BMC Research Notes

RESEARCH NOTE

Open Access

# Cubic non-polynomial spline on piecewise mesh for singularly perturbed reaction differential equations with robin type boundary conditions

Bethelhem Esayas Ayele[1], Tesfaye Aga Bullo[1*] and Gemechis File Duressa[1]

**Abstract**

**Objective** The main purpose of this work is to present cubic non-polynomial spline approximation method for solving Robin-type singularly perturbed reaction–diffusion problems.

**Results** The solution domain is first discretized using a piecewise mesh. The process begins by defining the cubic non-polynomial spline function and calculating its derivatives. These derivatives are then transformed into difference approximations, forming a linear system of algebraic equations in the form of a three-term recurrence relation, which is solved using an elimination algorithm. The stability and consistency of the method are analyzed, ensuring convergence. Numerical model examples are used to validate the proposed method, and the results are compared with those from other methods found in the literature. The maximum absolute error and the order of convergence for each example demonstrate the effectiveness and core contribution of the method.

**Keywords** Singularly perturbed, Robin type problems, Piecewise mesh, Accurate solution

**Mathematics Subject Classification** 34B08, 34D15, 34D20, 65L11

## Introduction

Mathematically modeled problems involve constructing a set of equations that consistently describe the characteristics or behavior of a physical system. Differential equations are often used to model various physical phenomena, where specific conditions are defined for the problem being studied [1]. A singularly perturbed differential equation arises when the highest order derivative is multiplied by a small positive parameter, called the perturbation parameter. These types of equations frequently appear in various fields of applied mathematics and engineering, including fluid mechanics, elasticity, quantum mechanics, chemical-reactor theory, aerodynamics, plasma dynamics, oceanography, meteorology, and the modeling of semiconductor devices [2–5].

The importance of these problems in real-world applications underscores the need for effective numerical methods to approximate their solutions. Numerical methods provide a practical approach when exact analytical solutions are either impossible or too complex to derive [2]. Among these, finite-difference methods are widely used for solving differential equations by approximating derivatives with finite differences. These methods transform differential equations into a system of algebraic equations that can be solved using iterative techniques.

Singularly perturbed problems are broadly categorized into reaction–diffusion and convection–diffusion types. As the perturbation parameter approaches zero, the order

*Correspondence:
Tesfaye Aga Bullo
tesfayeaga2@gmail.com
[1] Department of Mathematics, Jimma University, Jimma, Ethiopia

of the differential equation reduces by one or two [6–9]. These problems can also be classified based on boundary conditions, such as Dirichlet, Robin, or mixed conditions. In particular, second-order singularly perturbed reaction–diffusion problems with Robin boundary conditions exhibit boundary layers on both the left and right sides. This study focuses on the numerical solution of linear, second-order, singularly perturbed reaction–diffusion ordinary differential equations with Robin-type boundary conditions.

The challenge in solving singularly perturbed problems lies in the fact that the solution may vary rapidly in some parts of the domain and slowly in others, depending on the perturbation parameter [7, 8]. Standard numerical methods often fail to provide accurate results for these problems, especially as the perturbation parameter approaches zero. Therefore, it is crucial to develop numerical methods that are uniformly convergent and unaffected by the perturbation parameter. This study aims to develop such a method for Robin-type singularly perturbed reaction–diffusion problems.

Several numerical methods have been proposed in the literature for solving singularly perturbed reaction–diffusion problems [2, 5, 10–14], with many focusing on Dirichlet boundary conditions. However, more recent studies have addressed Robin-type boundary conditions [15–18], introducing higher-order finite difference methods for these problems. Also, the novel numerical methods, along with a detailed continuous solution analysis, are presented in ref. [15, 27, 28]. Despite these advances, there is still room for improvement in terms of the accuracy and convergence rates of the numerical solutions. Thus, the main objective of this work is to develop a higher-order, uniformly convergent numerical method for solving Robin-type singularly perturbed reaction–diffusion problems, along with a theoretical error analysis and numerical validation.

## Description of the method

This paper deals with the Robin type singularly perturbed reaction diffusion problem of the form:

$$\begin{cases} Ly(x) \equiv -\varepsilon y''(x) + b(x)y(x) = f(x), \; 0 < x < 1, \\ B_L y(0) \equiv a_0 y(0) - a_1\sqrt{\varepsilon} y'(0) = \eta_1, \\ B_R y(1) \equiv d_0 y(1) + d_1 \sqrt{\varepsilon} y'(1) = \eta_2, \end{cases} \tag{1}$$

where $0 < \varepsilon << 1$ is the singular perturbation parameter, and $a_0,\ a_1,\ d_0,\ d_1,\ \eta_0,\ \eta_1$ with $a_1,\ d_1 \ \neq 0$ are given constants. The main purpose of this condition is to ensure the applicability of Robin-type boundary conditions. If this condition is not fulfilled, the problem shifts to a form with Dirichlet boundary conditions, which are more widely addressed by existing methods in the literature. The differential operators are denoted by $L,\ B_L,\ B_R$.

The functions $b(x)$ and the source $f(x)$ function are assume to be sufficiently smooth functions such that $b(x) \geq b_0 > 0,\ \forall x \in [0,\ 1]$. Also, assume that the given constant values given in robin type boundary conditions satisfy $a_j,\ d_j \ \geq 0$ and $a_j + \ d_j \ > 0,\ j = 0, 1$ .Then, the problem in Eq. (1) has a unique solution $y(x) \in C^2(0,\ 1) \cap C^1[0,\ 1]$ that typically exhibits boundary layers both at $x = 0$ and $x = 1$ when the perturbation parameter sufficiently small. The detailed proofs of the existence and uniqueness of the continuous solution for this defined problem is provided in ref. [15].

## Piecewise mesh generation

We construct a piecewise mesh that contains more number of nodal points in the layer regions than non-layer region. The domain $[0, 1]$, $N \geq 4$ for $N$ is even multiple integer of 4, is divided into three subintervals, $[0, \tau]$, $[\tau, 1-\tau]$, $[1-\tau, 1]$ where the chosen transition parameter,

$$\tau = \min\left\{\frac{1}{4},\ 2\sqrt{\varepsilon}\ln(N)\right\}, \tag{2}$$

denotes the width of the boundary layers. The domain $[0, 1]^N$ is obtained by putting a non-uniform mesh with $\frac{N}{4}$ mesh elements in both the layer intervals and a uniform mesh with $\frac{N}{2}$ mesh elements in the outer layer region.

A general piecewise mesh $[0, 1]^N = \{0 = x_0, x_1, x_2, ..., x_N = 1\}$ with step size will be defined:

$$h_i = x_i - x_{i-1} = \begin{cases} \dfrac{4\tau}{N},\ i = 1, 2, ..., \dfrac{N}{4} \\ \dfrac{2(1-2\tau)}{N},\ i = \dfrac{N}{4} + 1, ..., \dfrac{3N}{4}, \\ \dfrac{4\tau}{N},\ i = \dfrac{3N}{4} + 1, ..., N. \end{cases} \tag{3}$$

## Formulation of the method (cubic non-polynomial spline approximation)

Now, in order to develop the non-polynomial cubic spline approximation for the problems in Eq. (1), the interval [0, 1] is divided into $N$ sub-intervals. Let $y(x_i)$ be the exact solution to the problem in Eq. (1), and $y_i$ be an approximation to $y(x_i)$, obtained by the segment $S(x)$ of the spline function passing through the points $(x_i,\ y_i)$ and $(x_{i+1},\ y_{i+1})$. Also, let us consider the non-polynomial cubic spline function $S(x)$ in subinterval $[x_i,\ x_{i+1}],\ \ i = 1,\ \ldots,\ N-1$, of the form:

$$S(x) = c_i + d_i(x - x_i) + u_i(e^{\omega(x-x_i)} - e^{-\omega(x-x_i)}) + v_i(e^{\omega(x-x_i)} + e^{-\omega(x-x_i)}) \tag{4}$$

where, $c_i,\ d_i,\ u_i$, and $v_i$ are unknown coefficients to be determined, and $\omega$ is a free parameter, which is used to raise the accuracy of the method.

To determine the unknown coefficient in Eq. (4), let us denote:

$$\begin{aligned} &S(x_i) = y_i, \quad S'(x_i) = m_i, \quad S''(x_i) = M_i, \\ &S(x_{i+1}) = y_{i+1}, \quad S'(x_{i+1}) = m_{i+1}, \quad S''(x_{i+1}) = M_{i+1}, \end{aligned} \tag{5}$$

Differentiating Eq. (4) successively, we get:

$$S'(x) = d_i + \omega u_i(e^{\omega(x-x_i)} + e^{-\omega(x-x_i)}) + \omega v_i(e^{\omega(x-x_i)} - e^{-\omega(x-x_i)}) \tag{6}$$

$$S''(x) = \omega^2 u_i(e^{\omega(x-x_i)} - e^{-\omega(x-x_i)}) + \omega^2 v_i(e^{\omega(x-x_i)} + e^{-\omega(x-x_i)}) \tag{7}$$

Using relations in Eqs. (5) and (7), we have:that results

$$\begin{aligned} S''(x_i) &= \omega^2 u_i(e^{\omega(x_i-x_i)} - e^{-\omega(x_i-x_i)}) + \omega^2 v_i(e^{\omega(x_i-x_i)} + e^{-\omega(x_i-x_i)}) = M_i \\ &= \omega^2 u_i(1-1) + \omega^2 v_i(1+1) = M_i \\ &= 2\omega^2 v_i = M_i \end{aligned}$$

$$v_i = \frac{M_i}{2\omega^2}. \tag{8}$$

Again, using the relation in Eqs. (5), (8) into Eq. (4) at the point $x_i$, yields:and it gives

$$\begin{aligned} S(x_i) &= c_i + d_i(x_i - x_i) + u_i(e^{\omega(x_i-x_i)} - e^{-\omega(x_i-x_i)}) + v_i(e^{\omega(x_i-x_i)} + e^{-\omega(x_i-x_i)}) = y_i \\ &= c_i + u_i(1-1) + v_i(1+1) = y_i \\ &= c_i + 2v_i = y_i \end{aligned}$$

$$c_i = y_i - \frac{M_i}{\omega^2} \tag{9}$$

Using the relation in Eqs. (5), (7) at the point $x_{i+1}$ and letting $\theta_2 = \omega h_{i+1}$, we have:

$$S''(x_{i+1}) = \omega^2 u_i(e^{\omega(x_{i+1}-x_i)} - e^{-\omega(x_{i+1}-x_i)}) + \omega^2 v_i(e^{\omega(x_{i+1}-x_i)} + e^{-\omega(x_{i+1}-x_i)}) = M_{i+1}$$

$$\begin{aligned} &\omega^2 u_i(e^{\omega h_{i+1}} - e^{-\omega h_{i+1}}) + \omega^2 v_i(e^{\omega h_{i+1}} + e^{-\omega h_{i+1}}) = M_{i+1} \\ &u_i(e^{\theta_2} - e^{-\theta_2}) + v_i(e^{\theta_2} + e^{-\theta_2}) = \frac{M_{i+1}}{\omega^2} \\ &u_i(e^{\theta_2} - e^{-\theta_2}) = \frac{M_{i+1}}{\omega^2} - v_i(e^{\theta_2} + e^{-\theta_2}), \qquad But,\ v_i = \frac{M_i}{2\omega^2} \end{aligned}$$

It gives

$$u_i = \frac{M_{i+1}}{\omega^2(e^{\theta_2} - e^{-\theta_2})} - \frac{M_i}{2\omega^2}\frac{(e^{\theta_2} + e^{-\theta_2})}{(e^{\theta_2} - e^{-\theta_2})} \quad (10)$$

Also, using the relation in Eqs. (5), Eq. (4) at the point $x_{i+1}$ and for $\theta_2 = \omega h_{i+1}$, we get:this results

$$S(x_{i+1}) = c_i + d_i(x_{i+1} - x_i) + u_i(e^{\omega(x_{i+1}-x_i)} - e^{-\omega(x_{i+1}-x_i)}) + v_i(e^{\omega(x_{i+1}-x_i)} + e^{-\omega(x_{i+1}-x_i)}) = y_{i+1}$$

$$y_{i+1} = c_i + d_i h_{i+1} + u_i(e^{\omega h_{i+1}} - e^{-\omega h_{i+1}}) + v_i(e^{\omega h_{i+1}} + e^{-\omega h_{i+1}})$$
$$y_{i+1} = c_i + d_i h_{i+1} + u_i(e^{\theta_2} - e^{-\theta_2}) + v_i(e^{\theta_2} + e^{-\theta_2})$$

$$y_{i+1} = y_i - \frac{M_i}{\omega^2} + d_i h_{i+1} + \frac{M_{i+1}}{\omega^2} - \frac{M_i}{2\omega^2}(e^{\theta_2} + e^{-\theta_2}) + \frac{M_i}{2\omega^2}(e^{\theta_2} + e^{-\theta_2})$$
$$y_{i+1} - y_i = d_i h_{i+1} + \frac{M_{i+1} - M_i}{\omega^2}$$

$$d_i = \frac{y_{i+1} - y_i}{h_{i+1}} - \frac{M_{i+1} - M_i}{h_{i+1}\omega^2} \quad (11)$$

Using the continuity condition of the first derivative at $x_i$, $S'_{\Delta-1}(x_i) = S'_{\Delta}(x_i)$, we obtain:

$$d_{i-1} + \omega u_{i-1}(e^{\theta_1} + e^{-\theta_1}) - \omega v_{i-1}(e^{\theta_1} - e^{-\theta_1}) = d_i + 2\omega u_i \quad (12)$$

Reducing indices of Eqs. (10) and (11) by one and substituting into Eq. (12), and after multiplying both sides by $\frac{2}{h_i+h_{i+1}}$, and due to the parameters $\theta_1 = \omega h_i$, and $\theta_2 = \omega h_{i+1}$, we have the parameter $\omega = \frac{\theta_1+\theta_2}{h_i+h_{i+1}}$, so that we get:

$$\frac{2}{h_i + h_{i+1}}\left[\frac{y_{i+1} - y_i}{h_{i+1}} - \frac{y_i - y_{i-1}}{h_i}\right] = \alpha M_{i-1} + 2\beta M_i + \alpha M_{i+1} \quad (13)$$

where the coefficients are denoted by:

$$\alpha = \frac{2}{h_i + h_{i+1}}\left(\frac{1}{h_i\omega^2} - \frac{e^{2\theta_1} + e^{-2\theta_1}}{\omega(e^{\theta_1} - e^{-\theta_1})}\right) = \frac{2}{h_i + h_{i+1}}\left(\frac{1}{h_{i+1}\omega^2} - \frac{2}{\omega(e^{\theta_2} - e^{-\theta_2})}\right),$$
$$\beta = \frac{1}{h_i + h_{i+1}}\left(\frac{(e^{\theta_1} + e^{-\theta_1})}{\omega(e^{\theta_1} - e^{-\theta_1})} + \frac{(e^{\theta_2} + e^{-\theta_2})}{\omega(e^{\theta_2} - e^{-\theta_2})} - \frac{h_i + h_{i+1}}{h_i h_{i+1}\omega^2}\right).$$

As $\theta_1 \to 0$, $\theta_2 \to 0$ in Eq. (13), we get the sum of the two constants, $\alpha + \beta = \frac{1}{2}$ and their values will be determined from local truncation error.

Considering the second order differential equation in Eq. (1) at the nodal point $x_i$ and using the relation in Eq. (5), corresponding to $M_i = y''_i = S''(x)$ which yields:

$$M_i = \frac{b_i y_i - f_i}{\varepsilon}$$
$$M_{i-1} = \frac{b_{i-1} y_{i-1} - f_{i-1}}{\varepsilon} \quad (14)$$
$$M_{i+1} = \frac{b_{i+1} y_{i+1} - f_{i+1}}{\varepsilon}$$

Thus, substituting Eq. (14) into Eq. (13) and also rearranging yields the finite difference scheme that re-written in three term recurrence relations:

$$E_i y_{i-1} + F_i y_i + G_i y_{i+1} = H_i, \quad i = 1, 2, \ldots, N-1 \quad (15)$$

where,

$$E_i = \frac{-2\varepsilon}{h_i(h_i + h_{i+1})} + \alpha b_{i-1}, \qquad F_i = \frac{2\varepsilon}{h_i h_{i+1}} + 2\beta b_i,$$
$$G_i = \frac{-2\varepsilon}{h_{i+1}(h_i + h_{i+1})} + \alpha b_{i+1}, \qquad H_i = \alpha f_{i-1} + 2\beta f_i + \alpha f_{i+1}.$$

### Truncation error (To determine the values of $\alpha$ and $\beta$)

For the formulated cubic spline method, the truncation error is obtained from Eq. (13) as:

$$T_i = \frac{2}{h_i + h_{i+1}}\left[\frac{y_{i+1} - y_i}{h_{i+1}} - \frac{y_i - y_{i-1}}{h_i}\right] - (\alpha M_{i-1} + 2\beta M_i + \alpha M_{i+1}) \tag{16}$$

From the relation in Eq. (5) and by expanding Eq. (16) in Taylor's series about $x_i$, we have the simplified truncation error:

$$T_i = (1 - 2(\alpha+\beta))y_i'' + (h_{i+1} - h_i)\left\{\frac{1}{3} - \alpha\right\}y_i''' + \left\{\frac{2}{h_i + h_{i+1}}\left(\frac{h_{i+1}^3 + h_i^3}{4!}\right) - \frac{\alpha}{2}\left(h_{i+1}^2 + h_i^2\right)\right\}y_i^{(4)} + \left\{\frac{2}{h_i + h_{i+1}}\left(\frac{h_{i+1}^4 - h_i^4}{5!}\right) - \frac{\alpha}{3!}\left(h_{i+1}^3 - h_i^3\right)\right\}y_i^{(5)} + \left\{\frac{2}{h_i + h_{i+1}}\left(\frac{h_{i+1}^6 + h_i^6}{6!}\right) - \frac{\alpha}{4!}\left(h_{i+1}^4 + h_i^4\right)\right\}y_i^{(6)} + \ldots \tag{17}$$

In order to obtain the higher order method, and for sufficiently cases of the mesh lengths, we choose the value of $\alpha$ and $\beta$ as:

$$\begin{cases} \alpha + \beta = \frac{1}{2}, \\ \frac{1}{3} - \alpha = 0. \end{cases}, \quad \alpha = \frac{1}{3}, \beta = \frac{1}{6},$$

$$\begin{aligned} \Rightarrow \frac{2}{h_i + h_{i+1}}\left(\frac{h_{i+1}^3 + h_i^3}{4!}\right) - \frac{\alpha}{2}\left(h_{i+1}^2 + h_i^2\right) &= \frac{2}{h_i + h_{i+1}}\left(\frac{h_{i+1}^3 + h_i^3}{4!}\right) - \frac{1}{6}\left(h_{i+1}^2 + h_i^2\right) \\ &= \frac{1}{6}\left\{\left(\frac{h_{i+1}^3 + h_i^3}{2(h_{i+1} + h_i)}\right) - \left(h_{i+1}^2 + h_i^2\right)\right\} \\ &= \frac{1}{6}\left\{\frac{h_{i+1}^3 + h_i^3 - 2(h_{i+1}^3 + h_i h_{i+1}^2 + h_{i+1} h_i^2 + h_i^3)}{2(h_{i+1} + h_i)}\right\} \\ &= \frac{1}{6}\left\{\frac{-h_{i+1}^3 - 2h_i h_{i+1}^2 - 2h_{i+1} h_i^2 - h_i^3)}{2(h_{i+1} + h_i)}\right\} \\ &\le (h_i^2 + h_{i+1}^2) \end{aligned}$$

and similarly, $\begin{cases} \alpha + \beta = \frac{1}{2}, \\ \frac{1}{12} - \alpha = 0. \end{cases}, \alpha = \frac{1}{12}, \beta = \frac{5}{12},$

This yields the second order and fourth order convergent respectively.

### Boundary conditions (end conditions)

For $i = 0$, at $x_0 = 0$, we have the discretized form of the first boundary condition, $B_L y(0)$ as

$$a_0 y_0 - a_1\sqrt{\varepsilon}y_0' = \eta_1 \tag{18}$$

In order to formulate the finite difference approximation to Eq. (18), adapting the Taylor series expansion,

$$y_{i+1} = y_i + h_{i+1}y_i' + \frac{h_{i+1}^2}{2}y_i'' + \frac{h_{i+1}^3}{3!}y_i''' + \frac{h_{i+1}^4}{4!}y_i^{(4)} + \frac{h_{i+1}^5}{5!}y_i^{(5)} + \frac{h_{i+1}^6}{6!}y_i^{(6)} + \ldots \tag{19}$$

Considering Eq. (19) at the nodal point $x_0$, to approximation the first derivative in boundary condition, we obtain

$$y'_0 = \frac{y_1 - y_0}{h_1} - \frac{h_1}{2}y''_0 - \frac{h_1^2}{3!}y'''_0 - \frac{h_1^3}{4!}y_0^{(4)} + TB_L, \tag{20}$$

where the local truncation error for the left boundary condition is.

$TB_L = -\frac{h_1^4}{5!}y_0^{(5)} - \frac{h_1^5}{6!}y_0^{(6)} + ....$

Differentiating the second order differential equation Eq. (1) twice with respect to the independent variable and considering it at the discretized points, we obtain

$$\begin{aligned} y''_i &= \frac{b_i y_i - f_i}{\varepsilon} \\ y'''_i &= \frac{b_i y'_i + b'_i y_i - f'_i}{\varepsilon}, \\ y_i^{(4)} &= \frac{b_i y''_i + 2b'_i y'_i + b''_i y_i - f''_i}{\varepsilon}. \end{aligned} \tag{21}$$

Hence, from Eq. (21) the values of $y''_0$, $y'''_0$ , $y_0^{(4)}$, obtained at the nodal point $x_0$ as:

$$\begin{aligned} y''_0 &= \frac{b_0 y_0 - f_0}{\varepsilon}, \\ y'''_0 &= \frac{b_0 y'_0 + b'_0 y_0 - f'_0}{\varepsilon}, \\ y_0^{(4)} &= \frac{b_0 y''_0 + 2b'_0 y'_0 + b''_0 y_0 - f''_0}{\varepsilon}. \end{aligned} \tag{22}$$

Substituting Eq. (22) into Eq. (20) and rearranging yields

$$y'_0 = \frac{\frac{1}{h_1}y_1 - \left\{\frac{h_1^2 b'_0}{3!\varepsilon} + \frac{h_1^3 b''_0}{4!\varepsilon} + \frac{b_0}{\varepsilon}\left(\frac{h_1^3 b_0}{4!\varepsilon} + \frac{h_1}{2}\right) + \frac{1}{h_1}\right\}y_0 + \frac{1}{\varepsilon}\left\{\frac{h_1^3 b_0}{4!\varepsilon} + \frac{h_1}{2}\right\}f_0 + \frac{h_1^2}{3!\varepsilon}f'_0 + \frac{h_1^3}{4!\varepsilon}f''_0}{1 + \frac{h_1^2 b_0}{3!\varepsilon} + \frac{2h_1^3 b'_0}{4!\varepsilon}}. \tag{23}$$

Hence, for $i = 0$, substituting Eq. (23) into Eq. (20), we get the scheme:

$$\begin{aligned} &\left[a_0\left(1 + \frac{h_1^2 b_0}{3!\varepsilon} + \frac{2h_1^3 b'_0}{4!\varepsilon}\right) + a_1\sqrt{\varepsilon}\left\{\frac{h_1^2 b'_0}{3!\varepsilon} + \frac{h_1^3 b''_0}{4!\varepsilon} + \frac{b_0}{\varepsilon}\left(\frac{h_1^3 b_0}{4!\varepsilon} + \frac{h_1}{2}\right) + \frac{1}{h_1}\right\}\right]y_0 - \frac{a_1\sqrt{\varepsilon}}{h_1}y_1 \\ &= \eta_1\left(1 + \frac{h_1^2 b_0}{3!\varepsilon} + \frac{2h_1^3 b'_0}{4!\varepsilon}\right) + a_1\sqrt{\varepsilon}\left\{\frac{1}{\varepsilon}\left\{\frac{h_1^3 b_0}{4!\varepsilon} + \frac{h_1}{2}\right\}f_0 + \frac{h_1^2}{3!\varepsilon}f'_0 + \frac{h_1^3}{4!\varepsilon}f''_0\right\}. \end{aligned} \tag{24}$$

Similarly, for $i = N$, at $x_N$, we have the discretized form of the second boundary condition, $B_R y(1)$ as

$$d_0 y_N + d_1\sqrt{\varepsilon}y'_N = \eta_2. \tag{25}$$

At any nodal point $x_i$, the Taylors' series expansion written as

$$y_{i-1} = y_i - h_i y'_i + \frac{h_i^2}{2}y''_i - \frac{h_i^3}{3!}y'''_i + \frac{h_i^4}{4!}y_i^{(4)} - \frac{h_i^5}{5!}y_i^{(4)} + \frac{h_i^6}{6!}y_i^{(6)} + ... \tag{26}$$

Then, adapting this expansion for the nodal point $x_N$, to approximation the first derivative in the second boundary condition, we get:

$$y'_N = \frac{y_N - y_{N-1}}{h_N} + \frac{h_N}{2}y''_N - \frac{h_N^2}{3!}y'''_N + \frac{h_N^3}{4!}y_N^{(4)} + TB_R, \tag{27}$$

where the local truncation error for the second boundary condition is

$TB_R = -\frac{h_N^4}{5!}y_N^{(5)} + \frac{h_N^5}{6!}y_N^{(6)} + ....$

Similar to Eq. (22), from Eq. (21) at the nodal point $x_N$, considering the values of $y''_N$, $y'''_N$ , $y_N^{(4)}$, and also substituting it into Eq. (27), we get:

$$y'_N = \frac{\frac{-y_{N-1}}{h_N} + \left\{\frac{1}{h_N} - \frac{h_N^2 b'_N}{3!\varepsilon} + \frac{h_N^3 b''_N}{4!\varepsilon} + \frac{h_N b_N}{2\varepsilon} + \frac{h_N^3 b_N^2}{4!\varepsilon^2}\right\} y_N - \left(\frac{h_N}{2\varepsilon} + \frac{h_N^3 b_N}{4!\varepsilon^2}\right) f_N + \frac{h_N^2 f'_N}{3!\varepsilon} - \frac{h_N^3 f''_N}{4!\varepsilon}}{1 + \frac{h_N^2 b_N}{3!\varepsilon} - \frac{2h_N^3 b'_N}{4!\varepsilon}}. \tag{28}$$

Now, substituting this value of $y'_N$ in Eq. (28) into Eq. (25) gives the scheme

$$\begin{aligned} &\frac{-d_1\sqrt{\varepsilon}}{h_N} y_{N-1} + \left[d_0\left(1 + \frac{h_N^2 b_N}{3!\varepsilon} - \frac{2h_N^3 b'_N}{4!\varepsilon}\right) + d_1\sqrt{\varepsilon}\left(\frac{1}{h_N} - \frac{h_N^2 b'_N}{3!\varepsilon} + \frac{h_N^3 b''_N}{4!\varepsilon} + \frac{h_N b_N}{2\varepsilon} + \frac{h_N^3 b_N^2}{4!\varepsilon^2}\right)\right] y_N \\ &= \eta_2\left(1 + \frac{h_N^2 b_N}{3!\varepsilon} - \frac{2h_N^3 b'_N}{4!\varepsilon}\right) + d_1\sqrt{\varepsilon}\left(\frac{h_N}{2\varepsilon} + \frac{h_N^3 b_N}{4!\varepsilon^2}\right) f_N - d_1\sqrt{\varepsilon}\frac{h_N^2 f'_N}{3!\varepsilon} + d_1\sqrt{\varepsilon}\frac{h_N^3 f''_N}{4!\varepsilon}. \end{aligned} \tag{29}$$

To conclude formulation of the method, the three equations in Eqs. (15), (24) and (29) constitute the system of $(N+1) \times (N+1)$ linear algebraic equations, that give the approximations $y_0, y_1, \ldots, y_N$ of the solution $y(x)$ at the nodal points, $x_0, x_1, \ldots, x_N$, with piecewise mesh length $h_i$. This can be re-written in matrix form

$$MY = R, \tag{30}$$

where the three matrices are defined

$$M = (m_{ij})_{(N+1)\times(N+1)} = \begin{bmatrix} B_0 & C_0 & 0 & \cdots & 0 \\ A_1 & B_1 & C_1 & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & A_{N-1} & B_{N-1} & C_{N-1} \\ 0 & \cdots & 0 & A_N & B_N \end{bmatrix}, \quad Y = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{N-1} \\ y_N \end{bmatrix}, \quad R = \begin{bmatrix} H_0 \\ H_1 \\ \vdots \\ H_{N-1} \\ H_N \end{bmatrix}$$

,

with the specific entries are denoting:

$$\begin{aligned} \text{For } i = 0, \quad B_0 &= a_0\left(1 + \frac{h_1^2 b_0}{3!\varepsilon} + \frac{2h_1^3 b'_0}{4!\varepsilon}\right) + a_1\sqrt{\varepsilon}\left(\frac{h_1^2 b'_0}{3!\varepsilon} + \frac{h_1^3 b''_0}{4!\varepsilon} + \frac{b_0}{\varepsilon}\left(\frac{h_1^3 b_0}{4!\varepsilon} + \frac{h_1}{2}\right) + \frac{1}{h_1}\right), \\ C_0 &= -\frac{a_1\sqrt{\varepsilon}}{h_1}, \\ H_0 &= \eta_1\left(1 + \frac{h_1^2 b_0}{3!\varepsilon} + \frac{2h_1^3 b'_0}{4!\varepsilon}\right) + a_1\sqrt{\varepsilon}\left\{\frac{1}{\varepsilon}\left(\frac{h_1^3 b_0}{4!\varepsilon} + \frac{h_1}{2}\right) f_0 + \frac{h_1^2}{3!\varepsilon} f'_0 + \frac{h_1^3}{4!\varepsilon} f''_0\right\}, \end{aligned}$$

For $i = 1, \ldots, N-1: \; A_i = E_i, \; B_i = F_i, \; C_i = G_i,$ and $H_i = \alpha f_{i-1} + 2\beta f_i + \alpha f_{i+1}$.

$$\text{For } i = N: \; A_N = \frac{-d_1\sqrt{\varepsilon}}{h_N},$$
$$B_N = d_0\left(1 + \frac{h_N^2 b_N}{3!\varepsilon} - \frac{2h_N^3 b_N'}{4!\varepsilon}\right) + d_1\sqrt{\varepsilon}\left(\frac{1}{h_N} - \frac{h_N^2 b_N'}{3!\varepsilon} + \frac{h_N^3 b_N''}{4!\varepsilon} + \frac{h_N b_N}{2\varepsilon} + \frac{h_N^3 b_N^2}{4!\varepsilon^2}\right),$$
$$H_N = \eta_2\left(1 + \frac{h_N^2 b_N}{3!\varepsilon} - \frac{2h_N^3 b_N'}{4!\varepsilon}\right) + d_1\sqrt{\varepsilon}\left(\frac{h_N}{2\varepsilon} + \frac{h_N^3 b_N}{4!\varepsilon^2}\right) f_N - d_1\sqrt{\varepsilon}\frac{h_N^2 f_N'}{3!\varepsilon} + d_1\sqrt{\varepsilon}\frac{h_N^3 f_N''}{4!\varepsilon}.$$

## Algorithm and its stability

The obtained finite difference approximation yields an $(N+1)\times(N+1)$ algebraic tri-diagonal system that can be solved by Gaussian elimination method. Hence, in using this method, we have the algorithm:

**Theorem 1: (Stability)** *Let M be a coefficient matrix of the tri-diagonal system, in Eq. (30). Then, the matrix M is an irreducible and diagonally dominant matrix and hence the scheme is stable [26].*

$$\text{Forward Elimination} \rightarrow \begin{cases} W_0 = \dfrac{C_0}{B_0}, \\ T_0 = \dfrac{R_0}{B_0}, \\ W_i = \dfrac{C_i}{B_i - A_i W_{i-1}}, \quad i = 1, 2, \ldots, N+1, \\ T_i = \dfrac{R_i - A_i T_{i-1}}{B_i - A_i W_{i-1}}, \quad i = 1, 2, \ldots, N+1. \end{cases}$$

$$\text{Backward Elimination} \rightarrow \begin{cases} \text{Setting } C_N = 0, \\ y_{N+1} = T_{N+1}, \\ y_i = T_i - W_i y_{i+1}, \quad i = N, N-1, \ldots 2, 1. \end{cases}$$

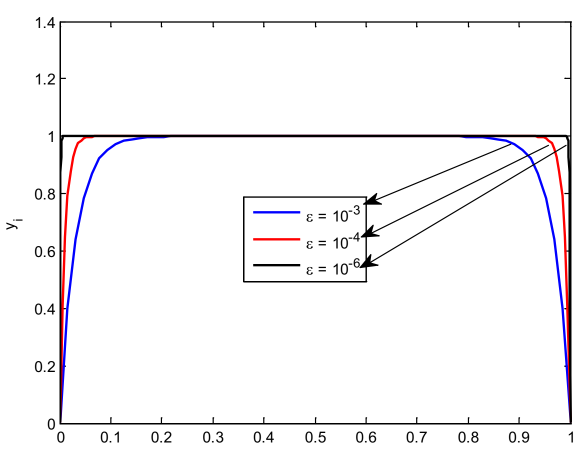


**Fig. 1** Effects of perturbation parameter on the solution for Example 1 when, $N = 64$

***Proof*** To estimate the stability of the formulated method, we show that the coefficient matrix of the method satisfy the diagonal dominance. First, to check this, consider that $b(x) \geq b_0 > 0, \; \forall x \in [0, 1]$ and for $i = 1, \ldots, N-1$, from Eq. (16), we have:

$$
\begin{aligned}
&B_i \ \geq \ A_i + C_i, \quad (F_i \ \geq \ E_i + G_i), \quad i = 1, 2, \ \dots, N-1,\\
&\frac{2\varepsilon}{h_i h_{i+1}} + 2\beta b_i \ \geq \frac{-2\varepsilon}{h_i(h_i + h_{i+1})} + \alpha b_{i-1} + \frac{-2\varepsilon}{h_{i+1}(h_i + h_{i+1})} + \alpha b_{i+1},\\
&\frac{2\varepsilon}{h_i h_{i+1}} + 2\beta b_0 \ \geq \frac{-2\varepsilon}{h_i(h_i + h_{i+1})} + \alpha b_0 + \frac{-2\varepsilon}{h_{i+1}(h_i + h_{i+1})} + \alpha b_0,\\
&\frac{2\varepsilon}{h_i h_{i+1}} + \frac{5b_0}{6} \ > \frac{-2\varepsilon}{h_i h_{i+1}} + \frac{b_0}{12}.
\end{aligned}
$$

Second, we consider the Robin boundary conditions at the two extremities $i = 0$ and $i = N$.

For $i = 0$, $B_0 > C_0$,

$$
B_0 = a_0\left(1 + \frac{h_1^2 b_0}{3!\varepsilon} + \frac{2h_1^3 b_0'}{4!\varepsilon}\right) + a_1\sqrt{\varepsilon}\left(\frac{h_1^2 b_0'}{3!\varepsilon} + \frac{h_1^3 b_0''}{4!\varepsilon} + \frac{b_0}{\varepsilon}\left(\frac{h_1^3 b_0}{4!\varepsilon} + \frac{h_1}{2}\right) + \frac{1}{h_1}\right), \quad C_0 = -\frac{a_1\sqrt{\varepsilon}}{h_1}.
$$

For $i = N:$ $B_N > A_N$,

$$
A_N = \frac{-d_1\sqrt{\varepsilon}}{h_N}, \ B_N = d_0\left(1 + \frac{h_N^2 b_N}{3!\varepsilon} - \frac{2h_N^3 b_N'}{4!\varepsilon}\right) + d_1\sqrt{\varepsilon}\left(\frac{1}{h_N} - \frac{h_N^2 b_N'}{3!\varepsilon} + \frac{h_N^3 b_N''}{4!\varepsilon} + \frac{h_N b_N}{2\varepsilon} + \frac{h_N^3 b_N^2}{4!\varepsilon^2}\right).
$$

Thus, this implies that for each row of $M$, the sum of the two off-diagonal elements is less than or equal to the modulus of the diagonal element. Therefore, $M$ is diagonally dominant. Hence, $M$ is irreducible matrix. Henceforth, the Thomas Algorithm (Solution of general Tri-diagonal system) or Gaussian elimination method for solving the formulated scheme in Eq. (30) is stable for the described method.

## Numerical examples and results

In order to test the validity of the proposed method and to demonstrate their convergence computationally, we have taken different model examples. The maximum absolute error is evaluated by:

$$
E_\varepsilon^N = \max_{x_i \in [0,1]^N} |y(x_i) - y_i|,
$$

where $y(x_i)$ and $y_i$ respectively, denotes the exact and approximate solutions. Further, the order of convergence calculated by the formula:

$$
R = \frac{\log(E_\varepsilon^N) - \log(E_\varepsilon^{2N})}{\log(2)}.
$$

***Example 1*** Consider the singularly perturbed problem [15]

$$
\begin{cases}
-\varepsilon y''(x) + (1 + x(1-x))y(x) = f(x), \quad 0 < x < 1,\\
y(0) - \sqrt{\varepsilon}y'(0) = -1 - \sqrt{\varepsilon} + e^{\frac{-1}{\sqrt{\varepsilon}}},\\
y(1) + \sqrt{\varepsilon}y'(1) = -1 - \sqrt{\varepsilon} + e^{\frac{-1}{\sqrt{\varepsilon}}}.
\end{cases}
$$

The source function is given by

$$
\begin{aligned}
f(x) =& 1 + x(1-x) + [2\sqrt{\varepsilon} - x(1-x)^2]e^{\frac{-x}{\sqrt{\varepsilon}}}\\
&+ [2\sqrt{\varepsilon} - x^2(1-x)]e^{\frac{-(1-x)}{\sqrt{\varepsilon}}}.
\end{aligned}
$$

The exact solution to the problem that satisfies the above boundary conditions is given by.

$$
y(x) = 1 + (x-1)e^{\frac{-x}{\sqrt{\varepsilon}}} - xe^{\frac{-(1-x)}{\sqrt{\varepsilon}}}.
$$

***Example 2*** Consider the singularly perturbed problem [16]

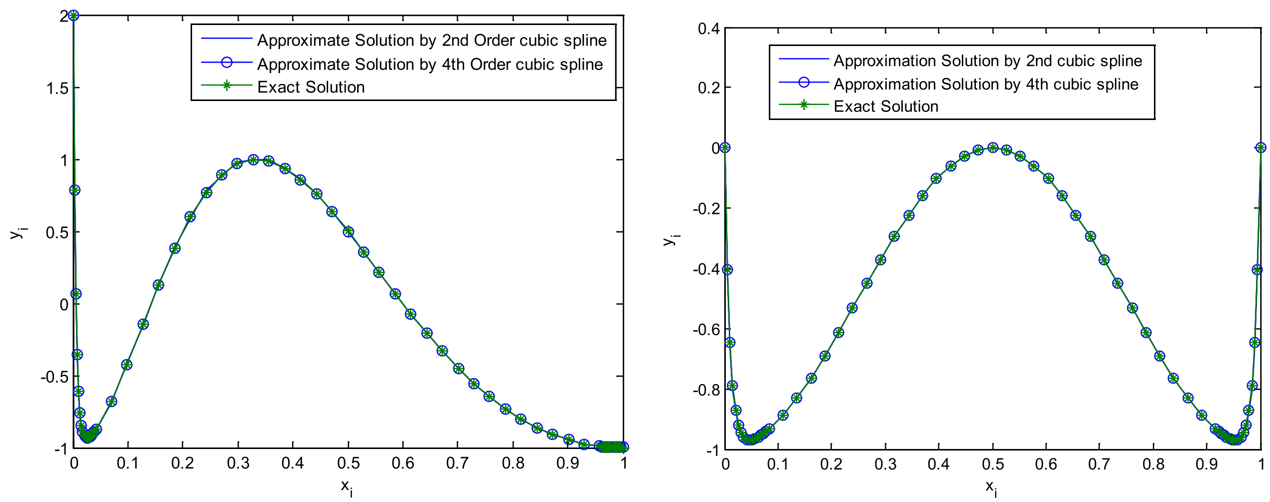


**Fig. 2** Numerical solution versus exact solution for Examples 2 and 3 respectively, when $N = 64, \ \varepsilon = 10^{-4}$

**Table 1** Computed maximum absolute errors for Example 1 by the proposed method on uniform mesh when $\alpha = \frac{1}{12}, \ \beta = \frac{5}{12}$

| $\varepsilon \downarrow N \rightarrow$ | 64 | 128 | 256 | 512 | 1024 | 2048 |
|---|---|---|---|---|---|---|
| $10^{-3}$ | 2.9485e−04 | 2.0460e−05 | 1.3407e−06 | 8.5692e−08 | 5.4143e−09 | 3.4022e−10 |
| $10^{-4}$ | 1.5204e−02 | 1.4654e−03 | 1.0970e−04 | 7.4149e−06 | 4.8043e−07 | 3.0548e−08 |
| $10^{-6}$ | 6.0163e−01 | 3.6419e−01 | 1.4058e−01 | 2.8215e−02 | 3.1066e−03 | 2.4622e−04 |
| $10^{-8}$ | 9.5028e−01 | 9.0287e−01 | 8.1506e−01 | 6.6455e−01 | 4.4341e−01 | 2.0299e−01 |
| $10^{-10}$ | 9.9491e−01 | 9.8983e−01 | 9.7975e−01 | 9.5989e−01 | 9.2137e−01 | 8.4894e−01 |
| $10^{-12}$ | 9.9949e−01 | 9.9898e−01 | 9.9796e−01 | 9.9591e−01 | 9.9184e−01 | 9.8375e−01 |
| $10^{-16}$ | 9.9999e−01 | 9.9999e−01 | 9.9998e−01 | 9.9996e−01 | 9.9992e−01 | 9.9984e−01 |

**Table 2** Computed maximum absolute errors for Example 1 by the proposed method on piecewise mesh

| $\varepsilon \downarrow N \rightarrow$ | 64 | 128 | 256 | 512 | 1024 | 2048 |
|---|---|---|---|---|---|---|
| $\alpha = \frac{1}{12}, \ \beta = \frac{5}{12}$ | | | | | | |
| $10^{-3}$ | 2.9485e−04 | 2.0460e−05 | 1.3407e−06 | 8.5692e−08 | 5.4143e−09 | 3.4022e−10 |
| $10^{-4}$ | 3.2535e−04 | 4.1312e−05 | 4.6331e−06 | 4.7689e−07 | 4.6128e−08 | 4.2567e−09 |
| $10^{-6}$ | 3.1203e−04 | 4.7604e−05 | 4.6331e−06 | 1.2766e−06 | 1.3469e−07 | 1.0771e−08 |
| $10^{-8}$ | 3.1070e−04 | 6.2649e−05 | 8.7563e−06 | 3.4607e−06 | 7.6676e−07 | 1.5275e−07 |
| $10^{-10}$ | 3.1057e−04 | 6.4374e−05 | 1.4857e−05 | 3.8392e−06 | 9.4147e−07 | 2.2909e−07 |
| $10^{-12}$ | 3.1055e−04 | 6.4549e−05 | 1.5662e−05 | 3.8792e−06 | 9.6101e−07 | 2.3864e−07 |
| $10^{-16}$ | 3.1055e−04 | 6.4568e−05 | 1.5754e−05 | 3.8836e−06 | 9.6318e−07 | 2.3971e−07 |
| $\alpha = \frac{1}{3}, \ \beta = \frac{1}{6}$ | | | | | | |
| $10^{-3}$ | 1.6639e−02 | 4.5091e−03 | 1.1885e−03 | 3.0595e−04 | 7.7666e−05 | 1.9568e−05 |
| $10^{-4}$ | 1.7402e−02 | 6.3386e−03 | 2.1634e−03 | 7.0684e−04 | 2.2270e−04 | 6.8177e−05 |
| $10^{-6}$ | 1.6980e−02 | 6.1878e−03 | 2.1102e−03 | 6.8962e−04 | 2.1718e−04 | 6.6467e−05 |
| $10^{-8}$ | 1.6941e−02 | 6.1728e−03 | 2.1049e−03 | 6.8790e−04 | 2.1663e−04 | 6.6296e−05 |
| $10^{-10}$ | 1.6936e−02 | 6.1713e−03 | 2.1043e−03 | 6.8772e−04 | 2.1658e−04 | 6.6279e−05 |
| $10^{-12}$ | 1.6936e−02 | 6.1711e−03 | 2.1043e−03 | 6.8771e−04 | 2.1657e−04 | 6.6277e−05 |
| $10^{-16}$ | 1.6936e−02 | 6.1711e−03 | 2.1043e−03 | 6.8771e−04 | 2.1657e−04 | 6.6277e−05 |

**Table 3** Computed maximum absolute errors for Example 2 by the proposed method

| $\varepsilon\downarrow N\rightarrow$ | 64 | 128 | 256 | 512 | 1024 | 2048 |
|---|---|---|---|---|---|---|
| $\alpha=\frac{1}{12},\ \beta=\frac{5}{12}$ | | | | | | |
| $10^{-1}$ | 2.2525e–05 | 1.9012e–06 | 1.8025e–07 | 1.8949e–08 | 2.1447e–09 | 2.5399e–10 |
| $10^{-3}$ | 1.6510e–03 | 2.1045e–04 | 2.3558e–05 | 2.4019e–06 | 2.2735e–07 | 2.0083e–08 |
| $10^{-6}$ | 1.2543e–03 | 1.7671e–04 | 3.8181e–05 | 7.3433e–06 | 1.1485e–06 | 1.4224e–07 |
| $10^{-9}$ | 1.2425e–03 | 1.9312e–04 | 4.6931e–05 | 1.1473e–05 | 2.7969e–06 | 6.7217e–07 |
| $10^{-12}$ | 1.2421e–03 | 1.9369e–04 | 4.7250e–05 | 1.1645e–05 | 2.8866e–06 | 7.1764e–07 |
| $\alpha=\frac{1}{3},\ \beta=\frac{1}{6}$ | | | | | | |
| $10^{-1}$ | 1.1943e–02 | 2.9936e–03 | 7.4887e–04 | 1.8727e–04 | 4.6820e–05 | 1.1705e–05 |
| $10^{-3}$ | 6.3931e–02 | 2.3480e–02 | 8.2158e–03 | 2.7294e–03 | 8.6616e–04 | 2.6643e–04 |
| $10^{-6}$ | 5.6933e–02 | 2.0894e–02 | 8.4326e–03 | 2.4112e–03 | 7.6410e–04 | 2.3476e–04 |
| $10^{-9}$ | 5.6713e–02 | 2.0813e–02 | 7.2498e–03 | 2.4013e–03 | 7.6092e–04 | 2.3378e–04 |
| $10^{-12}$ | 5.6706e–02 | 2.0811e–02 | 7.2489e–03 | 2.4010e–03 | 7.6082e–04 | 2.3375e–04 |

**Table 4** Computed maximum order of convergence for Example 2 by the proposed method

| $\varepsilon\downarrow N\rightarrow$ | 64 | 128 | 256 | 512 | 1024 |
|---|---|---|---|---|---|
| $\alpha=\frac{1}{12},\ \beta=\frac{5}{12}$ | | | | | |
| $10^{-1}$ | 3.5665 | 3.3988 | 3.2498 | 3.1433 | 3.0779 |
| $10^{-3}$ | 2.9718 | 3.1592 | 3.2940 | 3.4012 | 3.5009 |
| $10^{-6}$ | 2.8274 | 2.2105 | 2.3784 | 2.6767 | 3.0134 |
| $10^{-9}$ | 2.6857 | 2.0409 | 2.0323 | 2.0363 | 2.0569 |
| $10^{-12}$ | 2.6810 | 2.0354 | 2.0206 | 2.0123 | 2.0080 |
| $\alpha=\frac{1}{3},\ \beta=\frac{1}{6}$ | | | | | |
| $10^{-1}$ | 1.9962 | 1.9991 | 1.9996 | 1.9999 | 2.0000 |
| $10^{-3}$ | 1.4451 | 1.5150 | 1.5898 | 1.6559 | 1.7009 |
| $10^{-6}$ | 1.4462 | 1.3090 | 1.8062 | 1.6579 | 1.7026 |
| $10^{-9}$ | 1.4462 | 1.5215 | 1.5941 | 1.6580 | 1.7026 |
| $10^{-12}$ | 1.4462 | 1.5215 | 1.5941 | 1.6580 | 1.7026 |

$$\begin{cases} -\varepsilon y''(x)+\dfrac{4(1+\sqrt{\varepsilon}(1+x))}{(1+x)^4}y(x)=f(x), \quad 0<x<1,\\ y(0)-\sqrt{\varepsilon}y'(0)=2+\dfrac{6}{1-e^{\frac{-1}{\sqrt{\varepsilon}}}},\\ y(1)+\sqrt{\varepsilon}y'(1)=\dfrac{-2-e^{\frac{-1}{\sqrt{\varepsilon}}}}{2(1-e^{\frac{-1}{\sqrt{\varepsilon}}})},\end{cases}$$

With the source function is $f(x)=\frac{-4}{(1+x)^4}$ $[(1+\sqrt{\varepsilon}(1+x)+4\pi^2\varepsilon)\cos(\frac{4\pi x}{1+x})$ $-2\pi\varepsilon(1+x)\sin(\frac{4\pi x}{1+x})+3(1+\sqrt{\varepsilon}(1+x))\,\frac{e^{\frac{-1}{\sqrt{\varepsilon}}}}{1-e^{\frac{-1}{\sqrt{\varepsilon}}}}]$.

The mixed boundary conditions are chosen in a way, such that the exact solution for this problem becomes

$$y(x)=-\cos(\tfrac{4\pi x}{1+x})+3\,\frac{e^{\frac{-2x}{(1+x)\sqrt{\varepsilon}}}-e^{\frac{-1}{\sqrt{\varepsilon}}}}{1-e^{\frac{-1}{\sqrt{\varepsilon}}}}.$$

***Example 3*** Consider the singularly perturbed problem [15]

**Table 5** Comparison of maximum absolute errors and orders of convergence for Example 2, on the set for $\varepsilon\in S=\{2^{-2},\ \ldots,\ 2^{-40}\}$

| $N\rightarrow$ | 64 | 128 | 256 | 512 | 1024 | 2048 | 4096 |
|---|---|---|---|---|---|---|---|
| Present method | | | | | | | |
| $E^N$ | 1.2421e–03 | 1.9369e–04 | 4.7251e–05 | 1.1645e–05 | 2.8868e–06 | 7.1771e–07 | 1.7860e–07 |
| $R^N$ | 2.6810 | 2.0353 | 2.0206 | 2.0122 | 2.0080 | 2.0067 | – |
| Result in ref. [16] | | | | | | | |
| $E^N$ | 5.0900e–03 | 1.2696e–03 | 3.1841e–04 | 8.0216e–5 | 2.0094e–05 | 5.0358e–6 | 1.2601e–06 |
| $R^N$ | 2.0033 | 1.9954 | 1.9889 | 1.9971 | 1.9965 | 1.9987 | – |

**Table 6** Comparison of maximum absolute errors and orders of convergence for Example 3

| $\varepsilon \downarrow N \rightarrow$ | 64 | 128 | 256 | 512 | 1024 | 2048 |
|---|---|---|---|---|---|---|
| Present method | | | | | | |
| $10^{-1}$ | 1.5804e−07 | 9.5736e−09 | 6.0110e−10 | 3.8343e−11 | 2.4281e−12 | 1.7421e−13 |
| | 4.0101 | 3.9934 | 3.9706 | 3.9811 | 3.8009 | |
| $10^{-2}$ | 4.4426e−06 | 2.8574e−07 | 1.8107e−08 | 1.1394e−09 | 7.1458e−11 | 4.4797e−12 |
| | 3.9586 | 3.9801 | 3.9902 | 3.9950 | 3.9956 | |
| $10^{-3}$ | 3.8440e−04 | 2.6622e−05 | 1.7427e−06 | 1.1132e−07 | 7.0321e−09 | 4.4198e−10 |
| | 3.8519 | 3.9332 | 3.9685 | 3.9846 | 3.9919 | |
| $10^{-4}$ | 4.6569e−04 | 5.9110e−05 | 6.6270e−06 | 6.8198e−07 | 1.0174e−07 | 2.3644e−08 |
| | 2.9779 | 3.1570 | 3.2806 | 2.7448 | 2.1053 | |
| Result in ref. [15] | | | | | | |
| $10^{-1}$ | 3.208e−04 | 7.957e−05 | 2.005e−05 | 5.007e−06 | 1.252e−06 | 3.131e−07 |
| | 2.01 | 1.99 | 2.00 | 2.00 | 2.00 | |
| $10^{-2}$ | 4.066e−04 | 1.017e−04 | 2.544e−05 | 6.359e−06 | 1.590e−06 | 3.975e−07 |
| | 2.00 | 2.00 | 2.00 | 2.00 | 2.00 | |
| $10^{-3}$ | 3.998e−04 | 1.003e−04 | 2.509e−05 | 6.274e−06 | 1.569e−06 | 3.922e−07 |
| | 1.99 | 2.00 | 2.00 | 2.00 | 2.00 | |
| $10^{-4}$ | 3.859e−04 | 9.938e−05 | 2.504e−05 | 6.271e−06 | 1.568e−06 | 3.922e−07 |
| | 1.96 | 1.99 | 2.00 | 2.00 | 2.00 | |

**Table 7** Comparison of maximum absolute errors and orders of convergence for Example 4, on the set for $\varepsilon \in S = \left\{2^{-2}, \ldots, 2^{-40}\right\}$

| $N \rightarrow$ | 64 | 128 | 256 | 512 | 1024 | 2048 | 4096 |
|---|---|---|---|---|---|---|---|
| Present Method | | | | | | | |
| $E^N$ | 2.5673e−04 | 3.2707e−05 | 3.6738e−06 | 3.7844e−07 | 7.4034e−08 | 3.6837e−08 | 9.1670e−09 |
| $R^N$ | 2.9726 | 3.1543 | 3.2791 | 2.3538 | 1.0070 | 2.0066 | – |
| Result in ref. [16] | | | | | | | |
| $E^N$ | 8.3652e−04 | 1.9349e−04 | 4.6383e−05 | 1.1343e−05 | 2.8032e−06 | 6.9671e−07 | 1.7369e−07 |
| $R^N$ | 2.1122 | 2.0606 | 2.0319 | 2.0166 | 2.0085 | 2.0040 | – |

$$\begin{cases} -\varepsilon y''(x) + y(x) = -\cos^2(\pi x) = 2\varepsilon\pi^2\cos(2\pi x), \quad 0 < x < 1, \\ y(0) - 2\sqrt{\varepsilon}y'(0) = 2 - \dfrac{4e^{\frac{-1}{\sqrt{\varepsilon}}}}{1 + e^{\frac{-1}{\sqrt{\varepsilon}}}}, \\ y(1) + 3\sqrt{\varepsilon}y'(1) = \dfrac{3(1 - e^{\frac{-1}{\sqrt{\varepsilon}}})}{1 + e^{\frac{-1}{\sqrt{\varepsilon}}}}. \end{cases}$$

The exact solution that satisfies the given boundary conditions is given by

$$y(x) = \frac{e^{\frac{-(1-x)}{\sqrt{\varepsilon}}} + e^{\frac{-x}{\sqrt{\varepsilon}}}}{1 + e^{\frac{-1}{\sqrt{\varepsilon}}}} - \cos^2(\pi x)$$

***Example 4*** Consider singularly perturbed problem [16]

$$\begin{cases} -\varepsilon y''(x) + y(x) = -\cos^2(\pi x) - 2\varepsilon\pi^2\cos(2\pi x), \quad x \in (0,1) \\ y(0) - 2\sqrt{\varepsilon}y'(0) = 1, \\ y(1) + 3\sqrt{\varepsilon}y'(1) = 0. \end{cases}$$

As the exact solution for the Example 4 is not available, so the accuracy of its numerical solution will be computed using double mesh principle (Figs. 1, 2 and Tables 1, 2, 3, 4, 5, 6, 7).

## Conclusion

In conclusion, this paper presents a novel family of non-polynomial cubic spline approximation methods for solving singularly perturbed reaction–diffusion problems with Robin-type boundary conditions. By utilizing piecewise mesh lengths, this approach effectively addresses the challenge posed by boundary layers, resulting in a numerical scheme of either second- or fourth-order convergence. Through the transformation of boundary conditions into a fourth-order finite difference scheme and the construction of a tridiagonal system, the method achieves stability and consistency, ensuring convergence. The diagonally dominant nature of the coefficient matrix allows for efficient solution using Gaussian elimination, highlighting the method's computational effectiveness.

The numerical results further emphasize the advantages of this method, particularly in handling the small positive perturbation parameter that compromises accuracy on a uniform mesh. When applied to a piecewise mesh, finer discretization's in the boundary layer and coarser sizes in the outer region significantly improve accuracy, as demonstrated by numerical illustrations. The decrease in maximum absolute error with increasing mesh points substantiates the method's convergence. The proposed cubic spline approach outperforms existing methods in terms of accuracy and convergence, closely approximating the exact solution with fewer errors. Overall, this method offers a highly accurate and efficient solution for Robin-type singularly perturbed problems, aligning well with theoretical predictions.

## Limitations

Various approaches can be applied for the finite difference approximation of differential equations and boundary conditions. However, these differing methods may introduce inaccuracies when applied to the specific problem at hand.

**Acknowledgements**
The authors thank Department of Mathematics, Jimma University for their material supports.

**Author contributions**
BE led the development of the scheme. TAB and GFD were responsible for writing the MATLAB code for the scheme and contributed to drafting the manuscript. All authors reviewed and approved the final version of the manuscript.

**Funding**
No funding organization for this research work.

**Availability of data and materials**
No datasets were generated or analysed during the current study.

## Declarations

**Ethics approval and consent to participate**
Not applicable.

**Consent for publication**
Not applicable.

**Competing interests**
The authors declare no competing interests.

Received: 17 September 2024 Accepted: 21 January 2025
Published online: 26 February 2025

**References**

1. Rai P, Sharma KK. Numerical study of singularly perturbed differential–difference equation arising in the modeling of neuronal variability. Comput Math Appl. 2012;63(1):118–32.
2. Reda BT, Bullo TA, Duressa GF. Fourth-order fitted mesh scheme for semilinear singularly perturbed reaction–diffusion problems. BMC Res Notes. 2023;16(1):354.
3. Asrat T, File G, Aga T. Fourth-order stable central difference method for self-adjoint singular perturbation problems. Ethiop J Sci Technol. 2016;9(1):53–68.
4. Kumar M. A recent development of computer methods for solving singularly perturbed boundary value problems. Int J Differ Equ. 2011. https://doi.org/10.1155/2011/404276.
5. Wondimu F, File G, Aga T. Fourth order compact finite difference method for solving singularly perturbed 1D reaction diffusion equations with Dirichlet boundary conditions. Momona Ethiop J Sci. 2016;8(2):168–81.
6. Mariappan M, Tamilselvan A. Higher order numerical method for a semilinear system of singularly perturbed differential equations. Math Commun. 2021;26(1):41–52.
7. Miller JJH, O'riordan E, Shishkin GI. Fitted numerical methods for singular perturbation problems: error estimates in the maximum norm for linear problems in one and two dimensions. Singapore: World Scientific; 1996.
8. Roos H-G, Stynes M, Tobiska L. Robust numerical methods for singularly perturbed differential equations: convection-diffusion-reaction and flow problems, vol. 24. Berlin: Springer Science & Business Media; 2008.
9. Mishra P, Fairweather G, Sharma KK. A parameter uniform orthogonal spline collocation method for singularly perturbed semilinear reaction-diffusion problems in one dimension. Int J Comput Methods Eng Sci Mech. 2019;20(5):336–46.
10. Duru H, Güneş B. The finite difference method on adaptive mesh for singularly perturbed nonlinear 1D reaction diffusion boundary value problems. J Appl Math Comput Mech. 2020;19(4):45–56.
11. File G, Edosa F. Higher order compact finite difference method for singularly perturbed one dimensional reaction diffusion problems. J Niger Math Soc. 2017;36(3):491–502.
12. Galu FW, Duressa GF, Bullo TA. Tenth order compact finite difference method for solving singularly perturbed 1D reaction-diffusion equations. Int J Eng Appl Sci. 2016;8(3):15–24.
13. Gelu FW, Duressa GF, Bullo TA. Sixth-order compact finite difference method for singularly perturbed 1D reaction diffusion problems. J Taibah Univ Sci. 2017;11(2):302–8.
14. Siraj MK, Duressa GF, Bullo TA. Fourth-order stable central difference with Richardson extrapolation method for second-order self-adjoint singularly perturbed boundary value problems. J Egypt Math Soc. 2019;27(1):1–14.
15. Gupta A, Kaushik A. A robust spline difference method for robin-type reaction-diffusion problem using grid equidistribution. Appl Math Comput. 2021;390:125597.
16. Das P, Natesan S. Higher-order parameter uniform convergent schemes for Robin type reaction-diffusion problems using adaptively generated grid. Int J Comput Methods. 2012;9(04):1250052.
17. Liu LB, Long G, Ouyang A, Huang Z. Numerical solution of a singularly perturbed problem with Robin boundary conditions using particle swarm optimization algorithm. J Intell Fuzzy Syst. 2017;33(3):1785–95.
18. Natesan S, Bawa RK. Second-order numerical scheme for singularly perturbed reaction-diffusion Robin problems J. Numer Anal Ind Appl Math. 2007;2(3–4):177–92.
19. Prandtl L. Uber Flussigkeitsbewegung bei sehr kleiner Reibung. Verhandl. In: 3rd Int Math Kongr Heidelberg (1904), Leipzig. 1905.
20. Wasow WR. on boundary layer problems in the theory of ordinary differential equation. New Yor: New York University; 1942.
21. Kadalbajoo M, Reddy YN. Asymptotic and numerical analysis of singular perturbation problems: a survey. Appl Math Comput. 1989;30:223–59. https://doi.org/10.1016/0096-3003(89)90054-4.
22. Sharma KK, Rai P, Patidar KC. A review on singularly perturbed differential equations with turning points and interior layers. Appl Math Comput. 2013;219(22):10575–609.

23. Kadalbajoo MK, Gupta V. Numerical solution of singularly perturbed convection-diffusion problem using parameter uniform B-spline collocation method. J Math Anal Appl. 2009;355(1):439–52.
24. Barati A, Nabati M. A numerical technique for solving a class of nonlinear singularly perturbed boundary value problems. Filomat. 2021;35(10):3353–64. https://doi.org/10.2298/FIL2110353B.
25. Yamac K, Erdogan F. A numerical scheme for semilinear singularly perturbed reaction-diffusion problems. Appl Math Nonlinear Sci. 2020;5(1):405–12.
26. Isaacson E, Keller HB. Analysis of numerical methods. Chelmsford: Courier Corporation; 1994.
27. Gupta A, Kaushik A, Sharma M. A higher-order hybrid spline difference method on adaptive mesh for solving singularly perturbed parabolic reaction–diffusion problems with robin-boundary conditions. Numer Methods Partial Differ Equ. 2023;39(2):1220–50.
28. Sharma N, Kaushik A. A uniformly convergent difference method for singularly perturbed parabolic partial differential equations with large delay and integral boundary condition. J Appl Math Comput. 2023;69(1):1071–93.

## Publisher's Note