Mathematics

https://www.aimspress.com/journal/Math

AIMS Mathematics, 11(1): 2279–2312.
DOI: 10.3934/math.2026092
Received: 10 October 2025
Revised: 02 January 2026
Accepted: 12 January 2026
Published: 23 January 2026

Research article

A high-order nonuniform compact difference scheme coupled with adaptive mesh method for solving singularly perturbed convection-diffusion equations

Tao Wang[1,2,*], Jinhua Ma[1] and Qiong Chen[1]

[1] School of Mathematics and Information Science, North Minzu University, Yinchuan, 750021, China

[2] Ningxia Collaborative Innovation Center for Scientific Computing and Intelligent Information Processing, Yinchuan, 750021, China

* **Correspondence:** Email: wangtao@nmu.edu.cn.

Abstract: A high-order nonuniform compact difference scheme coupled with an adaptive mesh method for solving singularly perturbed convection-diffusion equations is proposed. The core idea of the proposed method lies in constructing a nonuniform grid scheme with the same fourth-order accuracy at both inner and boundary points, and this construction is based on an adaptive mesh method. This approach effectively eliminates numerical oscillation near the boundaries, a common challenge in singularly perturbed problems. Specifically, we first formulate the fourth-order compact scheme on nonuniform grids, then integrate it with the adaptive mesh method and elaborate on the corresponding numerical implementation procedure. Finally, numerical experiments are conducted against exact solutions, with the proposed scheme further compared against three benchmark methods: the fourth-order compact scheme on uniform grids, the identical scheme on adaptive nonuniform grids, and the other established methods reported in existing literature. Results of all test cases demonstrate that the proposed scheme generates accurate and stable numerical solutions and exhibits enhanced resolution for singularly perturbed convection-diffusion problems.

Keywords: singularly perturbed convection-diffusion equation; adaptive mesh method; consistent fourth-order nonuniform scheme; boundary layer
Mathematics Subject Classification: 65D25, 65N06, 65Y04, 65Z05

1. Introduction

The singularly perturbed convection-diffusion equation with parameters is a type of partial differential equation in which the convection term involves a very small parameter. The small parameters change the solution rapidly in a certain small region, such as near boundaries, forming what are known as boundary layers. These abrupt changes pose considerable challenges for traditional numerical methods, particularly when uniform grids are employed, as such grids may fail to accurately capture these fine-grained features. More and more researchers are paying attention to the numerical solution of this problem [1].

The main numerical methods for solving singularly perturbed convection-diffusion equation are the finite difference method [2], finite element method [3, 4], finite volume method [5, 6], spectral method [7, 8], and others. To overcome the difficulties caused by boundary layers, several effective solution strategies have been proposed. Common strategies include the fitted operator method [9], adaptive mesh method [10, 11], and layer-adapted method [12–14]. Each of these methods has its own advantages and disadvantages. The adaptive mesh method, which refines the mesh, moves the mesh, or increase the order of basis functions, better reflects the essence of the problem and has gradually gained favor among many scholars [15, 16]. The adaptive mesh method only needs to find a control function to continuously help optimize the mesh. In this paper, the adaptive mesh method is a grid generation algorithm that dynamically adjusts the distribution of grid nodes based on problem-solving requirements. Its core idea is to automatically refine the mesh in regions, where the solution changes rapidly and reduce mesh density in smooth regions by monitoring the change gradient of the solution on a nonuniform grid, thereby optimizing computational efficiency while ensuring calculation accuracy.

To couple with the adaptive mesh method, the first step is to construct a high-order finite difference scheme with a nonuniform grid. However, high-order difference schemes generally only achieve high order at the inner point, and it is much harder to achieve the same accuracy at the boundary. Wang Tao [17] et al. proposed the construction of a principle of a consistent high-order scheme. The consistent high-order scheme is used to solve the problem of the low-order scheme in the boundary that affects the calculation accuracy of the whole scheme. Therefore, coupled with the adaptive mesh method, we construct a nonuniform fourth-order compact difference scheme for solving the singularly perturbed convection-diffusion equation on nonuniform grids. It can efficiently alleviate or even eliminate numerical difficulties caused by boundary layers, making the numerical solution closer to the proper solution.

This work aims to develop a nonuniform scheme with the same fourth-order accuracy at both inner and boundary points, and this construction is coupled with the adaptive mesh method for solving the singularly perturbed convection-diffusion equation. The outline of the paper is as follows: In Section 2, we propose a consistent fourth-order accuracy scheme with nonuniform meshes for solving the singularly perturbed convection-diffusion equation. In Section 3, the accuracy of the proposed scheme is verified using an exact solution problem. We also simulate a numerically different singularly perturbed convection-diffusion problem based on the method proposed in this paper. Some concluding remarks are given in Section 4.

2. Numerical method

Let us consider the following one-dimensional singularly perturbed convection-diffusion equation:

$$\begin{cases} -\varepsilon u'' + b(x)u' + c(x)u = f(x), \quad x \in (a, b), \\ u(a) = u_a, u(b) = u_b, \end{cases} \tag{2.1}$$

where ε is constant, with $\varepsilon > 0$, and is the diffusion coefficient; $b(x)$ is the convection coefficient; $f(x)$ is the source term; $u(x)$ is the unknown quantity to be computed; $b(x)$, $c(x)$, $f(x)$, and $u(x)$ are all sufficiently smooth functions that are known to be sufficiently smooth functions of x.

2.1. Interior scheme

In this work, Eq (2.1) will be solved by the fourth-order compact difference scheme with a nonuniform grid.

We denote the nonuniform grid with step sizes h_i ($h_i \neq h_{i+1}$) and the grid node $x_i = a + ih_i$, $i = 0, ..., N$. We use u_i to represent the value of u at grid node x_i, $u_i^{''}$ and $u_i^{'}$ to represent approximations of $u^{''}$ and $u^{'}$ at node x_i, $i = 0, ..., N$. We note $k_i = \frac{h_{i+1}}{h_i}$. Figure 1 shows the computational stencil for interior nodes.

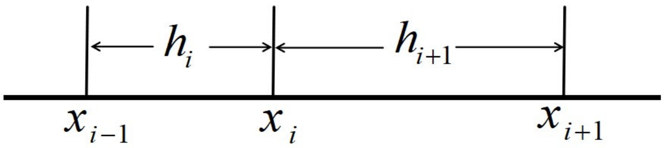


Figure 1. Computational stencil for interior nodes.

By the Taylor expansion, the fourth-order compact scheme with a nonuniform grid of the second derivative is given:

$$\begin{aligned}&-\frac{10k_i(k_i^2-k_i-1)}{k_i^3+4k_i^2+4k_i+1}u_{i-1}^{''}+10u_i^{''}+\frac{10(k_i^2+k_i-1)}{k_i^3+4k_i^2+4k_i+1}u_{i+1}^{''}\\ =&\frac{120}{(k_i+1)(k_i^2+3k_i+1)}\frac{u_{i+1}-u_i}{h_i^2}+\frac{120k_i}{(k_i+1)(k_i^2+3k_i+1)}\frac{u_{i-1}-u_i}{h_i^2},\quad i=2,3,...,N-2.\end{aligned}\tag{2.2}$$

It is noted that the truncation error of scheme (2.2) is

$$-\frac{k_i(3k_i^4+2k_i^3-7k_i^2+2k_i+3)}{12(k_i^2+3k_i+1)}h^4u_i^6+O(h^6).\tag{2.3}$$

When $k_i = 1$ in scheme (2.2), the scheme corresponds to the uniform grid scheme [18]. That is,

$$u_{i-1}^{''}+10u_i^{''}+u_{i+1}^{''}=\frac{12}{h^2}(u_{i-1}-2u_i+u_{i+1}),\quad i=2,3,...,N-2.\tag{2.4}$$

The truncation error of scheme (2.4) is $-\frac{1}{20}h^4u_i^{(6)}+O(h^6)$.

By the Taylor expansion, the fourth-order compact scheme with a nonuniform grid of the first derivative is given:

$$\begin{aligned}&\frac{4k_i^2}{(k_i+1)^2}u_{i-1}^{'}+4u_i^{'}+\frac{4}{(k_i+1)^2}u_{i+1}^{'}\\ =&\frac{8(2k_i+1)}{k_i(k_i+1)^3}\frac{(u_{i+1}-u_i)}{h_i}-\frac{8k_i^2(k_i+2)}{(k_i+1)^3}\frac{(u_{i-1}-u_i)}{h_i},\quad i=2,3,...,N-2.\end{aligned}\tag{2.5}$$

It is noted that the truncation error of scheme (2.5) is

$$-\frac{k_i^2}{30}h^4u_i^5+O(h^5).\tag{2.6}$$

When $k_i = 1$ in scheme (2.5), the scheme corresponds to the uniform grid scheme [19]. That is,

$$u'_{i-1} + 4u'_i + u'_{i+1} = \frac{3}{h}(-u_{i-1} + u_{i+1}), \quad i = 2, 3, ..., N-2. \tag{2.7}$$

The truncation error of scheme (2.7) is $-\frac{1}{60}h^4 u_i^5 + O(h^5)$.

2.2. *Boundary scheme*

To solve the singularly perturbed convection-diffusion equation with the above interior scheme, we must give the value for the first and second derivatives of the boundary points u at $i = 1, N-1$.

In Ref. [17], it is noted that the consistent fourth-order compact scheme used here means that the difference schemes applied have fourth-order accuracy and maintain this precision through the computational domain, even where the grid changes. So, we construct the consistent fourth-order boundary scheme with a nonuniform grid. The consistent boundary scheme must satisfy the following three conditions [17]: (i) The order of accuracy of the boundary scheme is the same as the interior scheme; (ii) The major truncation error term of the boundary scheme is identical to that of the interior scheme; (iii) The coefficient matrix of the algebraic system is strictly diagonally dominant. From the above definition, it is easy to conclude that the entire computation is numerically stable if a consistent scheme is applied.

2.2.1. Diffusion term boundary scheme

Theorem 2.1. *The consistent fourth-order boundary nonuniform scheme of the second derivative exists.*

Proof. By using Taylor series expansion, the fourth-order boundary scheme of the second derivative with boundary parameters is derived.

We give the fourth-order left boundary nonuniform scheme of the second derivative as

$$\alpha_1 u''_1 + \alpha_2 u''_2 + \alpha_3 u''_3 + \alpha_4 u''_4 + \alpha_5 u''_5 = \frac{1}{h_1^2}(a_0 u_0 + a_1 u_1 + a_2 u_2), \tag{2.8}$$

where the notation in Figure 2 is used.

Figure 2 shows the schematic of nonuniform left boundary nodes, where 0 represents the boundary node, 1 represents the near-boundary point, and 2 represents the internal node adjacent to the near-boundary point (i.e., the sub-near-boundary point). Let $h_i = x_i - x_{i-1}$, $k_i = \frac{h_{i+1}}{h_i}$, where $i = 1, 2, \ldots, 5$.

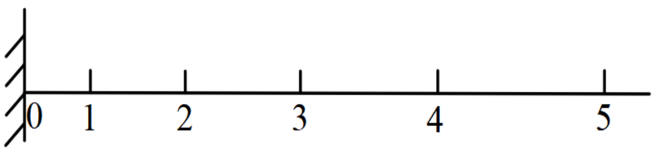


Figure 2. Computational stencil for left boundary nodes.

The expression (2.8) of the parameters is as follows:

$$
\begin{aligned}
a_0 &= A_{01}\alpha_4 + A_{02}\alpha_5,\\
a_1 &= A_{11}\alpha_4 + A_{12}\alpha_5,\\
a_2 &= A_{21}\alpha_4 + A_{22}\alpha_5,\\
\alpha_1 &= A_{31}\alpha_4 + A_{32}\alpha_5,\\
\alpha_2 &= A_{41}\alpha_4 + A_{42}\alpha_5,\\
\alpha_3 &= A_{51}\alpha_4 + A_{52}\alpha_5.
\end{aligned}
$$

The coefficients of the free unknown parameters are respectively

$$
\begin{aligned}
A_{01} &= \frac{-60k_1^3k_2^2k_3(k_3+1)(k_2+k_2k_3+1)}{(k_1+1)(7k_1+5k_1k_2+5k_1^2k_2-5k_1^3k_2+3k_1^2-3k_1^3+3)},\\
A_{02} &= \frac{60k_1^3k_2^2k_3(k_4+1)(k_3+k_3k_4+1)(k_2+k_2k_3+k_2k_3k_4+1)}{(k_1+1)(7k_1+5k_1k_2+5k_1^2k_2-5k_1^3k_2+3k_1^2-3k_1^3+3)},\\
A_{11} &= \frac{60k_1^2k_2^2k_3(k_3+1)(k_2+k_2k_3+1)}{(7k_1+5k_1k_2+5k_1^2k_2-5k_1^3k_2+3k_1^2-3k_1^3+3)},\\
A_{12} &= \frac{60k_1^2k_2^2k_3(k_4+1)(k_3+k_3k_4+1)(k_2+k_2k_3+k_2k_3k_4+1)}{(7k_1+5k_1k_2+5k_1^2k_2-5k_1^3k_2+3k_1^2-3k_1^3+3)},\\
A_{21} &= \frac{-60k_1^2k_2^2k_3(k_3+1)(k_2+k_2k_3+1)}{(k_1+1)(7k_1+5k_1k_2+5k_1^2k_2-5k_1^3k_2+3k_1^2-3k_1^3+3)},\\
A_{22} &= \frac{-60k_1^2k_2^2k_3(k_4+1)(k_3+k_3k_4+1)(k_2+k_2k_3+k_2k_3k_4+1)}{(k_1+1)(7k_1+5k_1k_2+5k_1^2k_2-5k_1^3k_2+3k_1^2-3k_1^3+3)},\\
A_{31} &= \frac{-k_2^2k_3(k_3+1)(a_{31}+3)}{(k_1+1)(7k_1+5k_1k_2+5k_1^2k_2-5k_1^3k_2+3k_1^2-3k_1^3+3)},\\
A_{32} &= \frac{k_2^2k_3(k_4+1)(k_3+k_3k_4+1)(a_{32}+3)}{(k_1+1)(7k_1+5k_1k_2+5k_1^2k_2-5k_1^3k_2+3k_1^2-3k_1^3+3)},\\
A_{41} &= \frac{k_3(k_2+k_2k_3+1)(a_{41}+3)}{(7k_1+5k_1k_2+5k_1^2k_2-5k_1^3k_2+3k_1^2-3k_1^3+3)},\\
A_{42} &= \frac{k_3(k_4+1)(k_2+k_2k_3+k_2k_3k_4+1)(a_{42}+3)}{(7k_1+5k_1k_2+5k_1^2k_2-5k_1^3k_2+3k_1^2-3k_1^3+3)},\\
A_{51} &= \frac{-(k_3+1)(k_2+k_2k_3+1)(a_{51}+3)}{(k_2+1)(7k_1+5k_1k_2+5k_1^2k_2-5k_1^3k_2+3k_1^2-3k_1^3+3)},\\
A_{52} &= \frac{-(k_3+k_3k_4+1)(k_2+k_2k_3+k_2k_3k_4+1)(a_{52}+3)}{(k_2+1)(7k_1+5k_1k_2+5k_1^2k_2-5k_1^3k_2+3k_1^2-3k_1^3+3)},
\end{aligned}
$$

where

$$
\begin{aligned}
a_{31} &= 12k_1+18k_1^2+12k_1^3+10k_1k_2+30k_1^2k_2+30k_1^3k_2+10k_1^2k_2^2+20k_1^3k_2^2+10k_1^2k_2^2k_3\\
&\quad +20k_1^3k_2^2k_3+5k_1k_2k_3+15k_1^2k_2k_3+15k_1^3k_2k_3+3,\\
a_{32} &= a_{31}+5k_1k_2k_3k_4+15k_1^2k_2k_3k_4+15k_1^3k_2k_3k_4+10k_1^2k_2^2k_3k_4+20k_1^3k_2^2k_3k_4,
\end{aligned}
$$

$$
\begin{aligned}
a_{41} &= 7k_1 + 3k_1^2 - 3k_1^3 + 10k_1k_2 + 10k_1^2k_2 - 10k_1^3k_2 + 10k_1^2k_2^2 - 10k_1^3k_2^2 + 10k_1^2k_2^2k_3 \\
&\quad - 10k_1^3k_2^2k_3 + 5k_1k_2k_3 + 5k_1^2k_2k_3 - 5k_1^3k_2k_3 + 3, \\
a_{42} &= a_{41} + 5k_1k_2k_3k_4 + 5k_1^2k_2k_3k_4 - 5k_1^3k_2k_3k_4 + 10k_1^2k_2^2k_3k_4 - 10k_1^3k_2^3k_3k_4, \\
a_{51} &= 7k_1 + 3k_1^2 - 3k_1^3 + 5k_1k_2 + 5k_1^2k_2 - 5k_1^3k_2 + 5k_1k_2k_3 + 5k_1^2k_2k_3 - 5k_1^3k_2k_3, \\
a_{52} &= a_{51} + 5k_1k_2k_3k_4 + 5k_1^2k_2k_3k_4 - 5k_1^3k_2k_3k_4.
\end{aligned}
$$

The truncation error of the scheme (2.8) is $(C_{11}\alpha_4 + C_{12}\alpha_5)h_1^4u_1^6 + O(h_1^5)$, where there are two free unknown α_4, α_5 for optional adjustments.

In order to ensure that the error of the boundary scheme is the same as the interior scheme, we let

$$
C_{11}\alpha_4 + C_{12}\alpha_5 = -\frac{k_i(3k_i^4 + 2k_i^3 - 7k_i^2 + 2k_i + 3)}{12(k_i^2 + 3k_i + 1)}.
$$

If we require the system to be stable when the boundary scheme is coupled with the above interior scheme, i.e., the coefficient matrix of the resultant system is diagonally dominant, we let

$$
A_{01}\alpha_4 + A_{02}\alpha_5 = \frac{120k_i}{(k_i + 1)(k_i^2 + 3k_i + 1)}.
$$

So, we get

$$
\begin{aligned}
\alpha_4 &= \frac{-(2k_ic_4)}{k_1^4k_2^3k_3^2k_4(k_i + 1)(k_3 + 1)(k_i^2 + 3k_i + 1)(k_2 + k_2k_3 + 1)}, \\
\alpha_5 &= \frac{2k_ic_5}{k_1^4k_2^3k_3^2k_4(k_i + 1)(k_4 + 1)(k_i^2 + 3k_i + 1)(k_3 + k_3k_4 + 1)(k_2 + k_2k_3 + k_2k_3k_4 + 1)}, \\
C_{11} &= \frac{-(k_1^3k_2^2k_3(k_3 + 1)(k_2 + k_2k_3 + 1)(c_1 + 2))}{24(7k_1 + 5k_1k_2 + 5k_1^2k_2 - 5k_1^3k_2 + 3k_1^2 - 3k_1^3 + 3)}, \\
C_{12} &= \frac{-(k_1^3k_2^2k_3(k_4 + 1)(k_3 + k_3k_4 + 1)(k_2 + k_2k_3 + k_2k_3k_4 + 1)(c_2 + 2))}{24(7k_1 + 5k_1k_2 + 5k_1^2k_2 - 5k_1^3k_2 + 3k_1^2 - 3k_1^3 + 3)},
\end{aligned}
$$

where

$$
\begin{aligned}
c_1 =& 7k_1 + 8k_1^2 + 2k_1^3 - 2k_1^4 + 6k_1k_2 + 14k_1^2k_2 + 6k_1^3k_2 - 6k_1^4k_2 + 5k_1^2k_2^2 + 5k_1^3k_2^2 - 5k_1^4k_2^2 \\
&+ 5k_1^2k_2^2k_3 + 5k_1^3k_2^2k_3 - 5k_1^4k_2^2k_3 + 3k_1k_2k_3 + 7k_1^2k_2k_3 + 3k_1^3k_2k_3 - 3k_1^4k_2k_3, \\
c_2 =& c_1 + 3k_1k_2k_3k_4 + 7k_1^2k_2k_3k_4 + 3k_1^3k_2k_3k_4 - 3k_1^4k_2k_3k_4 + 5k_1^2k_2^2k_3k_4 + 5k_1^3k_2^2k_3k_4 - 5k_1^4k_2^2k_3k_4, \\
c_5 =& 5k_i - 5k_i^2 - 5k_i^3 + 5k_i^4 + 3k_i^5 + 9k_1 + 15k_1^2 + 10k_1^3 - 2k_1^5 + 6k_1k_2 + 20k_1^2k_2 + 20k_1^3k_2 \\
&- 6k_1^5k_2 + 5k_1^2k_2^2 + 10k_1^3k_2^2 - 5k_1^5k_2^2 + 5k_1^2k_2^2k_3 + 10k_1^3k_2^2k_3 - 5k_1^5k_2^2k_3 + 3k_1k_2k_3 \\
&+ 10k_1^2k_2k_3 + 10k_1^3k_2k_3 - 3k_1^5k_2k_3 + 5, \\
c_4 =& c_5 + 3k_1k_2k_3k_4 + 10k_1^2k_2k_3k_4 + 10k_1^3k_2k_3k_4 - 3k_1^5k_2k_3k_4 + 5k_1^2k_2^2k_3k_4 + 10k_1^3k_2^2k_3k_4 - 5k_1^5k_2^2k_3k_4.
\end{aligned}
$$

Then, the consistent fourth-order left boundary scheme of the second derivative is

$$
\alpha_1u_1^{''} + \alpha_2u_2^{''} + \alpha_3u_3^{''} + \alpha_4u_4^{''} + \alpha_5u_5^{''} = \frac{1}{h_1^2}(a_0u_0 + a_1u_1 + a_2u_2). \tag{2.9}
$$

The coefficients of the free unknown parameters of scheme (2.9) are given (see Appendix A). The truncation error of scheme (2.9) is

$$-\frac{k_i(3k_i^4+2k_i^3-7k_i^2+2k_i+3)}{12(k_i^2+3k_i+1)}h_1^4u_1^{(6)}+O(h_1^6). \tag{2.10}$$

The boundary scheme is a uniform grid scheme with $k_i = 1$. Then, the boundary scheme with a uniform grid is

$$15u_1''-9u_2''+10u_3''-5u_4''+u_5''=\frac{12}{h^2}(u_0-2u_1+u_2). \tag{2.11}$$

Similarly, we develop the fourth-order right boundary nonuniform scheme as

$$\begin{aligned}&\alpha_{N-1}u_{N-1}''+\alpha_{N-2}u_{N-2}''+\alpha_{N-3}u_{N-3}''+\alpha_{N-4}u_{N-4}''+\alpha_{N-5}u_{N-5}''\\ =&\frac{1}{h_N^2}(a_Nu_N+a_{N-1}u_{N-1}+a_{N-2}u_{N-2}),\end{aligned} \tag{2.12}$$

where the notation in Figure 3 is used.

Figure 3 shows the schematic of nonuniform right boundary nodes, where N represents the boundary node, N-1 represents the near-boundary point, and N-2 represents the internal node adjacent to the near-boundary point (i.e., the sub-near-boundary point). Let $h_i = x_i - x_{i-1}$, $k_i = \frac{h_{i-1}}{h_i}$, where $i = N, N-1, \ldots, N-5$.

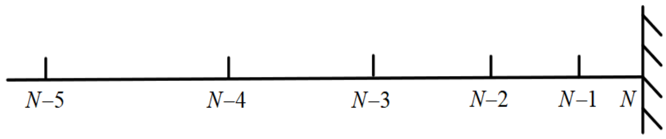


Figure 3. Computational stencil for right boundary nodes.

The expression (2.12) of the parameters is as follows

$$\begin{aligned}a_N&=A_{N01}\alpha_{N-4}+A_{N02}\alpha_{N-5},\\ a_{N-1}&=A_{N11}\alpha_{N-4}+A_{N12}\alpha_{N-5},\\ a_{N-2}&=A_{N21}\alpha_{N-4}+A_{N22}\alpha_{N-5},\\ \alpha_{N-1}&=A_{N31}\alpha_{N-4}+A_{N32}\alpha_{N-5},\\ \alpha_{N-2}&=A_{N41}\alpha_{N-4}+A_{N42}\alpha_{N-5},\\ \alpha_{N-3}&=A_{N51}\alpha_{N-4}+A_{N52}\alpha_{N-5}.\end{aligned}$$

The coefficients of the free unknown parameters are respectively

$$\begin{aligned}A_{N01}&=\frac{-(60k_N^3k_{N-1}^2k_{N-2})(k_{N-2}+1)(k_{N-1}k_{N-2}+1)}{(k_N+1)(7k_N+5k_{N-1}k_N+5k_N^2k_{N-1}-5k_N^3k_{N-1}+3k_N^2-3k_N^3+3)},\\ A_{N02}&=\frac{-(60k_N^3k_{N-1}^2k_{N-2}(k_{N-3}+1)}{(k_N+1))}\cdot\frac{(k_{N-2}+k_{N-2}k_{N-3}+1)(k_{N-1}+k_{N-1}k_{N-2}+k_{N-1}k_{N-2}k_{N-3}+1)}{(7k_N+5k_Nk_{N-1}+5k_N^2k_{N-1}-5k_N^3k_{N-1}+3k_N^2-3k_N^3+3)},\\ A_{N11}&=\frac{(60k_N^2k_{N-1}^2k_{N-2})(k_{N-2}+1)(k_{N-1}k_{N-2}+1)}{(7k_N+5k_{N-1}k_N+5k_N^2k_{N-1}-5k_N^3k_{N-1}+3k_N^2-3k_N^3+3)},\end{aligned}$$

$$
\begin{aligned}
A_{N12} &= \frac{(60k_N^2k_{N-1}^2k_{N-2})(k_{N-3}+1)(k_{N-2}+k_{N-2}k_{N-3}+1)}{(7k_N+5k_Nk_{N-1}+5k_N^2k_{N-1}-5k_N^3k_{N-1}+3k_N^2-3k_N^3+3)} \cdot \frac{(k_{N-1}+k_{N-1}k_{N-2}+k_{N-1}k_{N-2}k_{N-3}+1)}{1},\\
A_{N21} &= \frac{-(60k_N^2k_{N-1}^2k_{N-2})(k_{N-2}+1)(k_{N-1}k_{N-2}+1)}{(k_N+1)(7k_N+5k_Nk_{N-1}+5k_N^2k_{N-1}-5k_N^3k_{N-1}+3k_N^2-3k_N^3+3)},\\
A_{N22} &= \frac{-(60k_N^2k_{N-1}^2k_{N-2})(k_{N-3}+1)(k_{N-2}+k_{N-2}k_{N-3}+1)}{(k_N+1)}\\
&\quad \cdot \frac{(k_{N-1}+k_{N-1}k_{N-2}+k_{N-1}k_{N-2}k_{N-3}+1)}{(7k_N+5k_Nk_{N-1}+5k_N^2k_{N-1}-5k_N^3k_{N-1}+3k_N^2-3k_N^3+3)},\\
A_{N31} &= \frac{-(k_{N-1}^2k_{N-2})(k_{N-2}+1)(a_{N31}+3)}{(k_{N-1}+1)(7k_N+5k_Nk_{N-1}+5k_N^2k_{N-1}-5k_N^3k_{N-1}+3k_N^2-3k_N^3+3)},\\
A_{N32} &= \frac{-(k_{N-1}^2k_{N-2})(k_{N-3}+1)(k_{N-2}+k_{N-2}k_{N-3}+1)(a_{N32}+3)}{(k_{N-1}+1)(7k_N+5k_Nk_{N-1}+5k_N^2k_{N-1}-5k_N^3k_{N-1}+3k_N^2-3k_N^3+3)},\\
A_{N41} &= \frac{k_{N-2}(k_{N-1}+k_{N-1}k_{N-2}+1)(a_{N41}+3)}{7k_N+5k_Nk_{N-1}+5k_N^2k_{N-1}-5k_N^3k_{N-1}+3k_N^2-3k_N^3+3},\\
A_{N42} &= \frac{k_{N-2}(k_{N-3}+1)(k_{N-1}+k_{N-1}k_{N-2}+k_{N-1}k_{N-2}k_{N-3}+1)(a_{N42}+3)}{7k_N+5k_Nk_{N-1}+5k_N^2k_{N-1}-5k_N^3k_{N-1}+3k_N^2-3k_N^3+3},\\
A_{N51} &= \frac{-(k_{N-2}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)(a_{N51}+3)}{(k_{N-1}+1)(7k_N+5k_Nk_{N-1}+5k_N^2k_{N-1}-5k_N^3k_{N-1}+3k_N^2-3k_N^3+3}),\\
A_{N52} &= \frac{-(k_{N-2}+k_{N-2}k_{N-3}+1)(k_{N-1}+k_{N-1}k_{N-2}+k_{N-1}k_{N-2}k_{N-3}+1)(a_{N52}+3)}{(k_{N-1}+1)(7k_N+5k_Nk_{N-1}+5k_N^2k_{N-1}-5k_N^3k_{N-1}+3k_N^2-3k_N^3+3},
\end{aligned}
$$

where

$$
\begin{aligned}
a_{N31} =& 12k_N+18k_N^2+12k_N^3+10k_Nk_{N-1}+30k_N^2k_{N-1}+30k_N^3k_{N-1}+10k_N^2k_{N-1}^2\\
&+20k_N^3k_{N-1}^2+10k_N^2k_{N-1}^2k_{N-2}+20k_N^3k_{N-1}^2k_{N-2}+5k_Nk_{N-1}k_{N-2}\\
&+15k_N^2k_{N-1}k_{N-2}+15k_N^3k_{N-1}k_{N-2}+3,\\
a_{N32} =& a_{N31}+5k_Nk_{N-1}k_{N-2}k_{N-3}+15k_N^2k_{N-1}k_{N-2}k_{N-3}+15k_N^3k_{N-1}k_{N-2}k_{N-3}\\
&+10k_N^2k_{N-1}^2k_{N-2}k_{N-3}+20k_N^3k_{N-1}^2k_{N-2}k_{N-3},\\
a_{N41} =& 7k_N+3k_N^2-3k_N^3+10k_Nk_{N-1}+10k_N^2k_{N-1}-10k_N^3k_{N-1}+10k_N^2k_{N-1}^2\\
&-10k_N^3k_{N-1}^2+10k_N^2k_{N-1}^2k_{N-2}-10k_N^3k_{N-1}^2k_{N-2}+5k_Nk_{N-1}k_{N-2}\\
&+5k_N^2k_{N-1}k_{N-2}-5k_N^3k_{N-1}k_{N-2}+3,\\
a_{N42} =& a_{N41}+5k_Nk_{N-1}k_{N-2}k_{N-3}+5k_N^2k_{N-1}k_{N-2}k_{N-3}-5k_N^3k_{N-1}k_{N-2}k_{N-3}\\
&+10k_N^2k_{N-1}^2k_{N-2}k_{N-3}-10k_N^3k_{N-1}^3k_{N-2}k_{N-3},\\
a_{N51} =& 7k_N+3k_N^2-3k_N^3+5k_Nk_{N-1}+5k_N^2k_{N-1}-5k_N^3k_{N-1}+5k_Nk_{N-1}k_{N-2}\\
&+5k_N^2k_{N-1}k_{N-2}-5k_N^3k_{N-1}k_2,\\
a_{N52} =& a_{N51}+5k_Nk_{N-1}k_{N-2}k_{N-3}+5k_N^2k_{N-1}k_{N-2}k_{N-3}-5k_N^3k_{N-1}k_{N-2}k_{N-3}.
\end{aligned}
$$

Its truncation error is $(C_{N1}\alpha_{N-4}+C_{N2}\alpha_{N-5})h_{N-1}^4u_{N-1}^6+O(h_{N-1}^6)$, where there are two free unknowns $\alpha_{N-4},\alpha_{N-5}$ for optional adjustments.

In order to ensure that the error of the boundary scheme is the same as the interior scheme, we let

$$C_{N1}\alpha_{N-4}+C_{N2}\alpha_{N-5}=-\frac{k_i(3k_i^4+2k_i^3-7k_i^2+2k_i+3)}{12(k_i^2+3k_i+1)}.$$

If we require the system to be stable when the boundary scheme is coupled with the above interior scheme, i.e., the coefficient matrix of the resultant system is diagonally dominant, we let

$$A_{N1}\alpha_{N-4}+A_{N2}\alpha_{N-5}=\frac{120k_i}{(k_i+1)(k_i^2+3k_i+1)}.$$

So, we get

$$\begin{aligned}
\alpha_{N-4}&=\frac{-2k_i c_{N4}}{(k_N^4k_{N-1}^3k_{N-2}^2k_{N-3})(k_i+1)(k_{N-2}+1)(k_i^2+3k_i+1)(k_{N-1}+k_{N-1}k_{N-2}+1)},\\
\alpha_{N-5}&=\frac{2k_i c_{N5}}{(k_N^4k_{N-1}^3k_{N-2}^2k_{N-3})(k_i+1)(k_{N-3}+1)(k_i^2+3k_i+1)c_N},\\
C_{N1}&=\frac{-(k_N^3k_{N-1}^2k_{N-2})(k_{N-2}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)(c_{N1}+2)}{24(7k_N+5k_Nk_{N-1}+5k_N^2k_{N-1}-5k_N^3k_{N-1}+3k_N^2-3k_N^3+3)},\\
C_{N2}&=\frac{-(k_N^3k_{N-1}^2k_{N-2})(k_{N-3}+1)c_N(c_{N2}+2)}{24(7k_N+5k_Nk_{N-1}+5k_N^2k_{N-1}-5k_N^3k_{N-1}+3k_N^2-3k_N^3+3)},
\end{aligned}$$

where

$$\begin{aligned}
c_N=&(k_{N-2}+k_{N-2}k_{N-3}+1)(k_{N-1}+k_{N-1}k_{N-2}+k_{N-1}k_{N-2}k_{N-3}+1),\\
c_{N1}=&7k_N+8k_N^2+2k_N^3-2k_N^4+6k_Nk_{N-1}+14k_N^2k_{N-1}+6k_N^3k_{N-1}-6k_N^4k_{N-1}\\
&+5k_N^2k_{N-1}^2+5k_N^3k_{N-1}^2-5k_N^4k_{N-1}^2+5k_N^2k_{N-1}^2k_{N-2}+5k_N^3k_{N-1}^2k_{N-2}\\
&-5k_N^4k_{N-1}^2k_{N-2}+3k_Nk_{N-1}k_{N-2}+7k_N^2k_{N-1}k_{N-2}+3k_N^3k_{N-1}k_{N-2}-3k_N^4k_{N-1}k_{N-2},\\
c_{N2}=&c_{N1}+3k_Nk_{N-1}k_{N-2}k_{N-3}+7k_N^2k_{N-1}k_{N-2}k_{N-3}+3k_N^3k_{N-1}k_{N-2}k_{N-3}\\
&-3k_N^4k_{N-1}k_{N-2}k_{N-3}+5k_N^2k_{N-1}^2k_{N-2}k_{N-3}+5k_N^3k_{N-1}^2k_{N-2}k_{N-3}-5k_N^4k_{N-1}^2k_{N-2}k_{N-3},\\
c_{N5}=&5k_i-5k_i^2-5k_i^3+5k_i^4+3k_i^5+9k_N+15k_N^2+10k_N^3-2k_N^5+6k_Nk_{N-1}+20k_N^2k_{N-1}\\
&+20k_N^3k_{N-1}-6k_N^5k_{N-1}+5k_N^2k_{N-1}^2+10k_N^3k_{N-1}^2-5k_N^5k_{N-1}^2+5k_N^2k_{N-1}^2k_{N-2}\\
&+10k_N^3k_{N-1}^2k_{N-2}-5k_N^5k_{N-1}^2k_{N-2}+3k_Nk_{N-1}k_{N-2}+10k_N^2k_{N-1}k_{N-2}\\
&+10k_N^3k_{N-1}k_{N-2}-3k_N^5k_{N-1}k_{N-2}+5,\\
c_{N4}=&c_{N5}+3k_Nk_{N-1}k_{N-2}k_{N-3}+10k_N^2k_{N-1}k_{N-2}k_{N-3}+10k_N^3k_{N-1}k_{N-2}k_{N-3}\\
&-3k_N^5k_{N-1}k_{N-2}k_{N-3}+5k_N^2k_{N-1}^2k_{N-2}k_{N-3}+10k_N^3k_{N-1}^2k_{N-2}k_{N-3}-5k_N^5k_{N-1}^2k_{N-2}k_{N-3}.
\end{aligned}$$

Then, the consistent fourth-order right boundary scheme of the second derivative is

$$\begin{aligned}
&\alpha_{N-1}u''_{N-1}+\alpha_{N-2}u''_{N-2}+\alpha_{N-3}u''_{N-3}+\alpha_{N-4}u''_{N-4}+\alpha_{N-5}u''_{N-5}\\
=&\frac{1}{h_N^2}(a_Nu_N+a_{N-1}u_{N-1}+a_{N-2}u_{N-2}).
\end{aligned}\tag{2.13}$$

The coefficients of the free unknown parameters of scheme (2.13) are given (see Appendix B). The truncation error of scheme (2.13) is

$$-\frac{k_i(3k_i^4+2k_i^3-7k_i^2+2k_i+3)}{12(k_i^2+3k_i+1)}h_{N-1}^4 u_{N-1}^6 + O(h_{N-1}^6). \tag{2.14}$$

The boundary scheme is a uniform grid scheme with $k_i = 1$. Then, the boundary scheme with a uniform grid is

$$15u''_{N-1} - 9u''_{N-2} + 10u''_{N-3} - 5u''_{N-4} + u''_{N-5} = \frac{12}{h^2}(u_N - 2u_{N-1} + u_{N-2}). \tag{2.15}$$

□

By combining the interior scheme (2.2), the left boundary scheme (2.9), and the right boundary scheme (2.13) into a unified system, the matrix form is given as follows:

$$M_2^4 U'' = A_2^4 U + H_2^4, \tag{2.16}$$

where

$$M_2^4 = \begin{pmatrix} \alpha_1 & \alpha_2 & \alpha_3 & \cdots & 0 & 0 & 0 \\ \frac{-10k_2(k_2^2-k_2-1)}{k_2^3+4k_2^2+4k_2+1} & 10 & \frac{10(k_2^2+k_2-1)}{k_2^3+4k_2^2+4k_2+1} & \cdots & 0 & 0 & 0 \\ & & & \ddots & & & \\ 0 & 0 & 0 & \cdots & \frac{-10k_{N-2}(k_{N-2}^2-k_{N-2}-1)}{k_{N-2}^3+4k_{N-2}^2+4k_{N-2}+1} & 10 & \frac{10(k_{N-2}^2+k_{N-2}-1)}{k_{N-2}^3+4k_{N-2}^2+4k_{N-2}+1} \\ 0 & 0 & 0 & \cdots & \alpha_{N-3} & \alpha_{N-2} & \alpha_{N-1} \end{pmatrix},$$

$$U'' = \begin{pmatrix} u''_1 \\ u''_2 \\ u''_3 \\ \vdots \\ u''_{N-3} \\ u''_{N-2} \\ u''_{N-1} \end{pmatrix}, U = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{N-3} \\ u_{N-2} \\ u_{N-1} \end{pmatrix}, H_2^4 = \begin{pmatrix} a_0u_0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ a_Nu_N \end{pmatrix},$$

$$A_2^4 = \frac{12}{h_i^2}\begin{pmatrix} a_1 & a_2 & \cdots & 0 & 0 \\ \frac{120k_2}{(k_2+1)(k_2^2+3k_2+1)} & \frac{-120k_2-120}{(k_2+1)(k_2^2+3k_2+1)} & \cdots & 0 & 0 \\ & & \ddots & & \\ 0 & 0 & \cdots & \frac{-120k_{N-2}-120}{(k_{N-2}+1)(k_{N-2}^2+3k_{N-2}+1)} & \frac{120k_{N-2}}{(k_{N-2}+1)(k_{N-2}^2+3k_{N-2}+1)} \\ 0 & 0 & \cdots & a_{N-2} & a_{N-1} \end{pmatrix}.$$

The matrices M_2^4, A_2^4 are $(N-1)\times(N-1)$ dimensions, and the matrices U'', U, H_2^4 are $(N-1)\times 1$ dimensions.

When $k_i = 1$ $(i = 2, ..., N-2)$ in (2.16), the discretization employs a uniform grid, and the corresponding matrix is

$$M_2^4 = \begin{pmatrix} 15 & -9 & 10 & -5 & 1 & \cdots & 0 & 0 & 0 & 0 & 0 \\ 1 & 10 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 10 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ & & & \ddots & & & & & & & \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 10 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 & 10 & 1 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 & -5 & 10 & -9 & 15 \end{pmatrix},$$

$$U'' = \begin{pmatrix} u_1'' \\ u_2'' \\ u_3'' \\ \vdots \\ u_{N-3}'' \\ u_{N-2}'' \\ u_{N-1}'' \end{pmatrix}, U = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{N-3} \\ u_{N-2} \\ u_{N-1} \end{pmatrix}, H_2^4 = \begin{pmatrix} \frac{12}{h^2} u_0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ \frac{12}{h^2} u_N \end{pmatrix},$$

$$A_2^4 = \frac{12}{h^2} \begin{pmatrix} -2 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 1 & -2 & 1 & \cdots & 0 & 0 & 0 \\ 0 & 1 & -2 & \cdots & 0 & 0 & 0 \\ & & & \ddots & & & \\ 0 & 0 & 0 & \cdots & -2 & 1 & 0 \\ 0 & 0 & 0 & \cdots & 1 & -2 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 1 & -2 \end{pmatrix}.$$

2.2.2. Convection term boundary scheme

Theorem 2.2. *The consistent fourth-order boundary nonuniform scheme of the first derivative exists.*

Proof. By using Taylor series expansion, the fourth-order boundary scheme of the first-order derivative with boundary parameters is derived.

We give the fourth-order left boundary nonuniform scheme of the first derivative as

$$\beta_1 u_1' + \beta_2 u_2' = \frac{1}{h_1}(b_0 u_0 + b_1 u_1 + b_2 u_2 + b_3 u_3 + b_4 u_4 + b_5 u_5), \tag{2.17}$$

where the notation in Figure 2 is used.

The expression (2.17) of the right parameter is as follows:

$$\begin{aligned} b_0 &= B_{01} b_5 + B_{02} \beta_1 + B_{03} \beta_2, \\ b_1 &= B_{11} b_5 + B_{12} \beta_1 + B_{13} \beta_2, \\ b_2 &= B_{21} b_5 + B_{22} \beta_1 + B_{23} \beta_2, \\ b_3 &= B_{31} b_5 + B_{32} \beta_1 + B_{33} \beta_2, \\ b_4 &= B_{41} b_5 + B_{42} \beta_1 + B_{43} \beta_2. \end{aligned}$$

The coefficients of the free unknowns are respectively given out as follows:

$$
\begin{aligned}
B_{01} &= \frac{-(k_1^4k_2^3k_3^2k_4)(k_4+1)(k_3+k_3k_4+1)(k_2+k_2k_3+k_2k_3k_4+1)}{(k_1+1)(k_1+k_1k_2+1)(k_1+k_1k_2+k_1k_2k_3+1)},\\
B_{02} &= \frac{-k_1^3(k_2+1)(k_2+k_2k_3+1)}{(k_1+1)(k_1+k_1k_2+1)(k_1+k_1k_2+k_1k_2k_3+1)},\\
B_{03} &= \frac{k_1^3k_2^2(k_3+1)}{(k_1+1)(k_1+k_1k_2+1)(k_1+k_1k_2+k_1k_2k_3+1)},\\
B_{11} &= \frac{(k_2^3k_3^2k_4(k_4+1)(k_3+k_3k_4+1)(k_1+k_1k_2+k_1k_2k_3+k_1k_2k_3k_4+1)}{(k_2+1)(k_2+k_2k_3+1)},\\
B_{12} &= \frac{(k_1-4k_2+2k_1k_2-2k_2k_3+k_1k_2^2-k_2^2k_3-k_2^2+k_1k_2k_3+k_1k_2^2k_3-3)}{k_1(k_2+1)(k_2+k_2k_3+1)},\\
B_{13} &= \frac{-k_2^2(k_1+1)(k_3+1)}{k_1(k_2+1)(k_2+k_2k_3+1)},\\
B_{21} &= \frac{-k_3^2k_4(k_4+1)(k_2+k_2k_3+k_2k_3k_4+1)(k_1+k_1k_2+k_1k_2k_3+k_1k_2k_3k_4+1)}{(k_1+1)(k_3+1)},\\
B_{22} &= \frac{(k_2+1)(k_2+k_2k_3+1)}{(k_1k_2^2)(k_1+1)(k_3+1)},\\
B_{23} &= \frac{-(2k_1-k_2+k_3-2k_1k_2+k_1k_3-2k_1k_2k_3+2)}{k_1k_2(k_1+1)(k_3+1)},\\
B_{31} &= \frac{k_4(k_3+k_3k_4+1)(k_2+k_2k_3+k_2k_3k_4+1)(k_1+k_1k_2+k_1k_2k_3+k_1k_2k_3k_4+1)}{(k_2+1)(k_1+k_1k_2+1)},\\
B_{32} &= \frac{-(k_2+k_2k_3+1)}{k_1k_2^2k_3(k_2+1)(k_1+k_1k_2+1)},\\
B_{33} &= \frac{(k_1+1)(k_3+1)}{k_1k_2k_3(k_2+1)(k_1+k_1k_2+1)},\\
B_{41} &= \frac{-(k_4+1)(k_3+k_3k_4+1)(k_2+k_2k_3+k_2k_3k_4+1)(k_1+k_1k_2+k_1k_2k_3+k_1k_2k_3k_4+1)}{(k_3+1)(k_2+k_2k_3+1)(k_1+k_1k_2+k_1k_2k_3+1)},\\
B_{42} &= \frac{(k_2+1)}{(k_1k_2^2k_3)(k_3+1)(k_2+k_2k_3+1)(k_1+k_1k_2+k_1k_2k_3+1)},\\
B_{43} &= \frac{-(k_1+1)}{(k_1k_2k_3)(k_3+1)(k_2+k_2k_3+1)(k_1+k_1k_2+k_1k_2k_3+1)}.
\end{aligned}
$$

The truncation error of the scheme (2.17) is $(C_{01}b_5 + C_{02}\beta_1 + C_{03}\beta_2)h_1^4u_1^5 + O(h_1^5)$, where are three free unknowns b_5, β_1, β_2 for optional adjustment.

In order to ensure that the error of the boundary scheme is the same as the interior scheme, we let

$$C_{01}b_5 + C_{02}\beta_1 + C_{03}\beta_2 = -\frac{k_i^2}{30}.$$

If we require the system to be stable when the boundary scheme is coupled with the above interior scheme, i.e., the coefficient matrix of the resultant system is diagonally dominant, it is required that $\beta_1 = 4,\ \beta_2 = \frac{4}{(k_i+1)^2}$.

Then, we get

$$b_5 = -\frac{120}{k_1^4k_2^3k_3^2k_4(k_4+1)(k_3+k_3k_4+1)(k_2+k_2k_3+k_2k_3k_4+1)} \cdot \frac{(\frac{k_i^2}{30} + \frac{k_1^3(k_2+1)(k_2+k_2k_3+1)}{30} - \frac{k_1^3k_2^2(k_1+1)(k_3+1)}{30(k_i+1)^2})}{(k_1+k_1k_2+k_1k_2k_3+k_1k_2k_3k_4+1)},$$

where

$$\begin{aligned} C_{01} =& \frac{-(k_1^4k_2^3k_3^2k_4)(k_4+1)(k_3+k_3k_4+1)(k_2+k_2k_3+k_2k_3k_4+1)}{120} \\ &\cdot \frac{(k_1+k_1k_2+k_1k_2k_3+k_1k_2k_3k_4+1)}{1}, \\ C_{02} =& \frac{k_1^3(k_2+1)(k_2+k_2k_3+1)}{120}, \\ C_{03} =& \frac{-(k_1^3k_2^2)(k_1+1)(k_3+1)}{120}. \end{aligned}$$

Then, the consistents fourth-order left boundary scheme of the first derivative is

$$\beta_1 u_1^{'} + \beta_2 u_2^{'} = \frac{1}{h_1}(b_0u_0 + b_1u_1 + b_2u_2 + b_3u_3 + b_4u_4 + b_5u_5). \tag{2.18}$$

The coefficients of the free unknown parameters of scheme (2.18) are given (see Appendix C). The truncation error of scheme (2.18) is

$$-\frac{k_i^2}{30}h_1^4u_1^5 + O(h_1^5). \tag{2.19}$$

The boundary scheme is a uniform grid scheme with $k_i = 1$. Then, the boundary scheme with a uniform grid is

$$4u_1^{'} + u_2^{'} = \frac{1}{h}(-5u_1 + 8u_2 - \frac{10}{3}u_3 + \frac{5}{4}u_4 - \frac{1}{5}u_5) - \frac{43}{60h}u_0. \tag{2.20}$$

Similarly, we develop the fourth-order right boundary nonuniform scheme as

$$\beta_{N-1}u_{N-1}^{'} + \beta_{N-2}u_{N-2}^{'} = \frac{1}{h}(b_Nu_N + b_{N-1}u_{N-1} + b_{N-2}u_{N-2} + b_{N-3}u_{N-3} + b_{N-4}u_{N-4} + b_{N-5}u_{N-5}), \tag{2.21}$$

where the notation in Figure 3 is used.

The expression (2.21) of the right parameter is as follows:

$$\begin{aligned} b_N &= B_{N1}b_{N-5} + B_{N2}\beta_{N-1} + B_{N3}\beta_{N-2}, \\ b_{N-1} &= B_{N11}b_{N-5} + B_{N12}\beta_{N-1} + B_{N13}\beta_{N-2}, \\ b_{N-2} &= B_{N21}b_{N-5} + B_{N22}\beta_{N-1} + B_{N23}\beta_{N-2}, \\ b_{N-3} &= B_{N31}b_{N-5} + B_{N32}\beta_{N-1} + B_{N33}\beta_{N-2}, \\ b_{N-4} &= B_{N41}b_{N-5} + B_{N42}\beta_{N-1} + B_{N43}\beta_{N-2}. \end{aligned}$$

The coefficients of the free unknowns are respectively given out as follows:

$$B_{N1} = \frac{-(k_N^4k_{N-1}^3k_{N-2}^2k_{N-3})(k_{N-3}+1)(k_{N-2}+k_{N-2}k_{N-3}+1)}{(k_N+1)(k_N+k_Nk_{N-1}+1)}$$

$$
\begin{aligned}
&\cdot \frac{(k_{N-1}+k_{N-1}k_{N-2}+k_{N-1}k_{N-2}k_{N-3}+1)}{(k_N+k_Nk_{N-1}+k_Nk_{N-1}k_{N-2}+1)},\\
B_{N2} &= \frac{4k_N^3(k_{N-1}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)}{(k_N+1)(k_N+k_Nk_{N-1}+1)(k_N+k_Nk_{N-1}+k_Nk_{N-1}k_{N-2}+1)},\\
B_{N3} &= \frac{-(k_N^3k_{N-1}^2)(k_{N-2}+1)}{(k_N+1)(k_N+k_Nk_{N-1}+1)(k_N+k_Nk_{N-1}+k_Nk_{N-1}k_{N-2}+1)},\\
B_{N11} &= \frac{(k_{N-1}^3k_{N-2}^2k_{N-3})(k_{N-3}+1)(k_{N-2}+k_{N-2}k_{N-3}+1)}{(k_{N-1}+1)}\\
&\cdot \frac{(k_N+k_Nk_{N-1}+k_Nk_{N-1}k_{N-2}+k_Nk_{N-1}k_{N-2}k_{N-3}+1)}{(k_{N-1}+k_{N-1}k_{N-2}+1)},\\
B_{N12} &= \frac{-(k_N-4k_{N-1}+2k_Nk_{N-1}-2k_{N-1}k_{N-2}+k_Nk_{N-1}^2-k_{N-1}^2k_{N-2}-k_{N-1}^2)}{k_N(k_{N-1}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)}\\
&- \frac{(k_Nk_{N-1}k_{N-2})k_Nk_{N-1}^2k_{N-2}-3)}{k_N(k_{N-1}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)},\\
B_{N13} &= \frac{k_{N-1}^2(k_N+1)(k_{N-2}+1)}{k_N(k_{N-1}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)},\\
B_{N21} &= \frac{-(k_{N-2}^2k_{N-3})(k_{N-3}+1)(k_{N-1}+k_{N-1}k_{N-2}+k_{N-1}k_{N-2}k_{N-3}+1)}{(k_N+1)}\\
&\cdot \frac{(k_N+k_Nk_{N-1}+k_Nk_{N-1}k_{N-2}+k_Nk_{N-1}k_{N-2}k_{N-3}+1)}{(k_{N-2}+1)},\\
B_{N22} &= \frac{-(k_{N-1}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)}{k_Nk_{N-1}^2(k_N+1)(k_{N-2}+1)},\\
B_{N23} &= \frac{(2k_N-k_{N-1}+k_{N-2}-2k_Nk_{N-1}+k_Nk_{N-2}-2k_Nk_{N-1}k_{N-2}+2)}{k_Nk_{N-1}(k_N+1)(k_{N-2}+1)},\\
B_{N31} &= \frac{k_{N-3}(k_{N-2}+k_{N-2}k_{N-3}+1)(k_{N-1}+k_{N-1}k_{N-2}+k_{N-1}k_{N-2}k_{N-3}+1)}{(k_{N-1}+1)}\\
&\cdot \frac{(k_N+k_Nk_{N-1}+k_Nk_{N-1}k_{N-2}+k_Nk_{N-1}k_{N-2}k_{N-3}+1)}{(k_N+k_Nk_{N-1}+1)},\\
B_{N32} &= \frac{(k_{N-1}+k_{N-1}k_{N-2}+1)}{(k_Nk_{N-1}^2k_{N-2})(k_{N-1}+1)(k_N+k_Nk_{N-1}+1)},\\
B_{N33} &= \frac{-(k_N+1)(k_{N-2}+1)}{k_Nk_{N-1}k_{N-2}(k_{N-1}+1)(k_N+k_Nk_{N-1}+1)},\\
B_{N41} &= \frac{-(k_{N-3}+1)(k_{N-2}+k_{N-2}k_{N-3}+1)(k_{N-1}+k_{N-1}k_{N-2}+k_{N-1}k_{N-2}k_{N-3}+1)}{(k_{N-2}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)}\\
&\cdot \frac{(k_N+k_Nk_{N-1}+k_Nk_{N-1}k_{N-2}+k_Nk_{N-1}k_{N-2}k_{N-3}+1)}{(k_Nk_Nk_{N-1}+k_Nk_{N-1}k_{N-2}+1)},\\
B_{N42} &= \frac{-(k_{N-1}+1)}{k_Nk_{N-1}^2k_{N-2}(k_{N-2}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)(k_N+k_Nk_{N-1}+k_Nk_{N-1}k_{N-2}+1)},\\
B_{N43} &= \frac{(k_N+1)}{k_Nk_{N-1}k_{N-2}(k_{N-2}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)(k_N+k_Nk_{N-1}+k_Nk_{N-1}k_{N-2}+1)}.
\end{aligned}
$$

The truncation error is

$$(C_{N1}b_{N-5} + C_{N2}\beta_{N-1} + C_{N3}\beta_{N-2})h_{N-1}^4 u_{N-1}^5 + O(h_{N-1}^5),$$

where there are three free unknowns $b_{N-5}, \beta_{N-1}, \beta_{N-2}$ for optional adjustment.

In order to ensure that the error of the boundary scheme is the same as the interior scheme, we let

$$C_{N1}b_{N-5} + C_{N2}\beta_{N-1} + C_{N3}\beta_{N-2} = -\frac{k_i^2}{30}.$$

If we require the system to be stable when the boundary scheme is coupled with the above interior scheme, i.e., the coefficient matrix of the resultant system is diagonally dominant, it is required that $\beta_{N-1} = 4, \beta_{N-2} = \frac{4}{(k_i+1)^2}$.

So, we get

$$b_{N-5} = \frac{120(\frac{k_i^2}{30} + \frac{k_N^3(k_{N-1}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)}{30} - \frac{k_N^3 k_{N-1}^2(k_{N-2}+1)(k_N+1)}{30(k_i+1)^2}}{b_{51}},$$

where

$$\begin{aligned}
C_{N1} &= \frac{-(k_N^4 k_{N-1}^3 k_{N-2}^2 k_{N-3})(k_{N-3}+1)(k_{N-2}+k_{N-2}k_{N-3}+1)}{120} \cdot \frac{(k_{N-1}+k_{N-1}k_{N-2}+k_{N-1}k_{N-2}k_{N-3}+1)}{1} \\
&\quad \cdot \frac{(k_N + k_{N-1}k_N + k_{N-1}k_{N-2}k_N + k_{N-1}k_{N-2}k_{N-3}k_N + 1)}{1}, \\
C_{N2} &= \frac{k_N^3(k_{N-1}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)}{120}, \\
C_{N3} &= \frac{-(k_N^3 k_{N-1}^2)(k_{N-2}+1)(k_N+1)}{120}, \\
b_{51} &= k_N^4 k_{N-1}^3 k_{N-2}^2 k_{N-3}(k_{N-3}+1)(k_{N-2}+k_{N-2}k_{N-3}+1)(k_{N-1}+k_{N-1}k_{N-2} \\
&\quad + k_{N-1}k_{N-2}k_{N-3}+1)(k_N + k_{N-1} + k_N + k_{N-1}k_{N-2}k_N + k_{N-1}k_{N-2}k_{N-3}k_N + 1).
\end{aligned}$$

Then, the consistent fourth-order right boundary scheme of the first derivative is

$$\beta_{N-1}u'_{N-1} + \beta_{N-2}u'_{N-2} = \frac{1}{h}(b_N u_N + b_{N-1}u_{N-1} + b_{N-2}u_{N-2} + b_{N-3}u_{N-3} + b_{N-4}u_{N-4} + b_{N-5}u_{N-5}). \quad (2.22)$$

The coefficients of the free unknown parameters of scheme (2.22) are given (see Appendix D). The truncation error of scheme (2.22) is

$$-\frac{k_i^2}{30}h_{N-1}^4 u_{N-1}^5 + O(h_{N-1}^5). \quad (2.23)$$

The boundary scheme is a uniform grid scheme with $k_i = 1$. Then, the boundary scheme with a uniform grid is

$$4u'_{N-1} + u'_{N-2} = \frac{1}{h}(5u_{N-1} - 8u_{N-2} + \frac{10}{3}u_{N-3} - \frac{5}{4}u_{N-4} + \frac{1}{5}u_{N-5}) + \frac{43}{60h}u_N. \quad (2.24)$$

□

By combining the interior scheme (2.5), the left boundary scheme (2.18), and the right boundary scheme (2.22) into a unified system, we obtain the matrix form as follows:

$$A_1^4 U' = M_1^4 U + H_1^4, \tag{2.25}$$

where,

$$A_1^4 = \begin{pmatrix} 4 & \frac{4}{(k_1+1)^2} & 0 & \cdots & 0 & 0 & 0 \\ \frac{4k_2^2}{(k_2+1)^2} & 4 & \frac{4}{(k_2+1)^2} & \cdots & 0 & 0 & 0 \\ & & & \ddots & & & \\ 0 & 0 & 0 & \cdots & \frac{4k_{N-2}^2}{(k_{N-2}+1)^2} & 4 & \frac{4}{(k_{N-2}+1)^2} \\ 0 & 0 & 0 & \cdots & 0 & \frac{4}{(k_{N-1}+1)^2} & 4 \end{pmatrix},$$

$$U^{'} = \begin{pmatrix} u_1^{'} \\ u_2^{'} \\ u_3^{'} \\ \vdots \\ u_{N-3}^{'} \\ u_{N-2}^{'} \\ u_{N-1}^{'} \end{pmatrix}, U = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{N-3} \\ u_{N-2} \\ u_{N-1} \end{pmatrix}, H_1^4 = \begin{pmatrix} b_0 u_0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ b_N u_N \end{pmatrix},$$

$$M_1^4 = \frac{1}{h_i} \begin{pmatrix} b_1 & b_2 & \cdots & 0 & 0 \\ \frac{-8k_2^2(k_2+2)}{(k_2+1)^3} & \frac{8\left(k_2^3(k_2+2)-(2k_2+1)\right)}{k_2(k_2+1)^3} & \cdots & 0 & 0 \\ & & \ddots & & \\ 0 & 0 & \cdots & \frac{8\left(k_{N-2}^3(k_{N-2}+2)-(2k_{N-2}+1)\right)}{k_{N-2}(k_{N-2}+1)^3} & \frac{8(2k_{N-2}+1)}{k_{N-2}(k_{N-2}+1)^3} \\ 0 & 0 & \cdots & b_{N-2} & b_{N-1} \end{pmatrix}.$$

The matrices A_1^4, M_1^4 are $(N-1)\times(N-1)$ dimensions, and the matrices $U^{'}$, U, H_1^4 are $(N-1)\times 1$ dimensions.

When $k_i = 1$ $(i = 2, ..., N-2)$ in (2.25), the discretization employs a uniform grid, and the corresponding matrix is:

$$A_1^4 = \begin{pmatrix} 4 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 1 & 4 & 1 & \cdots & 0 & 0 & 0 \\ 0 & 1 & 4 & \cdots & 0 & 0 & 0 \\ & & & \ddots & & & \\ 0 & 0 & 0 & \cdots & 4 & 1 & 0 \\ 0 & 0 & 0 & \cdots & 1 & 4 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 1 & 4 \end{pmatrix}, U' = \begin{pmatrix} u_1' \\ u_2' \\ u_3' \\ \vdots \\ u_{N-3}' \\ u_{N-2}' \\ u_{N-1}' \end{pmatrix}, H_1^4 = \begin{pmatrix} -\frac{43}{60h} u_0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ \frac{43}{60h} u_N \end{pmatrix},$$

$$M_1^4 = \frac{1}{h}\begin{pmatrix} -5 & 8 & -\frac{10}{3} & \frac{5}{4} & -\frac{1}{5} & \cdots & 0 & 0 & 0 & 0 & 0 \\ -3 & 0 & 3 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ 0 & -3 & 0 & 3 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ & & & \ddots & \ddots & \ddots & & & & & \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & -3 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & -3 & 0 & 3 \\ 0 & 0 & 0 & 0 & 0 & \cdots & \frac{1}{5} & -\frac{5}{4} & \frac{10}{3} & -8 & 5 \end{pmatrix}.$$

2.3. The adaptive mesh algorithm description

The adaptive mesh method is an algorithm that automatically adjusts the solution accuracy in regions with significant variations. Generally, the numerical solutions are defined on a predefined computational grid. In many numerical simulations, fine grids are necessary in regions with steep gradients, whereas coarser grids can be used in smooth regions with slowly varying solutions. The adaptive mesh method provides a dynamic environment that can achieve different solution accuracies based on different requirements.

The basis of the proposed adaptive mesh algorithm lies in the derivative change rate of the numerical solution obtained from the fourth-order compact scheme on uniform grids. The basis of our algorithm is described as follows: The derivative change rate directly reflects the smoothness of the solution large absolute values indicate rapid solution changes (e.g., boundary layers), where dense grids are needed to capture details; and small values correspond to smooth regions, allowing coarser grids to optimize efficiency. This basis aligns with the core demand of singularly perturbed convection-diffusion equations, namely, accurately resolving boundary layers while maintaining computational efficiency. It avoids numerical oscillations caused by inappropriate grid density in traditional uniform schemes. First, the fourth-order compact scheme with uniform grids is used throughout the entire region to obtain a preliminary numerical solution. Then, based on the derivative change rate of the numerical solution, the grid step size is adjusted to obtain a nonuniform grid distribution. Finally, updated numerical results are obtained using the high-order scheme proposed in this paper. This process is repeated until the grid distribution reaches the optimal distribution effect.

3. Numerical experiments

In this section, we perform numerical experiments to demonstrate the accuracy, dependability, and effectiveness of the proposed nonuniform fourth-order compact difference scheme coupled with the adaptive mesh method. All computations were conducted on a PC with Intel core i7-1075H CPU and 8GB memory.

$L^\infty - error$ is defined by

$$L^\infty - error = max|u_{ie} - u_{in}|,$$

where u_{ie} is the exact solution of u, and u_{in} is the numerical solution of u.

$L^2 - error$ is defined by

$$L^2 - error = \sqrt{\frac{1}{N+1}\sum_{i=1}^{N+1}(u_{ie} - u_{in})},$$

where u_{ie} is the exact solution of u, and u_{in} is the numerical solution of u.

The rate of convergence is defined by

$$Rate = \frac{Ln(err(N_1)/err(N_2))}{Ln(N_2/N_1)},$$

where $err(N_1)$ and $err(N_2)$ are the maximum absolute errors computed with the nodes N_1 and N_2, respectively.

Based on the above discussion, we present the following iterative algorithm. The specific iterative algorithm steps are as follows:

1. The adaptive mesh algorithm based on the fourth-order nonuniform compact difference scheme.

1: Input the value of $h_{min}^{(0)}$, $h_{max}^{(0)}$, u'_{min}, ϵ.
2: Resolve on uniform grids:
(a) Generate a uniform grid with a step size of $h_{max}^{(0)}$.
(b) Resolve the Eq (2.1) using the fourth-order uniform compact difference schemes (2.4), (2.7), (2.11), (2.15), (2.20), and (2.24) get the numerical solution $u_i^{(0)}$.
3: Resolve the optimal h_i by iterative method:
(a) For $k = 0, 1, 2, ...$ until a certain convergence criterion is met.
(b) Calculate $u'(x_i) = \frac{u_{i+1}^{(k)} - u_i^{(k)}}{h_{max}}$.
(c) Determine the derivative threshold. If $|u'(x_i)| > u'_{min}$, then the grid encryption will be performed within the interval $[x_i, x_{i+1}]$ where this node is located.
(d) Generate nonuniform grids. The encrypted step size h_i is determined using the minimum step size h_{min} and the sensitivity parameter p ($p = 4^{(floor)(\log_2(|u'(x_i)|))}$) as follows:

$$h_i^{(k+1)} = \max(h_{min}^{(k)}, \frac{h_{max}^{(k)}}{p}).$$

(e) Calculate a new $h_{min}^{(k+1)}$, $h_{max}^{(k+1)}$.
4: Resolve on nonuniform grids:
Resolve the Eq (2.1) by the fourth-order compact difference nonuniform schemes (2.2), (2.5), (2.9), (2.13), (2.18), and (2.22) get the numerical solution $u_i^{(k+1)}$.
5: Determine step size:
If $|h_i^{(k+1)} - h_i^{(k)}| < \epsilon$, then
Stop;
else
go to 4.
end if
6: Output the numerical solution $u_i^{(k+1)}$.

3.1. Example 1

We consider the following constant coefficient singularly perturbed problem:

$$\begin{cases} -\varepsilon u'' + u' = \varepsilon\pi^2 \sin(\pi x) + \pi\cos(\pi x), & x \in (0, 1), \\ u(0) = 0, u(1) = 1. \end{cases}$$

The exact solution to this problem is given as

$$u(x) = \sin(\pi x) + \frac{e^{\frac{x}{\varepsilon}} - 1}{e^{\frac{1}{\varepsilon}} - 1}.$$

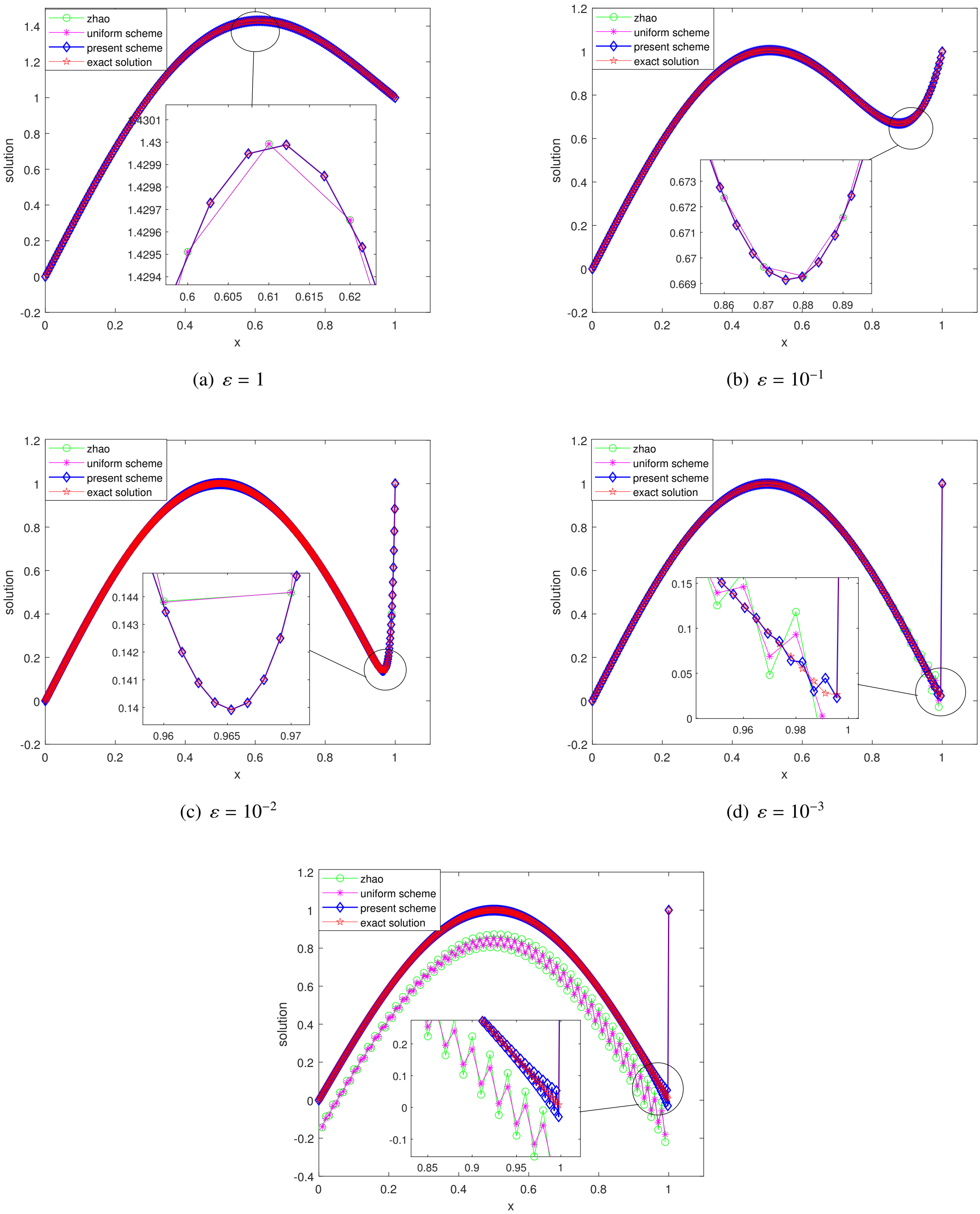
When ε is small, the solution to the problem has a boundary layer at $x = 1$. Table 1 gives the $L^\infty-$error and the convergence rate for Example 1 by using the present scheme, uniform scheme, and Zhao's fourth-order scheme. It is observed that the present scheme, uniform scheme, and Zhao's fourth-order scheme are stable and effective when $\varepsilon = 1, 10^{-1}, 10^{-2}$, but when $\varepsilon = 10^{-3}$, $\varepsilon = 10^{-4}$, the errors of the Zhao [20] and uniform schemes are large and no longer decrease as the number of grids increases.That is, the two schemes don't achieve fourth-order accuracy, while the present scheme can obtain very exact solutions with enough computational nodes distributed in the boundary layer because of coupling with the adaptive mesh and can reach the theoretical fourth order, thereby demonstrating superior computational accuracy and resolution.

Table 1. The L^∞-errors and rate of the convergence under different ε for Example 1.

N	Zhao [20]		uniform scheme		present scheme	
	L^∞-error	*Rate*	L^∞-error	*Rate*	L^∞-error	*Rate*
$\varepsilon = 1$						
40	1.557(-7)	–	1.613(-7)	–	7.864(-9)	–
60	3.129(-8)	3.958	3.166(-8)	4.015	1.479(-9)	4.115
80	9.949(-9)	3.983	1.000(-8)	4.006	4.893(-10)	3.845
100	4.082(-9)	3.992	4.095(-9)	4.001	1.927(-10)	4.176
$\varepsilon = 10^{-1}$						
40	4.091(-6)	–	2.480(-6)	–	6.866(-8)	–
60	5.677(-7)	4.170	4.347(-7)	4.294	1.129(-8)	4.452
80	1.566(-7)	4.016	1.326(-7)	4.127	3.218(-9)	4.299
120	2.781(-8)	4.263	2.556(-8)	4.060	7.023(-10)	3.800
$\varepsilon = 10^{-2}$						
40	1.151(-2)	–	7.291(-3)	–	4.240(-6)	–
100	1.157(-3)	2.507	1.200(-3)	1.969	1.455(-7)	3.680
120	6.416(-4)	3.234	6.080(-4)	3.757	7.209(-8)	3.852
160	2.274(-4)	3.601	1.782(-4)	4.248	2.192(-8)	4.138
$\varepsilon = 10^{-3}$						
200	3.394(-2)	–	1.942(-2)	–	2.209(-3)	–
400	1.151(-2)	1.560	7.291(-3)	1.413	1.843(-4)	3.583
520	6.541(-3)	2.154	6.166(-3)	0.638	6.943(-5)	3.721
$\varepsilon = 10^{-4}$						
250	7.598(-2)	–	3.892(-2)	–	1.522(-2)	–
300	7.520(-2)	0.057	3.852(-2)	0.057	7.808(-3)	3.700

Note: $1.557(-7) = 1.557 \times 10^{-7}$.

From Figure 4, when the perturbation parameters gradually decrease, the present scheme has obvious advantages over the other two schemes in terms of computational accuracy and resolution.



(a) $\varepsilon = 1$

(b) $\varepsilon = 10^{-1}$

(c) $\varepsilon = 10^{-2}$

(d) $\varepsilon = 10^{-3}$

(e) $\varepsilon = 10^{-4}$

Figure 4. The exact and numerical solutions with three schemes with $N = 100$ under different ε.

Table 2 indicates that when the perturbation parameters ε gradually decrease, the error of the present scheme is much smaller than that of the other two schemes, and the error of the present scheme is 1–2 orders of magnitude smaller than those of the other two schemes. This is consistent with the numerical results in Figure 4. It fully demonstrates the superiority of the scheme calculation with the adaptive mesh method. Consequently, the present scheme employs a non-uniform adaptive mesh algorithm to mitigate the oscillation within the boundary layer.

Table 2. The errors with three schemes under different ε for Example 1.

ε	Zhao [20]		uniform scheme		present scheme	
	L^∞-error	L^2-error	L^∞-error	L^2-error	L^∞-error	L^2-error
1	4.082(-9)	2.854(-9)	4.095(-9)	2.864(-9)	1.927(-10)	1.354(-10)
10^{-1}	5.995(-8)	2.642(-8)	5.334(-8)	2.384(-8)	1.553(-9)	6.977(-10)
10^{-2}	1.157(-3)	1.583(-4)	1.200(-3)	1.489(-4)	1.455(-7)	1.976(-8)
10^{-3}	5.948(-2)	1.134(-2)	3.042(-2)	6.061(-3)	4.678(-3)	2.267(-4)
10^{-4}	2.520(-1)	1.658(-1)	2.114(-1)	1.644(-1)	2.116(-2)	7.573(-4)

3.2. Example 2

We consider the following variable coefficient singularly perturbed problem:

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

The exact solution to this problem is given as

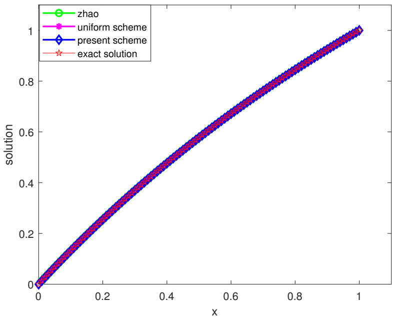
$$u(x) = xe^{-\frac{1}{\sqrt{\varepsilon}}} - e^{-\frac{x}{\sqrt{\varepsilon}}} + 1.$$

When ε is small, the solution to the problem has a boundary layer at $x = 1$. We calculate the solution for parameter $\varepsilon = 1, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}$ using the present scheme, uniform scheme, and Zhao's fourth-order scheme. As shown in Table 3, when $\varepsilon = 1, 10^{-1}, 10^{-2}, 10^{-3}$ the computational accuracy of all schemes can reach the theoretical fourth-order accuracy. However, when $\varepsilon = 10^{-4}, 10^{-5}, 10^{-6}$ the computational errors of the Zhao [20] and uniform schemes are large, while the present scheme can still obtain the fourth-order accuracy solution.

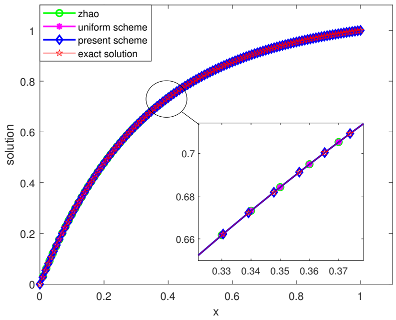
As shown in Figure 5 (a)–(c), the numerical solution is in good agreement with the exact solution by using the three schemes. As shown in Figure 5 (d)–(g), the numerical solutions obtained with the Zhao [20] and uniform schemes all produce nonphysical oscillations near $x = 1$, while the present scheme matches well with the exact solution because there are enough computational nodes distributed in the boundary layer.

Table 3. The L^∞-errors and rate of the convergence under different ε for Example 2.

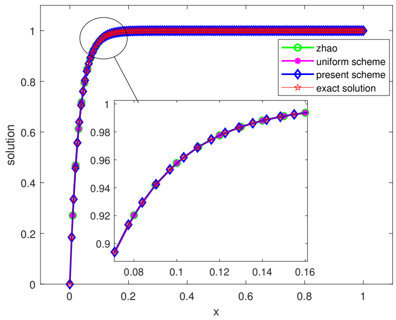
N	Zhao [20]		uniform scheme		present scheme	
	L^∞-error	*Rate*	L^∞-error	*Rate*	L^∞-error	*Rate*
$\varepsilon = 1$						
40	1.020(-10)	–	1.022(-10)	–	1.023(-10)	–
60	2.013(-11)	4.002	2.014(-11)	4.006	2.014(-11)	4.008
80	6.323(-12)	4.025	6.319(-12)	4.029	6.374(-12)	3.996
100	2.612(-12)	3.962	2.646(-12)	3.901	2.548(-12)	4.112
$\varepsilon = 10^{-1}$						
40	7.369(-8)	–	7.868(-8)	–	1.780(-8)	–
60	1.508(-8)	3.912	1.554(-8)	4.003	3.513(-9)	4.002
80	4.832(-9)	3.956	4.014(-9)	4.002	1.112(-9)	3.999
100	1.991(-9)	3.973	2.013(-9)	3.999	4.554(-10)	4.001
$\varepsilon = 10^{-2}$						
40	1.643(-5)	–	1.249(-5)	–	6.461(-8)	–
80	8.323(-7)	4.303	8.804(-7)	3.827	4.549(-9)	3.828
120	1.666(-7)	3.967	1.748(-7)	3.987	8.640(-10)	4.097
160	5.352(-8)	3.947	5.547(-8)	3.989	2.629(-10)	4.136
$\varepsilon = 10^{-3}$						
100	5.439(-5)	–	3.494(-5)	–	1.223(-8)	–
140	1.423(-5)	3.985	1.044(-5)	3.576	2.704(-9)	4.485
220	2.278(-6)	4.053	1.902(-6)	3.778	3.717(-10)	4.390
$\varepsilon = 10^{-4}$						
150	7.789(-4)	–	3.456(-4)	-	5.154(-7)	–
190	3.846(-4)	2.985	1.942(-4)	2.438	1.866(-7)	4.298
300	6.866(-5)	3.778	4.181(-5)	3.211	3.329(-8)	3.774
$\varepsilon = 10^{-5}$						
40	1.372(0)	–	1.337(0)	–	3.804(-5)	–
60	1.261(0)	0.298	1.171(0)	0.327	1.026(-5)	3.232
70	1.099(0)	0.656	1.048(0)	0.719	6.219(-6)	3.174
80	0.971(0)	0.927	0.913(0)	1.033	3.982(-6)	3.339
$\varepsilon = 10^{-6}$						
80	1.406(0)	–	1.363(0)	–	5.817(-6)	–
90	1.356(0)	0.044	1.399(0)	0.042	3.365(-6)	4.647
100	1.348(0)	0.056	1.391(0)	0.054	2.310(-6)	3.570
110	1.383(0)	0.061	1.338(0)	0.078	1.637(-6)	3.613



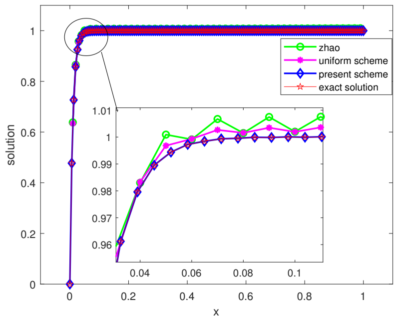
(a) $\varepsilon = 1$



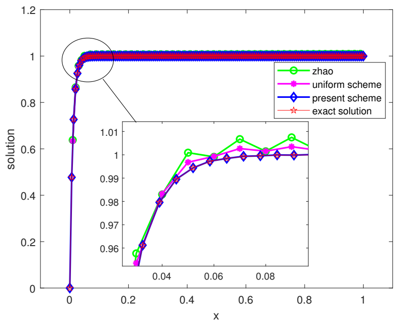
(b) $\varepsilon = 10^{-1}$



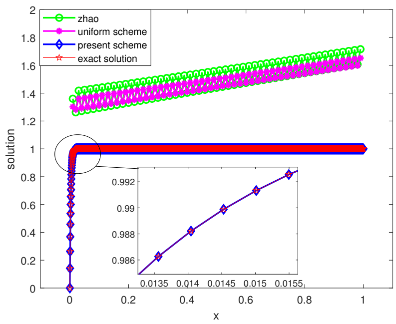
(c) $\varepsilon = 10^{-2}$



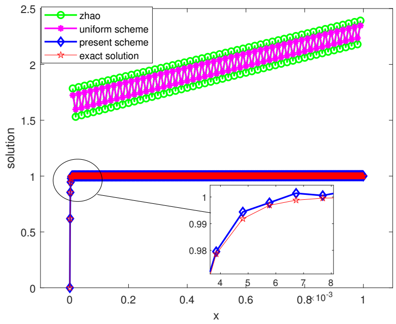
(d) $\varepsilon = 10^{-3}$



(e) $\varepsilon = 10^{-4}$



(f) $\varepsilon = 10^{-5}$



(g) $\varepsilon = 10^{-6}$

Figure 5. The exact and numerical solutions with three schemes with $N = 100$ under different ε.

As further shown in Table 4, which presents the $L^{\infty}-$ and L^2- errors for Example 2, using the present scheme, the uniform scheme, and Zhao's fourth-order scheme [20], Table 4 indicates that when $10^{-1} < \varepsilon < 1$, the error of the present scheme is similar to that of the uniform scheme and Zhao's fourth-order scheme [20]. When $10^{-6} < \varepsilon < 10^{-2}$, the error of the present scheme is much smaller than that of the other two schemes, and the error of the present scheme is 2–3 orders of magnitude smaller than those of the other two schemes. This is consistent with the numerical results in Figure 5.

Table 4. The errors with three schemes under different ε for Example 2.

ε	Zhao [20]		uniform scheme		present scheme	
	L^{∞}-error	L^2-error	L^{∞}-error	L^2-error	L^{∞}-error	L^2-error
1	2.662(-12)	1.904(-12)	2.646(-12)	1.892(-12)	2.548(-12)	1.823(-12)
10^{-1}	1.991(-9)	1.086(-9)	2.013(-9)	1.100(-9)	4.554(-10)	2.498(-10)
10^{-2}	3.339(-7)	9.225(-8)	3.585(-7)	9.836(-8)	2.371(-9)	6.497(-10)
10^{-3}	5.439(-5)	6.818(-6)	3.494(-5)	5.474(-6)	1.223(-8)	1.693(-9)
10^{-4}	9.410(-3)	6.796(-3)	5.266(-3)	3.845(-3)	2.295(-6)	1.725(-7)
10^{-5}	7.160(-1)	5.057(-1)	6.524(-1)	4.751(-1)	2.659(-6)	1.121(-7)
10^{-6}	1.392(0)	9.823(-1)	1.348(0)	0.981(-1)	2.310(-6)	5.453(-8)

3.3. Example 3

We consider the following nonlinear singularly perturbed problem:

$$\begin{cases} -\varepsilon u'' + uu' = f(x), & x \in (0,1), \\ u(0) = 1 + e + e^{-\frac{1}{\varepsilon}}, & u(1) = e^{-1} + 1. \end{cases}$$

The exact solution to this problem is given as

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

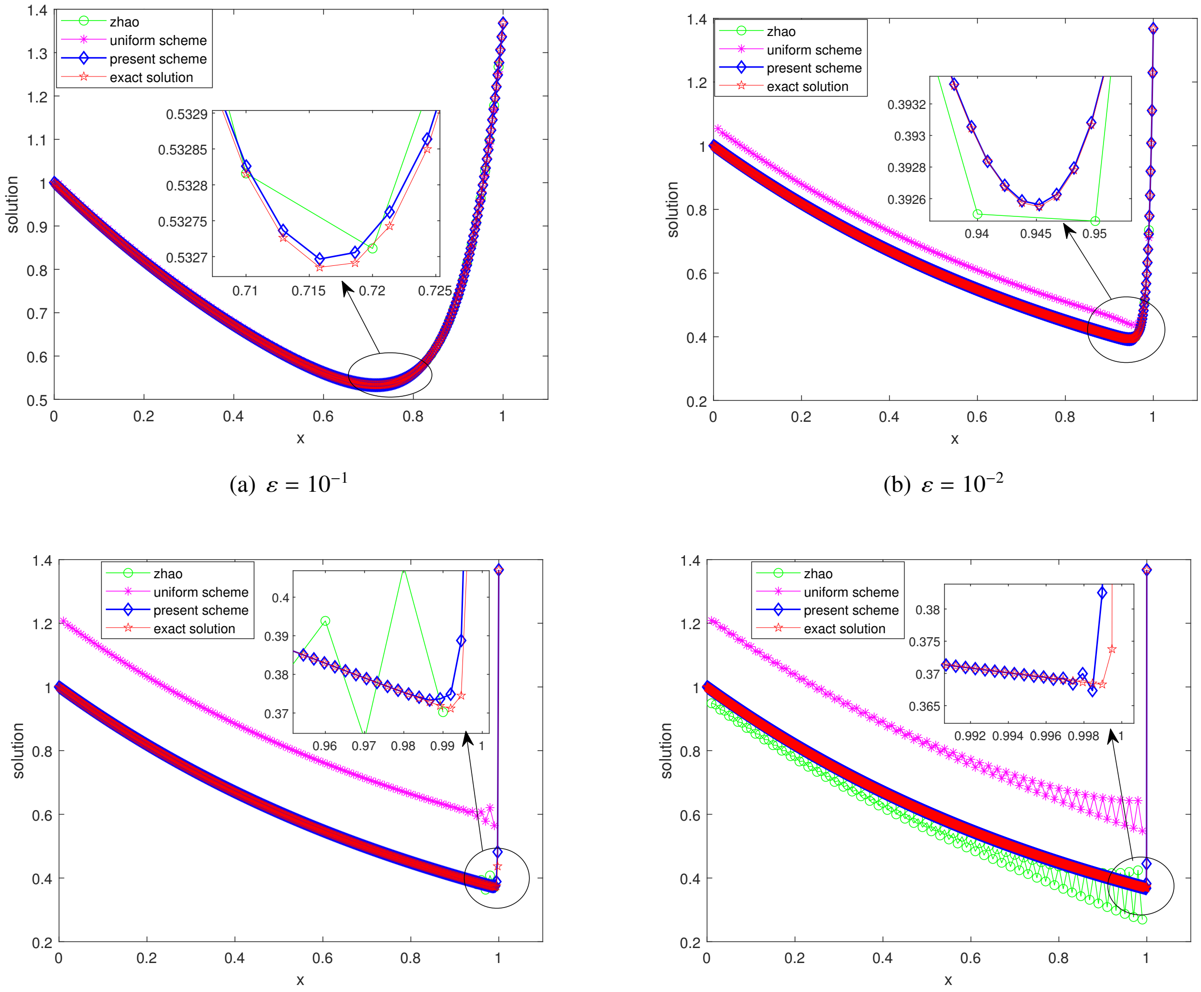
When ε is small, the solution to the equation exhibits a boundary layer at $x = 1$. Table 5 lists the computational error and convergence order when different values of parameters ε. As shown in Table 5, when $\varepsilon = 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}$, the errors of the Zhao [20] and the uniform schemes fail to achieve fourth-order accuracy, while the present scheme can obtain highly accurate solutions and reach the theoretical fourth–order convergence. Furthermore, the present scheme in this paper for solving nonlinear singularly perturbed convection–diffusion equations also has advantages over the uniform scheme in terms of computational accuracy and resolution.

Table 5. The L^∞-errors and rate of the convergence under different ε for Example 3.

N	Zhao [20]		uniform scheme		present scheme	
	L^∞-error	*Rate*	L^∞-error	*Rate*	L^∞-error	*Rate*
$\varepsilon = 10^{-1}$						
5	4.768(-2)	–	–	–	6.817(-3)	–
10	8.824(-4)	5.765	5.784(-2)	–	5.260(-4)	3.696
20	1.918(-4)	2.202	3.315(-2)	0.803	4.343(-5)	3.598
30	5.300(-5)	3.172	2.541(-2)	0.656	1.013(-5)	3.590
$\varepsilon = 10^{-2}$						
190	3.203(-4)	–	3.649(-2)	–	7.025(-5)	–
300	6.072(-5)	3.641	2.681(-2)	0.675	1.383(-5)	3.558
450	7.085(-6)	5.298	2.157(-2)	0.536	3.165(-6)	3.637
750	1.381(-6)	3.199	1.505(-2)	0.705	2.131(-6)	3.661
$\varepsilon = 10^{-3}$						
160	3.956(-2)	–	2.067(-1)	–	2.987(-2)	–
320	4.370(-2)	-0.144	1.412(-1)	0.550	2.318(-3)	3.688
850	1.816(-3)	3.256	7.325(-2)	0.672	5.587(-5)	3.814
1280	1.015(-3)	1.421	5.212(-2)	0.831	9.744(-6)	4.266
$\varepsilon = 10^{-4}$						
8	7.108(+0)	–	8.701(+0)	–	2.052(+0)	–
20	1.554(+0)	1.659	1.025(+0)	2.334	3.510(-2)	4.440

As shown in Figure 6 (a) and (b),the numerical solution is in good agreement with the exact solution by using the three schemes. As shown in Figure 6 (c) and (d), the numerical solutions obtained with the Zhao [20] and the uniform schemes all produce nonphysical oscilations near $x = 1$, while the present scheme matches well with the exact solution because there are enough computational nodes distributed in the boundary layer.

Table 6 shows the errors of the present scheme are 3–4 orders of magnitude smaller than those of the other two schemes. This is consistent with the numerical results in Figure 6. It fully demonstrates the superiority of the scheme calculation under the adaptive mesh method.



(a) $\varepsilon = 10^{-1}$ (b) $\varepsilon = 10^{-2}$

(c) $\varepsilon = 10^{-3}$ (d) $\varepsilon = 10^{-4}$

Figure 6. The exact and numerical solutions with three schemes with $h = 1/100$ under different ε.

Table 6. The errors with different ε for Example 3.

ε	Zhao [20]		uniform scheme		present scheme	
	L^∞-error	L^2-error	L^∞-error	L^2-error	L^∞-error	L^2-error
10^{-1}	3.893(-4)	1.625(+0)	1.176(-2)	5.777(-3)	1.963(-5)	6.131(-6)
10^{-2}	1.469(-3)	2.559(+3)	6.364(-2)	5.843(-2)	9.432(-5)	2.794(-5)
10^{-3}	3.265(-2)	3.845(+4)	2.458(-1)	2.112(-1)	4.495(-2)	2.445(-3)
10^{-4}	1.021(-1)	4.889(+5)	2.682(-1)	2.135(-1)	7.115(-2)	1.648(-3)

4. Concluding remarks

In this paper, we developed a consistent fourth-order nonuniform compact scheme coupled with an adaptive method for solving singularly perturbed convection-diffusion equations with boundary layers.

(1) A fourth-order nonuniform compact difference scheme coupled with an adaptive mesh method for solving singularly perturbed convection-diffusion equations was developed. The method can efficiently alleviate or even eliminate numerical difficulties caused by boundary layers, making the numerical solution closer to the proper solution.

(2) To efficiently solve singularly perturbed convection-diffusion equations and boundary-layer problems, we extended the obtained difference scheme on uniform grids and coupled them with an adaptive mesh method. It is shown that the calculation results of the present scheme in this paper are and thus better suited to solving singularly perturbed convection-diffusion equations well.

(3) Several typical numerical cases were selected for solutions to demonstrate the advantages of the present scheme proposed in this paper in terms of accuracy validity, and stability. Numerical experiments show that the present scheme has better scale resolution and is more fit to solve convection-dominated singularly perturbed problems precisely.

In follow-up research, we aim to focus on the following three aspects: (i) conducting high-accuracy numerical simulations of flow and heat transfer problems based on the present scheme for convection–diffusion equations; (ii) theoretically analyzing the stability and convergence of the proposed difference scheme; and (iii) Extending the present scheme to two-dimensional cases and solving two-dimensional singularly perturbed convection–diffusion equations. We look forward to reporting the results of these studies in future work.

Author contributions

Tao Wang: Conceptualization, methodology, software, formal analysis, writing—original draft, visualization, project administration, funding acquisition; Jinhua Ma: Conceptualization, methodology, validation, formal analysis, writing—review and editing; Qiong Chen: Conceptualization, software, data curation, writing—original draft, visualization. All authors have read and approved the final version of the manuscript for publication.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No.12162001, 12001015) and the Construction Project of First-Class Disciplines in Ningxia Higher Education (Grant No. NXYLXK2017B09).

Conflict of interest

On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

1. K. Kumar, P. C. Pidola, A new stable finite difference scheme and its error analysis for two-dimensional singularly perturbed convection–diffusion equations, *Numer. Methods Partial Differential Equations*, **38** (2020), 1215–1231. https://doi.org/10.1002/num.22732
2. S. Elango, Second order singularly perturbed delay differential equations with non-local boundary condition, *J. Comput. Appl. Math.*, **417** (2023), 114498. https://doi.org/10.1016/j.cam.2022.114498
3. J. Zhang, X. W. Liu, Supercloseness of linear finite element method on Bakhvalov-type meshes for singularly perturbed convection–diffusion equation in 1D, *Appl. Math. Lett.*, **111** (2021), 106624. https://doi.org/10.1016/j.aml.2020.106624
4. X. Zhao, J. Zhang, Uniform convergence of finite element method on Vulanovi-Bakhvalov mesh for singularly perturbed convection–diffusion equation in 2D, *Comput. Math. Appl.*, **188** (2025), 183–194. https://doi.org/10.1016/j.camwa.2025.04.007
5. Y. Wang, Y. H. Li, X. Y. Meng, An upwind finite volume element method on a Shishkin mesh for singularly perturbed convection–diffusion problems, *J. Comput. Appl. Math.*, **438** (2024), 115493. https://doi.org/10.1016/j.cam.2023.115493
6. X. Y. Meng, M. Stynes, Energy-norm and balanced-norm supercloseness error analysis of a finite volume method on Shishkin meshes for singularly perturbed reaction–diffusion problems, *Calcolo*, **60** (2023), 40. https://doi.org/10.1007/s10092-023-00535-3
7. J. W. Zhou, Z. W. Jiang, H. T. Xie, H. F. Niu, The error estimates of spectral methods for 1-dimension singularly perturbed problem, *Appl. Math. Lett.*, **100** (2020), 106001. https://doi.org/10.1016/j.aml.2019.106001
8. R. S. Temsah, Spectral methods for some singularly perturbed third order ordinary differential equations, *Numer. Algor.*, **47** (2020), 63–80. https://doi.org/10.1007/s11075-007-9147-6
9. D. M. Tefera, A. A. Tiruneh, G. A. Derese, Fitted operator method using multiple fitting factors for two parameters singularly perturbed parabolic problems, *Math. Probl. Eng.*, **2022** (2022), 6267522. https://doi.org/10.1155/2022/6267522
10. L. B. Liu, L. Xu, Y. Xu, Z. Huang, An adaptive grid method for a two-parameter singularly perturbed problem with non-smooth data, *Appl. Math. Lett.*, **157** (2024), 109200. https://doi.org/10.1016/j.aml.2024.109200
11. A. G. R. Turnquist, Adaptive mesh methods on compact manifolds via optimal transport and optimal information transport, *J. Comput. Phys.*, **500** (2024), 112726. https://doi.org/10.1016/j.jcp.2023.112726
12. S. Franz, Convergence of local projection stabilisation finite element methods for convection–diffusion problems on layer-adapted meshes, *Bit Numer. Math.*, **57** (2017), 771–786. https://doi.org/10.1007/s10543-017-0652-2
13. F. Erdogan, M. G. Saldn, O. Saldır, A finite difference method on layer-adapted mesh for singularly perturbed delay differential equations, *Appl. Math. Nonlinear Sci.*, **5** (2020), 425–436. http://dx.doi.org/10.2478/AMNS.2020.1.00040

14. N. S. Yadav, M. Kaushik, Stability and error analysis of an efficient numerical method for convection dominated parabolic PDEs with jump discontinuity in source function on modified layer-adapted mesh, *Comput. Math. Math. Phys.*, **64** (2024), 509–536. https://doi.org/10.1134/S0965542524030102

15. J. Mackenzie, Uniform convergence analysis of an upwind finite-difference approximation of a convection-diffusion boundary value problem on an adaptive grid, *IMA J. Numer. Anal.*, **19** (1999), 233–249. https://doi.org/10.1093/imanum/19.2.233

16. H. G. Roos, M. Stynes, L. Tobiska, Robust numerical methods for singularly perturbed differential equations, In: *Springer series in computational mathematics*, Berlin: Springer, **24** (2008). https://doi.org/10.1007/978-3-540-34467-4

17. T. Wang, T. Liu, A consistent fourth-order compact finite difference scheme for solving vorticity-stream function form of incompressible Navier-Stokes equations, *Numer. Math. Theory Methods Appl.*, **12** (2019), 312–330. https://doi.org/10.4208/nmtma.OA-2018-0043

18. S. K. Lete, T. G. Liu, Compact finite difference schemes with spectral-like resolution, *J. Comput. Phys.*, **103** (1992), 16–42. https://doi.org/10.1016/0021-9991(92)90324-R

19. Z. F. Tian, Y. A. Li, Numerical solution of the incompressible Navier-Stokes equations with a three-point fourth-order upwind compact difference schemes, In: *Proceedings of fourth international conference nonlinear mechanic*, 2002, 942–946.

20. J. C. Zhao, T. Zhang, R. M. Corless, Convergence of the compact finite difference method for second-order elliptic equations, *Appl. Math. Comput.*, **182** (2006), 1451–1469. https://doi.org/10.1016/j.amc.2006.05.033

A. Appendix A

The consistent fourth-order left boundary scheme of the second derivative is

$$\alpha_1 u_1'' + \alpha_2 u_2'' + \alpha_3 u_3'' + \alpha_4 u_4'' + \alpha_5 u_5'' = \frac{1}{h_1^2}(a_0 u_0 + a_1 u_1 + a_2 u_2).$$

The coefficients of the free unknown parameters in the uniform fourth-order left boundary scheme of the second derivative are respectively

$$\begin{aligned}
a_0 &= \frac{120k_i}{k_i^3 + 4k_i^2 + 4k_i + 1}, \\
a_1 &= -\frac{120k_i(k_1 + 1)}{k_1(k_i^3 + 4k_i^2 + 4k_i + 1)}, \\
a_2 &= \frac{120k_i}{k_1(k_i^3 + 4k_i^2 + 4k_i + 1)},
\end{aligned}$$

$$
\begin{aligned}
\alpha_1 &= \frac{2k_iA_1}{k_1^4(k_2+1)(k_2+k_2k_3+1)(k_i^3+4k_i^2+4k_i+1)(k_2+k_2k_3+k_2k_3k_4+1)},\\
\alpha_2 &= \frac{2k_iA_2}{k_1^4k_2^3(k_3+1)(k_3+k_3k_4+1)(k_i^3+4k_i^2+4k_i+1)},\\
\alpha_3 &= \frac{2k_iA_3}{k_1^4k_2^3k_3^2(k_2+1)(k_4+1)(k_i^3+4k_i^2+4k_i+1)},
\end{aligned}
$$

where

$$
\begin{aligned}
A_3 =& 5k_i - 5k_i^2 - 5k_i^3 + 5k_i^4 + 3k_i^5 + 9k_1 + 15k_1^2 + 10k_1^3 - 2k_1^5 + 6k_1k_2 + 20k_1^2k_2 + 20k_1^3k_2\\
&- 6k_1^5k_2 + 5k_1^2k_2^2 + 10k_1^3k_2^2 - 5k_1^5k_2^2 + 10k_1^2k_2^2k_3 + 20k_1^3k_2^2k_3 - 10k_1^5k_2^2k_3 + 6k_1k_2k_3\\
&+ 5k_1^2k_2^2k_3^2 + 10k_1^3k_2^2k_3^2 - 5k_1^5k_2^2k_3^2 + 20k_1^2k_2k_3 + 20k_1^3k_2k_3 - 6k_1^5k_2k_3 + 3k_1k_2k_3k_4\\
&+ 5k_1^2k_2^2k_3^2k_4 + 10k_1^3k_2^2k_3^2k_4 - 5k_1^5k_2^2k_3^2k_4 + 10k_1^2k_2k_3k_4 + 10k_1^3k_2k_3k_4 - 3k_1^5k_2k_3k_4\\
&+ 5k_1^2k_2^2k_3k_4 + 10k_1^3k_2^2k_3k_4 - 5k_1^5k_2^2k_3k_4 + 5,\\
A_2 =& A_3 + 3k_1k_2 + 10k_1^2k_2 + 10k_1^3k_2 - 3k_1^5k_2 + 10k_1^2k_2^2 + 20k_1^3k_2^2 - 10k_1^3k_2^3 - 10k_1^5k_2^2 - 10k_1^5k_2^3\\
&+ 10k_1^2k_2^2k_3 + 20k_1^3k_2^2k_3 + 20k_1^3k_2^3k_3 - 10k_1^5k_2^2k_3 - 20k_1^5k_2^3k_3 - 10k_1^5k_2^3k_3^2 + 10k_1^3k_2^3k_3^2k_4\\
&- 10k_1^5k_2^3k_3^2k_4 + 5k_1^2k_2^2k_3k_4 + 10k_1^3k_2^2k_3k_4 + 10k_1^3k_2^3k_3k_4 - 5k_1^5k_2^2k_3k_4 - 10k_1^5k_2^3k_3k_4,\\
A_1 =& A_2 + 3k_1 + 15k_1^2 + 30k_1^3 + 30k_1^4 + 12k_1^5 + 15k_1^2k_2 + 60k_1^3k_2 + 90k_1^4k_2 + 45k_1^5k_2 + 30k_1^3k_2^2\\
&+ 90k_1^4k_2^2 + 30k_1^4k_2^3 + 60k_1^5k_2^2 + 30k_1^5k_2^3 + 40k_1^3k_2^2k_3 + 120k_1^4k_2^2k_3 + 60k_1^4k_2^3k_3 + 80k_1^5k_2^2k_3\\
&+ 60k_1^5k_2^3k_3 + 10k_1^3k_2^2k_3^2 + 30k_1^4k_2^2k_3^2 + 30k_1^4k_2^3k_3^2 + 20k_1^5k_2^2k_3^2 + 30k_1^5k_2^3k_3^2 + 10k_1^2k_2k_3\\
&+ 40k_1^3k_2k_3 + 60k_1^4k_2k_3 + 30k_1^5k_2k_3 + 10k_1^3k_2^2k_3^2k_4 + 30k_1^4k_2^2k_3^2k_4 + 30k_1^4k_2^3k_3^2k_4 + 20k_1^5k_2^2k_3^2k_4\\
&+ 30k_1^5k_2^3k_3^2k_4 + 5k_1^2k_2k_3k_4 + 20k_1^3k_2k_3k_4 + 30k_1^4k_2k_3k_4 + 15k_1^5k_2k_3k_4 + 20k_1^3k_2^2k_3k_4\\
&+ 60k_1^4k_2^2k_3k_4 + 30k_1^4k_2^3k_3k_4 + 40k_1^5k_2^2k_3k_4 + 30k_1^5k_2^3k_3k_4.
\end{aligned}
$$

B. Appendix B

The consistent fourth-order right boundary scheme of the second derivative is

$$
\alpha_{N-1}u''_{N-1} + \alpha_{N-2}u''_{N-2} + \alpha_{N-3}u''_{N-3} + \alpha_{N-4}u''_{N-4} + \alpha_{N-5}u''_{N-5} = \frac{1}{h_N^2}(a_Nu_N + a_{N-1}u_{N-1} + a_{N-2}u_{N-2}).
$$

The coefficients of the free unknown parameters in the uniform fourth-order right boundary scheme of the second derivative are respectively

$$
\begin{aligned}
a_N &= \frac{120k_i}{k_i^3+4k_i^2+4k_i+1},\\
a_{N-1} &= -\frac{120k_i(k_N+1)}{k_N(k_i^3+4k_i^2+4k_i+1)},\\
a_{N-2} &= \frac{120k_i}{k_N(k_i^3+4k_i^2+4k_i+1)},\\
\alpha_{N-1} &= \frac{2k_iA_{N1}}{k_N^4(k_{N-1}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)}\cdot\frac{1}{(k_i^3+4k_i^2+4k_i+1)(k_{N-1}+k_{N-1}k_{N-2}+k_{N-1}k_{N-2}k_{N-3}+1)},\\
\alpha_{N-2} &= \frac{2k_iA_{N2}}{k_N^4k_{N-1}^3(k_{N-2}+1)(k_{N-2}+k_{N-2}k_{N-3}+1)(k_i^3+4k_i^2+4k_i+1)},\\
\alpha_{N-3} &= \frac{2k_iA_{N3}}{k_N^4k_{N-1}^3k_{N-2}^2(k_{N-1}+1)(k_{N-3}+1)(k_i^3+4k_i^2+4k_i+1)},
\end{aligned}
$$

where

$$
\begin{aligned}
A_{N3} =& 5k_i-5k_i^2-5k_i^3+5k_i^4+3k_i^5+9k_N+15k_N^2+10k_N^3-2k_N^5+6k_Nk_{N-1}+20k_N^2k_{N-1}\\
&+20k_N^3k_{N-1}-6k_N^5k_{N-1}+5k_N^2k_{N-1}^2+10k_N^3k_{N-1}^2-5k_N^5k_{N-1}^2+10k_N^2k_{N-1}^2k_{N-2}\\
&+20k_N^3k_{N-1}^2k_{N-2}-10k_N^5k_{N-1}^2k_{N-2}+6k_Nk_{N-1}k_{N-2}+5k_N^2k_{N-1}^2k_{N-2}^2\\
&+10k_N^3k_{N-1}^2k_{N-2}^2-5k_N^5k_{N-1}^2k_{N-2}^2+20k_N^2k_{N-1}k_{N-2}+20k_N^3k_{N-1}k_{N-2}\\
&-6k_N^5k_{N-1}k_{N-2}+3k_Nk_{N-1}k_{N-2}k_{N-3}+5k_N^2k_{N-1}^2k_{N-2}^2k_{N-3}\\
&+10k_N^3k_{N-1}^2k_{N-2}^2k_{N-3}-5k_N^5k_{N-1}^2k_{N-2}^2k_{N-3}+10k_N^2k_{N-1}k_{N-2}k_{N-3}\\
&+10k_N^3k_{N-1}k_{N-2}k_{N-3}-3k_N^5k_{N-1}k_{N-2}k_{N-3}+5k_N^2k_{N-1}^2k_{N-2}k_{N-3}\\
&+10k_N^3k_{N-1}^2k_{N-2}k_{N-3}-5k_N^5k_{N-1}^2k_{N-2}k_{N-3}+5,\\
A_{N2} =& A_{N3}+3k_Nk_{N-1}+10k_N^2k_{N-1}+10k_N^3k_{N-1}-3k_N^5k_{N-1}+10k_N^2k_{N-1}^2+20k_N^3k_{N-1}^2\\
&-10k_N^3k_{N-1}^3-10k_N^5k_{N-1}^2-10k_N^5k_{N-1}^3+10k_N^2k_{N-1}^2k_{N-2}+20k_N^3k_{N-1}^2k_{N-2}\\
&+20k_N^3k_{N-1}^3k_{N-2}-10k_N^5k_{N-1}^2k_{N-2}-20k_N^5k_{N-1}^3k_{N-2}-10k_N^5k_{N-1}^3k_{N-2}^2\\
&+10k_N^3k_{N-1}^3k_{N-2}^2k_{N-3}-10k_N^5k_{N-1}^3k_{N-2}^2k_{N-3}+5k_N^2k_{N-1}^2k_{N-2}k_{N-3}\\
&+10k_N^3k_{N-1}^2k_{N-2}k_{N-3}+10k_N^3k_{N-1}^3k_{N-2}k_{N-3}-5k_N^5k_{N-1}^2k_{N-2}k_{N-3}\\
&-10k_N^5k_{N-1}^3k_{N-2}k_{N-3},\\
A_{N1} =& A_{N2}+3k_N+15k_N^2+30k_N^3+30k_N^4+12k_N^5+15k_N^2k_{N-1}+60k_N^3k_{N-1}+90k_N^4k_{N-1}\\
&+45k_N^5k_{N-1}+30k_N^3k_{N-1}^2+90k_N^4k_{N-1}^2+30k_N^4k_{N-1}^3+60k_N^5k_{N-1}^2+30k_N^5k_{N-1}^3\\
&+40k_N^3k_{N-1}^2k_{N-2}+120k_N^4k_{N-1}^2k_{N-2}+60k_N^4k_{N-1}^3k_{N-2}+80k_N^5k_{N-1}^2k_{N-2}\\
&+60k_N^5k_{N-1}^3k_{N-2}+10k_N^3k_{N-1}^2k_{N-2}^2+30k_N^4k_{N-1}^2k_{N-2}^2+30k_N^4k_{N-1}^3k_{N-2}^2\\
&+20k_N^5k_{N-1}^2k_{N-2}^2+30k_N^5k_{N-1}^3k_{N-2}^2+10k_N^2k_{N-1}k_{N-2}+40k_N^3k_{N-1}k_{N-2}\\
&+60k_N^4k_{N-1}k_{N-2}+30k_N^5k_{N-1}k_{N-2}+10k_N^3k_{N-1}^2k_{N-2}^2k_{N-3}+30k_N^4k_{N-1}^2k_{N-2}^2k_{N-3}\\
&+30k_N^4k_{N-1}^3k_{N-2}^2k_{N-3}+20k_N^5k_{N-1}^2k_{N-2}^2k_{N-3}+30k_N^5k_{N-1}^3k_{N-2}^2k_{N-3}\\
&+5k_N^2k_{N-1}k_{N-2}k_{N-3}+20k_N^3k_{N-1}k_{N-2}k_{N-3}+30k_N^4k_{N-1}k_{N-2}k_{N-3}
\end{aligned}
$$

$$+ 15k_N^5k_{N-1}k_{N-2}k_{N-3} + 20k_N^3k_{N-1}^2k_{N-2}k_{N-3} + 60k_N^4k_{N-1}^2k_{N-2}k_{N-3}$$
$$+ 30k_N^4k_{N-1}^3k_{N-2}k_{N-3} + 40k_N^5k_{N-1}^2k_{N-2}k_{N-3} + 30k_N^5k_{N-1}^3k_{N-2}k_{N-3}.$$

C. Appendix C

The consistent fourth-order left boundary scheme of the first derivative is

$$\beta_1 u_1^{'} + \beta_2 u_2^{'} = \frac{1}{h_1}(b_0u_0 + b_1u_1 + b_2u_2 + b_3u_3 + b_4u_4 + b_5u_5).$$

The coefficients of the free unknown parameters in the fourth-order left boundary scheme of the first-order derivative are respectively

$$\begin{aligned}
\beta_1 =& 4,\\
\beta_2 =& \frac{4}{(k_i+1)^2},\\
b_0 =& \frac{4k_i^2 + 4k_1^3(k_2+1)(k_2+k_2k_3+1) - \frac{4k_1^3k_2^2(k_1+1)(k_3+1)}{(k_i+1)^2}}{(k_1+1)(k_1+k_1k_2+1)(k_1+k_1k_2+k_1k_2k_3+1)(k_1+k_1k_2+k_1k_2k_3+k_2k_3k_4k_1+1)}\\
&- \frac{4k_1^3(k_2+1)(k_2+k_2k_3+1)}{(k_1+1)(k_1+k_1k_2+1)(k_1+k_1k_2+k_1k_2k_3+1)}\\
&+ \frac{4k_1^3k_2^2(k_3+1)}{(k_1+1)(k_i+1)^2(k_1+k_1k_2+1)(k_1+k_1k_2+k_1k_2k_3+1)},\\
b_1 =& \frac{4(k_1 - 4k_2 - 2k_2k_3 + 2k_1k_2 - k_2^2k_3 + k_1k_2^2 - k_2^2 + k_1k_2k_3 + k_1k_2^2k_3 - 3)}{k_1(k_2+1)(k_2+k_2k_3+1)}\\
&- \frac{4k_i^2 + 4k_1^3(k_2+1)(k_2+k_2k_3+1) - \frac{4k_1^3k_2^2(k_1+1)(k_3+1)}{(k_i+1)^2}}{k_1^4(k_2+1)(k_2+k_2k_3+1)(k_2+k_2k_3+k_2k_3k_4+1)}\\
&- \frac{4k_2^2(k_1+1)(k_3+1)}{k_1(k_2+1)(k_i+1)^2(k_2+k_2k_3+1)},\\
b_2 =& \frac{4(k_2+1)(k_2+k_2k_3+1)}{k_1k_2^2(k_1+1)(k_3+1)} + \frac{4k_i^2 + 4k_1^3(k_2+1)(k_2+k_2k_3+1) - \frac{4k_1^3k_2^2(k_1+1)(k_3+1)}{(k_i+1)^2}}{k_1^4k_2^3(k_1+1)(k_3+1)(k_3+k_3k_4+1)}\\
&- \frac{4(2k_1 - k_2 + k_3 - 2k_1k_2 - k_2k_3 + k_1k_3 - 2k_1k_2k_3 + 2)}{k_1k_2(k_1+1)(k_3+1)(k_i+1)^2},\\
b_3 =& \frac{4(k_1+1)(k_3+1)}{k_1k_2k_3(k_2+1)(k_1+k_1k_2+1)(k_i+1)^2} - \frac{4(k_2+k_2k_3+1)}{k_1k_2^2k_3(k_2+1)(k_1+k_1k_2+1)}\\
&- \frac{4k_i^2 + 4k_1^3(k_2+1)(k_2+k_2k_3+1) - \frac{4k_1^3k_2^2(k_1+1)(k_3+1)}{(k_i+1)^2}}{k_1^4k_2^3k_3^2(k_2+1)(k_4+1)(k_1+k_2k_1+1)},\\
b_4 =& \frac{4(k_2+1)}{k_1k_2^2k_3(k_3+1)(k_2+k_2k_3+1)(k_1+k_1k_2+k_1k_2k_3+1)}\\
&+ \frac{4k_i^2 + 4k_1^3(k_2+1)(k_2+k_2k_3+1) - \frac{4k_1^3k_2^2(k_1+1)(k_3+1)}{(k_i+1)^2}}{k_1^4k_2^3k_3^2k_4(k_3+1)(k_2+k_2k_3+1)(k_1+k_1k_2+k_1k_2k_3+1)}
\end{aligned}$$

$$-\frac{4(k_1+1)}{k_1k_2k_3(k_3+1)(k_i+1)^2(k_2+k_2k_3+1)(k_1+k_1k_2+k_1k_2k_3+1)}.$$

D. Appendix D

The consistent fourth-order right boundary scheme of the first derivative is

$$\beta_{N-1}u'_{N-1}+\beta_{N-2}u'_{N-2}=\frac{1}{h}(b_Nu_N+b_{N-1}u_{N-1}+b_{N-2}u_{N-2}+b_{N-3}u_{N-3}+b_{N-4}u_{N-4}+b_{N-5}u_{N-5}).$$

The coefficients of the free unknown parameters in the uniform fourth-order right boundary scheme of the first derivative are respectively

$$\begin{aligned}
\beta_{N-1}&=4,\\
\beta_{N-2}&=\frac{4}{(k_i+1)^2},\\
b_N&=\frac{4k_N^3(k_{N-1}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)}{(k_N+1)(k_N+k_Nk_{N-1}+1)(k_N+k_Nk_{N-1}+k_Nk_{N-1}k_{N-2}+1)}\\
&\quad-\frac{4k_i^2+4k_N^3(k_{N-1}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)-\frac{4k_N^3k_{N-1}^2(k_{N-2}+1)(k_N+1)}{(k_i+1)^2}}{b_{N1}}\\
&\quad-\frac{4k_N^3k_{N-1}^2(k_{N-2}+1)}{(k_N+1)(k_i+1)^2(k_N+k_{N-1}k_N+1)(k_N+k_{N-1}k_N+k_{N-1}k_{N-2}k_N+1)},\\
b_{N-1}&=\frac{4b_{N-11}}{k_N(k_{N-1}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)}\\
&\quad+\frac{4k_i^2+2k_N^3(k_{N-1}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)-\frac{4k_N^3k_{N-1}^2(k_N+1)(k_{N-2}+1)}{(k_i+1)^2}}{k_N^4(k_{N-1}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)(k_{N-1}+k_{N-1}k_{N-2}+k_{N-1}k_{N-2}k_{N-3}+1)}\\
&\quad+\frac{4k_{N-1}^2(k_{N-2}+1)(k_N+1)}{k_N(k_{N-1}+1)(k_i+1)^2(k_{N-1}+k_{N-1}k_{N-2}+1)},\\
b_{N-2}&=-\frac{4(k_{N-1}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)}{k_Nk_{N-1}^2(k_{N-2}+1)(k_N+1)}\\
&\quad-\frac{4k_i^2+4k_N^3(k_{N-1}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)-\frac{4k_N^3k_{N-1}^2(k_{N-2}+1)(k_N+1)}{(k_i+1)^2}}{k_N^4k_{N-1}^3(k_{N-2}+1)(k_N+1)(k_{N-2}+k_{N-2}k_{N-3}+1)}\\
&\quad-\frac{4(k_{N-1}-k_{N-2}-2k_N+2k_{N-1}k_N+k_{N-1}k_{N-2}-k_{N-2}k_N+2k_{N-1}k_{N-2}k_N-2}{k_{N-1}k_N(k_{N-2}+1)(k_N+1)(k_i+1)^2},\\
b_{N-3}&=\frac{4(k_{N-1}+k_{N-1}k_{N-2}+1)}{k_{N-1}^2k_{N-2}k_N(k_{N-1}+1)(k_N+k_{N-1}k_N+1)}\\
&\quad+\frac{4k_i^2+4k_N^3(k_{N-1}+1)(k_{N-1}+k_{N-1}k_{N-2}+1)-\frac{4k_N^3k_{N-1}^2(k_{N-2}+1)(k_N+1)}{(k_i+1)^2}}{k_N^4k_{N-1}^3k_{N-2}^2(k_{N-1}+1)(k_{N-3}+1)(k_N+k_{N-1}k_N+1)}\\
&\quad-\frac{4(k_{N-2}+1)(k_N+1)}{k_{N-1}k_{N-2}k_N(k_{N-1}+1)(k_N+k_Nk_{N-1}+1)(k_i+1)^2},\\
b_{N-4}&=\frac{4(k_N+1)}{k_{N-1}k_{N-2}k_N(k_{N-2}+1)(k_i+1)^2(k_{N-1}+k_{N-1}k_{N-2}+1)}
\end{aligned}$$

$$
\begin{aligned}
&\cdot \frac{1}{(k_N + k_{N-1}k_N + k_{N-1}k_{N-2}k_N + 1)} - \frac{4k_i^2 + 4k_N^3(k_{N-1}+1)}{k_N^4 k_{N-1}^3 k_{N-2}^2 k_{N-3}(k_{N-2}+1)} \\
&\cdot \frac{(k_{N-1} + k_{N-1}k_{N-2} + 1) - \frac{4k_N^3 k_{N-1}^2 (k_{N-2}+1)(k_N+1)}{(k_i+1)^2}}{(k_{N-1} + k_{N-1}k_{N-2} + 1)(k_N + k_{N-1}k_N + k_{N-1}k_{N-2}k_N + 1)} \\
&- \frac{4(k_{N-1}+1)}{k_{N-1}^2 k_{N-2} k_N (k_{N-2}+1)(k_{N-1} + k_{N-1}k_{N-2} + 1)(k_N + k_{N-1}k_N + k_{N-1}k_{N-2}k_N + 1)},
\end{aligned}
$$

where

$$
\begin{aligned}
b_{N1} = {} & (k_N + 1)(k_N + k_N k_{N-1} + 1)(k_N + k_N k_{N-1} + k_{N-1}k_{N-2}k_N + 1)(k_N + k_{N-1}k_N \\
& + k_{N-1}k_{N-2}k_N + k_{N-1}k_{N-2}k_{N-3}k_N + 1), \\
b_{N-11} = {} & 4k_{N-1} - k_N + 2k_{N-1}k_{N-2} - 2k_{N-1}k_N + k_{N-2}k_{N-1}^2 - k_{N-1}^2 k_N + k_{N-1}^2 \\
& - k_{N-1}k_{N-2}k_N - k_N k_{N-1}^2 + 3.
\end{aligned}
$$

AIMS Press