lambda_1 - 1) h_{i+1} \gamma - 4\lambda_1 h_i \gamma}{2 h_i \hat{h}_i} + \frac{\lambda_1 h_i \eta}{\hat{h}_i} \right|, \qquad \text{taking } \lambda_1 = \frac{1}{12} \\
&= \left| \frac{\varepsilon}{h_i \hat{h}_i} - \frac{(2h_{i+1} - h_i)\gamma}{6 h_i \hat{h}_i} + \frac{h_i \eta}{12 \hat{h}_i} \right| > 0,
\end{aligned}
$$

provided that $12\varepsilon + h_i^2 \eta > 2(2h_{i+1} + h_i)\gamma$, for all ε, assuming the lower bound of $\gamma = a(x) = x^p$ and $\eta = \frac{b(x)}{2} + \frac{1}{\Delta t} = \frac{b(x)}{2} + N_t$. Similarly, we have

$$
\begin{aligned}
|\mathscr{L}_i^+| &= \left| \frac{\varepsilon}{h_{i+1} \hat{h}_i} - \frac{\lambda_1 h_i^2}{h_{i+1} \hat{h}_i^2} a_{i-1} + \frac{\lambda_2 h_i}{h_{i+1} \hat{h}_i} a_i + \frac{\lambda_1 (h_i + 2h_{i+1})}{\hat{h}_i^2} a_{i+1} - \frac{\lambda_1 h_{i+1}}{\hat{h}_i} c_{i+1} \right|, \\
&= \left| \frac{\varepsilon}{h_{i+1} \hat{h}_i} + \frac{(\lambda_2 - \lambda_1) h_i^2 \gamma + (\lambda_1 + \lambda_2) h_i h_{i+1} \gamma + 2\lambda_1 h_{i+1}^2 \gamma}{h_{i+1} \hat{h}_i^2} - \frac{\lambda_1 h_{i+1} \eta}{\hat{h}_i} \right|, \qquad \text{since } \lambda_1 + \lambda_2 = \frac{1}{2}, \\
&= \left| \frac{\varepsilon}{h_{i+1} \hat{h}_i} + \frac{(1 - 4\lambda_1) h_i \gamma + 4\lambda_1 h_{i+1} \gamma}{2 h_{i+1} \hat{h}_i} - \frac{\lambda_1 h_{i+1} \eta}{\hat{h}_i} \right|, \qquad \text{taking } \lambda_1 = \frac{1}{12} \\
&= \left| \frac{\varepsilon}{h_{i+1} \hat{h}_i} + \frac{(h_{i+1} + 2h_i)\gamma}{6 h_{i+1} \hat{h}_i} - \frac{h_{i+1} \eta}{12 \hat{h}_i} \right| > 0,
\end{aligned}
$$

which is always true since the mesh parameters and the coefficients are bounded. Finally, we have

$$
\begin{aligned}
|\mathscr{L}_i^c| &= \left| \frac{-\varepsilon}{h_i h_{i+1}} + \frac{\lambda_1}{h_{i+1}} a_{i-1} + \frac{\lambda_2 (h_{i+1} - h_i)}{h_i h_{i+1}} a_i - \frac{\lambda_1}{h_i} a_{i+1} - \lambda_2 c_i \right| \\
&= \left| \frac{-\varepsilon}{h_i h_{i+1}} - \frac{2(h_i - h_{i+1})\gamma}{6 h_i h_{i+1}} - \frac{5}{12}\eta \right|, \\
&= \frac{\varepsilon}{h_i h_{i+1}} + \frac{2(h_i - h_{i+1})\gamma}{6 h_i h_{i+1}} + \frac{5}{12}\eta, \\
|\mathscr{L}_i^-| + |\mathscr{L}_i^+| &= \frac{\varepsilon}{h_i \hat{h}_i} - \frac{(2h_{i+1} + h_i)\gamma}{6 h_i \hat{h}_i} + \frac{h_i \eta}{12 \hat{h}_i} + \frac{\varepsilon}{h_{i+1} \hat{h}_i} + \frac{(h_{i+1} + 2h_i)\gamma}{6 h_{i+1} \hat{h}_i} - \frac{h_{i+1} \eta}{12 \hat{h}_i} \\
&= \frac{\varepsilon}{h_i h_{i+1}} + \frac{(h_i - h_{i+1})\gamma}{3 h_i h_{i+1}} + \frac{(h_i - h_{i+1})\eta}{12 \hat{h}_i}.
\end{aligned}
$$

From the above derivation, it easily follows that $|\mathscr{L}_i^c| \geq |\mathscr{L}_i^-| + |\mathscr{L}_i^+|$ for all mesh parameters and bounded coefficients. The above conditions guarantee that the system of linear equations is diagonally dominant for sufficiently small h_i. This implies that the tridiagonal system is non-singular and has a unique solution. Therefore, the derived method for solving the tri-diagonal system given by Eq. (18) is stable. In the following step, we use the truncation error to show consistency and convergence analysis.

To do this, we consider space direction error analysis only at $(n+1)$. For the sake of simplicity, considering space direction in Eq. (20) at $(n+1)$ time level only and the function in the right hand side in Eq. (21) only, the local truncation error is given by

$$\mathscr{E}_i = \mathscr{L}_i^{-}\Phi_{i-1} + \mathscr{L}_i^{c}\Phi_i + \mathscr{L}_i^{+}\Phi_{i+1} - \mathscr{F}_i, \tag{22}$$

where $\mathscr{F}_i = -\frac{\lambda_1 h_i}{\hat{h}_i} f_{i-1} - \lambda_2 f_i - \frac{\lambda_1 h_{i+1}}{\hat{h}_i} f_{i+1}$. From Eq. (3), we get

$$\begin{aligned}
&\varepsilon(\Phi_{xx})_{i-1} + a_{i-1}(\Phi_x)_{i-1} - b_{i-1}\Phi_{i-1} = f_{i-1},\\
&\varepsilon(\Phi_{xx})_{i} + a_{i}(\Phi_x)_{i} - b_{i}\Phi_{i} = f_{i},\\
&\varepsilon(\Phi_{xx})_{i+1} + a_{i+1}(\Phi_x)_{i+1} - b_{i+1}\Phi_{i+1} = f_{i+1}.
\end{aligned} \tag{23}$$

Multiplying Eq. (23) with different expressions, we obtain

$$\begin{aligned}
&-\frac{\lambda_1 h_i}{\hat{h}_i}\left[\varepsilon(\Phi_{xx})_{i-1} + a_{i-1}(\Phi_x)_{i-1} - b_{i-1}\Phi_{i-1}\right] = -\frac{\lambda_1 h_i}{\hat{h}_i} f_{i-1},\\
&-\lambda_2\left[\varepsilon(\Phi_{xx})_{i} + a_{i}(\Phi_x)_{i} - b_{i}\Phi_{i}\right] = -\lambda_2 f_i,\\
&-\frac{\lambda_1 h_{i+1}}{\hat{h}_i}\left[\varepsilon(\Phi_{xx})_{i+1} + a_{i+1}(\Phi_x)_{i+1} - b_{i+1}\Phi_{i+1}\right] = -\frac{\lambda_1 h_{i+1}}{\hat{h}_i} f_{i+1}.
\end{aligned} \tag{24}$$

Plugging Eq. (24) in Eq. (22), we have

$$\begin{aligned}
\mathscr{E}_i = {} & \mathscr{L}_i^{-}\Phi_{i-1} + \mathscr{L}_i^{c}\Phi_i + \mathscr{L}_i^{+}\Phi_{i+1} + \frac{\varepsilon\lambda_1 h_i}{\hat{h}_i}(\Phi_{xx})_{i-1} + \frac{\lambda_1 h_i a_{i-1}}{\hat{h}_i}(\Phi_x)_{i-1} - \frac{\lambda_1 h_i b_{i-1}}{\hat{h}_i}\Phi_{i-1}\\
& + \varepsilon\lambda_2(\Phi_{xx})_i + \lambda_2 a_i(\Phi_x)_i - \lambda_2 b_i\Phi_i + \frac{\varepsilon\lambda_1 h_{i+1}}{\hat{h}_i}(\Phi_{xx})_{i+1} + \frac{\lambda_1 h_{i+1} a_{i+1}}{\hat{h}_i}(\Phi_x)_{i+1} - \frac{\lambda_1 h_{i+1} b_{i+1}}{\hat{h}_i}\Phi_{i+1}.
\end{aligned} \tag{25}$$

By replacing the Taylor's expansion of terms $\Phi_{i-1}, \Phi_{i+1}, (\Phi_x)_{i-1}, (\Phi_x)_{i+1}, (\Phi_{xx})_{i-1}$, and $(\Phi_{xx})_{i+1}$ in Eq. (25) in space direction only up to the third-order derivatives, the error at the $(n+1)^{th}$ time level is expressed as follows:

$$\mathscr{E}_i = \mathscr{E}_{0,i}\Phi_i + \mathscr{E}_{1,i}(\Phi_x)_i + \mathscr{E}_{2,i}(\Phi_{xx})_i + \mathscr{E}_{3,i}(\Phi_{xxx})_i, \tag{26}$$

where

$$\begin{aligned}
\mathscr{E}_{0,i} &= \mathscr{L}_i^{-} + \mathscr{L}_i^{c} + \mathscr{L}_i^{+} - \frac{\lambda_1 h_i}{\hat{h}_i} b_{i-1} - \lambda_2 b_i - \frac{\lambda_1 h_{i+1}}{\hat{h}_i} b_{i+1},\\
\mathscr{E}_{1,i} &= h_{i+1}\mathscr{L}_i^{+} - h_i\mathscr{L}_i^{-} + \frac{\lambda_1 h_i}{\hat{h}_i} a_{i-1} + \frac{\lambda_1 h_i^2}{\hat{h}_i} b_{i-1} + \lambda_2 a_i + \frac{\lambda_1 h_{i+1}}{\hat{h}_i} a_{i+1} - \frac{\lambda_1 h_{i+1}^2}{\hat{h}_i} b_{i+1},\\
\mathscr{E}_{2,i} &= \frac{h_i^2}{2}\mathscr{L}_i^{-} + \frac{h_{i+1}^2}{2}\mathscr{L}_i^{+} - \frac{\lambda_1 h_i^2}{\hat{h}_i} a_{i-1} - \frac{\lambda_1 h_i^3}{2\hat{h}_i} b_{i-1} + \varepsilon(\lambda_1 + \lambda_2) + \frac{\lambda_1 h_{i+1}^2}{\hat{h}_i} a_{i+1} - \frac{\lambda_1 h_{i+1}^3}{2\hat{h}_i} b_{i+1},\\
\mathscr{E}_{3,i} &= \frac{h_{i+1}^3}{6}\mathscr{L}_i^{+} - \frac{h_i^3}{6}\mathscr{L}_i^{-} + \frac{\lambda_1 h_i^3}{2\hat{h}_i} a_{i-1} + \frac{\lambda_1 h_i^4}{6\hat{h}_i} b_{i-1} + \varepsilon\lambda_1(h_{i+1} - h_i) + \frac{\lambda_1 h_{i+1}^3}{2\hat{h}_i} a_{i+1} - \frac{\lambda_1 h_{i+1}^4}{6\hat{h}_i} b_{i+1}.
\end{aligned} \tag{27}$$

From Eq. (18), we have

$$\mathscr{L}_i^- + \mathscr{L}_i^c + \mathscr{L}_i^+ = \frac{\lambda_1 h_i}{\hat{h}_i} b_{i-1} + \lambda_2 b_i + \frac{\lambda_1 h_{i+1}}{\hat{h}_i} b_{i+1}. \tag{28}$$

Using Eq. (20) in first equation of Eq. (27), we obtain $\mathscr{E}_{0,i} = 0$. It is straightforward that using $\mathscr{L}_i^+$ and $\mathscr{L}_i^-$ in Eq. (27), we obtain $\mathscr{E}_{1,i} = 0$ and $\mathscr{E}_{2,i} = 0$. The last expression in Eq. (27) can be simplified as

$$\mathscr{E}_{3,i} = \varepsilon(h_{i+1} - h_i)\left(\lambda_1 - \frac{1}{6}\right) + \frac{1}{6}(\lambda_1 h_i^2 a_{i-1} - \lambda_2 h_i h_{i+1} a_i + \lambda_1 h_{i+1}^2 a_{i+1}). \tag{29}$$

The truncation error in Eq. (26) becomes

$$\mathscr{E}_i = \left[\varepsilon(h_{i+1} - h_i)\left(\lambda_1 - \frac{1}{6}\right) + \frac{1}{6}(\lambda_1 h_i^2 a_{i-1} - \lambda_2 h_i h_{i+1} a_i + \lambda_1 h_{i+1}^2 a_{i+1})\right](\Phi_{xxx})_i. \tag{30}$$

As $h_i, h_{i+1} \to 0$, then the truncation error $\mathscr{E}_i \to 0$, demonstrating the consistency of the method.

Theorem 3. *We consider $\phi(x_i)$ as the continuous solution and Φ_i as the discrete solution. The space direction error satisfies the following bound:*

$$\sup_{0<\varepsilon\ll 1} |\Phi_i - \phi(x_i)| \leq C N_x^{-2} \ln^2 N_x,$$

where C is a generic constant independent of ε and N_x.

Proof. To prove this theorem, we look at two cases:
Case i): For $\sigma = \frac{1}{2}$, the mesh is uniform, and the classical approach is used to carry out the error analysis. The mesh size is $h_i = \frac{1}{N_x}$ and $\frac{2\sqrt{\varepsilon}}{\sqrt{\sigma}} \geq \frac{1}{2}$. This gives $\varepsilon^{-1/2} \leq C \ln N_x$. Since $a(x) = x^p$, $a(0) = 0, a(x) > 0$ for $x > 0$, at $x = 0$, there is a turning point and $x = 0$ is left boundary. Therefore, we assume $a(x) \geq 0$ on $[0,1]$ and $a(x) \geq \alpha > 0$ only on $[\sigma, 1]$, where σ is the transition point of Shishkin mesh. Use $h_i := h_{i+1} = \frac{1}{N_x}$ in Eq. (30), the truncation error becomes

$$|\mathscr{E}_i| \leq C N_x^{-2} (\Phi_{xxx})_i,$$

where $C = \frac{1}{6}(2\lambda_1 - \lambda_2)\alpha$. Using a classical bound in Theorem (1), we obtain the bound

$$|\mathscr{E}_i| \leq C N_x^{-2} \ln^3 N_x, \qquad \text{for} \quad i = 1, 2, \ldots, N_x - 1.$$

Case ii): For $\sigma = \frac{2\sqrt{\varepsilon}}{\sqrt{\beta}} \ln N_x$, the mesh is piecewise uniform, with mesh size $h_i = \frac{2\sigma}{N_x}$ in sub-domain $(0, \sigma)$ and $h_i = \dfrac{2(1-\sigma)}{N_x}$ in sub-domain $(\sigma, 1)$. Using $h_i = h_{i+1} = \frac{2(1-\sigma)}{N_x} \leq C N_x^{-1}$ in Eq. (30) on sub-domain $(\sigma, 1)$, we get

$$|\mathscr{E}_i| \leq C N_x^{-2} (\Phi_{xxx}),$$

where $C = \frac{1}{6}(2\lambda_1 - \lambda_2)\alpha$ and $a(x) \geq \alpha > 0$ only on $[0,1]$. Applying the fact $\sqrt{\varepsilon} \leq C N_x^{-1}$ and using bound in Theorem 2 for regular component, we obtain

$$|\mathscr{E}_i| \leq C N_x^{-2}(1 + \sqrt{\varepsilon}) \leq C N_x^{-2}, \qquad i = \frac{N_x}{2}, \ldots, N_x - 1. \tag{31}$$

On sub-domain $(0,\sigma)$, using $h_i = h_{i+1} = \frac{2\sigma}{N_x}$ in Eq. (30), we have

$$|\mathscr{E}_i| \leq Ch_i^2(\Phi_{xxx})_i,$$

where $C = \frac{1}{6}(2\lambda_1 - \lambda_2)\alpha$ and $a(x) \geq \alpha > 0$ only on $[0,1]$. Using the fact that $e^{-x_i\sqrt{\frac{\beta}{\varepsilon}}} \leq e^{-\sigma\sqrt{\frac{\beta}{\varepsilon}}} = e^{-2\frac{\sqrt{\varepsilon}}{\sqrt{\beta}}\ln N_x\sqrt{\frac{\beta}{\varepsilon}}} = e^{-2\ln N_x} = N_x^{-2}$ and applying the bound in Theorem 2 for singular component, we have

$$|\mathscr{E}_i| \leq (C\sqrt{\varepsilon}N_x^{-1}\ln N_x)^2\varepsilon^{-3/2}e^{-x_i\sqrt{\frac{\beta}{\varepsilon}}} \leq CN_x^{-2}\ln^2 N_x, \tag{32}$$

assuming $\sqrt{\varepsilon} \leq CN_x^{-1}$ [see [7], page 3]. Combining Eqs. (31) and (32) yields the truncation error

$$|\mathscr{E}_i| \leq CN_x^{-2}\ln^2 N_x, \qquad 1 \leq i \leq N_x - 1,$$

as required. □

Theorem 4. *Assume $\phi(x_i,t_n)$ be the continuous solution and Φ_i^n be the discrete solution. Then, the parameter-uniform error bound is given by*

$$|\Phi_i^n - \phi(x_i,t_n)| \leq C(\Delta t^2 + N_x^{-2}\ln^2 N_x),$$

where C is a generic constant regardless of ε, N_x and N_t.

Proof. Combining the global time discretization bound established in Lemma 4 with the global space discretization bound from Theorem 3, we obtain the result. □

5 Numerical illustrations

For numerical experimentation and simulation, we used MATLAB R2020a software packages. All numerical results and graphs are computed by taking parameters $\lambda_1 = \frac{1}{12}; \lambda_2 = \frac{5}{12}$.

Example 1. Firstly, we consider $a_0(x) = 1, b(x) = 1, \vartheta(x) = (1-x)^2, \psi_L(t) = 1 + t^2, \psi_R(t) = 0$ and $f(x,t) = x^2 - 1$ in Eqs. (1)-(2).

Example 2. Secondly, we consider $a_0(x) = 1, b(x) = (x+p), f(x,t) = pe^{-t}(x^2-1), \vartheta(x) = (1-x)^2, \psi_L(t) = 1 + t^2$ and $\psi_R(t) = 0$ in Eqs. (1)-(2).

Due to the lack of an analytical solution for Example 1 and Example 2, the double mesh concept is utilized to compute the maximum point-wise errors and numerical rate of convergence via

$$E_\varepsilon^{N_x,N_t} = \max_{0\leq i\leq N_x; 0\leq n\leq N_t} \left|\Phi^{N_x,N_t}(x_i,t_n) - \Phi^{2N_x,2N_t}(x_{2i},t_{2n})\right| \quad \text{and} \quad r_\varepsilon^{N_x,N_t} = \log_2\left(\frac{E_\varepsilon^{N_x,N_t}}{E_\varepsilon^{2N_x,2N_t}}\right),$$

where $\Phi^{N_x,N_t}(x_i,t_n)$ is the numerical solution whereas $\Phi^{2N_x,2N_t}(x_{2i},t_{2n})$ is the double mesh numerical solution.

Table 1: Comparisons of our E^{N_x}, r^{N_x} with [17] and [9] for Example 1 taking $N_x = N_t$ and $p = 2$

$\varepsilon \downarrow$	$N_x = 16$	32	64	128	256	512	1024
Our result							
10^{-2}	1.1234e-3	1.5963e-4	3.2204e-5	8.2609e-6	4.7519e-6	2.5577e-6	1.2161e-6
10^{-4}	2.8602e-3	8.6909e-4	2.5430e-4	6.7496e-5	1.7964e-5	4.5147e-6	1.1113e-6
10^{-6}	3.3531e-3	9.8713e-4	2.8377e-4	8.5340e-5	2.3703e-5	6.4176e-6	1.6988e-6
10^{-8}	3.3699e-3	1.0140e-3	2.8412e-4	8.6841e-5	2.3857e-5	6.5249e-6	1.7427e-6
10^{-10}	3.3713e-3	1.0164e-3	2.8546e-4	8.6870e-5	2.3903e-5	6.5603e-6	1.7426e-6
10^{-12}	3.3715e-3	1.0167e-3	2.8559e-4	8.6872e-5	2.3914e-5	6.5633e-6	1.7437e-6
10^{-14}	3.3715e-3	1.0167e-3	2.8561e-4	8.6872e-5	2.3916e-5	6.5636e-6	1.7438e-6
10^{-16}	3.3715e-3	1.0167e-3	2.8561e-4	8.6872e-5	2.3916e-5	6.5636e-6	1.7438e-6
10^{-18}	3.3715e-3	1.0167e-3	2.8561e-4	8.6872e-5	2.3916e-5	6.5636e-6	1.7438e-6
10^{-20}	3.3715e-3	1.0167e-3	2.8561e-4	8.6872e-5	2.3916e-5	6.5636e-6	1.7438e-6
E^{N_x}	3.3715e-3	1.0167e-3	2.8561e-4	8.6872e-5	2.3916e-5	6.5636e-6	1.7438e-6
r^{N_x}	1.7295	1.8318	1.7171	1.8609	1.8654	1.9123	-
CPU(sec)	0.127256	0.273973	0.883163	3.275063	12.488072	69.609164	299.47704
In [17]							
E^{N_x}	-	1.0995e-2	6.2049e-3	3.3378e-3	1.7433e-3	8.9528e-4	-
r^{N_x}	-	0.8254	0.8945	0.9371	0.9614	-	-
In [9]							
E^{N_x}	-	7.6969e-3	3.6222e-3	1.3930e-3	4.7347e-4	1.5194e-4	4.7125e-5
r^{N_x}	-	1.0874	1.3786	1.5569	1.6398	1.6890	-

Example 3. Lastly, we consider Eqs. (1)-(2) with $a_0(x) = 1, b(x) = 1$. The functions $\psi_L(t) = 0, \psi_R(t) = 0$, $\vartheta(x) = 1 - x + xe^{-1/\sqrt{\varepsilon}} - e^{-x/\sqrt{\varepsilon}}$ and $f(x,t) = e^{-t}[x^p(e^{-1/\sqrt{\varepsilon}} + \frac{e^{-x/\sqrt{\varepsilon}}}{\sqrt{\varepsilon}} - 1) - e^{-x/\sqrt{\varepsilon}}]$ are chosen from the analytical solution

$$\phi(x,t) = e^{-t}\left(1 - e^{\frac{-x}{\sqrt{\varepsilon}}}\right) + xe^{-t}\left(e^{\frac{-1}{\sqrt{\varepsilon}}} - 1\right).$$

As we have the analytical solution for the Example 3, we use the following formula to compute the maximum point-wise errors:

$$E_\varepsilon^{N_x,N_t} = \max_{0\le i\le N_x;0\le n\le N_t} \left|\Phi^{N_x,N_t}(x_i,t_n) - \phi^{N_x,N_t}(x_i,t_n)\right| \quad \text{and} \quad r_\varepsilon^{N_x,N_t} = \log_2\left(\frac{E_\varepsilon^{N_x,N_t}}{E_\varepsilon^{2N_x,2N_t}}\right),$$

where $\Phi^{N_x,N_t}(x_i,t_n)$ is the numerical solution whereas $\phi^{N_x,N_t}(x_i,t_n)$ is the analytical solution. The parameter-uniform error and rate of convergence are computed by

$$E^{N_x,N_t} = \max_\varepsilon E_\varepsilon^{N_x,N_t} \quad \text{and} \quad r^{N_x,N_t} = \log_2\left(\frac{E^{N_x,N_t}}{E^{2N_x,2N_t}}\right).$$

The results presented in Tables 1 $(p = 2)$ and 2 $(p = 1)$ clearly show that currently used method improved the findings reported in the literature for Example 1. These results indicate that using $p = 2$

Table 2: Comparisons of our E^{N_x}, r^{N_x} with [17], [5], [13] and [14] for Example 1 taking $N_x = N_t$ and $p = 1$

$\varepsilon \downarrow$	$N_x = 16$	32	64	128	256	512	1024
Our result							
10^{-2}	6.5979e-3	1.4887e-3	3.6050e-4	8.9364e-5	2.2300e-5	5.5715e-6	1.3928e-6
10^{-4}	8.8940e-3	3.0898e-3	1.0372e-3	3.4276e-4	1.1161e-4	3.5260e-5	1.0873e-5
10^{-6}	8.8949e-3	3.0902e-3	1.0373e-3	3.4281e-4	1.1162e-4	3.5264e-5	1.0874e-5
10^{-8}	8.8949e-3	3.0902e-3	1.0373e-3	3.4281e-4	1.1162e-4	3.5264e-5	1.0874e-5
10^{-10}	8.8949e-3	3.0902e-3	1.0373e-3	3.4281e-4	1.1162e-4	3.5264e-5	1.0874e-5
10^{-12}	8.8949e-3	3.0902e-3	1.0373e-3	3.4281e-4	1.1162e-4	3.5264e-5	1.0874e-5
10^{-14}	8.8949e-3	3.0902e-3	1.0373e-3	3.4281e-4	1.1162e-4	3.5264e-5	1.0874e-5
10^{-16}	8.8949e-3	3.0902e-3	1.0373e-3	3.4281e-4	1.1162e-4	3.5264e-5	1.0874e-5
10^{-18}	8.8949e-3	3.0902e-3	1.0373e-3	3.4281e-4	1.1162e-4	3.5264e-5	1.0874e-5
10^{-20}	8.8949e-3	3.0902e-3	1.0373e-3	3.4281e-4	1.1162e-4	3.5264e-5	1.0874e-5
E^{N_x}	8.8949e-3	3.0902e-3	1.0373e-3	3.4281e-4	1.1162e-4	3.5264e-5	1.0874e-5
r^{N_x}	1.5253	1.5749	1.5974	1.6188	1.6623	1.6973	
CPU(sec)	0.133344	0.271822	0.841862	8.416789	12.311128	48.345393	242.90517
In [17]							
E^{N_x}	-	9.1522e-3	4.9429e-3	2.5826e-3	1.3245e-3	6.7190e-4	-
r^{N_x}	-	0.8888	0.9366	0.9634	0.9791		
In [5]							
E^{N_x}	-	3.0430e-2	1.1938e-2	6.1610e-3	3.2113e-3	1.6509e-3	8.1908e-4
r^{N_x}	-	1.3499	0.9543	0.9400	0.9599	1.0112	
In [13]							
E^{N_x}	-	8.0607e-3	4.4718e-3	2.4120e-3	1.2744e-3	-	-
r^{N_x}	-	0.8501	0.8906	0.9204	-		-
In [14]							
E^{N_x}	-	4.9556e-3	2.6327e-3	1.3664e-3	6.9824e-4	4.0519e-4	-
r^{N_x}	-	0.91254	0.9461	0.9686	0.9820	1.6890	-

yields more accurate maximum point-wise errors and rates than using $p = 1$. The numerical findings for $p = 1$ in Tables 3 and 4 reveal that the current method improves the findings that were published in the literature for Example 2 and Example 3, respectively. The numerical simulations for Examples 1–3 are shown in Figures 1–3 depicting the strong boundary layer formation at $x = 0$ as $\varepsilon \to 0$. In Figure 4, the maximum absolute errors for Examples 1–3 are displayed on a log-log scale, illustrating parameter-uniform convergence and the theoretical order of convergence. The reported CPU times increase with refinement of mesh since the linear system size becomes larger, resulting in increasing computational costs. The method exhibits the expected scalability for direct solutions, with CPU time increasing in proportion to problem size. Furthermore, CPU times are influenced by the computing environment, which includes the processor, memory, MATLAB version, and system load.

Table 3: Comparisons of our E^{N_x}, r^{N_x} with [13] and [14] for Example 2 taking $N_x = N_t$, $p = 1$

$\varepsilon \downarrow$	$N_x = 16$	32	64	128	256	512	1024
Our result							
10^{-2}	7.9221e-3	1.8600e-3	4.4941e-4	1.1198e-4	2.7928e-5	6.9783e-6	1.7445e-6
10^{-4}	1.0932e-2	3.8711e-3	1.3038e-3	4.3396e-4	1.4130e-4	4.4617e-5	1.3769e-5
10^{-6}	1.0957e-2	3.8770e-3	1.3053e-3	4.3448e-4	1.4146e-4	4.4665e-5	1.3784e-5
10^{-8}	1.0959e-2	3.8775e-3	1.3055e-3	4.3452e-4	1.4147e-4	4.4668e-5	1.3785e-5
10^{-10}	1.0960e-2	3.8776e-3	1.3055e-3	4.3453e-4	1.4147e-4	4.4669e-5	1.3785e-5
10^{-12}	1.0960e-2	3.8776e-3	1.3055e-3	4.3453e-4	1.4147e-4	4.4669e-5	1.3785e-5
10^{-14}	1.0960e-2	3.8776e-3	1.3055e-3	4.3453e-4	1.4147e-4	4.4669e-5	1.3785e-5
10^{-16}	1.0960e-2	3.8776e-3	1.3055e-3	4.3453e-4	1.4147e-4	4.4669e-5	1.3785e-5
10^{-18}	1.0960e-2	3.8776e-3	1.3055e-3	4.3453e-4	1.4147e-4	4.4669e-5	1.3785e-5
10^{-20}	1.0960e-2	3.8776e-3	1.3055e-3	4.3453e-4	1.4147e-4	4.4669e-5	1.3785e-5
E^{N_x}	1.0960e-2	3.8776e-3	1.3055e-3	4.3453e-4	1.4147e-4	4.4669e-5	1.3785e-5
r^{N_x}	1.4990	1.5706	1.5871	1.6189	1.6632	1.6962	-
CPU(sec)	0.117904	0.283026	0.845376	3.249200	12.352646	54.954900	240.758146
In [13]							
E^{N_x}	-	5.8935e-3	3.1304e-3	1.6083e-3	8.1279e-4	4.0825e-4	-
r^{N_x}	-	0.9128	0.9608	0.9846	0.9934	-	-
In [14]							
E^{N_x}	-	4.2861e-3	2.2276e-3	1.1347e-3	5.7213e-4	2.8721e-4	-
r^{N_x}	-	0.9442	0.9732	0.9879	0.9942	-	-

6 Conclusion

A uniformly convergent method has been developed to solve the singularly perturbed parabolic boundary turning point problem. The proposed method is stable and parameter-uniformly convergent with an error bound of order $O(N_x^{-2}\ln^2 N_x + N_t^2)$. Since we used Shishkin mesh, we don't expect a full second-order method in the space direction. This is due to the logarithmic factor in the Shishkin mesh, which reduces both the accuracy and the rate of convergence. The numerical results presented in the Tables reveal that the current method outperforms methods reported in the literature. Numerical experiments also indicate that the method performs well for problems with non-smooth data. Future work will consider applying the present method to a singularly perturbed problem with non-smooth data and higher dimensions.

Acknowledgments

The authors are grateful to the Editor-in-Chief and reviewers for their constructive feedback.

Financial disclosure

This work was funded by the Foreign Expert Project at Nanjing University of Information Science and Technology in China [No. Y20240261].

Table 4: Comparisons of our E^{N_x}, r^{N_x} with [17] and [14] for Example 3 taking $N_x = N_t$, $p = 1$

$\varepsilon \downarrow$	$N_x = 16$	32	64	128	256	512	1024
Our result							
10^{-2}	3.0609e-3	7.0638e-4	1.7414e-4	4.3309e-5	1.0819e-5	2.7037e-6	6.7589e-7
10^{-4}	3.6957e-3	1.3887e-3	4.9147e-4	1.6584e-4	5.4147e-5	1.7123e-5	5.2871e-6
10^{-6}	9.6260e-3	2.2955e-3	4.9137e-4	1.6582e-4	5.4141e-5	1.7121e-5	5.2875e-6
10^{-8}	1.1412e-2	3.0051e-3	8.0532e-4	2.0989e-4	5.4140e-5	1.7121e-5	5.2876e-6
10^{-10}	1.3995e-2	3.2386e-3	8.5076e-4	2.3382e-4	6.3850e-5	1.7161e-5	5.2876e-6
10^{-12}	1.4614e-2	3.5234e-3	8.9211e-4	2.3700e-4	6.5299e-5	1.7964e-5	5.2876e-6
10^{-14}	1.4691e-2	3.5719e-3	9.1783e-4	2.4357e-4	6.5798e-5	1.8048e-5	5.2876e-6
10^{-16}	1.4699e-2	3.5772e-3	9.2120e-4	2.4554e-4	6.6566e-5	1.8154e-5	5.2876e-6
10^{-18}	1.4699e-2	3.5778e-3	9.2155e-4	2.4577e-4	6.6702e-5	1.8221e-5	5.2876e-6
10^{-20}	1.4699e-2	3.5778e-3	9.2158e-4	2.4579e-4	6.6717e-5	1.8229e-5	5.2876e-6
E^{N_x}	1.4699e-2	3.5778e-3	9.2159e-4	2.4579e-4	6.6718e-5	1.8230e-5	5.2876e-6
r^{N_x}	2.0386	1.9569	1.9067	1.8813	1.8718	1.7856	-
CPU(sec)	0.069908	0.101305	0.252813	0.838704	3.137878	18.438645	99.490726
In [17]							
E^{N_x}	-	1.6200e-2	1.0147e-2	5.7757e-3	3.1179e-3	1.6272e-3	-
r^{N_x}	-	0.6750	0.8130	0.8894	0.9382	-	-
In [14]							
E^{N_x}	-	5.3773e-3	2.7828e-3	1.4144e-3	7.1277e-4	3.5763e-4	-
r^{N_x}	-	0.9503	0.9764	0.9887	0.9950	-	-

Conflict of interest

The authors have not reported any potential conflicts of interest.

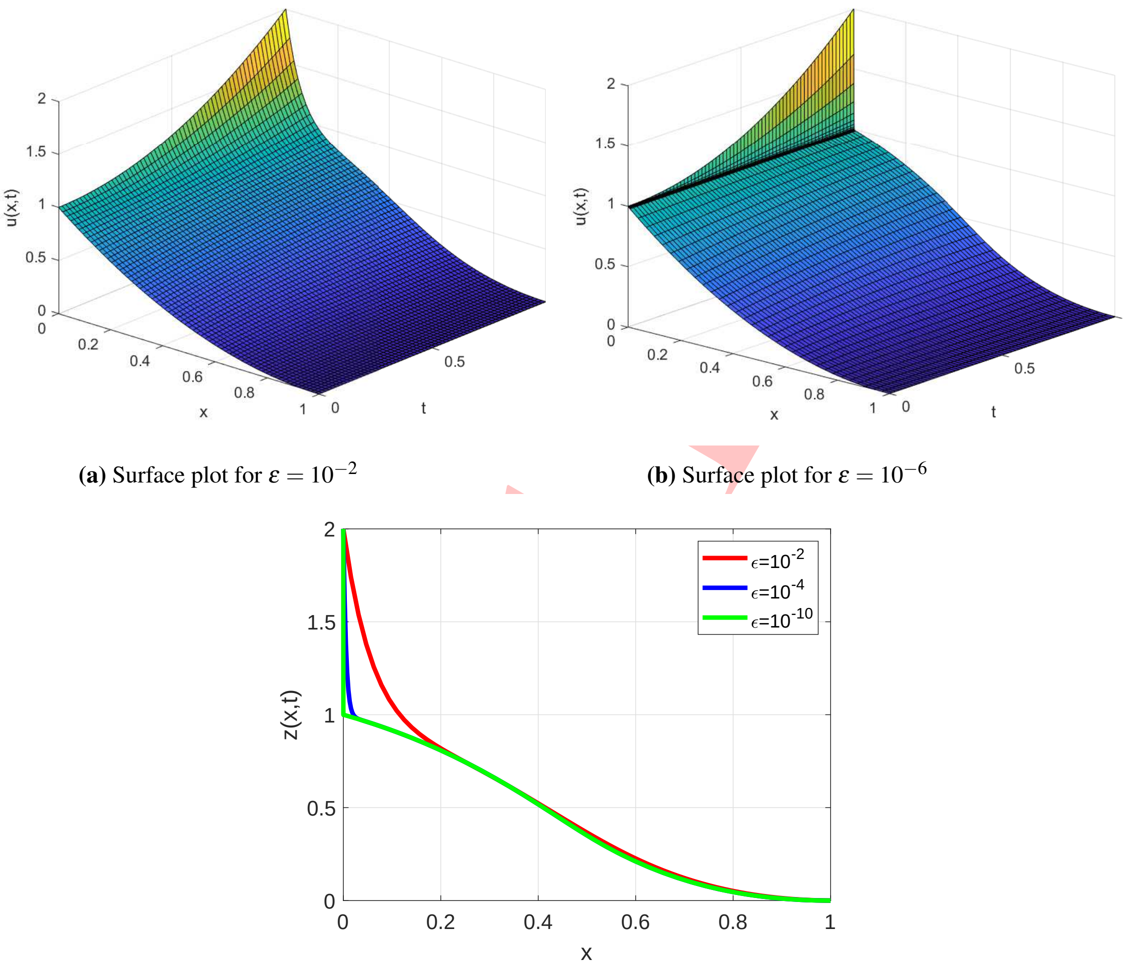
References

[1] T.C. Hanks, *Model relating heat-flow values near, and vertical velocities of mass transport beneath oceanic rises,* J. Geophys. Res. **76** (1971) 537–544.

[2] H. Schliting, Boundary Layer Theory, McGraw-Hill, New York. 1979.

[3] R.K. Dunne, E. O'Riordan, G.I. Shishkin, *A fitted mesh method for a class of singularly perturbed parabolic problems with boundary turning point,* Comput. Methods Appl. Math. **3** (2003) 361–372.

[4] V. Gupta, M.K. Kadalbajoo, *A layer adaptive B-spline collocation method for singularly perturbed one-dimensional parabolic problem with a boundary turning point,* Numer. Methods Partial Differ. Equ. **27** (2011) 1143–1164.



(a) Surface plot for $\varepsilon = 10^{-2}$

(b) Surface plot for $\varepsilon = 10^{-6}$

(c) Effect of the parameter ε

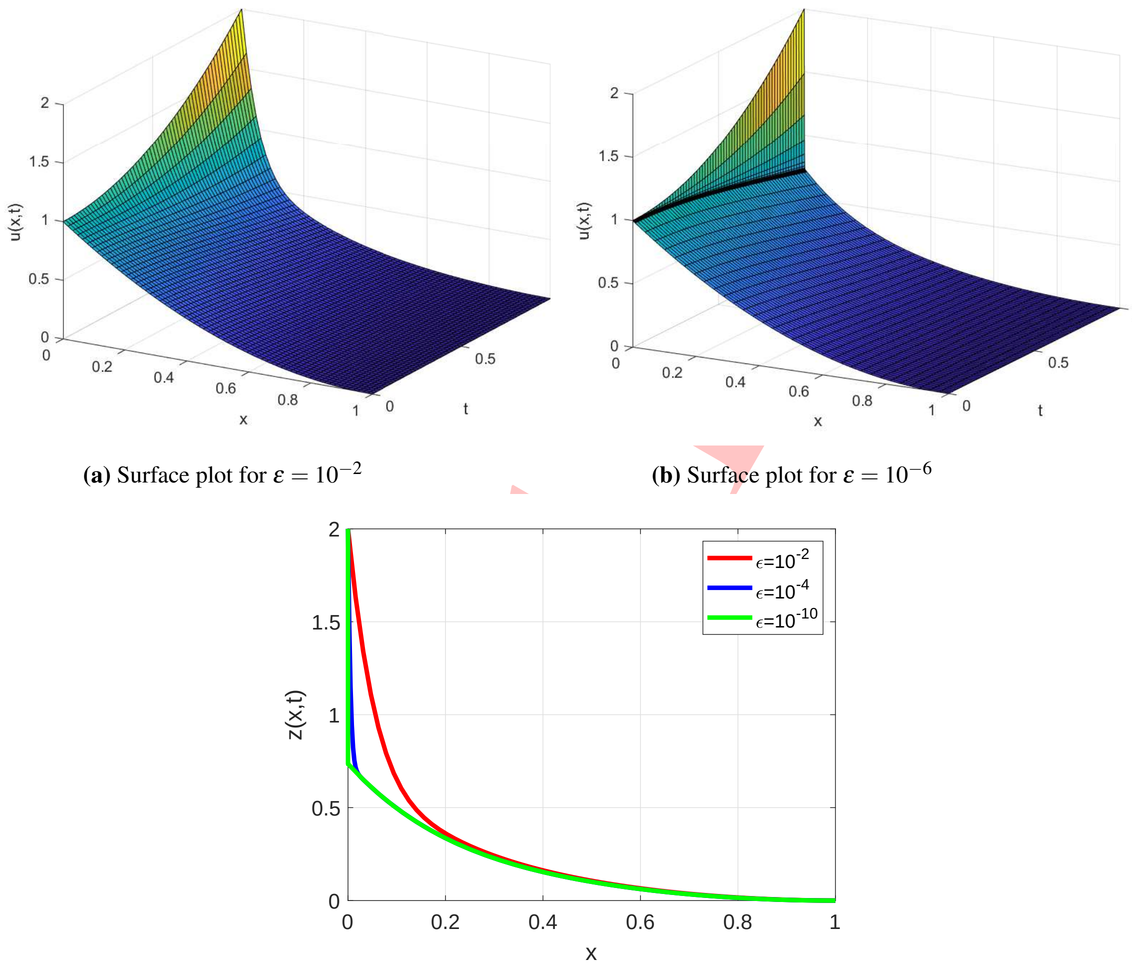
Figure 1: This figure depicts the formation of strong layer for Example 1 at $N_x = 64,\ N_t = 64,\ p = 2$ as $\varepsilon \to 0$.

[5] S. Kumar, Sumit, J. Vigo-Aguiar, *A parameter-uniform grid equidistribution method for singularly perturbed degenerate parabolic convection-diffusion problems,* J. Comput. Appl. Math. **404** (2022) 113273.

[6] A. Majumdar, S. Natesan, *Second-order uniformly convergent Richardson extrapolation method for singularly perturbed degenerate parabolic PDEs,* Int. J. Appl. Comput. Math. **3** (2017) 31–53.

[7] A. Majumdar, S. Natesan, *An $\varepsilon-$uniform hybrid numerical scheme for a singularly perturbed degenerate parabolic convection diffusion problem,* Int. J. Appl. Comput. Math. **96** (2018) 1313–1334.

[8] N.A. Mbroh, S.C. Noutchie, R.Y. Massoukou, *A second-order fitted operator finite difference scheme for a singularly perturbed degenerate parabolic problem,* Int. J. Nonlinear Anal. Appl.



(a) Surface plot for $\varepsilon = 10^{-2}$

(b) Surface plot for $\varepsilon = 10^{-6}$

(c) Effect of the parameter ε

Figure 2: This figure depicts the formation of strong layer for Example 2 at $N_x = 64,\ N_t = 64,\ p = 1$ as $\varepsilon \to 0$.

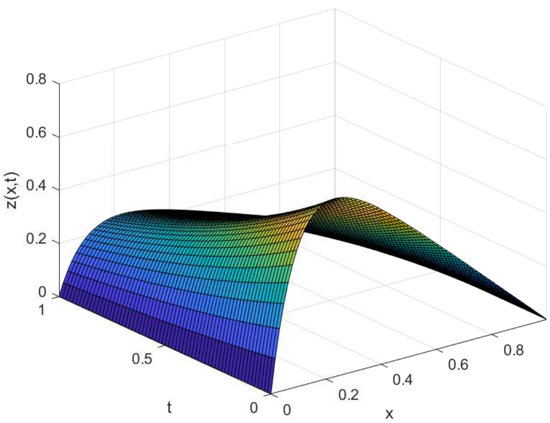
12 (2021) 677–687.

[9] S.K. Sahoo, V. Gupta, *Second-order parameter-uniform finite difference scheme for singularly perturbed parabolic problem with a boundary turning point,* J. Differ. Equ. Appl. **27** (2021) 223–240.

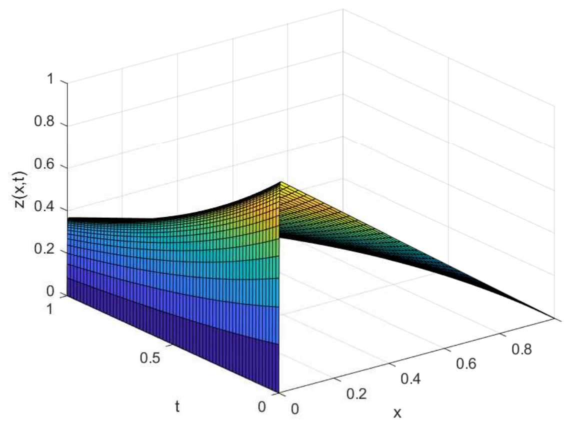
[10] S. Singh, D. Kumar, H. Ramos, *A uniformly convergent quadratic B-spline based scheme for singularly perturbed degenerate parabolic problems,* Math. Comput. Simul. **195** (2022) 88–106.

[11] S. Yadav, P. Rai, K.K. Sharma, *A higher order uniformly convergent method for singularly perturbed parabolic turning point problems,* Numer. Methods Partial Differ. Equ. **36** (2019) 342–368.

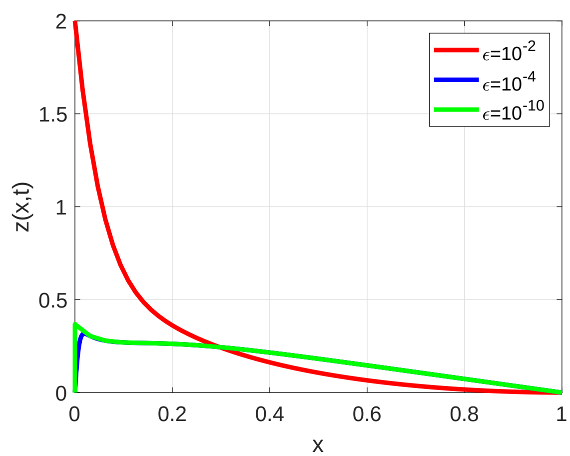
[12] P. Kumari, D. Kumar, H. Ramos, *Parameter independent scheme for singularly perturbed problems including a boundary turning point of multiplicity* ≥ 1*,* J. Appl. Anal. Comput. **13** (2023) 1304–1320.



(a) Surface plot for $\varepsilon = 10^{-2}$



(b) Surface plot for $\varepsilon = 10^{-6}$



(c) Effect of the parameter ε

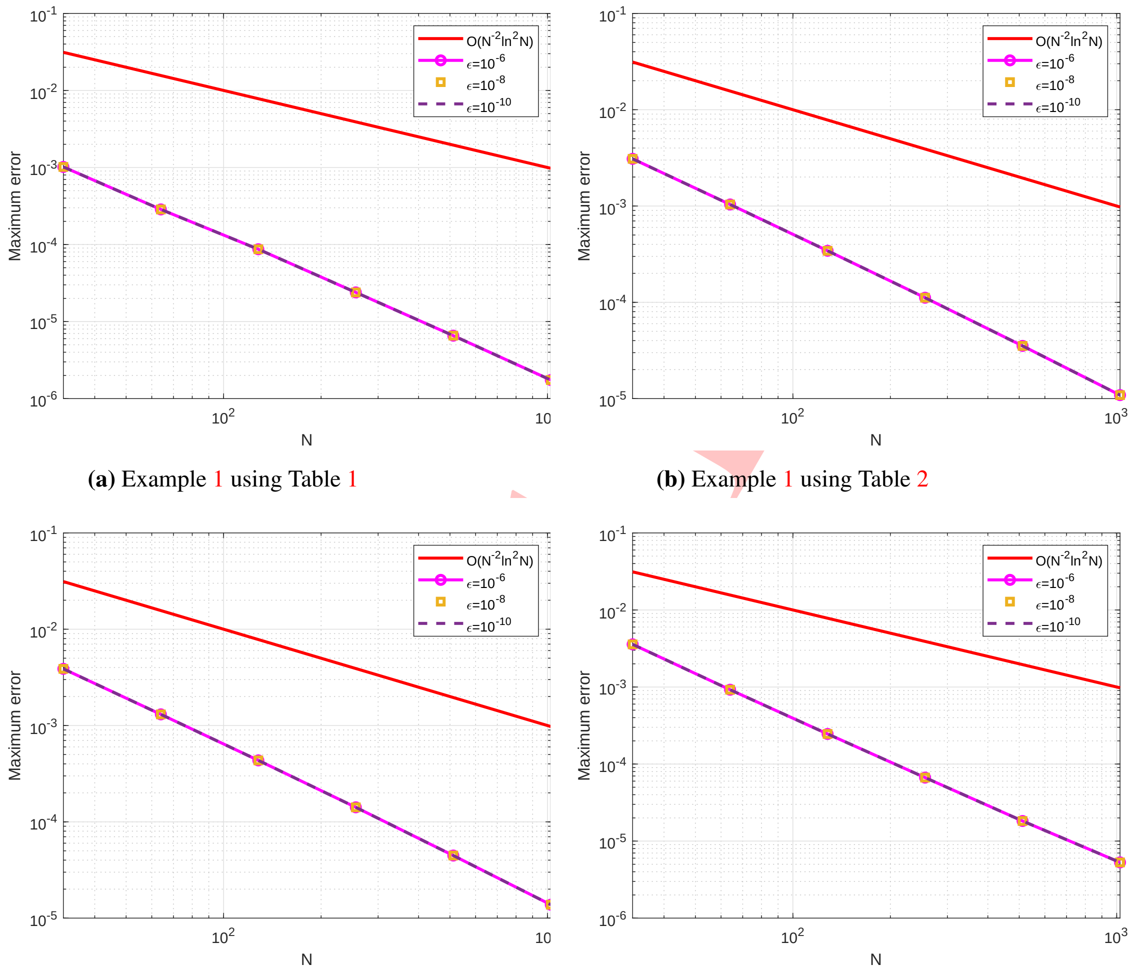
Figure 3: This figure depicts the formation of strong layer for Example 3 at $N_x = 64, N_t = 64, p = 1$ as $\varepsilon \to 0$.

[13] S.K. Tesfaye, G.F. Duressa, M.M. Woldaregay, T.G. Dinka, *A uniformly convergent numerical scheme for singularly perturbed parabolic turning point problem,* J. Math. Model. **12(3)** (2024) 501–516.

[14] Y.M. Kebede, A.A. Tiruneh, E.G. Tsega, *Crank-Nicolson method for singularly perturbed unsteady parabolic problem with multiple boundary turning points,* Adv. Math. Phys. **2024** (2024) 4895771.

[15] J.G. Janani, A. Tamilselvan, *An ε−uniform method for a class of singularly perturbed parabolic problems with Robin boundary conditions having boundary turning point,* Asian Eur. J. Math. **13** (2020) 2050025.

[16] T. Aziz, A. Khan, *A spline method for second-order singularly perturbed boundary-value problems,* J. Comput. Appl. Math. **147** (2002) 445–452.



(a) Example 1 using Table 1 **(b)** Example 1 using Table 2

(c) Example 2 using Table 3 **(d)** Example 3 using Table 4

Figure 4: This figure depicts the maximum errors in terms of log-log scale.

[17] S. Kumar, S. Kumar, H. Ramos, Kuldeep. *A parameter uniform numerical method on a Bakhvalov type mesh for singularly perturbed degenerate parabolic convection-diffusion problems,* J. Appl. Math. Comput. **162** (2020), 49–78.

[18] S. Yadav, P. Rai, *A parameter uniform scheme for delay parabolic singularly perturbed turning point problem,* Differ Equ Dyn Syst. **32** (2024) 421–436.

[19] J.B. Munyakazi, K.C. Patidar, M.T. Sayi, *A fitted numerical method for parabolic turning point singularly perturbed problems with an interior layer,* Numer. Methods Partial Differ. Equ. **35** (2020) 2407–2422.

[20] P. Rai, S. Yadav, *Robust numerical schemes for singularly perturbed delay parabolic convection-diffusion problems with degenerate coefficient,* Int. J. Comput. Math. **98** (2021) 195–221.

[21] F.W. Gelu, G.F. Duressa, *Parameter-uniform numerical scheme for singularly perturbed parabolic convection-diffusion Robin type problems with a boundary turning point,* Results Appl. Math. **15** (2022) 100324.

[22] F.W. Gelu, G.F. Duressa, *A parameter-uniform numerical method for singularly perturbed Robin type parabolic convection-diffusion turning point problems,* Appl. Numer. Math. **190** (2023) 50–64.

[23] H.J. Al Salman, F.W. Gelu, A.A. Al Ghafli, *A fitted mesh robust numerical method and analysis for the singularly perturbed parabolic PDEs with a degenerate coefficient,* Results Appl. Math. **24** (2024) 100519.

[24] N.K. Nichols, *On the numerical integration of a class of singular perturbation problems,* J. Optim. Theory Appl. **60** (1989) 439–452.