RESEARCH ARTICLE

WILEY

An efficient hybrid numerical method based on an additive scheme for solving coupled systems of singularly perturbed linear parabolic problems

Sunil Kumar[1] | Kuldeep[1] | Higinio Ramos[2,3] | Joginder Singh[1]

[1]Department of Mathematical Sciences, Indian Institute of Technology (BHU) Varanasi, Varanasi, Uttar Pradesh, India

[2]Scientific Computing Group, Universidad de Salamanca, Salamanca, Spain

[3]Escuela Politécnica Superior de Zamora, Campus Viriato, Zamora, Spain

Correspondence
Higinio Ramos, Scientific Computing Group, Universidad de Salamanca, Plaza de la Merced, 37008 Salamanca, Spain.
Email: higra@usal.es

Communicated by: T. Monovasilis

We construct an efficient hybrid numerical method for solving coupled systems of singularly perturbed linear parabolic problems of reaction-diffusion type. The discretization of the coupled system is based on the use of an additive or splitting scheme on a uniform mesh in time and a hybrid scheme on a layer-adapted mesh in space. It is proven that the developed numerical method is uniformly convergent of first order in time and third order in space. The purpose of the additive scheme is to decouple the components of the vector approximate solution at each time step and thus make the computation more efficient. The numerical results confirm the theoretical convergence result and illustrate the efficiency of the proposed strategy.

KEYWORDS

additive scheme, generalized Shishkin mesh, hybrid scheme, singularly perturbed parabolic problem, uniform convergence

MSC CLASSIFICATION

65M06, 65M12, 65M15, 65M50

1 | INTRODUCTION

Singularly perturbed differential equations are typically characterized by a small perturbation parameter multiplied with the highest order derivative term. They have attracted a lot of researchers in the last few decades due to their widespread applications in many areas of sciences and applied mathematics. Singularly perturbed model problems appear for example in computational fluid dynamics, hydrodynamics, chemical reactor theory, financial modelling, and mathematical biology.[1] Unlike regularly perturbed problems, their solutions (or their derivatives) approach a discontinuous limit as perturbation parameter approaches zero. In general, the solutions of singularly perturbed problems exhibit multiscale phenomena; that is, solutions vary rapidly within some parts of the domain and behave smoothly away from them. Due to this reason conventional numerical methods on uniform meshes do not produce satisfactory numerical approximations for small values of the perturbation parameter. This has lead to the concept of a robust/uniformly convergent numerical method, which is a numerical method suitable for these problems and in which the error bound is independent of the size of the perturbation parameter (see previous works[1–13] and the references therein for more details).

In this work, we construct an efficient hybrid numerical method for solving a coupled system of singularly perturbed linear parabolic problems with appropriate initial and boundary conditions. Such systems appear for example in modelling

of saturated flow in fractured porous media[14] and diffusion process in bones.[15] We consider the following problem:

$$\begin{cases} \mathcal{L}_{\vec{\varepsilon}}\vec{u}(x,t) := (\partial_t + \mathcal{L}_{x,\vec{\varepsilon}})\vec{u}(x,t) = \vec{f}(x,t), \ (x,t) \in S = \Omega \times (0,T], \\ \vec{u}(x,0) = \vec{0}, \ \forall x \in \Omega, \ \vec{u}(0,t) = \vec{u}(1,t) = \vec{0}, \ \forall t \in [0,T], \end{cases} \quad (1)$$

where $\Omega = (0,1)$ and $\mathcal{L}_{x,\vec{\varepsilon}}$ is defined by

$$\mathcal{L}_{x,\vec{\varepsilon}} := \mathcal{D}\partial_x^2 + \mathcal{B}, \ \mathcal{B} = \begin{pmatrix} b_{11}(x,t) & b_{12}(x,t) \\ b_{21}(x,t) & b_{22}(x,t) \end{pmatrix}, \ \mathcal{D} = \text{diag}(-\varepsilon_1, -\varepsilon_2). \quad (2)$$

Further, $\vec{f}(x,t) = (f_1(x,t), f_2(x,t))^T$, $\vec{u}(x,t) = (u_1(x,t), u_2(x,t))^T$, and $\vec{\varepsilon} = (\varepsilon_1, \varepsilon_2)^T$ is the vectorial perturbation parameter with $0 < \varepsilon_1 \leq \varepsilon_2 \leq 1$. We denote $\gamma_0 = \{(x,0) | x \in \Omega\}$, $\gamma_1 = \{(x,t) | t \in [0,T], x = 0, 1\}$, and $\gamma = \gamma_0 \cup \gamma_1$. We also assume that

$$b_{ij}(x,t) \leq 0, \ \text{if} \ i \neq j, \quad (3)$$

$$b_{ii}(x,t) > 0, \ b_{i1}(x,t) + b_{i2}(x,t) \geq 0, \ i = 1,2, \ \text{for all} \ (x,t) \in \bar{S} = [0,1] \times [0,T]. \quad (4)$$

Observe that the assumption (4) is not a restriction, as one can achieve this by considering the transformation $\vec{y}(x,t) = \vec{u}(x,t)e^{\alpha t}$, with α sufficiently large. Under these assumptions, $\mathcal{L}_{\vec{\varepsilon}}$ satisfies a maximum principle, which has been used to establish that the solution $\vec{u}(x,t)$ of (1) exhibits overlapping layers near the subset γ_1 of the boundary of the domain (see Franklin et al[16] and Shishkin[17]).

Numerical methods for problem (1) have been developed in previous studies[8,12,16,17] using finite differences on rectangular meshes adapted to the boundary layers. For some recent results on numerical methods for nonlinear reaction-diffusion problems, we refer the reader to previous works.[18-21] In Munyakazi and Patidar,[22] a fitted operator approach is used, while in previous studies,[23-25] it is considered a domain decomposition approach. All these methods considered the standard backward Euler scheme in time. To solve problem (1) the idea of using a standard discretization technique in time, such as the backward Euler scheme, is not an efficient approach because it requires solving a linear system at each time level, which is too costly because of the coupling of the components of the discrete vector solution. This has lead to additive (or splitting) schemes, but so far they are examined with a lower order central difference scheme in space.[25-29] As a general rule, higher order convergent methods are favoured, as they provide better numerical approximations with a low computational cost. So, in this paper, our objective is to propose an efficient high order convergent method for problem (1) by combining an additive scheme on a uniform mesh in time and a high order hybrid scheme on generalized Shishkin mesh mesh in space. The additive scheme decouples the components of the vector approximate solution at each time step, and hence resulting in a reduction of the computational time. We provide an error analysis for the proposed method and prove that it is uniformly convergent, being first order accurate in time and third order accurate in space. It is also shown that the proposed method is computationally more efficient compared to the standard discretization approach. Also, it is noteworthy to mention that this is the first time that an additive scheme with a higher order spatial approach has been used for solving singularly perturbed parabolic coupled reaction-diffusion systems.

We have chosen the generalized Shishkin mesh here because of its superiority over Shishkin mesh in constructing high order schemes. To the best of our knowledge, we are not aware of any high order numerical method based on Shishkin mesh for problem (1). For a stationary version of (1), a high order numerical scheme on a Shishkin mesh is developed in.[30] Note that the analysis of that scheme is not easy, it requires special barrier functions, and the accuracy of the method is proven to be $O(N^{-3}\ln^4 N)$. Nevertheless, for a stationary version of (1), the present scheme on generalized Shishkin mesh can prove to be $O(N^{-4}\ln N + N^{-4}\ln^4 N + N^{-5}\ln^3 N)$, which is better than the earlier mentioned convergence rate.

The paper is structured as follows. In Section 2, we describe bounds on the derivatives of $\vec{u}$. In Section 3, we introduce an additive-based hybrid numerical method for solving the problem (1). In Section 4, the error analysis is given and robust convergence of the method is demonstrated. In Section 5, we present some numerical results by taking two test examples, in support of the theory and to illustrate the efficiency of the present method.

Notation: Throughout the article, C is used as a generic positive constant independent of N, M, and $\varepsilon_k, k = 1, 2$. We have used $\vec{C} = (C, C)^T$. Further, $\|.\|_Q$, where Q is a closed and bounded set, is used to denote the maximum norm. The corresponding discrete maximum norm is denoted by $\|.\|_{Q^{N,M}}$, where $Q^{N,M}$ is a selected discretized domain.

2 | DERIVATIVE BOUNDS

We describe the asymptotic behaviour of the solution $\vec{u}$ of problem (1). We consider the decomposition $\vec{u} = \vec{v} + \vec{w}$, where $\vec{v} = (v_1, v_2)^T$ is the regular part satisfying $\mathcal{L}_{\vec{\varepsilon}}\vec{v} = \vec{f}$ on S, $\vec{v}(x,0) = \vec{0}$ on γ_0, and $\vec{v}(0,t)$ and $\vec{v}(1,t)$ on γ_1 are chosen appropriately.

The singular part $\vec{w} = (w_1, w_2)^T$ satisfies $\mathcal{L}_{\vec{\varepsilon}} \vec{w} = \vec{0}$ on S, and $\vec{w} = \vec{u} - \vec{v}$ on γ. We now recall some bounds on the derivatives of the different components. One can see a detailed proof of these bounds in Clavero and Gracia.[12] For the regular part $\vec{v} = (v_1, v_2)^T$, it holds

$$\|\partial_t^{l_0} \vec{v}\|_{\bar{S}} \leq C, \, 0 \leq l_0 \leq 2, \qquad \|\partial_x^l v_1\|_{\bar{S}} \leq C, \, 0 \leq l \leq 4,$$

$$\|\partial_x^5 v_1\|_{\bar{S}} \leq C(1 + \varepsilon_1^{-1/2} \varepsilon_2), \, \|\partial_x^6 v_1\|_{\bar{S}} \leq C(1 + \varepsilon_1^{-1} \varepsilon_2), \tag{5}$$

$$\|\partial_x^l v_2\|_{\bar{S}} \leq C, \, 0 \leq l \leq 6.$$

For the singular part $\vec{w} = (w_1, w_2)^T$, it holds

$$|\partial_t^{l_0} w_1(x,t)| \leq C\mathcal{A}_{\varepsilon_2}(x), |\partial_t^{l_0} w_2(x,t)| \leq C\mathcal{A}_{\varepsilon_2}(x), \, l_0 = 0, 1, 2, 3,$$

$$|\partial_x^l w_1(x,t)| \leq C(\varepsilon_1^{-l/2} \mathcal{A}_{\varepsilon_1}(x) + \varepsilon_2^{-l/2} \mathcal{A}_{\varepsilon_2}(x)), \, l = 1, \ldots, 6,$$

$$|\partial_x^l w_2(x,t)| \leq C\varepsilon_2^{-l/2} \mathcal{A}_{\varepsilon_2}(x), \, l = 1, 2, \tag{6}$$

$$|\partial_x^l w_2(x,t)| \leq C\varepsilon_2^{-1}(\varepsilon_1^{(2-l)/2} \mathcal{A}_{\varepsilon_1}(x) + \varepsilon_2^{(2-l)/2} \mathcal{A}_{\varepsilon_2}(x)), \, l = 3, 4, 5, 6,$$

for all $(x,t) \in \bar{S}$, where $\mathcal{A}_{\varepsilon_k}(x) = e^{-x/\sqrt{\varepsilon_k}} + e^{-(1-x)/\sqrt{\varepsilon_k}}$, $k = 1, 2$.

Further, considering $\varepsilon_1 < \varepsilon_2$, the components of the singular part $\vec{w} = (w_1, w_2)^T$ can be decomposed as

$$w_1 = w_{1,\varepsilon_1} + w_{1,\varepsilon_2}, \, w_2 = w_{2,\varepsilon_1} + w_{2,\varepsilon_2},$$

$$w_1 = z_{1,\varepsilon_1} + z_{1,\varepsilon_2}, \, w_2 = z_{2,\varepsilon_1} + z_{2,\varepsilon_2}, \tag{7}$$

where

$$|\partial_x^2 w_{1,\varepsilon_1}(x,t)| \leq C\varepsilon_1^{-1} \mathcal{A}_{\varepsilon_1}(x), \, |\partial_x^3 w_{1,\varepsilon_2}(x,t)| \leq C\varepsilon_2^{-3/2} \mathcal{A}_{\varepsilon_2}(x),$$

$$|\partial_x^2 w_{2,\varepsilon_1}(x,t)| \leq C\varepsilon_2^{-1} \mathcal{A}_{\varepsilon_1}(x), \, |\partial_x^3 w_{2,\varepsilon_2}(x,t)| \leq C\varepsilon_2^{-3/2} \mathcal{A}_{\varepsilon_2}(x),$$

$$|\partial_x^2 z_{1,\varepsilon_1}(x,t)| \leq C\varepsilon_1^{-1} \mathcal{A}_{\varepsilon_1}(x), \, |\partial_x^5 z_{1,\varepsilon_2}(x,t)| \leq C\varepsilon_2^{-5/2} \mathcal{A}_{\varepsilon_2}(x), \tag{8}$$

$$|\partial_x^2 z_{2,\varepsilon_1}(x,t)| \leq C\varepsilon_2^{-1} \mathcal{A}_{\varepsilon_1}(x), \, |\partial_x^5 z_{2,\varepsilon_2}(x,t)| \leq C\varepsilon_2^{-5/2} \mathcal{A}_{\varepsilon_2}(x).$$

3 | ADDITIVE-BASED HYBRID NUMERICAL METHOD

We introduce an additive-based hybrid numerical method to approximate the solution of (1). For this purpose, we first consider the discretization of (1) in time on a uniform mesh $\bar{\theta}^M = \{t_j = j\Delta t, 0 \leq j \leq M, \Delta t = T/M\}$, using the additive scheme in Vabishchevich[26] and Samarskii and Vabishchevich[28] as follows:

$$\begin{cases} \left(I + \Delta t \mathcal{L}_{x,\vec{\varepsilon}}^{\mathcal{P}^j}\right) \vec{y}(x, t_j) = \Delta t \vec{f}(x, t_j) + (I + \Delta t \mathcal{Q}^j) \vec{y}(x, t_{j-1}), x \in \Omega, \\ \vec{y}(x, 0) = \vec{u}(x, 0) = \vec{0}, x \in \Omega, \\ \vec{y}(0, t_j) = \vec{y}(1, t_j) = \vec{0}, \\ \text{for } j = 1, \ldots, M, \end{cases} \tag{9}$$

where $\vec{y}$ denotes the approximate values of $\vec{u}$, I denotes the identity operator and $\mathcal{L}_{x,\vec{\varepsilon}}^{\mathcal{P}^j} := \mathcal{D}\partial_x^2 + \mathcal{P}^j$ with $\mathcal{B}^j = \mathcal{P}^j - \mathcal{Q}^j$ and $\mathcal{B}^j = \mathcal{B}(x, t_j)$. Here, in order to decouple the system and reduce the computational cost, we consider $\mathcal{P}^j$ as follows:

$$\mathcal{P}^j = \begin{pmatrix} b_{11}(x, t_j) & 0 \\ b_{21}(x, t_j) & b_{22}(x, t_j) \end{pmatrix}. \tag{10}$$

When $\mathcal{P}^j = \mathcal{B}^j$, the discretization reduces to the implicit Euler scheme in time, which leads to a discretization with a higher computational cost. To obtain the full-discrete method, we discretize (9) using a hybrid finite difference scheme on a specific generalized Shishkin mesh $\bar{\Omega}^N = \{x_i\}_0^N$. We first describe the generalized Shishkin mesh, which is generated using a mesh generating function χ. For a positive integer d, assuming $N = 8d$ and choosing the transition points

$$\tau_2 = \min\left\{1/4, 4\sqrt{\varepsilon_2}\ln N\right\}, \; \tau_1 = \min\left\{\tau_2/2, 4\sqrt{\varepsilon_1}\ln N\right\}, \tag{11}$$

the mesh is defined by $x_i = \chi(i/N)$ for $i = 0, 1, \ldots, N/2$, with $\chi \in C^1[0, 1/2]$. Further, to complete the mesh, we consider the symmetric points of the above around $x = 1/2$. Note that if we take $\tau_2 = 1/4$ and $\tau_1 = 1/8$, the mesh is uniform. We consider $\tau_1 = 4\sqrt{\varepsilon_1}\ln N$, which is a more interesting case of overlapping layers from a theoretical as well as a practical viewpoint. Now, we differentiate two possible cases:

1. If $\tau_2 = 4\sqrt{\varepsilon_2}\ln N$, the mesh is constructed by

$$\chi(\eta) = \begin{cases} 8\eta\tau_1, & \text{if } \eta \in [0, 1/8], \\ q_1(\eta - 1/8)^3 + 8\tau_1(\eta - 1/8) + \tau_1, & \text{if } \eta \in [1/8, 1/4], \\ q_2(\eta - 1/4)^3 + q_3(\eta - 1/4) + \tau_2, & \text{if } \eta \in [1/4, 1/2], \end{cases} \tag{12}$$

where q_1, q_2, and q_3 are determined such that $\chi \in C^1[0, 1/2]$, $\chi(1/4) = \tau_2$, $\chi(1/2) = 1/2$. Thus, it is easy to verify that

$$q_1 = 512(\tau_2 - 2\tau_1), q_2 = 64(1/2 - 7\tau_2 + 10\tau_1), q_3 = (24\tau_2 - 40\tau_1).$$

Note that the mesh generated using χ is uniform on $[0, \tau_1] \cup [1 - \tau_1, 1]$ and nonuniform on $[\tau_1, 1 - \tau_1]$. Further, it is imposed that $\chi'(\eta) \geq 0$ and $\chi''(\eta) \geq 0$ so that χ is increasing and $h_i \geq h_{i-1}, i = 2, \ldots, N/2$, where $h_i = x_i - x_{i-1}$. So, it is required that $(1/2 - 7\tau_2 + 10\tau_1) \geq 0$. We have $h_i \leq CN^{-1}, 1 \leq i \leq N$. The Mean Value Theorem gives

$$h_{i+1} - h_i = \chi\left(\frac{i+1}{N}\right) - 2\chi\left(\frac{i}{N}\right) + \chi\left(\frac{i-1}{N}\right) = N^{-2}\chi''(\gamma_i), \text{ for some } \gamma_i \in \left(\frac{i-1}{N}, \frac{i+1}{N}\right).$$

Thus, using (12), we have

$$|h_{i+1} - h_i| \leq \begin{cases} CN^{-2}\sqrt{\varepsilon_2}\ln N, & N/8 \leq i \leq N/4 - 1, \\ CN^{-2}, & N/4 \leq i \leq N/2 - 1. \end{cases} \tag{13}$$

2. If $\tau_2 = 1/4$ or $(1/2 - 7\tau_2 + 10\tau_1) < 0$, the mesh definition is changed and the mesh is generated using $\hat{\chi} \in C^2[0, 1/2]$ given by

$$\hat{\chi}(\eta) = \begin{cases} 8\eta\tau_1, & \text{if } \eta \in [0, 1/8], \\ \hat{q}_1(\eta - 1/8)^3 + 8\tau_1(\eta - 1/8) + \tau_1, & \text{if } \eta \in [1/8, 1/2], \end{cases} \tag{14}$$

where $\hat{q}_1 = 512(1/2 - 4\tau_1)/27$ is obtained after imposing that $\hat{\chi}(1/2) = 1/2$. The mesh points are given by $\hat{x}_i = \hat{\chi}(i/N), 0 \leq i \leq N/2$, and the mesh generated is symmetric about 1/2. The local mesh sizes $\hat{h}_i = \hat{x}_i - \hat{x}_{i-1}$ satisfy $|\hat{h}_i - \hat{h}_{i-1}| \leq CN^{-2}, N/8 + 1 \leq i \leq N/2$.

The computational domain $\bar{S}$ is discretized as $\bar{S}^{N,M} = \bar{\Omega}^N \times \bar{\theta}^M$, where $\bar{\theta}^M$ is the uniform mesh in time and $\bar{\Omega}^N$ is the generalized Shishkin mesh in space generated by χ or $\hat{\chi}$ as described above. We define $S^{N,M} = \bar{S}^{N,M} \cap S$ and $\gamma^{N,M} = \bar{S}^{N,M} \backslash S^{N,M}$. Then, if we denote by $\vec{U}$ the approximate values of $\vec{u}$ at the corresponding points of $\bar{S}^{N,M}$, the full-discrete method is defined as follows:

- At the initial time and boundaries, $\vec{U}(x_i, 0) = \vec{0}, 0 \leq i \leq N$, $\vec{U}(0, t_j) = \vec{U}(1, t_j) = \vec{0}, 0 \leq j \leq M$.

- On $S^{N,M}$, we define the hybrid finite difference scheme as follows:

$$\mathcal{L}_k^{N,M} \vec{U}(x_i, t_j) := \Gamma_k^{N,M}(D_t^- U_k(x_i, t_j)) + \mathcal{L}_{x,k}^{N,M} \vec{U}(x_i, t_j) = \Gamma_k^{N,M}(f_k(x_i, t_j)),$$
$$k = 1, 2; 1 \leq i \leq N - 1, 1 \leq j \leq M, \tag{15}$$

where the backward Euler operator D_t^- is defined by

$$D_t^- Y(x_i, t_j) := (Y(x_i, t_j) - Y(x_i, t_{j-1}))/\Delta t,$$

and the discrete operators $\mathcal{L}_{x,k}^{N,M}, k = 1, 2$, are defined as follows:

$$\begin{cases} \mathcal{L}_{x,1}^{N,M} \vec{U}(x_i, t_j) := s_{1,i}^{-,j} U_1(x_{i-1}, t_j) + s_{1,i}^{c,j} U_1(x_i, t_j) + s_{1,i}^{+,j} U_1(x_{i+1}, t_j) + \Gamma_1^{N,M} \left(b_{1,2}(x_i, t_j) U_2(x_i, t_{j-1}) \right), \\ \mathcal{L}_{x,2}^{N,M} \vec{U}(x_i, t_j) := s_{2,i}^{-,j} U_2(x_{i-1}, t_j) + s_{2,i}^{c,j} U_2(x_i, t_j) + s_{2,i}^{+,j} U_2(x_{i+1}, t_j) + \Gamma_2^{N,M} \left(b_{2,1}(x_i, t_j) U_1(x_i, t_j) \right). \end{cases} \tag{16}$$

The operators $\Gamma_{k,i}^{N,M}, k = 1, 2$, are defined by

$$\Gamma_k^{N,M} \left(Y(x_i, t_j) \right) := p_{k,i}^- Y(x_{i-1}, t_j) + p_{k,i}^c Y(x_i, t_j) + p_{k,i}^+ Y(x_{i+1}, t_j), \tag{17}$$

and the values of the coefficients $s_{k,i}^{*,j}, *= \{-, c, +\}$ are given by

$$s_{k,i}^{-,j} = -2\varepsilon_k/(h_i(h_i + h_{i+1})) + p_{k,i}^- b_{k,k}(x_{i-1}, t_j),$$

$$s_{k,i}^{+,j} = -2\varepsilon_k/(h_{i+1}(h_i + h_{i+1})) + p_{k,i}^+ b_{k,k}(x_{i+1}, t_j), \tag{18}$$

$$s_{k,i}^{c,j} = 2\varepsilon_k/h_i h_{i+1} + p_{k,i}^c b_{k,k}(x_i, t_j).$$

The values of $p_{k,i}^*, *= \{-, c, +\}$ are chosen so that the matrix associated to the discrete scheme is an M-matrix and the resulting numerical approximation is of higher order. We choose between the standard central difference scheme (CDS) (which corresponds to the choice $p_{k,i}^- = p_{k,i}^+ = 0$ and $p_{k,i}^c = 1$) and the high order scheme (HOS), which corresponds to

$$p_{k,i}^- = \frac{1}{6} \left(\frac{h_j^2 + h_j h_{j+1} - h_{j+1}^2}{h_j(h_j + h_{j+1})} \right), \quad p_{k,i}^+ = \frac{1}{6} \left(\frac{h_{j+1}^2 + h_j h_{j+1} - h_j^2}{h_{j+1}(h_j + h_{j+1})} \right), \quad p_{k,i}^c = 1 - p_{k,i}^- - p_{k,i}^+. \tag{19}$$

The numerical scheme uses either the HOS or the CDS approach depending on the regions and the relation between the mesh widths and the perturbation parameters according to the following:

- If $\bar{S}^{N,M}$ is generated using χ, the additive-based hybrid numerical method is defined by

$$\begin{array}{ll} \text{HOS,} & \text{if } x_i \in (0, \tau_1) \cup (1 - \tau_1, 1) \text{ and } k = 1, 2, \\ \text{CDS,} & \text{if } x_i \in [\tau_1, \tau_2) \cup (1 - \tau_2, 1 - \tau_1], k = 1, \text{and } h^2(\|b_{11}\|_{\bar{S}} + \Delta t^{-1}) \geq 6\varepsilon_1, \\ \text{HOS,} & \text{if } x_i \in [\tau_1, \tau_2) \cup (1 - \tau_2, 1 - \tau_1], k = 1, \text{and } h^2(\|b_{11}\|_{\bar{S}} + \Delta t^{-1}) < 6\varepsilon_1, \\ \text{HOS,} & \text{if } x_i \in [\tau_1, \tau_2) \cup (1 - \tau_2, 1 - \tau_1] \text{ and } k = 2, \\ \text{CDS,} & \text{if } x_i \in [\tau_2, 1 - \tau_2], k = 1, \text{ and } H^2(\|b_{11}\|_{\bar{S}} + \Delta t^{-1}) \geq 6\varepsilon_1, \\ \text{HOS,} & \text{if } x_i \in [\tau_2, 1 - \tau_2], k = 1, \text{ and } H^2(\|b_{11}\|_{\bar{S}} + \Delta t^{-1}) < 6\varepsilon_1, \\ \text{CDS,} & \text{if } x_i \in [\tau_2, 1 - \tau_2], k = 2, \text{ and } H^2(\|b_{22}\|_{\bar{S}} + \Delta t^{-1}) \geq 6\varepsilon_1, \\ \text{HOS,} & \text{if } x_i \in [\tau_2, 1 - \tau_2], k = 2, \text{ and } H^2(\|b_{22}\|_{\bar{S}} + \Delta t^{-1}) < 6\varepsilon_1, \end{array}$$

where $h = \max\limits_{N/8+1 \leq i \leq N/4} \{h_i\}$ and $H = \max\limits_{N/4+1 \leq i \leq N/2} \{h_i\}$.

- If $\bar{S}^{N,M}$ is generated using $\hat{\chi}$, the additive-based hybrid numerical method is defined by

$$
\begin{cases}
\text{HOS,} & \text{if } \hat{x}_i \in (0, \tau_1) \cup (1 - \tau_1, 1) \text{ and } k = 1, 2, \\
\text{CDS,} & \text{if } \hat{x}_i \in [\tau_1, 1 - \tau_1], k = 1, \text{ and } \hat{H}^2(\|b_{11}\|_{\bar{S}} + \Delta t^{-1}) \geq 6\varepsilon_1, \\
\text{HOS,} & \text{if } \hat{x}_i \in [\tau_1, 1 - \tau_1], k = 1, \text{ and } \hat{H}^2(\|b_{11}\|_{\bar{S}} + \Delta t^{-1}) < 6\varepsilon_1, \\
\text{HOS,} & \text{if } \hat{x}_i \in [\tau_1, 1 - \tau_1] \text{ and } k = 2,
\end{cases}
$$

where $\hat{H} = \max_{N/8+1 \leq i \leq N/2} \{\hat{h}_i\}$.

The above choices are taken in order to ensure that

$$
s_{k,i}^{-,j} \leq 0, \ s_{k,i}^{+,j} \leq 0, \ s_{k,i}^{c,j} > 0, \ s_{k,i}^{-,j} + s_{k,i}^{c,j} + s_{k,i}^{+,j} > 0, \ p_{k,i}^* \geq 0, \ *= \{-, c, +\}. \tag{20}
$$

We also require the mild assumption $M \leq CN^2/\ln^2 N$; then the HOS suffices so that (20) holds for the coefficients of the second equation (i.e., for (15) with $k = 2$) in $[\tau_1, 1 - \tau_1]$. Considering the sign pattern (20), one can observe that the matrix corresponding to the additive-based hybrid numerical method is an M-matrix. Hence, the following discrete comparison principle is satisfied and the method is uniformly stable.

Lemma 3.1. *Suppose $\vec{Y}_1$ and $\vec{Y}_2$ are two discrete approximations of $\vec{u}$ such that $\mathcal{L}^{N,M}\vec{Y}_1 \geq \mathcal{L}^{N,M}\vec{Y}_2$ on $S^{N,M}$ and $\vec{Y}_1 \geq \vec{Y}_2$ on $\gamma^{N,M}$. Then, $\vec{Y}_1 \geq \vec{Y}_2$ on $\bar{S}^{N,M}$.*

Remark 3.2. The assumption $M \leq CN^2/\ln^2 N$ is very mild and can be easily satisfied. The values of N and M in Tables 1–9 satisfy this assumption. Further, the reasonability of this assumption can also be seen from the fact that such an assumption is prevalent in the construction of high-order numerical methods for time-dependent singularly perturbed differential equations. See, for example, Bujanda et al.,[31, Lemma 3] Clavero et al.,[32, Lemma 4.1] Kumar and Rao,[33, Lemma 4] and Clavero and Gracia.[34, Lemma 11]

4 | CONVERGENCE ANALYSIS

We now give a detailed uniform convergence analysis of the additive-based hybrid numerical method in the case where the discrete mesh $\bar{S}^{N,M}$ is generated using the mesh generating function χ. The analysis of the method in the case where the mesh $\bar{S}^{N,M}$ is generated using the function $\hat{\chi}$ can be done using similar arguments.

Theorem 4.1. *Suppose $\vec{u}$ is the solution of (1) and $\vec{U}$ is the solution of the additive-based hybrid numerical method on a rectangular mesh $\bar{S}^{N,M}$ generated using χ. Then, we have*

$$
\|\vec{U} - \vec{u}\|_{\bar{S}^{N,M}} \leq C(M^{-1} + MN^{-4}\ln N + N^{-4}\ln^4 N + MN^{-5}\ln^3 N). \tag{21}
$$

Proof. The truncation error for the additive-based hybrid numerical method is given by

$$
\begin{aligned}
\left[\mathcal{L}_k^{N,M}(\vec{U} - \vec{u})\right]_{i,j} &= \Gamma_k^{N,M}\left(f_k(x_i, t_j)\right) - \mathcal{L}_k^{N,M}\vec{u}(x_i, t_j) = \Gamma_k^{N,M}(\mathcal{L}_{\bar{\varepsilon}}\vec{u}(x_i, t_j)) - \mathcal{L}_k^{N,M}\vec{u}(x_i, t_j) \\
&= \Gamma_k^{N,M}\left(\partial_t u_k(x_i, t_j) - D_t^- u_k(x_i, t_j)\right) + \varepsilon_k\left(\delta_x^2 u_k(x_i, t_j) - \Gamma_k^{N,M}(\partial_x^2 u_k(x_i, t_j))\right) \\
&\quad + \psi(k)\left(\Gamma_k^{N,M}\left(b_{k,3-k}(x_i, t_j)u_{3-k}(x_i, t_j)\right) - b_{k,3-k}(x_i, t_j)u_{3-k}(x_i, t_{j-1})\right), k = 1, 2,
\end{aligned}
$$

where $\psi(1) = 1, \psi(2) = 0$. Here, δ_x^2 is the central difference operator defined by

$$
\delta_x^2 \xi(x_i, t_j) = \frac{2}{h_i + h_{i+1}}\left(\frac{\xi(x_{i+1}, t_j) - \xi(x_i, t_j)}{h_{i+1}} - \frac{\xi(x_i, t_j) - \xi(x_{i-1}, t_j)}{h_i}\right).
$$

$\square$

TABLE 1 Errors and convergence rates using the additive-based hybrid numerical method (15)–(16) for Example 1

		$N = 64$ $M = 4$	$N = 128$ $M = 8$	$N = 256$ $M = 16$	$N = 512$ $M = 32$	$N = 1024$ $M = 64$
$\varepsilon_1 = 2^{-17}$	$e_1^{N,M}$	9.2382e−03	4.9854e−03	2.5960e−03	1.3336e−03	6.7621e−04
	$r_1^{N,M}$	0.890	0.941	0.961	0.980	
$\varepsilon_2 = 2^{-12}$	$e_2^{N,M}$	6.4983e−03	3.9160e−03	2.1705e−03	1.1475e−03	5.9015e−04
	$r_2^{N,M}$	0.731	0.851	0.919	0.959	
$\varepsilon_1 = 2^{-19}$	$e_1^{N,M}$	9.2406e−03	4.9854e−03	2.6238e−03	1.3485e−03	6.8382e−04
	$r_1^{N,M}$	0.890	0.926	0.960	0.980	
$\varepsilon_2 = 2^{-14}$	$e_2^{N,M}$	6.8114e−03	4.0415e−03	2.2481e−03	1.1854e−03	6.0892e−04
	$r_2^{N,M}$	0.753	0.846	0.923	0.961	
$\varepsilon_1 = 2^{-23}$	$e_1^{N,M}$	9.2405e−03	5.0175e−03	2.6443e−03	1.3592e−03	6.8927e−04
	$r_1^{N,M}$	0.881	0.924	0.960	0.979	
$\varepsilon_2 = 2^{-17}$	$e_2^{N,M}$	7.0393e−03	4.1532e−03	2.3072e−03	1.2130e−03	6.2304e−04
	$r_2^{N,M}$	0.761	0.848	0.927	0.961	
$\varepsilon_1 = 2^{-10}$	$e_1^{N,M}$	9.1184e−03	4.9076e−03	2.5505e−03	1.3007e−03	6.5687e−04
	$r_1^{N,M}$	0.894	0.944	0.971	0.986	
$\varepsilon_2 = 2^{-5}$	$e_2^{N,M}$	3.9570e−03	2.5798e−03	1.5020e−03	8.2331e−04	4.3117e−04
	$r_2^{N,M}$	0.617	0.780	0.867	0.933	
$\varepsilon_1 = 10^{-15}$	$e_1^{N,M}$	9.3617e−03	5.0439e−03	2.6525e−03	1.3632e−03	6.9129e−04
	$r_1^{N,M}$	0.892	0.927	0.960	0.980	
$\varepsilon_2 = 10^{-6}$	$e_2^{N,M}$	7.1071e−03	4.2020e−03	2.3299e−03	1.2248e−03	6.2864e−04
	$r_2^{N,M}$	0.758	0.851	0.928	0.962	
$\varepsilon_1 = 10^{-8}$	$e_1^{N,M}$	6.1188e−03	3.6447e−03	2.0364e−03	1.0851e−03	5.6206e−04
	$r_1^{N,M}$	0.747	0.840	0.908	0.949	
$\varepsilon_2 = 1$	$e_2^{N,M}$	1.1498e−03	6.8237e−04	3.9229e−04	2.5576e−04	1.4872e−04
	$r_2^{N,M}$	0.753	0.799	0.617	0.782	

Now using Taylor expansions and derivative bounds, we have

$$|\Gamma_k^{N,M}\left(\partial_t u_k(x_i, t_j) - D_t^- u_k(x_i, t_j)\right)| \le C(t_j - t_{j-1})\|\partial_t^2 u_k(x_i, .)\|_{[t_{j-1}, t_j]} \le CM^{-1},$$

and

$$|\Gamma_k^{N,M}\left(b_{k,3-k}(x_i, t_j)u_{3-k}(x_i, t_j)\right) - b_{k,3-k}(x_i, t_j)u_{3-k}(x_i, t_{j-1})| \le C(t_j - t_{j-1})\|\partial_t u_{3-k}(x_i, .)\|_{[t_{j-1}, t_j]}$$
$$\le CM^{-1}.$$

For brevity we denote $\varepsilon_k|\Gamma_k^{N,M}(\partial_x^2 u_k(x_i, t_j)) - \delta_x^2 u_k(x_i, t_j)|$ by $\varepsilon_k|[\Gamma_k^{N,M}\left(\partial_x^2 u_k\right) - \delta_x^2 u_k]_{i,j}|$. Now suppose the CDS is activated, then Taylor expansions give

$$\varepsilon_k|[\Gamma_k^{N,M}\left(\partial_x^2 u_k\right) - \delta_x^2 u_k]_{i,j}| \le \begin{cases} C\varepsilon_k h_i^2\|\partial_x^4 u_k(., t_j)\|_{[x_{i-1}, x_{i+1}]}, & \text{if } h_i = h_{i+1}, \\ C\varepsilon_k|h_{i+1} - h_i|\|\partial_x^3 u_k(., t_j)\|_{[x_{i-1}, x_{i+1}]}, & \text{if } h_i \ne h_{i+1}. \end{cases} \tag{22}$$

Otherwise, when the HOS is activated, then Taylor expansions give

$$\varepsilon_k|[\Gamma_k^{N,M}\left(\partial_x^2 u_k\right) - \delta_x^2 u_k]_{i,j}| \le \begin{cases} C\varepsilon_k h_i^4\|\partial_x^6 u_k(., t_j)\|_{[x_{i-1}, x_{i+1}]}, & \text{if } h_i = h_{i+1}, \\ C\varepsilon_k|h_{i+1} - h_i| \max\left\{h_i^2, h_{i+1}^2\right\}\|\partial_x^5 u_k(., t_j)\|_{[x_{i-1}, x_{i+1}]}, & \text{if } h_i \ne h_{i+1}. \end{cases} \tag{23}$$

TABLE 2 Errors and convergence rates using the additive-based hybrid numerical method (15)–(16) for Example 1

		$N = 64$ $M = 4$	$N = 128$ $M = 16$	$N = 256$ $M = 64$	$N = 512$ $M = 256$	$N = 1024$ $M = 1024$
$\varepsilon_1 = 2^{-17}$	$e_1^{N,M}$	9.2382e−03	2.5959e−03	6.7618e−04	1.7086e−04	4.2830e−05
	$r_1^{N,M}$	1.831	1.941	1.985	1.996	
$\varepsilon_2 = 2^{-12}$	$e_2^{N,M}$	6.4983e−03	2.1703e−03	5.9016e−04	1.5072e−04	3.7883e−05
	$r_2^{N,M}$	1.582	1.879	1.969	1.992	
$\varepsilon_1 = 2^{-19}$	$e_1^{N,M}$	9.2406e−03	2.6236e−03	6.8379e−04	1.7280e−04	4.3317e−05
	$r_1^{N,M}$	1.816	1.940	1.984	1.996	
$\varepsilon_2 = 2^{-14}$	$e_2^{N,M}$	6.8114e−03	2.2481e−03	6.0893e−04	1.5541e−04	3.9055e−05
	$r_2^{N,M}$	1.599	1.884	1.970	1.992	
$\varepsilon_1 = 2^{-23}$	$e_1^{N,M}$	9.2405e−03	2.6443e−03	6.8926e−04	1.7418e−04	4.3664e−05
	$r_1^{N,M}$	1.805	1.940	1.984	1.996	
$\varepsilon_2 = 2^{-17}$	$e_2^{N,M}$	7.0393e−03	2.3072e−03	6.2304e−04	1.5892e−04	3.9933e−05
	$r_2^{N,M}$	1.609	1.889	1.971	1.993	
$\varepsilon_1 = 2^{-10}$	$e_1^{N,M}$	9.1184e−03	2.5504e−03	6.5687e−04	1.6546e−04	4.1443e−05
	$r_1^{N,M}$	1.838	1.957	1.989	1.997	
$\varepsilon_2 = 2^{-5}$	$e_2^{N,M}$	3.9570e−03	1.5020e−03	4.3117e−04	1.1179e−04	2.8209e−05
	$r_2^{N,M}$	1.397	1.800	1.947	1.987	
$\varepsilon_1 = 10^{-15}$	$e_1^{N,M}$	9.3617e−03	2.6690e−03	6.9256e−04	1.7478e−04	4.3798e−05
	$r_1^{N,M}$	1.810	1.946	1.986	1.997	
$\varepsilon_2 = 10^{-6}$	$e_2^{N,M}$	7.1071e−03	2.3301e−03	6.2868e−04	1.6031e−04	4.0281e−05
	$r_2^{N,M}$	1.609	1.890	1.971	1.993	
$\varepsilon_1 = 10^{-8}$	$e_1^{N,M}$	6.1188e−03	2.0364e−03	5.6206e−04	1.4448e−04	3.6381e−05
	$r_1^{N,M}$	1.587	1.857	1.960	1.990	
$\varepsilon_2 = 1$	$e_2^{N,M}$	7.7574e−04	3.9229e−04	1.4871e−04	4.2070e−05	1.0869e−05
	$r_2^{N,M}$	0.984	1.399	1.822	1.953	

To get a bound for $\varepsilon_k|[\Gamma_k^{N,M}\left(\partial_x^2 u_k\right) - \delta_x^2 u_k]_{i,j}|$, we consider the decomposition $u_k = v_k + w_k$ and use the triangle inequality as follows

$$\varepsilon_k|[\Gamma_k^{N,M}\left(\partial_x^2 u_k\right) - \delta_x^2 u_k]_{i,j}| \leq \varepsilon_k|[\Gamma_k^{N,M}\left(\partial_x^2 v_k\right) - \delta_x^2 v_k]_{i,j}| + \varepsilon_k|[\Gamma_k^{N,M}\left(\partial_x^2 w_k\right) - \delta_x^2 w_k]_{i,j}|. \tag{24}$$

The first term on the right hand side of (24) is bounded by considering the following cases.

(a) If $x_i \in [\tau_1, \tau_2) \cup (1 - \tau_2, 1 - \tau_1]$, and the CDS is used. In this case $\varepsilon_1 \leq CN^{-2}M$ and $|h_{i+1} - h_i| \leq C\sqrt{\varepsilon_2}N^{-2}\ln N$. Thus, we get

$$\varepsilon_1|[\Gamma_1^{N,M}\left(\partial_x^2 v_1\right) - \delta_x^2 v_1]_{i,j}| \leq C\varepsilon_1|h_{i+1} - h_i|\|\partial_x^3 v_1(.,t_j)\|_{[x_{i-1},x_{i+1}]} \leq CMN^{-4}\ln N.$$

(b) If $x_i \in [\tau_2, 1 - \tau_2]$, and the CDS is applied. In this case, $\varepsilon_k \leq CN^{-2}M, k = 1, 2$, and $|h_{i+1} - h_i| \leq CN^{-2}$. Hence, we have

$$\varepsilon_k|[\Gamma_k^{N,M}\left(\partial_x^2 v_k\right) - \delta_x^2 v_k]_{i,j}| \leq C\varepsilon_k|h_{i+1} - h_i|\|\partial_x^3 v_k(.,t_j)\|_{[x_{i-1},x_{i+1}]} \leq CMN^{-4}.$$

(c) If the HOS is applied, then we get

$$\varepsilon_k|[\Gamma_k^{N,M}\left(\partial_x^2 u_k\right) - \delta_x^2 u_k]_{i,j}| \leq C\varepsilon_k|h_{i+1} - h_i|\max\{h_i^2, h_{i+1}^2\}\|\partial_x^5 u_k(.,t_j)\|_{[x_{i-1},x_{i+1}]} + C\varepsilon_k h_i^4\|\partial_x^6 u_k(.,t_j)\|_{[x_{i-1},x_{i+1}]}$$
$$\leq C\varepsilon_2 N^{-4}.$$

On combining the bounds for the regular part we have

$$\varepsilon_k|[\Gamma_k^{N,M}\left(\partial_x^2 v_k\right) - \delta_x^2 v_k]_{i,j}| \leq CMN^{-4}\ln N. \tag{25}$$

		$N = 64$ $M = 4$	$N = 128$ $M = 32$	$N = 256$ $M = 256$	$N = 512$ $M = 2048$	$N = 1024$ $M = 16\,384$
$\varepsilon_1 = 2^{-17}$	$e_1^{N,M}$	9.2382e−03	1.3332e−03	1.7084e−04	2.1422e−05	2.6789e−06
	$r_1^{N,M}$	2.793	2.964	2.995	2.999	
$\varepsilon_2 = 2^{-12}$	$e_2^{N,M}$	6.4983e−03	1.1473e−03	1.5072e−04	1.8960e−05	2.3718e−06
	$r_2^{N,M}$	2.502	2.928	2.991	2.999	
$\varepsilon_1 = 2^{-19}$	$e_1^{N,M}$	9.2406e−03	1.3483e−03	1.7278e−04	2.1666e−05	2.7094e−06
	$r_1^{N,M}$	2.777	2.964	2.995	2.999	
$\varepsilon_2 = 2^{-14}$	$e_2^{N,M}$	6.8114e−03	1.1854e−03	1.5541e−04	1.9545e−05	2.4450e−06
	$r_2^{N,M}$	2.522	2.931	2.991	2.999	
$\varepsilon_1 = 2^{-23}$	$e_1^{N,M}$	9.2405e−03	1.3591e−03	1.7418e−04	2.1841e−05	2.7313e−06
	$r_1^{N,M}$	2.765	2.964	2.995	2.999	
$\varepsilon_2 = 2^{-17}$	$e_2^{N,M}$	7.0393e−03	1.2129e−03	1.5892e−04	1.9983e−05	2.4998e−06
	$r_2^{N,M}$	2.537	2.932	2.991	2.999	
$\varepsilon_1 = 2^{-10}$	$e_1^{N,M}$	9.1184e−03	1.3007e−03	1.6546e−04	2.0728e−05	2.5918e−06
	$r_1^{N,M}$	2.809	2.975	2.997	3.000	
$\varepsilon_2 = 2^{-5}$	$e_2^{N,M}$	3.9570e−03	8.2324e−04	1.1179e−04	1.4127e−05	1.7682e−06
	$r_2^{N,M}$	2.265	2.880	2.984	2.998	
$\varepsilon_1 = 10^{-15}$	$e_1^{N,M}$	9.3617e−03	1.3857e−03	1.7642e−04	2.2024e−05	2.7473e−06
	$r_1^{N,M}$	2.756	2.973	3.002	3.003	
$\varepsilon_2 = 10^{-6}$	$e_2^{N,M}$	7.1071e−03	1.2250e−03	1.6035e−04	2.0159e−05	2.5217e−06
	$r_2^{N,M}$	2.536	2.933	2.992	2.999	
$\varepsilon_1 = 10^{-8}$	$e_1^{N,M}$	6.1188e−03	1.0843e−03	1.4447e−04	1.8212e−05	2.2789e−06
	$r_1^{N,M}$	2.496	2.908	2.988	2.998	
$\varepsilon_2 = 1$	$e_2^{N,M}$	7.7574e−04	2.5576e−04	4.2070e−05	5.4642e−06	6.8640e−07
	$r_2^{N,M}$	1.601	2.604	2.945	2.993	

TABLE 3 Errors and convergence rates using the additive-based hybrid numerical method (15)–(16) for Example 1

Next we find the bound for the layer part $\vec{w}$. Let $x_i \in (0, \tau_1) \cup (1 - \tau_1, 1)$. Then, using the bound (23) and bounds on the derivatives with $h_i = h_{i+1} = 4N^{-1}\sqrt{\varepsilon_1}\ln N$, we get

$$\varepsilon_k |\Gamma_k^{N,M}\left(\partial_x^2 w_k(x_i, t_j)\right) - \delta_x^2 w_k(x_i, t_j)| \le C\varepsilon_k h_i^4 \|\partial_x^6 w_k(., t_j)\|_{[x_{i-1}, x_{i+1}]} \le CN^{-4}\ln^4 N.$$

Now suppose $x_i \in [\tau_1, \tau_2] \cup (1 - \tau_2, 1 - \tau_1]$ and the CDS is used. Using the decomposition $w_1 = w_{1,\varepsilon_1} + w_{1,\varepsilon_2}$, $\varepsilon_1 \le CN^{-2}M$, and $|h_{i+1} - h_i| \le C\sqrt{\varepsilon_2}N^{-2}\ln N$, we obtain

$$\varepsilon_1 |[\Gamma_1^{N,M}\left(\partial_x^2 w_1\right) - \delta_x^2 w_1]_{i,j}| \le C\varepsilon_1 \left(\|\partial_x^2 w_{1,\varepsilon_1}(., t_j)\|_{[x_{i-1}, x_{i+1}]} + |h_{i+1} - h_i| \|\partial_x^3 w_{1,\varepsilon_2}(., t_j)\|_{[x_{i-1}, x_{i+1}]} \right)$$

$$\le C\mathcal{A}_{\varepsilon_1}(\tau_1) + C\varepsilon_1 \varepsilon_2^{1/2} N^{-2} \ln N \varepsilon_2^{-3/2} \mathcal{A}_{\varepsilon_2}(\tau_1) \le CN^{-4} + CN^{-2} \ln N(\sigma^2 N^{-4\sigma}),$$

where $\sigma = \sqrt{\varepsilon_1/\varepsilon_2}$. We have $2\sigma N^{-2\sigma} \le N^{-1}$ for $2\sigma \le 1$. Hence, we obtain

$$\varepsilon_1 |[\Gamma_1^{N,M}\left(\partial_x^2 w_1\right) - \delta_x^2 w_1]_{i,j}| \le CN^{-4} + CN^{-4} \ln N.$$

In the other case, that is $2\sigma > 1$, we use the inequality $\varepsilon_1 \varepsilon_2^{-1} \le CMN^{-2}\ln^2 N$ to get

$$\varepsilon_1 |[\Gamma_1^{N,M}\left(\partial_x^2 w_1\right) - \delta_x^2 w_1]_{i,j}| \le CN^{-4} + CMN^{-5}\ln^3 N.$$

TABLE 4 Errors and convergence rates using the implicit Euler-based hybrid numerical method[12] for Example 1

		$N = 64$ $M = 4$	$N = 128$ $M = 32$	$N = 256$ $M = 256$	$N = 512$ $M = 2048$	$N = 1024$ $M = 16384$
$\varepsilon_1 = 2^{-17}$	$e_1^{N,M}$	3.4438e−03	7.3255e−04	1.0156e−04	1.2869e−05	1.6114e−06
	$r_1^{N,M}$	2.233	2.850	2.980	2.997	
$\varepsilon_2 = 2^{-12}$	$e_2^{N,M}$	8.6254e−03	1.3627e−03	1.7592e−04	2.2083e−05	2.7617e−06
	$r_2^{N,M}$	2.662	2.953	2.994	2.999	
$\varepsilon_1 = 2^{-19}$	$e_1^{N,M}$	3.4437e−03	7.3294e−04	1.0156e−04	1.2869e−05	1.6113e−06
	$r_1^{N,M}$	2.232	2.851	2.980	2.997	
$\varepsilon_2 = 2^{-14}$	$e_2^{N,M}$	9.0207e−03	1.4174e−03	1.8275e−04	2.2934e−05	2.8681e−06
	$r_2^{N,M}$	2.670	2.955	2.994	2.999	
$\varepsilon_1 = 2^{-23}$	$e_1^{N,M}$	3.4437e−03	7.3293e−04	1.0156e−04	1.2869e−05	1.6113e−06
	$r_1^{N,M}$	2.232	2.851	2.980	2.997	
$\varepsilon_2 = 2^{-17}$	$e_2^{N,M}$	9.3602e−03	1.4582e−03	1.8784e−04	2.3570e−05	2.9477e−06
	$r_2^{N,M}$	2.682	2.957	2.994	2.999	
$\varepsilon_1 = 2^{-10}$	$e_1^{N,M}$	3.4586e−03	7.3516e−04	1.0181e−04	1.2898e−05	1.6150e−06
	$r_1^{N,M}$	2.234	2.852	2.981	2.997	
$\varepsilon_2 = 2^{-5}$	$e_2^{N,M}$	4.7273e−03	8.8817e−04	1.1816e−04	1.4893e−05	1.8636e−06
	$r_2^{N,M}$	2.412	2.910	2.988	2.998	
$\varepsilon_1 = 10^{-15}$	$e_1^{N,M}$	3.4437e−03	7.3208e−04	1.0157e−04	1.2868e−05	1.6113e−06
	$r_1^{N,M}$	2.234	2.850	2.981	2.997	
$\varepsilon_2 = 10^{-6}$	$e_2^{N,M}$	9.4901e−03	1.4738e−03	1.8988e−04	2.3827e−05	2.9796e−06
	$r_2^{N,M}$	2.687	2.956	2.994	2.999	
$\varepsilon_1 = 10^{-8}$	$e_1^{N,M}$	4.3882e−03	8.5090e−04	1.1455e−04	1.4461e−05	1.8098e−06
	$r_1^{N,M}$	2.367	2.8930	2.986	2.998	
$\varepsilon_2 = 1$	$e_2^{N,M}$	4.2684e−04	2.5880e−04	4.2287e−05	5.4871e−06	6.8920e−07
	$r_2^{N,M}$	0.722	2.613	2.946	2.993	

Now suppose $x_i \in [\tau_1, \tau_2] \cup (1 - \tau_2, 1 - \tau_1]$ and the HOS is used. Then using the decomposition $w_k = z_{k,\varepsilon_1} + z_{k,\varepsilon_2}$ and the inequality $|h_{i+1} - h_i| \leq C\sqrt{\varepsilon_2}N^{-2} \ln N$ with $\varepsilon_2^{-1} \leq C\ln^2 N$, we have

$$\varepsilon_k |[\Gamma_k^{N,M}\left(\partial_x^2 w_k\right) - \delta_x^2 w_k]_{i,j}| \leq C\varepsilon_k \left(\|\partial_x^2 z_{k,\varepsilon_1}(.,t_j)\|_{[x_{i-1},x_{i+1}]} + |h_{i+1} - h_i| \max(h_i^2, h_{i+1}^2)\|\partial_x^5 z_{k,\varepsilon_2}(.,t_j)\|_{[x_{i-1},x_{i+1}]}\right)$$
$$\leq C\mathcal{A}_{\varepsilon_1}(\tau_1) + C\varepsilon_k\varepsilon_2^{-1}N^{-4}\ln^3 N \leq CN^{-4} + CN^{-4}\ln^3 N.$$

Finally, for $x_i \in [\tau_2, 1 - \tau_2] = [\tau_2, 1/2] \cup [1/2, 1 - \tau_2]$, we first assume $x_i \in [\tau_2, 1/2]$. Then, we have

$$\varepsilon_k |[\Gamma_k^{N,M}\left(\partial_x^2 w_k\right) - \delta_x^2 w_k]_{i,j}| \leq C\varepsilon_k \|\partial_x^2 w_k(.,t_j)\|_{[x_{i-1},x_{i+1}]} \leq C\mathcal{A}_{\varepsilon_2}(\tau_2) \leq CN^{-4}.$$

Likewise, we have

$$\varepsilon_k |[\Gamma_k^{N,M}\left(\partial_x^2 w_k\right) - \delta_x^2 w_k]_{i,j}| \leq CN^{-4} \text{ for } x_i \in [1/2, 1 - \tau_2].$$

Collecting the bounds for the singular part we have

$$\varepsilon_k \left|\left[\Gamma_k^{N,M}\left(\partial_x^2 w_k\right) - \delta_x^2 w_k\right]_{i,j}\right| \leq CN^{-4}\ln^4 N + CMN^{-5}\ln^3 N. \qquad (26)$$

Thus, we get the following bound for the truncation error:

$$\left|\left[\mathcal{L}_k^{N,M}(\vec{U} - \vec{u})\right]_{i,j}\right| \leq C(M^{-1} + MN^{-4}\ln N + N^{-4}\ln^4 N + MN^{-5}\ln^3 N), k = 1, 2. \qquad (27)$$

Hence, using Lemma 3.1 with a constant barrier function, we get the desired result.

		$N = 32$ $M = 16$	$N = 64$ $M = 64$	$N = 128$ $M = 256$	$N = 256$ $M = 1052$	$N = 512$ $M = 4096$
$\varepsilon_1 = 2^{-17}$	$e_1^{N,M}$	1.7060e−03	1.1375e−03	3.8660e−04	6.7587e−05	6.9543e−06
	$r_1^{N,M}$	0.585	1.557	2.516	3.281	
$\varepsilon_2 = 2^{-12}$	$e_2^{N,M}$	3.5804e−04	1.1843e−04	2.0427e−05	2.9303e−06	3.8249e−07
	$r_2^{N,M}$	1.596	2.535	2.801	2.938	
$\varepsilon_1 = 2^{-19}$	$e_1^{N,M}$	1.7128e−03	1.1402e−03	3.8708e−04	6.7628e−05	6.9566e−06
	$r_1^{N,M}$	0.587	1.559	2.517	3.281	
$\varepsilon_2 = 2^{-14}$	$e_2^{N,M}$	3.5782e−04	1.1846e−04	2.0428e−05	2.9700e−06	3.8310e−07
	$r_2^{N,M}$	1.595	2.536	2.782	2.954	
$\varepsilon_1 = 2^{-23}$	$e_1^{N,M}$	1.7233e−03	1.1426e−03	3.8807e−04	6.7947e−05	7.0215e−06
	$r_1^{N,M}$	0.593	1.558	2.514	3.274	
$\varepsilon_2 = 2^{-17}$	$e_2^{N,M}$	2.0068e−04	6.4268e−05	1.1981e−05	1.6990e−06	2.2504e−07
	$r_2^{N,M}$	1.643	2.423	2.818	2.916	
$\varepsilon_1 = 10^{-15}$	$e_1^{N,M}$	2.0671e−03	1.1997e−03	3.9754e−04	6.9535e−05	7.2686e−06
	$r_1^{N,M}$	0.785	1.593	2.515	3.258	
$\varepsilon_2 = 10^{-6}$	$e_2^{N,M}$	5.2379e−03	1.0323e−03	8.3511e−05	7.7518e−06	9.5226e−07
	$r_2^{N,M}$	2.343	3.628	3.429	3.025	

TABLE 5 Errors and convergence rates using the additive-based hybrid numerical method (15)–(16) for Example 2

		$N = 64$ $M = 4$	$N = 128$ $M = 32$	$N = 256$ $M = 256$	$N = 512$ $M = 2048$	$N = 1024$ $M = 16384$
$\varepsilon_1 = 2^{-17}$	$e_1^{N,M}$	2.0907e−03	2.8582e−04	6.0305e−05	6.6010e−06	6.2502e−07
	$r_1^{N,M}$	2.871	2.245	3.191	3.401	
$\varepsilon_2 = 2^{-12}$	$e_2^{N,M}$	6.5854e−04	9.2251e−05	1.1714e−05	1.4673e−06	1.8347e−07
	$r_2^{N,M}$	2.836	2.977	2.997	2.999	
$\varepsilon_1 = 2^{-19}$	$e_1^{N,M}$	2.1313e−03	2.9161e−04	6.0331e−05	6.6025e−06	6.2514e−07
	$r_1^{N,M}$	2.870	2.273	3.192	3.401	
$\varepsilon_2 = 2^{-14}$	$e_2^{N,M}$	6.6747e−04	9.3463e−05	1.1866e−05	1.4862e−06	1.8583e−07
	$r_2^{N,M}$	2.836	2.978	2.997	3.000	
$\varepsilon_1 = 2^{-23}$	$e_1^{N,M}$	2.1636e−03	3.0226e−04	6.2270e−05	6.7481e−06	6.4575e−07
	$r_1^{N,M}$	2.840	2.279	3.206	3.385	
$\varepsilon_2 = 2^{-17}$	$e_2^{N,M}$	6.7388e−04	9.4343e−05	1.1976e−05	1.4999e−06	1.8753e−07
	$r_2^{N,M}$	2.836	2.978	2.997	3.000	
$\varepsilon_1 = 2^{-10}$	$e_1^{N,M}$	1.6347e−03	2.2147e−04	2.7974e−05	3.5121e−06	4.4048e−07
	$r_1^{N,M}$	2.884	2.985	2.994	2.995	
$\varepsilon_2 = 2^{-5}$	$e_2^{N,M}$	5.7027e−04	8.0181e−05	1.0188e−05	1.2762e−06	1.5959e−07
	$r_2^{N,M}$	2.830	2.976	2.997	2.999	
$\varepsilon_1 = 10^{-15}$	$e_1^{N,M}$	2.1929e−03	3.9576e−04	6.9337e−05	7.2579e−06	7.1691e−07
	$r_1^{N,M}$	2.470	2.513	3.256	3.340	
$\varepsilon_2 = 10^{-6}$	$e_2^{N,M}$	1.4844e−03	1.4296e−04	1.4354e−05	1.5713e−06	1.8824e−07
	$r_2^{N,M}$	3.376	3.316	3.191	3.061	
$\varepsilon_1 = 10^{-8}$	$e_1^{N,M}$	1.2133e−03	3.9567e−04	6.9329e−05	7.2574e−06	7.1685e−07
	$r_1^{N,M}$	1.616	2.513	3.256	3.340	
$\varepsilon_2 = 1$	$e_2^{N,M}$	1.0604e−04	1.4120e−05	1.7760e−06	2.2216e−07	2.7773e−08
	$r_2^{N,M}$	2.909	2.991	2.999	3.000	

TABLE 6 Errors and convergence rates using the additive-based hybrid numerical method (15)–(16) for Example 2

5 | NUMERICAL RESULTS

We now present some numerical results to test the theoretical error bound given in Section 4. The values of N and M taken in this section satisfy the assumption $M \leq CN^2/\ln^2 N$ of Section 3.

TABLE 7 Errors and convergence rates using the implicit Euler-based hybrid numerical method[12] for Example 2

		$N = 64$ $M = 4$	$N = 128$ $M = 32$	$N = 256$ $M = 256$	$N = 512$ $M = 2048$	$N = 1024$ $M = 16384$
$\varepsilon_1 = 2^{-17}$	$e_1^{N,M}$	1.2248e−03	3.9822e−04	6.9593e−05	7.2747e−06	7.1829e−07
	$r_1^{N,M}$	1.621	2.516	3.258	3.340	
$\varepsilon_2 = 2^{-12}$	$e_2^{N,M}$	1.3125e−04	2.2490e−05	2.7803e−06	2.8697e−07	2.7392e−08
	$r_2^{N,M}$	2.545	3.016	3.276	3.389	
$\varepsilon_1 = 2^{-19}$	$e_1^{N,M}$	1.2282e−03	3.9873e−04	6.9639e−05	7.2776e−06	7.1859e−07
	$r_1^{N,M}$	1.623	2.517	3.258	3.340	
$\varepsilon_2 = 2^{-14}$	$e_2^{N,M}$	1.3136e−04	2.2499e−05	2.7808e−06	2.8703e−07	2.7397e−08
	$r_2^{N,M}$	2.546	3.016	3.276	3.389	
$\varepsilon_1 = 2^{-23}$	$e_1^{N,M}$	1.2160e−03	3.9685e−04	6.9453e−05	7.2648e−06	7.1735e−07
	$r_1^{N,M}$	1.615	2.514	3.257	3.340	
$\varepsilon_2 = 2^{-17}$	$e_2^{N,M}$	7.2692e−05	1.2263e−05	1.5439e−06	1.5870e−07	1.5017e−08
	$r_2^{N,M}$	2.567	2.990	3.282	3.402	
$\varepsilon_1 = 2^{-10}$	$e_1^{N,M}$	1.4628e−03	1.9576e−04	2.4712e−05	3.1121e−06	3.9142e−07
	$r_1^{N,M}$	2.902	2.986	2.989	2.991	
$\varepsilon_2 = 2^{-5}$	$e_2^{N,M}$	2.8909e−04	4.0189e−05	5.1004e−06	6.3829e−07	7.9902e−08
	$r_2^{N,M}$	2.891	2.978	2.998	2.998	
$\varepsilon_1 = 10^{-15}$	$e_1^{N,M}$	1.2146e−03	3.9579e−04	6.9339e−05	7.2581e−06	7.1693e−07
	$r_1^{N,M}$	1.618	2.513	3.256	3.340	
$\varepsilon_2 = 10^{-6}$	$e_2^{N,M}$	1.0420e−03	7.9452e−05	6.2103e−06	4.7142e−07	3.6096e−08
	$r_2^{N,M}$	3.713	3.677	3.720	3.707	
$\varepsilon_1 = 10^{-8}$	$e_1^{N,M}$	1.2144e−03	3.9575e−04	6.9336e−05	7.2579e−06	7.1691e−07
	$r_1^{N,M}$	1.618	2.513	3.256	3.340	
$\varepsilon_2 = 1$	$e_2^{N,M}$	6.4632e−06	8.4174e−07	1.0560e−07	1.3206e−08	1.6506e−09
	$r_2^{N,M}$	2.941	2.995	2.999	3.000	

TABLE 8 The used CPU times in seconds for Example 1 with $\varepsilon_1 = 2^{-17}, \varepsilon_2 = 2^{-12}$

| Scheme↓	$N = 64$ $M = 4$	$N = 128$ $M = 32$	$N = 256$ $M = 256$	$N = 512$ $M = 2048$	$N = 1024$ $M = 16384$
Additive-based hybrid method	0.093764	0.193491	1.366630	39.766969	1023.793446
Implicit Euler-based hybrid method[12]	0.098180	0.206715	2.747457	68.490184	2662.148163

TABLE 9 The used CPU times in seconds for Example 2 with $\varepsilon_1 = 2^{-17}, \varepsilon_2 = 2^{-12}$

Scheme↓	$N = 64$ $M = 4$	$N = 128$ $M = 32$	$N = 256$ $M = 256$	$N = 512$ $M = 2048$	$N = 1024$ $M = 16384$
Additive-based hybrid method	0.104230	0.348862	3.801144	76.147485	1630.298301
Implicit Euler-based hybrid method[12]	0.109387	0.359951	5.092575	104.299862	3265.088412

Example 1. Consider the following coupled system:

$$
\begin{cases}
(\partial_t + \mathcal{D}\partial_x^2 + \mathcal{B})\vec{u}(x,t) = \vec{f}(x,t), \ (x,t) \in S = \Omega \times (0,1], \\
\vec{u}(x,0) = \vec{0}, \ x \in \bar{\Omega}, \ \vec{u}(0,t) = \vec{0}, \ \vec{u}(1,t) = \vec{0}, \ t \in (0,1],
\end{cases}
$$

where $\mathcal{B} = \begin{pmatrix} 2(x+1)^2 & -(x^3+1) \\ -2\cos(\pi x/4) & 2.2e^{-x+1} \end{pmatrix}, \ \vec{f} = \begin{pmatrix} x\cos(\pi x/2) \\ t \end{pmatrix}.$
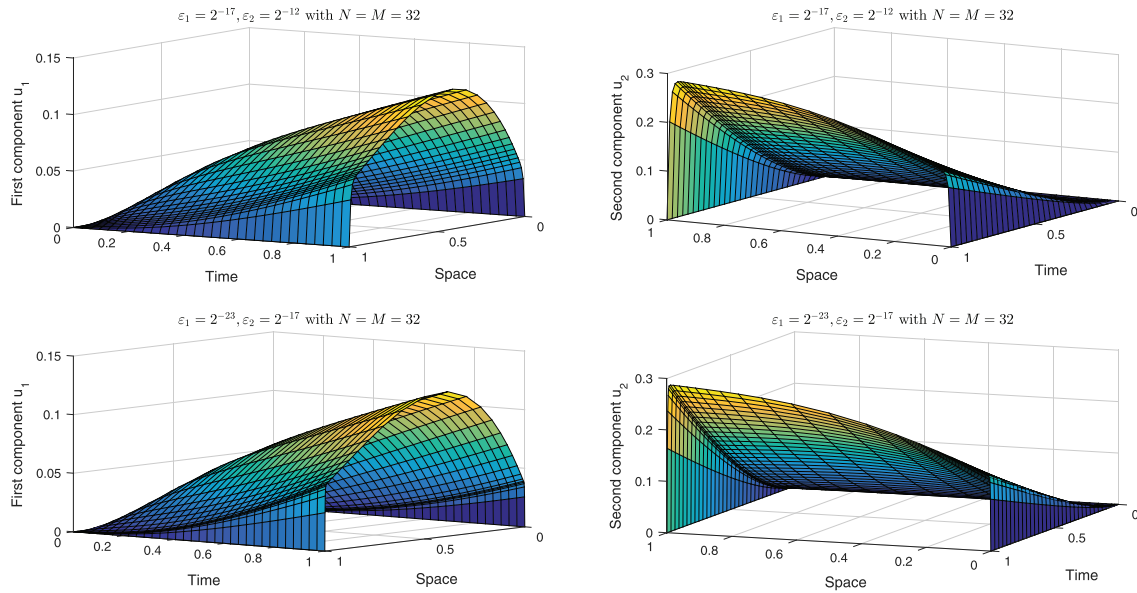
FIGURE 1 Solution plot for Example 1 [Colour figure can be viewed at wileyonlinelibrary.com]

The solution plot is given in Figure 1, which clearly shows the two boundary layers at $x = 0$ and $x = 1$. The exact solution for this test example is not available, so we use a variant of the double mesh principle (see Farrell and Hegarty[35]) to find the maximum pointwise errors. We compute two approximate solutions $\vec{U}^{N,M}$ and $\vec{U}^{2N,2M}$; the approximate solution $\vec{U}^{2N,2M}$ is obtained on a rectangular mesh by bisecting the spatial and time meshes corresponding to $\vec{U}^{N,M}$. We then compute the component-wise maximum pointwise errors in the following way:

$$e_k^{N,M} = \|U_k^{N,M} - U_k^{2N,2M}\|_{\bar{S}^{N,M}}, \ k = 1, 2.$$

We then calculate the rates of convergence by

$$r_k^{N,M} = \frac{\log(e_k^{N,M}/e_k^{2N,sM})}{\log 2}, \ k = 1, 2, \ s = 2, 4, 8.$$

We consider five different pairs of values of the perturbation parameters to present the results:

- $\varepsilon_1 = 2^{-17}$ and $\varepsilon_2 = 2^{-12}$; for this choice, $\tau_2 = 1/4$, $\tau_1 < \tau_2/2$, and $1/2 - 7\tau_2 + 10\tau_1 < 0$ hold.
- $\varepsilon_1 = 2^{-19}$ and $\varepsilon_2 = 2^{-14}$; for this choice, we have $\tau_2 < 1/4$, $\tau_1 < \tau_2/2$ and $1/2 - 7\tau_2 + 10\tau_1 < 0$.
- $\varepsilon_1 = 2^{-23}$ and $\varepsilon_2 = 2^{-17}$; for this choice, the inequality $1/2 - 7\tau_2 + 10\tau_1 \geq 0$ holds with $\tau_2 < 1/4$, $\tau_1 < \tau_2/2$.
- $\varepsilon_1 = 2^{-10}$ and $\varepsilon_2 = 2^{-5}$; for this choice the mesh is uniform.
- $\varepsilon_1 = 10^{-15}$ and $\varepsilon_2 = 10^{-6}$; for this choice, again we have $1/2 - 7\tau_2 + 10\tau_1 \geq 0$ and $\tau_2 < 1/4$, $\tau_1 < \tau_2/2$.
- $\varepsilon_1 = 10^{-8}$ and $\varepsilon_2 = 1$; for this choice, $\tau_2 = 1/4$, $\tau_1 < \tau_2/2$ hold with $1/2 - 7\tau_2 + 10\tau_1 < 0$.

The results corresponding to these pairs illustrate how the hybrid numerical method and the meshes work in practice. For other pairs of perturbation parameters, we observed similar results, and the conclusions mentioned below are the same.

Table 1 lists the componentwise maximum pointwise errors computed using the additive-based hybrid numerical method (15)–(16) for Example 1 with M and N doubled. Here, one can observe first order convergence, which is in line with Theorem 4.1. To observe the influence of the spatial error on the global error of the discretization, we consider doubling the number of mesh points in space, while the number of mesh points in time is multiplied by a factor of eight or four. The corresponding results are displayed in Tables 2 and 3. From these tables, one can observe second and third order convergence, depending on the ratio of time stepping. These numerical results validate the high order convergence in space of the proposed numerical method. Table 4 displays the componentwise maximum pointwise errors and conver-
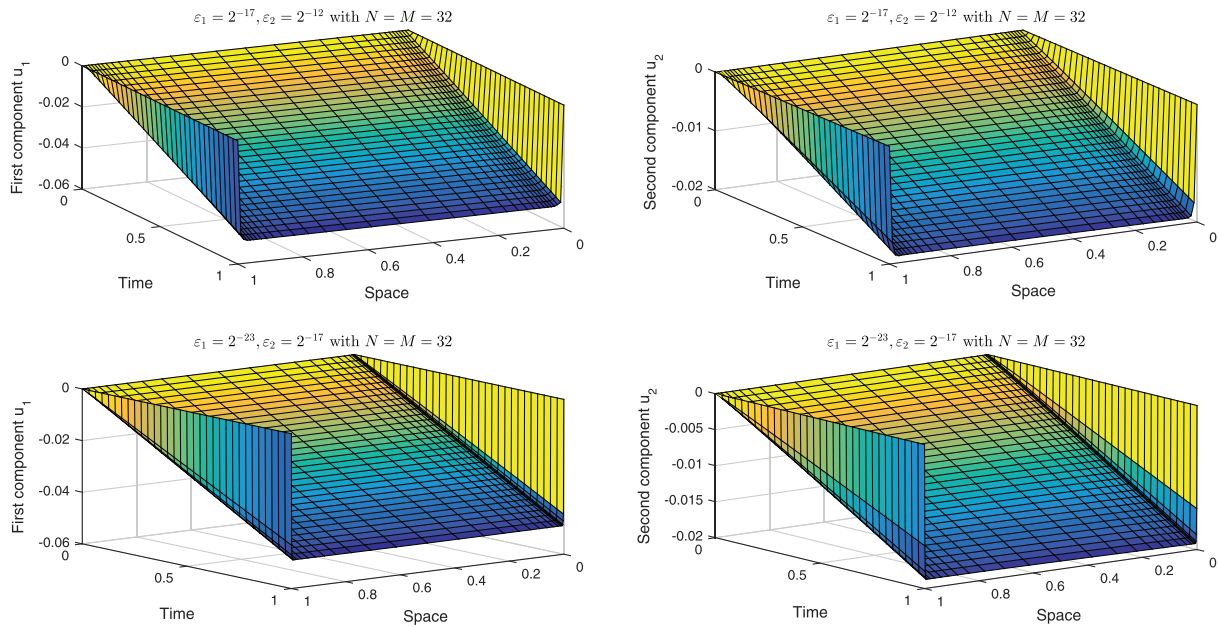
FIGURE 2 Solution plot for Example 2 [Colour figure can be viewed at wileyonlinelibrary.com]

gence rates for Example 1 computed using the implicit Euler-based hybrid numerical method. Tables 3 and 4 show that both methods give similar results providing high order numerical approximations.

Example 2. Consider the following coupled system:

$$
\begin{cases}
(\partial_t + \mathcal{D}\partial_x^2 + \mathcal{B})\vec{u}(x,t) = \vec{f}(x,t), \ (x,t) \in S = \Omega \times (0,1], \\
\vec{u}(x,0) = \vec{0}, \ x \in \Omega, \ \vec{u}(0,t) = \vec{0}, \ \vec{u}(1,t) = \vec{0}, \ t \in (0,1],
\end{cases}
\tag{28}
$$

where $\mathcal{B} = \begin{pmatrix} 2(x+1)^2 & -(x^3+1) \\ -2\cos(\pi x/4) & 2.2e^{-x+1} \end{pmatrix}$, $f = \begin{pmatrix} \cos(\pi x/2) \\ x \end{pmatrix}$,

and the exact solution $\vec{u} = (u_1, u_2)^T$ is given by

$$
u_1(x,t) = (1 - e^{-t/40}) \left(\frac{e^{-2x/\sqrt{\varepsilon_1}} + e^{-2(1-x)/\sqrt{\varepsilon_1}}}{1 + e^{-2/\sqrt{\varepsilon_1}}} + \frac{e^{-3x/\sqrt{\varepsilon_2}} + e^{-3(1-x)/\sqrt{\varepsilon_2}}}{1 + e^{-3/\sqrt{\varepsilon_2}}} - 2 \right),
$$

$$
u_2(x,t) = (1 - e^{-t/50}) \left(\varepsilon_1 \left(\frac{e^{-2x/\sqrt{\varepsilon_1}} + e^{-2(1-x)/\sqrt{\varepsilon_1}}}{1 + e^{-2/\sqrt{\varepsilon_1}}} - 1 \right) + \frac{e^{-3x/\sqrt{\varepsilon_2}} + e^{-3(1-x)/\sqrt{\varepsilon_2}}}{1 + e^{-3/\sqrt{\varepsilon_2}}} - 1 \right).
$$

The solution plot is given in Figure 2, which clearly shows the two boundary layers at $x = 0$ and $x = 1$. This test example is specially designed so that the spatial error is the main contribution to the global error. Since the exact solution is known, we calculate the component-wise maximum pointwise errors using

$$
e_k^{N,M} = \|U_k^{N,M} - u_k\|_{\bar{S}^{N,M}}, \ k = 1, 2,
$$

and the rates of convergence are calculated by

$$
r_k^{N,M} = \frac{\log \left(e_k^{N,M}/e_k^{2N,sM} \right)}{\log 2}, \ k = 1, 2, \ s = 4, 8.
$$

When M is multiplied by a factor of two, the global error of the discretization is associated with time error and we observed first-order convergence results. Here, we display the numerical results when M is multiplied by a factor of four or eight. Tables 5 and 6 display the componentwise maximum pointwise errors and convergence rates for Example 2 computed using the additive-based hybrid numerical method (15)–(16), when M is multiplied by four and eight, respectively. Table 7 displays the componentwise maximum pointwise errors and convergence rates for Example 2 computed using

the implicit Euler-based hybrid numerical method. From these tables, it is clear that the obtained results are similar for both methods having high order accuracy. Further, to observe the contribution of the spatial error to the global error, it is not required to multiply M by eight (as required in the first test example).

In order to compare the efficiency of the additive-based hybrid numerical method with the implicit Euler-based hybrid numerical method, Tables 8 and 9 display the used CPU times for different values of N and M, taking $\varepsilon_1 = 2^{-17}$, $\varepsilon_2 = 2^{-12}$ for Examples 1 and 2, respectively. In these tables the used CPU time in seconds is reported by averaging the used CPU time from several executions in MATLAB R2015b on a 64 bit Windows 10 machine, with Intel(R) Core(TM) i5-9400 processor running at 2.9GHz and 8.00Gb RAM. These results show the good performance of proposed method, being more efficient than the standard implicit Euler-based hybrid method.

CONFLICT OF INTEREST

This work does not have any conflicts of interest.

ORCID

Sunil Kumar https://orcid.org/0000-0001-9991-1012
Higinio Ramos https://orcid.org/0000-0003-2791-6230

REFERENCES

1. Kadalbajoo MK, Gupta V. A brief survey on numerical methods for solving singularly perturbed problems. *Appl Math Comput*. 2010;217(8):3641-3716.
2. Kumar K, Chakravarthy PP, Vigo-Aguiar J. Numerical solution of time-fractional singularly perturbed convection-diffusion problems with a delay in time. *Math Methods Appl Sci*. 2021;44(4):3080-3097.
3. Gharibi Z, Dehghan M. Convergence analysis of weak Galerkin flux-based mixed finite element method for solving singularly perturbed convection-diffusion-reaction problem. *Appl Numer Math*. 2021;163:303-316.
4. Farrell P, Hegarty A, Miller JJH, O'Riordan E, Shishkin GI. *Robust Computational Techniques for Boundary Layers*. CRC Press; 2000.
5. Munyakazi JB. A uniformly convergent nonstandard finite difference scheme for a system of convection–diffusion equations. *Comput Appl Math*. 2015;34:1153-1165.
6. Das P, Rana S, Vigo-Aguiar J. Higher order accurate approximations on equidistributed meshes for boundary layer originated mixed type reaction diffusion systems with multiple scale nature. *Appl Numerical Math*. 2020;148:79-97.
7. Das P, Vigo-Aguiar J. Parameter uniform optimal order numerical approximation of a class of singularly perturbed system of reaction diffusion problems involving a small perturbation parameter. *J Comput Appl Math*. 2019;354:533-544.
8. Shakti D, Mohapatra J, Das P, Vigo-Aguiar J. A moving mesh refinement based optimal accurate uniformly convergent computational method for a parabolic system of boundary layer originated reaction-diffusion problems with arbitrary small diffusion terms. *J Comput Appl Math*. 2020:113167.
9. Gupta V, Kadalbajoo MK. A layer adaptive B-spline collocation method for singularly perturbed one-dimensional parabolic problem with a boundary turning point. *Numer Methods Partial Differ Equ*. 2011;27(5):1143-1164.
10. Lubuma JM-S, Patidar KC. Solving singularly perturbed advection-reaction equations via non-standard finite difference methods. *Math Methods Appl Sci*. 2007;30(14):1627-1637.
11. Kumar S, Kumar M. An analysis of overlapping domain decomposition methods for singularly perturbed reaction-diffusion problems. *J Comput Appl Math*. 2015;281:250-262.
12. Clavero C, Gracia JL. An improved uniformly convergent schemes in space for 1D parabolic reaction-diffusion systems. *Appl Math Comput*. 2014;243:57-73.
13. Kumar S, Kumar S, Sumit. High-order convergent methods for singularly perturbed quasilinear problems with integral boundary conditions. *Math Methods Appl Sci*. 2020:1-14. doi:10.1002/mma.6854
14. Barenblatt G, Zheltov I, Kochina I. Basic concepts in the theory of seepage of homogeneous liquids in fissured rocks [strata]. *J Appl Math Mech*. 1960;24(5):1286-1303.
15. Cowin SC. Bone poroelasticity. *J Biomech*. 1999;32:217-238.
16. Franklin V, Paramasivam M, Miller JJH, Valarmathi S. Second order parameter-uniform convergence for a finite difference method for a singularly perturbed linear parabolic system. *Int J Numer Anal Model*. 2013;10:178-202.

17. Shishkin GI. Mesh approximation of singularly perturbed boundary-value problems for systems of elliptic and parabolic equations. *Comput Math Math Phys*. 1995;4(35):429-446.

18. Kaltenbacher B, Rundell W. The inverse problem of reconstructing reaction–diffusion systems. *Inverse Probl*. 2020;36(6):065011.

19. Lukyanenko D, Grigorev VB, Volkov VT, Shishlenin MA. Solving of the coefficient inverse problem for a nonlinear singularly perturbed two-dimensional reaction–diffusion equation with the location of moving front data. *Comput Math Appl*. 2019;77(5):1245-1254.

20. Levashova N, Gorbachev A, Argun R, Lukyanenko D. The problem of the non-uniqueness of the solution to the inverse problem of recovering the symmetric states of a bistable medium with data on the position of an autowave front. *Symmetry*. 2021;13(5):860.

21. Averós JC, Llorens JP, Uribe-Kaffure R. Numerical simulation of non-linear models of reaction-diffusion for a DGT sensor. *Algorithms*. 2020;13(4):98.

22. Munyakazi JB, Patidar KC. A new fitted operator finite difference method to solve systems of evolutionary reaction-diffusion equations. *Quaest Math*. 2015;38:121-138.

23. Kumar S, Singh J, Kumar M. A robust domain decomposition method for singularly perturbed parabolic reaction-diffusion systems. *J Math Chem*. 2019;57(5):1557-1578.

24. Kumar S, Rao SCS. A robust overlapping Schwarz domain decomposition algorithm for time-dependent singularly perturbed reaction-diffusion problems. *J Comput Appl Math*. 2014;261:127-138.

25. Aakansha J, Singh S. Kumar, Additive schemes based domain decomposition algorithm for solving singularly perturbed parabolic reaction-diffusion systems. *Comput Appl Math*. 2021;40:82.

26. Vabishchevich PN. Additive schemes for certain operator-differential equations. *Comput Math Math Phys*. 2010;50:2033-2043.

27. Clavero C, Gracia JL. Uniformly convergent additive finite difference schemes for singularly perturbed parabolic reaction-diffusion systems. *Comput Math Appl*. 2014;67:655-670.

28. Samarskii AA, Vabishchevich PN. *Computational Heat Transfer: The Finite Difference Methodology*. John Wiley & Sons Inc.; 1995.

29. Aakansha, Kumar S, Singh J. An efficient numerical method for coupled systems of singularly perturbed parabolic delay problems. *Comput Appl Math*. 2022;41:29.

30. Clavero C, Gracia J, Lisbona F. An almost third order finite difference scheme for singularly perturbed reaction-diffusion systems. *J Comput Appl Math*. 2010;234(8):2501-2515.

31. Bujanda B, Clavero C, Gracia JL, Jorge JC. A high order uniformly convergent alternating direction scheme for time dependent reaction-diffusion singularly perturbed problems. *Numer Math*. 2007;107:1-25.

32. Clavero C, Gracia J, Jorge J. Second order numerical methods for one dimensional parabolic singularly perturbed problems with regular layers. *Numer Methods Partial Differ Equ*. 2005;21:149-169.

33. Kumar M, Rao SCS. High order parameter-robust numerical method for time dependent singularly perturbed reaction-diffusion problems. *Computing*. 2010;90:15-38.

34. Clavero C, Gracia J. A high order HODIE finite difference scheme for 1D parabolic singularly perturbed reaction-diffusion problems. *Appl Math Comput*. 2012;218(9):5067-5080.

35. Farrell PA, Hegarty A. On the determination of the order of uniform convergence. In: Proceedings of IMACS, Vol. 91; 1991:501-502.