

Richardson Extrapolation for Singularly Perturbed Fredholm Integro Differential Equations

P. Antony Prince

Department of Mathematics,
Amrita School of Physical Science, Coimbatore, Amrita Vishwa Vidyapeetham, Tamil Nadu, India.
E-mail: p_antonyprince@cb.students.amrita.edu

L. Govindarao

Department of Mathematics,
Amrita School of Physical Science, Coimbatore, Amrita Vishwa Vidyapeetham, Tamil Nadu India.
E-mail: l_govindarao@cb.amrita.edu

Sekar Elango

Department of Mathematics,
Amrita School of Physical Science, Coimbatore, Amrita Vishwa Vidyapeetham, Tamil Nadu India.
Corresponding author: esekar@cb.amrita.edu

(Received on May 21, 2025; Revised on May 23, 2025 & June 20, 2025; Accepted on July 4, 2025)

Abstract

This study numerically derived the higher order convergence for a class of singularly perturbed Fredholm integro differential equations with reaction diffusion and convection diffusion type problems. A non-standard finite difference approach is used to approximate the derivatives. The trapezoidal rule determines the integral term. The suggested numerical technique achieves a uniform convergence rate independently of the perturbation parameter. Implementing the Richardson extrapolation technique achieves a fourth order convergence rate for reaction diffusion type problems and a second order convergence rate for convection diffusion type problems. Specific numerical examples are provided to corroborate in practice the effectiveness of the theoretical findings.

Keywords- Singular perturbation, Extrapolation, Fitted operator, Fredholm integral, Boundary layer.

1. Introduction

The diffusion equation is a partial differential equation (PDE) that describes the behavior of a substance (e.g. heat, particles, or chemicals) as it spreads out over space and time due to random motion or concentration gradients. The standard form of the diffusion equation in one spatial dimension is (Crank, 1975).

$$u_t = Du_{xx},$$

where, $u = u(x, t)$ is the concentration or temperature at a position x and time t , D is the positive diffusion coefficient. The diffusion equation is split into the reaction-diffusion equation and the convection-diffusion equation.

The general form of the one-dimensional reaction-diffusion steady-state problems is,

$$D \frac{d^2 u(x)}{dx^2} + R(u) = 0,$$

where, $R(u)$ is the reaction term.

The general form of the one-dimensional convection-diffusion steady-state problems is

$$D \frac{d^2 u(x)}{dx^2} - v \frac{du(x)}{dx} = 0,$$

where, v is the convection velocity. These types of diffusion equations have physical applications like chemical kinetics, population dynamics, biological pattern formation, fluid flow, and heat transfer.

A differential equation where a small parameter multiplies the higher derivative term is typically called a singularly perturbed differential equations (SPDEs). The solutions to these equations have a very thin layer, referred to as either an interior layer or a boundary layer, depending on the location in the domain. SPDEs can be solved using various numerical methods, especially fitted mesh methods and fitted operator methods. Both approaches dealt with SPDEs in one and multidimensional problems (Govindarao and Mohapatra, 2019; Sekar, 2023), also Das and Natesan (2017) solved parabolic singularly perturbed delay PDEs using a Shishkin mesh, addressing challenges associated with small perturbation and delay effects. Udupa et al. (2022) explored blood flow through a stenosed artery under body acceleration, employing a singular perturbation approach combined with Shishkin mesh discretization to simulate realistic physiological responses. Das and Natesan (2018) extended their earlier work by introducing a fractional-step higher-order method for 2D convection-diffusion parabolic problems, achieving improved accuracy on non-uniform meshes. Izadi and Yuzbasi (2022) proposed a hybrid numerical technique for parabolic singularly perturbed convection-diffusion problems, offering uniform convergence and improved treatment of boundary layers. Ansari et al. (2024) numerically solved the two small parameters singularly perturbed parabolic convection–diffusion–reaction problems and demonstrated significant performance improvements. Izadi and Zeidan (2022) developed a hybrid scheme for nonlinear diffusion equations, offering reliable convergence and accurate results even for stiff and nonlinear systems. Govindarao and Sekar (2023) numerically solved the RLC closed series circuit with small inductance values. Mohapatra et al. (2025) developed a numerical scheme for solving 2D time-dependent SPDE with improved accuracy and stability. Integro-differential equations (IDEs) have significance in various domains, including engineering, physics, and biology (Rahman, 2007). IDEs are categorized into two types based on their elements. Fredholm integro-differential equations (FIDEs), which contain integral components with a fixed range and Volterra integro-differential equations (VIDEs), which feature integral components constrained by specific variables. In the literature, there are many integro-differential models, here present a mathematical model of chemical reaction-diffusion processes that include the effects of a catalyst (Chadam and Yin, 1994).

$$u_t - \Delta u = a \int_{\Omega} H(u(\cdot, y)) dy, \quad t > 0, x \in \Omega \subset \mathbb{R}^d,$$

with homogeneous Dirichlet or Neumann boundary conditions on $\partial\Omega$. This type of model gave the motivation to construct singularly perturbed Fredholm integro-differential equations (SPFIDEs).

Consider a class of linear second-order SPFIDEs of the form

$$\begin{cases} L := L_1 + L_2 = f(x), & x \in (0,1) = \Omega, \\ z(0) = A, & z(1) = B, \end{cases} \quad (1)$$

where, $L_1 = -\epsilon z''(x) + a(x)z'(x) + b(x)z(x)$, $L_2 = \lambda \int_0^1 K(x,s)z(s) ds$, $0 < \epsilon \ll 1$. The functions $a(x) \geq \alpha > 0$, $b(x) \geq \beta > 0$ and $f(x)$ are differentiable functions, $K(x,s)$ is a kernel function, A, B are constants, and λ is a given parameter.

Equation (1) is split into two types of problems. If the coefficient $a(x) = 0$, it denotes the Reaction-diffusion type problems and if $a(x) \neq 0$, it is known as the Convection-diffusion type problems. In literature, Lange and Smith (1993) derived the existence and uniqueness of SPFIDEs. Amiraliyev et al. (2020) solved SPFIDEs with Shishkin mesh. Durmaz and Amiraliyev (2021) tackled the second-order reaction-diffusion SPFIDEs utilizing a fitted mesh and showed second-order convergence. In articles (Sekar et al., 2024; Sekar et al., 2025) successfully solved the second-order reaction-diffusion and convection-diffusion SPFIDEs, respectively, applying a standard difference method for the derivative part and integral used by the trapezoidal rule of non-uniform meshes, also applying a post-processing method to increase the convergence rate. Govindarao et al. (2024) investigated the reaction-diffusion SPFIDEs with non-local boundaries and they succeeded in attaining a second-order convergence rate and also applying the extrapolation technique, a fourth-order convergence rate was obtained. Elango et al. (2025) solved the system of SPFIDE with a non-uniform mesh.

Fitted mesh finite difference methods need experience with the layer's position and size. Fulfilling this need might seem challenging at times. However, the fitted operator finite difference (FOFD) methods do not impose these requirements. Inspired by the previously discussed research, implementing the FOFD approach and the composite trapezoidal rule are effectively addressed in Equation (1).

The objective of this article is to achieve a fourth and second-order convergence rate with Richardson extrapolation in reaction-diffusion and convection-diffusion SPFIDEs, respectively. Initially, the derivative part is handled using a non-standard finite difference (NSFD) scheme, while the trapezoidal rule is applied to an integral section of the uniform mesh. Afterwards, a post-processing technique implementing the convergence rate increases the second order to the fourth order in reaction-diffusion problems and order one to the second order in convection-diffusion problems. As a result, the global rate of convergence appears computationally.

This article is organized like this, Section 2 shows the numerical discretization of reaction-diffusion and convection-diffusion SPFIDEs, and shows the solution bounds Section 3 explains the post-process method and improves the numerical solutions accuracy and the rate of convergence, Section 4 shows error estimates with the extrapolation technique, and Section 5 examines computational simulations for reaction-diffusion and convection-diffusion SPFIDEs, Section 6 shows the results and discussion of the comparison with and without the post-process method.

Notations: In this study, C stands for a generic positive constant that is independent of the mesh parameter (Δh) and perturbation parameter (ϵ) . The space of real-valued functions that are continuously differentiable n times on $[0,1]$ is represented by the letter $C^n([0,1])$, $\Re = \max_{x \in [0,1]} \int_0^1 |K(x,s)| ds$ and $z_i = z(x_i)$ indicate an approximation by the Z_i .

2. Numerical Discretization

On $[0,1]$, the uniform mesh step (Δh) is used to discretize the interval. Here $(\Delta h) = \frac{1}{N}$ such that $x_i = i(\Delta h)$, where N is the total number of sub-intervals. For each mesh point, Equation (1) becomes

$$L^{(\Delta h)} := L_1^{(\Delta h)} + L_2^{(\Delta h)} = f(x_i), \quad i = 0, 1, 2, \dots, N \quad (2)$$

where,

$$L_1^{(\Delta h)} = -\epsilon z''(x_i) + a(x_i)z'(x_i) + b(x_i)z(x_i),$$

$$L_2^{(\Delta h)} = \lambda \int_0^1 K(x_i, s)z(s) ds.$$

Apply the NSFD method for the differential part (Lubuma and Patidar, 2007) in Equation (2), then,

$$L_1^{(\Delta h)} = \begin{cases} -\epsilon \frac{z_{i-1} - 2z_i + z_{i+1}}{\psi_i^2} + \overline{b}(x_i)z(x_i), & \text{if } a(x) = 0, \\ -\epsilon \frac{z_{i-1} - 2z_i + z_{i+1}}{\phi_i^2} + \overline{a}(x_i) \frac{z_i - z_{i-1}}{\Delta h} + \overline{b}(x_i)z(x_i), & \text{if } a(x) \neq 0, \end{cases}$$

$$\text{where, } \overline{\psi}_i^2 = \frac{4}{\rho_i^2} \left(\frac{\overline{\rho}_i(\Delta h)}{2} \right), \overline{\rho}_i = \left(\frac{\overline{a}_i}{\epsilon} \right)^{\frac{1}{2}}, \overline{\phi}_i^2 = \frac{\epsilon(\Delta h)}{\overline{a}_i} \left(\left(\frac{\overline{a}_i(\Delta h)}{\epsilon} \right) - 1 \right), \overline{a}_i = \frac{a_i + a_{i+1}}{2}, \overline{b}_i = \frac{b_{i-1} + b_i + b_{i+1}}{3}.$$

Apply the composite trapezoidal rule to determine the integral component (Kress, 1998) in Equation (2), then,

$$L_2^{(\Delta h)} = \lambda \sum_{j=0}^N \theta_j(\Delta h) K(x_i, s_j) z(s_j),$$

$$\text{where, } \theta_j = \begin{cases} \frac{1}{2} & \text{for } j = 0, N, \\ 1 & \text{for } j = 1, 2, 3, \dots, N-1. \end{cases}$$

Lemma 2.1 Let $z(x_i)$ and Z_i be the functions of L_1 and $L_1^{(\Delta h)}$, then the bounds are

- (i) $\max_{0 \leq i \leq N} |z(x_i) - Z_i| \leq C(\Delta h)^2$ if $a(x) = 0$,
- (ii) $\max_{0 \leq i \leq N} |z(x_i) - Z_i| \leq C(\Delta h)$ if $a(x) \neq 0$.

Proof. The proof of (i) can be found in (Munyakazi and Patidar, 2008) and (ii) can be found in (Lubuma and Patidar, 2007).

Lemma 2.2 Let $z(x_i)$ and Z_i be the functions of L_2 and $L_2^{(\Delta h)}$, then the bound is

$$\text{i.e., } \max_{0 \leq x \leq 1} |z(x_i) - Z_i| \leq C(\Delta h)^2.$$

Proof. This Lemma is proved in two cases:

Case (i): Reaction-diffusion ($a(x) = 0$).

Let $|\lambda| < \frac{\beta}{\mathfrak{R}}$, then

$$\begin{aligned} \left| L_2^{(\Delta h)}(z(x_i) - Z_i) \right| &= \left| \lambda \int_0^1 K(x_i, s)z(s) ds - \lambda \sum_{j=0}^N \theta_j(\Delta h) K(x_i, s_j) z(s_j) \right|, \\ &\leq \frac{1}{12} |\lambda| (\Delta h)^2 \max_{0 \leq x_i, s \leq 1} \left| \frac{\partial^2}{\partial s^2} [K(x_i, s)z(s)] \right|, \\ &\leq C(\Delta h)^2. \end{aligned}$$

Then it follows that

$$\max_{0 \leq x \leq 1} |z(x_i) - Z_i| \leq C(\Delta h)^2.$$

Case (ii): Convection-diffusion ($a(x) \neq 0$).

Let $|\lambda| < \frac{\alpha}{\Re}$, then

$$\begin{aligned} \left| L_2^{(\Delta h)}(z(x_i) - Z_i) \right| &= \left| \lambda \int_0^1 K(x_i, s) z(s) ds - \lambda \sum_{j=0}^N \theta_j(\Delta h) K(x_i, s_j) z(s_j) \right|, \\ &\leq \frac{1}{12} |\lambda| (\Delta h)^2 \max_{0 \leq x_i, s \leq 1} \left| \frac{\partial^2}{\partial s^2} [K(x_i, s) z(s)] \right|, \\ &\leq C(\Delta h)^2. \end{aligned}$$

Then it follows that

$$\max_{0 \leq x \leq 1} |z(x_i) - Z_i| \leq C(\Delta h)^2.$$

Theorem 2.3 Let $z(x_i)$ and Z_i be the solutions of Equation (1) and Equation (2), then

$$\begin{aligned} \sup_{0 < \epsilon \leq 1} \max_{0 \leq i \leq N} |z(x_i) - Z_i| &\leq C(\Delta h)^2 \quad \text{if } a(x) = 0, \\ \sup_{0 < \epsilon \leq 1} \max_{0 \leq i \leq N} |z(x_i) - Z_i| &\leq C(\Delta h) \quad \text{if } a(x) \neq 0. \end{aligned}$$

Proof. By applying Lemmas 2.1 and 2.2, it follows.

3. Post-process Method

The Richardson extrapolation improves the numerical solution accuracy and the rate of convergence (Zlatev et al., 2017). Let the meshes $\bar{\Omega}^N = \{x_i\}, x_0 = 0, x_N = 1, \Delta h = (x_i - x_{i-1})$ and $\bar{\Omega}^{2N} = \{\bar{x}_i\}, \bar{x}_0 = 0, \bar{x}_N = 1, (\bar{x}_i - \bar{x}_{i-1}) = \bar{\Delta h} = \Delta h/2$. The discrete Equation (2) is initially solved in the NSFD method and composite trapezoidal rule with uniform mesh $\bar{\Omega}^N, \bar{\Omega}^{2N}$.

Let Z_i and $\bar{Z}_i$ be the solutions of $L^{(\Delta h)}, \bar{L}^{(\Delta h)}$ with the meshes $\bar{\Omega}^N$ and $\bar{\Omega}^{2N}$, respectively.

Now take reaction-diffusion type ($a(x) = 0$) problems then Theorem 2.3 gives

$$\begin{aligned} |z_i - Z_i| &\leq C(\Delta h)^2, \quad i = 1, \dots, N-1, \\ |z_i - \bar{Z}_i| &\leq C(\Delta h/2)^2, \quad i = 1, \dots, 2N-1. \end{aligned}$$

Therefore, $z_i - Z_i = C(\Delta h)^2 + R_N(x_i), \forall x_i \in \bar{\Omega}^N$ and $z_i - \bar{Z}_i = C(\Delta h/2)^2 + R_{2N}(\bar{x}_i), \forall x_i \in \bar{\Omega}^{2N}$, where the remainders, $R_N(x_i)$ and $R_{2N}(\bar{x}_i)$ are in $\mathcal{O}((\Delta h)^2)$. Removing $\mathcal{O}((\Delta h)^2)$ is necessary, hence using the following expression to get higher order $(z_i - Z_i) - 4(z_i - \bar{Z}_i) = R_N(x_i) - 4R_{2N}(\bar{x}_i), \forall x_i \in \bar{\Omega}^N$.

Therefore, the post-process method formula is,

$$Z_i^{exp} := \frac{4\bar{Z}_i - Z_i}{3}, \quad i = 1, \dots, N-1.$$

Similarly, convection-diffusion type ($a(x) \neq 0$) implies that

$$z_i - Z_i = C(\Delta h) + R_N(x_i), \forall x_i \in \bar{\Omega}^N \text{ and } z_i - \bar{Z}_i = C(\Delta h/2) + R_{2N}(\bar{x}_i), \forall x_i \in \bar{\Omega}^{2N},$$

where, the remainders, $R_N(x_i)$ and $R_{2N}(\bar{x}_i)$ are in $\mathcal{O}((\Delta h))$. Hence, $z_i - (2\bar{Z}_i - Z_i) = \mathcal{O}((\Delta h)), \forall x_i \in \bar{\Omega}^N$, in the post-process method, using this formula,

$$Z_i^{exp} := 2\bar{Z}_i - Z_i, \quad i = 1, \dots, N-1.$$

4. Error Estimate after Extrapolation

Here, $z(x_i)$ is the continuous solution and $Z^{exp}(x_i)$ is the Richardson extrapolation method solution for each mesh point of the problem (1). The solution is split into the differential part and an integral part and then the error estimation follows like

$$|L^{(\Delta h)}(z - Z^{exp})(x_i)| = |L_1^{(\Delta h)}(z - Z^{exp})(x_i) + L_2^{(\Delta h)}(z - Z^{exp})(x_i)|.$$

Now the error bound with extrapolation follows that

Theorem 4.1 Let z_i be the solution of L and $Z_i^{exp}, i = 0, 1, \dots, N$ be the Richardson extrapolation method solution of $L^{(\Delta h)}$ with $z_0 = Z_0^{exp} = A$ and $z_N = Z_N^{exp} = B$, then

- (i) $\sup_{0 < \epsilon \leq 1} \max_{0 \leq i \leq N} |z_i - Z_i^{exp}| \leq C(\Delta h)^4, \quad \text{if } a(x) = 0,$
- (ii) $\sup_{0 < \epsilon \leq 1} \max_{0 \leq i \leq N} |z_i - Z_i^{exp}| \leq C(\Delta h)^2, \quad \text{if } a(x) \neq 0.$

Proof. The error bound appears in the form.

$$|L^{(\Delta h)}(z - Z^{exp})(x_i)| = |L_1^{(\Delta h)}(z - Z^{exp})(x_i) + L_2^{(\Delta h)}(z - Z^{exp})(x_i)|.$$

If $a(x) = 0$, then the differential part L_1 operator bound is $|L_1^{(\Delta h)}(z - Z^{exp})(x_i)| \leq C(\Delta h)^4$.

The proof of this bound is provided in (Munyakazi and Patidar, 2008).

If $a(x) \neq 0$, then the differential part L_1 operator bound is $|L_1^{(\Delta h)}(z - Z^{exp})(x_i)| \leq C(\Delta h)^2$.

The proof of this bound is provided in (Lubuma and Patidar, 2007).

Now, to prove the error bound of the operator L_2 . By Theorem 2.3 gives

$$\begin{aligned} L_2^{(\Delta h)}(z - Z)(x_i) &\leq C(\Delta h)^2, \quad \text{for } x_i \in \bar{\Omega}^N, \\ L_2^{(\Delta h)}(z - \bar{Z})(x_i) &\leq C(\bar{\Delta h})^2, \quad \text{for } x_i \in \bar{\Omega}^{2N}. \end{aligned}$$

From the extrapolation formula of the reaction-diffusion case

$$\left(L_2^{(\Delta h)}(z - Z^{exp})(x_i) \right) = \frac{4}{3} \left(L_2^{(\bar{\Delta h})}(z - \bar{Z})(x_i) \right) - \frac{1}{3} \left(L_2^{(\Delta h)}(z - Z)(x_i) \right).$$

Then, after simplification, it implies that

$$|L_2^{(\Delta h)}(z - Z^{exp})| \leq C(\Delta h)^4.$$

Similarly, the extrapolation formula for the convection-diffusion case

$$\left(L_2^{(\Delta h)}(z - Z^{exp})(x_i)\right) = 2 \left(L_2^{(\overline{\Delta h})}(z - \bar{Z})(x_i)\right) - \left(L_2^{(\Delta h)}(z - Z)(x_i)\right).$$

Then, it implies

$$|L_2^{(\Delta h)}(z - Z^{exp})| \leq C(\Delta h)^2.$$

Combine L_1 bound and L_2 bound

$$\max_{1 \leq i \leq N} |L^{(\Delta h)}(z - Z^{exp})(x_i)| \leq \begin{cases} C(\Delta h)^4, & (a(x) = 0), \\ C(\Delta h)^2, & (a(x) \neq 0). \end{cases} \quad (3)$$

Now apply the uniform stability (Munyakazi and Patidar, 2008) in Equation (3), then

$$\max_{1 \leq i \leq N} |z_i - Z_i^{exp}| \leq \begin{cases} C(\Delta h)^4, & \text{for } a(x) = 0, \\ C(\Delta h)^2, & \text{for } a(x) \neq 0. \end{cases}$$

Then follows

$$\sup_{0 < \epsilon \leq 1} \max_{0 < i \leq N} |z_i - Z_i^{exp}| \leq \begin{cases} C(\Delta h)^4, & \text{for } a(x) = 0, \\ C(\Delta h)^2, & \text{for } a(x) \neq 0. \end{cases}$$

5. Computational Simulations

The developed method, according to theoretical analysis, exhibits a uniform convergence rate with the perturbation parameter ϵ . Numerical computations were performed to evaluate the efficiency of the current methodology, utilizing the given instance.

Example 5.1 Consider the example in the format of Equation (1)

$$\begin{aligned} a(x) &\equiv 0, b(x) = (1 + x(1 - x)), \lambda = 1, K(x, s) = x, \\ f(x) &= e^{-\frac{1+x}{\sqrt{\epsilon}}} \left[e^{\frac{1}{\sqrt{\epsilon}}} \left(-(x-1)^2 x + 2\sqrt{\epsilon} \right) + e^{\frac{2x}{\sqrt{\epsilon}}} \left((x-1)x^2 + 2\sqrt{\epsilon} \right) \right. \\ &\quad \left. + e^{\frac{x}{\sqrt{\epsilon}}} \left(-2\epsilon + e^{\frac{1}{\sqrt{\epsilon}}} (2 + x - x^2 - 2\sqrt{\epsilon} + 2\epsilon) \right) \right], \\ z(0) &= 0, z(1) = 0. \end{aligned}$$

Example 5.2 Consider the example in the format of Equation (1)

$$\begin{aligned} a(x) &= 1 - \frac{x^2}{2}, b(x) = 0, \lambda = \frac{1}{4}, K(x, s) = x, f(x) = 1, \\ z(0) &= 0, z(1) = 0. \end{aligned}$$

Example 5.3 Consider the example in the format of Equation (1)

$$\begin{aligned} a(x) &= \left(\frac{x+1}{2}\right), b(x) = 2(1+x), \lambda = \frac{1}{2}, K(x, s) = x + s, f(x) = \left(1 - \frac{x}{2}\right)(3-x), \\ z(0) &= 0, z(1) = 0. \end{aligned}$$

Exact solution of Example 5.1 is $z(x) = 1 + (x - 1)e^{-\frac{x}{\sqrt{\epsilon}}} - xe^{-\frac{(1-x)}{\sqrt{\epsilon}}}$. Before extrapolation, the error estimate is $E_{\epsilon}^N = \max_i |z(x_i) - Z_i|$, where $z(x_i)$ is the exact solution and Z_i is an approximation solution, and Example 5.2 and Example 5.3 do not possess exact solutions. Consequently, an error estimate is followed by a double mesh error analysis. The error obtained by

$$E_{\epsilon}^N = \max_i |Z_i^{\epsilon, N} - \tilde{Z}_i^{\epsilon, 2N}|,$$

where, $\tilde{Z}_i^{\epsilon, 2N}$ is the Computational solution of the associated approach on the $2N$ points.

After extrapolation, the error estimation of Example 5.1 is

$$E_{\epsilon}^N = \max_i |z(x_i) - Z_i^{exp}|,$$

where, $z(x_i)$ is an exact solution and Z_i^{exp} is an approximation solution after extrapolation is applied, Example 5.2 and Example 5.3 error estimation is

$$E_{\epsilon}^N = \max_i |Z_i^{exp, N} - \tilde{Z}_i^{exp, 2N}|.$$

The convergence rate is defined by $R_{\epsilon}^N = \log_2 \left(\frac{E_{\epsilon}^N}{E_{\epsilon}^{2N}} \right)$. The computed ϵ –uniform pointwise maximum error $E_{\epsilon}^N = \max_{\epsilon} E_{\epsilon}^N$ and also define the ϵ –uniform rate of convergence as $R^N = \log_2 \left(\frac{E^N}{E^{2N}} \right)$.

6. Result Discussion

- For various values of ϵ , a numerical and exact solution of Example 5.1 is plotted in **Figure 1** this figure shows that if ϵ is very small, the layer attains the boundary of $x = 0$ and $x = 1$.
- **Figure 2** and **Figure 3** shows computational solutions of various ϵ of Example 5.2 and Example 5.3. These figures indicate that when ϵ diminishes, a layer emerges around $x = 1$.
- **Figure 4** illustrates the error plot of Example 5.1 without extrapolation, while **Figure 5** displays the error plot with extrapolation. These plots illustrate the highest pointwise error at the boundary layers of $x = 0$ and $x = 1$.
- Similarly, **Figure 6** displays the error plot of Example 5.2 without extrapolation, while **Figure 7** exhibits the error plot with extrapolation. These pictures illustrate the maximum pointwise error at the layer boundary of $x = 1$.
- **Table 1**, **Table 3**, and **Table 5** display the maximum pointwise error and convergence order for Example 5.1, Example 5.2, and Example 5.3 without extrapolation, with **Table 1** indicating a maximum convergence of second-order convergence for each ϵ value. **Table 3** and **Table 5** demonstrate first-order convergence.
- Similarly, **Table 2**, **Table 4**, and **Table 6** shows the maximum pointwise error and convergence order for Example 5.1, Example 5.2, and Example 5.3 with post-processing, with **Table 2** presenting the fourth-order convergence, **Table 4** and **Table 6** showing second-order convergence for each ϵ value.
- All the tables indicate that as the values of N increase, the error decreases and the rate of convergence approaches the theoretically predicted value.
- **Table 7**, **Table 8** and **Table 9** present a comparison of the ϵ –uniform maximum errors and convergence rates before and after extrapolation.

- **Figure 8** and **Figure 10** plot the numerical convergence rates on a log-log scale, providing a graphical representation of Example 5.1 and Example 5.2 before extrapolation, while **Figure 9** and **Figure 11** show the corresponding plots after extrapolation.
- A log-log plot is a graphical representation where both the horizontal and vertical axes are on a logarithmic scale. This plot is particularly useful for identifying power-law relationships of the rate of convergence, as straight lines on a log-log plot indicate a consistent rate of error reduction with mesh refinement.

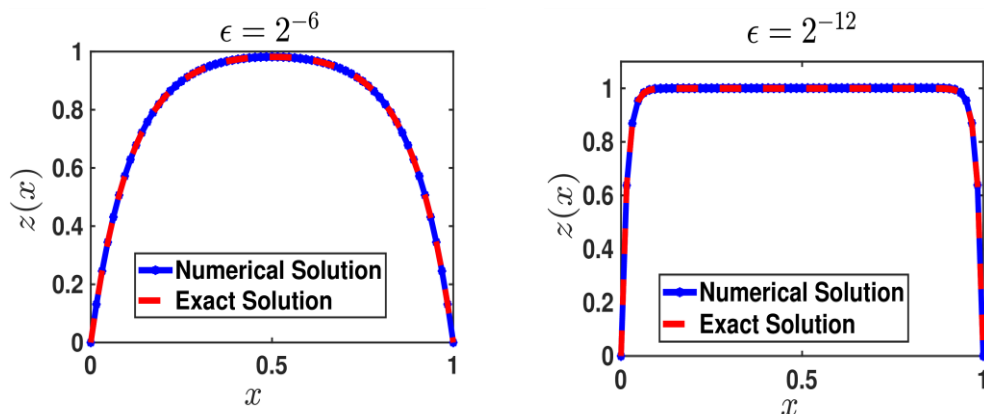


Figure 1. Comparison solution plots for example 5.1 with $N = 64$.

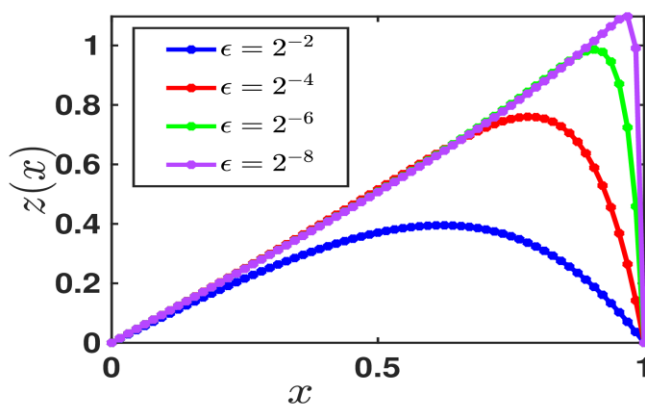


Figure 2. Solution plots of example 5.2 with corresponding values of ϵ .

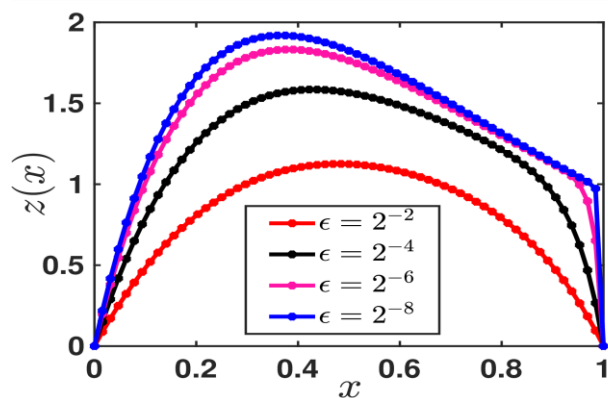


Figure 3. Solution plots of example 5.3 with corresponding values of ϵ .

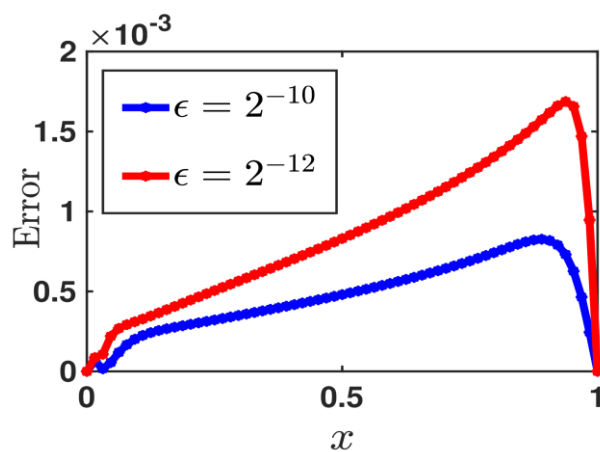


Figure 4. Error plot of example 5.1 before extrapolation, corresponding to the values of ϵ .

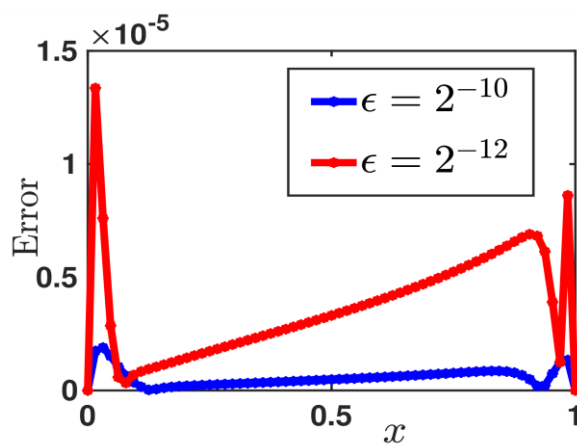


Figure 5. Error plot of example 5.1 after extrapolation, corresponding to the values of ϵ .

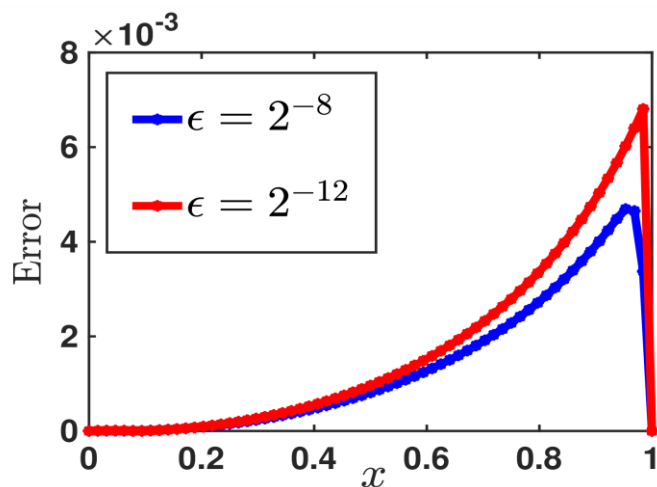


Figure 6. Error plot of example 5.2 before extrapolation, corresponding to the values of ϵ .

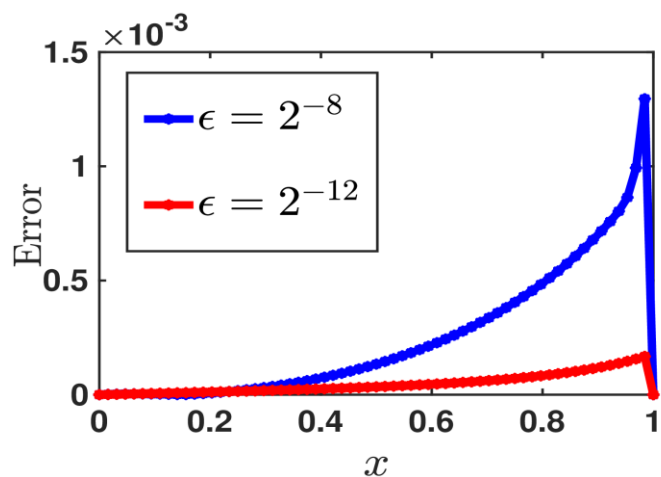


Figure 7. Error plot of example 5.2 after extrapolation, corresponding to the values of ϵ .

Table 1. E_ϵ^N and R_ϵ^N of example 5.1.

$\epsilon \downarrow$	Number of mesh N					
	32	64	128	256	512	1024
2^{-2}	1.2354e-4 1.9990	3.0905e-5 2.0001	7.7256e-6 2.0000	1.9313e-6 2.0000	4.8283e-7 2.0000	1.2071e-7 1.9992
2^{-4}	4.2340e-4 2.0007	1.0580e-4 2.0002	2.6447e-5 2.0000	6.6116e-6 2.0000	1.6529e-6 2.0000	4.1323e-7 2.0001
2^{-6}	8.7423e-4 1.9998	2.1858e-4 1.9996	5.4661e-5 2.0000	1.3665e-5 2.0000	3.4164e-6 2.0000	8.5409e-7 2.0000
2^{-8}	1.6293e-3 1.9958	4.0852e-4 1.9991	1.0220e-4 1.9996	2.5555e-5 2.0000	6.3891e-6 2.0000	1.5973e-6 2.0000
2^{-14}	1.1516e-2 1.7936	3.3219e-3 1.9399	8.6581e-4 1.9835	2.1894e-4 1.9955	5.4907e-5 1.9989	1.3737e-5 1.9997
2^{-16}	1.5904e-2 1.4482	5.8283e-3 1.7896	1.6859e-3 1.9372	4.4024e-4 1.9820	1.1144e-4 1.9956	2.7945e-5 1.9989

Table 2. E_ϵ^N and R_ϵ^N of example 5.1 with extrapolation.

	Number of mesh N					
$\epsilon \downarrow$	16	32	64	128	256	512
2^{-2}	2.7105e-7 3.9965	1.6982e-8 3.9985	1.0624e-9 3.9986	6.6466e-11 3.9038	4.4404e-12 0.3572	3.4666e-12
2^{-4}	1.3601e-6 3.9831	8.6008e-8 3.9964	5.3889e-9 3.9971	3.3748e-10 3.7984	2.4256e-11 1.4812	8.6880e-12
2^{-6}	7.7453e-6 3.8855	5.2408e-7 3.9872	3.3046e-8 3.9969	2.0698e-9 3.9782	1.3133e-10 3.9342	8.5916e-12
2^{-8}	5.2708e-5 3.8042	3.7730e-6 3.9072	2.5147e-7 3.9873	1.5856e-8 3.9939	9.9520e-10 3.9934	6.2487e-11
2^{-14}	4.1424e-3 2.8107	5.9042e-4 3.5263	5.1243e-5 2.9340	6.7052e-6 3.8158	4.7614e-7 3.9363	3.1102e-8
2^{-16}	8.0355e-3 1.8595	2.2143e-3 2.8591	3.0519e-4 3.5320	2.6384e-5 2.9731	3.3600e-6 3.8170	2.3841e-7

Table 3. E_ϵ^N and R_ϵ^N of example 5.2.

	Number of mesh N					
$\epsilon \downarrow$	32	64	128	256	512	1024
2^{-2}	4.0824e-4 0.8788	2.2201e-4 0.9402	1.1570e-4 0.9720	5.8984e-5 0.9860	2.9779e-5 0.9931	1.4960e-5 0.9966
2^{-4}	2.3835e-3 1.1413	1.0806e-3 1.0703	5.1460e-4 1.0358	2.5100e-4 1.0180	1.2394e-4 1.0091	6.1582e-5 1.0045
2^{-6}	6.0606e-3 1.2604	2.5298e-3 1.1478	1.1418e-3 1.0769	5.4123e-4 1.0397	2.6326e-4 1.0202	1.2980e-4 1.0101
2^{-8}	1.1940e-2 1.3474	4.6924e-3 1.2935	1.9143e-3 1.2006	8.3289e-4 1.1159	3.8429e-4 1.0621	1.8406e-4 1.0321
2^{-14}	1.3449e-2 0.9820	6.8085e-3 0.9911	3.4254e-3 0.9955	1.7180e-3 0.9980	8.6021e-4 1.0130	4.2625e-4 1.1134
2^{-16}	1.3449e-2 0.9820	6.8085e-3 0.9911	3.4254e-3 0.9955	1.7180e-3 0.9978	8.6033e-4 0.9989	4.3049e-4 0.9996
2^{-18}	1.3449e-2 0.9820	6.8085e-3 0.9911	3.4254e-3 0.9955	1.7180e-3 0.9978	8.6033e-4 0.9989	4.3049e-4 0.9994

Table 4. E_ϵ^N and R_ϵ^N of example 5.2 with extrapolation.

	Number of mesh N					
$\epsilon \downarrow$	16	32	64	128	256	512
2^{-2}	2.1609e-4 2.0615	5.1770e-5 2.0307	1.2670e-5 2.0153	3.1340e-6 2.0079	7.7925e-7 2.0036	1.9433e-7
2^{-4}	1.3368e-3 2.0666	3.1912e-4 2.0319	7.8034e-5 2.0169	1.9281e-5 2.0091	4.7900e-6 2.0045	1.1938e-6
2^{-6}	5.3708e-3 1.9175	1.4218e-3 1.9536	3.6705e-4 2.0052	9.1432e-5 2.0063	2.2759e-5 2.0036	5.6755e-6
2^{-8}	3.9338e-3 0.5755	2.6397e-3 1.0267	1.2957e-3 1.8276	3.6503e-4 1.9056	9.7428e-5 1.9922	2.4489e-5
2^{-14}	2.4172e-3 1.9049	6.4548e-4 1.9515	1.6689e-4 1.9755	4.2437e-5 1.9580	1.0923e-5 0.0429	1.0603e-5
2^{-16}	2.4172e-3 1.9049	6.4548e-4 1.9515	1.6689e-4 1.9755	4.2437e-5 1.9777	1.0700e-5 1.9938	2.6866e-6
2^{-18}	2.4172e-3 1.9049	6.4548e-4 1.9515	1.6689e-4 1.9755	4.2437e-5 1.9877	1.0700e-5 1.9938	2.6865e-6

Table 5. E_ϵ^N and R_ϵ^N of example 5.3.

	Number of mesh N					
$\epsilon \downarrow$	32	64	128	256	512	1024
2^{-2}	1.5030e-3 1.2897	6.1479e-4 1.1663	2.7393e-4 1.089	1.2877e-4 1.0462	6.2357e-5 1.0236	3.0673e-5 1.0119
2^{-4}	5.9033e-3 1.4053	2.2288e-3 1.2511	9.3637e-4 1.1427	4.2410e-4 1.0764	2.0111e-4 1.0396	9.7837e-5 1.0202
2^{-6}	1.8899e-2 1.6105	6.1894e-3 1.4649	2.2421e-3 1.3043	9.0790e-4 1.1787	4.0106e-4 1.0978	1.8739e-4 1.0513
2^{-8}	4.8675e-2 1.5149	1.7032e-2 1.6572	5.4001e-3 1.5807	1.8054e-3 1.4237	6.7296e-4 1.2696	2.7912e-4 1.1554
2^{-14}	5.8936e-2 0.9307	3.0918e-2 0.9633	1.5857e-2 0.9816	8.0301e-3 0.9911	4.0399e-3 1.0275	1.9818e-3 1.2503
2^{-16}	5.8936e-2 0.9307	3.0918e-2 0.9633	1.5857e-2 0.9816	8.0301e-3 0.9907	4.0411e-3 0.9953	2.0271e-3 0.9981

Table 6. E_ϵ^N and R_ϵ^N of example 5.3 with extrapolation.

	Number of mesh N					
$\epsilon \downarrow$	16	32	64	128	256	512
2^{-2}	1.2765e-3 2.0021	3.1868e-4 2.0017	7.9575e-5 2.0017	1.9870e-5 2.0010	4.9642e-6 2.0005	1.2406e-6 2.0004
2^{-4}	5.9039e-3 1.9725	1.5044e-3 1.9960	3.7714e-4 2.0003	9.4268e-5 2.0007	2.3556e-5 2.0005	5.8869e-6 2.0003
2^{-6}	2.1482e-2 1.6972	6.6248e-3 1.9258	1.7436e-3 1.9825	4.4122e-4 1.9962	1.1060e-4 1.9993	2.7663e-5 2.0000
2^{-16}	1.0849e-2 1.7825	3.1537e-3 1.8980	8.4619e-4 1.9458	2.1964e-4 1.9729	5.5949e-5 1.9864	1.4120e-5 2.3011
2^{-18}	1.0849e-2 1.7825	3.1537e-3 1.8980	8.4619e-4 1.9458	2.1964e-4 1.9729	5.5949e-5 1.9864	1.4120e-5 1.9932
2^{-20}	1.0849e-2 1.7825	3.1537e-3 1.8980	8.4619e-4 1.9458	2.1964e-4 1.9729	5.5949e-5 1.9864	1.4120e-5 1.9932

Table 7. ϵ –uniform maximum error and convergence rate E^N, R^E for example 5.1.

	Number of mesh N					
		32	64	128	256	512
Before extrapolation	E^N	1.8459e-2	8.0736e-3	2.9338e-3	8.4972e-4	2.2261e-4
	R^E	1.1930	1.4604	1.7877	1.9325	1.9830
After extrapolation	E^N	4.2808e-3	1.1443e-3	1.5734e-4	1.3386e-5	1.6820e-6
	R^E	1.9034	2.8625	3.5551	2.9925	3.8176

Table 8. ϵ –uniform maximum error and convergence rate E^N, R^E for example 5.2.

	Number of mesh N					
		32	64	128	256	512
Before extrapolation	E^N	1.3449e-2	6.8085e-3	3.4254e-3	1.7180e-3	8.6033e-4
	R^E	0.9820	0.9911	0.9955	0.9978	0.9989
After extrapolation	E^N	2.6397e-3	1.2957e-3	3.6503e-4	9.7428e-5	2.4489e-5
	R^E	1.0266	1.8276	1.9056	1.9922	1.9994

Table 9. ϵ –uniform maximum error and convergence rate E^N, R^E for example 5.3.

	Number of mesh N					
		32	64	128	256	512
Before extrapolation	E^N	5.8936e-2	3.0918e-2	1.5857e-2	8.0301e-3	4.0411e-3
	R^E	0.9307	0.9633	0.9816	0.9907	0.9953
After extrapolation	E^N	6.6248e-3	1.7436e-3	4.4122e-4	1.1060e-4	2.7663e-5
	R^E	1.9258	1.9825	1.9961	1.9993	1.9999

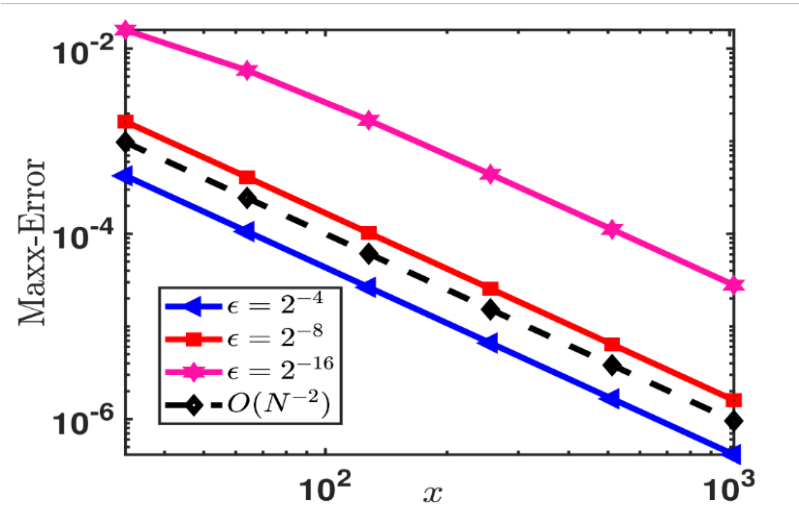


Figure 8. Log-log plots of example 5.1 before extrapolation with corresponding to various values of ϵ .

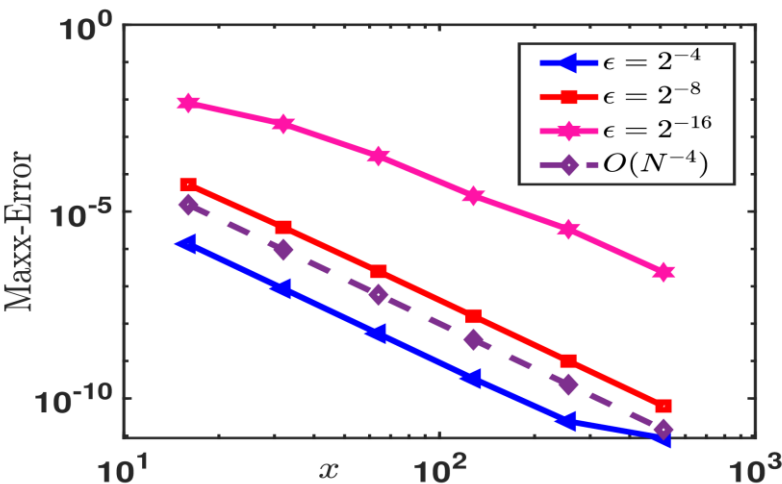


Figure 9. Log-log plots of example 5.1 after extrapolation with corresponding to various values of ϵ .

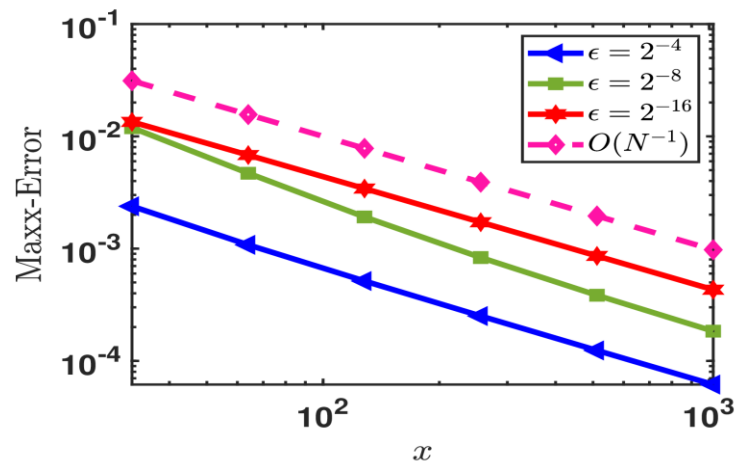


Figure 10. Log-log plots of example 5.2 before extrapolation with corresponding to various values of ϵ .

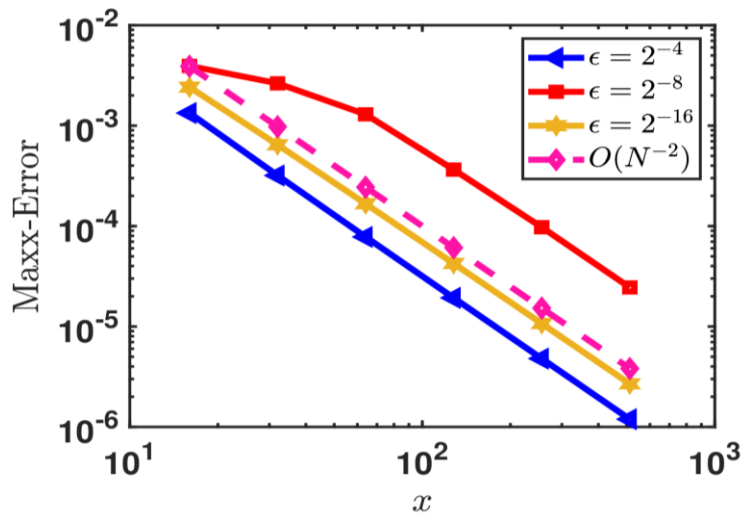


Figure 11. Log-log plots of example 5.2 after extrapolation with corresponding to various values of ϵ .

7. Conclusion

This article addresses the solution of a second-order Fredholm integro-differential equation both theoretically and numerically. A non-standard finite difference method is employed for the differential part, while the trapezoidal rule is used to approximate the integral term on a uniform mesh. The numerical analysis shows that singularly perturbed Fredholm integro-differential equations exhibit uniform convergence of order one for convection-diffusion types and order two for reaction-diffusion types. Upon applying a post-processing technique, the convergence rate improves from first to second order when $a(x) \neq 0$, and from second to fourth order when $a(x) = 0$ in Equation (1). Theoretical findings are validated through numerical experiments, which confirm the predicted rates. The post-processing approach significantly enhances accuracy and enables higher-order convergence, marking a noteworthy advancement in this research area. The key contributions of this work are as follows:

- Modelling layer behavior within the solution using a uniform mesh, demonstrating the accurate resolution of boundary layers through appropriate numerical techniques.
- Efficient application of the non-standard finite difference method for differential terms and the trapezoidal rule for integral components in solving SPFIDEs.
- Enhanced convergence rates to second and fourth order through Richardson extrapolation, marking a significant advancement over previous approaches.

Conflict of Interest

The authors declare that they have no conflicts of interest regarding this publication.

Acknowledgments

The authors express their gratitude for the APC funding received from Amrita Vishwa Vidyapeetham, Coimbatore, India.

AI Disclosure

The author(s) declare that no assistance is taken from generative AI to write this article.

References

- Amiraliyev, G.M., Durmaz, M.E., & Kudu, M. (2020). Fitted second order numerical method for a singularly perturbed Fredholm integro-differential equation. *Bulletin of the Belgian Mathematical Society-Simon Stevin*, 27(1), 71-88.
- Ansari, K.J., Izadi, M., & Noeiaghdam, S. (2024). Enhancing the accuracy and efficiency of two uniformly convergent numerical solvers for singularly perturbed parabolic convection–diffusion–reaction problems with two small parameters. *Demonstratio Mathematica*, 57(1), 20230144. <https://doi.org/10.1515/dema-2023-0144>.
- Chadam, J.M., & Yin, H.M. (1994). A diffusion equation with localized chemical reactions. *Proceedings of the Edinburgh Mathematical Society*, 37(1), 101-118.
- Crank, J. (1975). *The mathematics of diffusion*. Clarendon Press. Oxford, UK.
- Das, A., & Natesan, S. (2017). Second-order uniformly convergent numerical method for singularly perturbed delay parabolic partial differential equations. *International Journal of Computer Mathematics*, 95(3), 490-510. <https://doi.org/10.1080/00207160.2016.1269327>.
- Das, A., & Natesan, S. (2018). Higher-order convergence with fractional-step method for singularly perturbed 2D parabolic convection–diffusion problems on Shishkin mesh. *Computers & Mathematics with Applications*, 75(7), 2387-2403. <https://doi.org/10.1016/j.camwa.2017.11.029>.
- Durmaz, M.E., & Amiraliyev, G.M. (2021). A robust numerical method for a singularly perturbed Fredholm integro-differential equation. *Mediterranean Journal of Mathematics*, 18(1), 24. <https://doi.org/10.1007/s00009-020-01660-z>.
- Elango, S., Govindarao, L., Awadalla, M., & Zaway, H. (2025). Efficient numerical methods for reaction–diffusion problems governed by singularly perturbed Fredholm integro-differential equations. *Mathematics*, 13(9), 1511. <https://doi.org/10.3390/math13091511>.
- Govindarao, L., & Mohapatra, J. (2019). A second-order weighted numerical scheme on nonuniform meshes for convection-diffusion parabolic problems. *European Journal of Computational Mechanics*, 28(5), 467-497.
- Govindarao, L., & Sekar, E. (2023). B-spline method for second order RLC closed series circuit with small inductance value. In *Journal of Physics: Conference Series* (Vol. 2646, No. 1, p. 012039). IOP Publishing. UK. <https://doi.org/10.1088/1742-6596/2646/1/012039>.

- Govindarao, L., Ramos, H., & Elango, S. (2024). Numerical scheme for singularly perturbed Fredholm integro-differential equations with non-local boundary conditions. *Computational and Applied Mathematics*, 43(3), 126. <https://doi.org/10.1007/s40314-024-02530-3>.
- Izadi, M., & Yuzbasi, S. (2022). A hybrid approximation scheme for 1-D singularly perturbed parabolic convection-diffusion problems. *Mathematical Communications*, 27(1), 47-62.
- Izadi, M., & Zeidan, D. (2022). A convergent hybrid numerical scheme for a class of nonlinear diffusion equations. *Computational and Applied Mathematics*, 41(7), 318. <https://doi.org/10.1007/s40314-022-01906-9>.
- Kress, R. (1998). *Numerical analysis*. Springer, New York.
- Lange, C.G., & Smith, D.R. (1993). Singular perturbation analysis of integral equations. *Studies in Applied Mathematics*, 90(1), 1-74.
- Lubuma, J.M.S., & Patidar, K.C. (2007). Non-standard methods for singularly perturbed problems possessing oscillatory/layer solutions. *Applied Mathematics and Computation*, 187(2), 1147-1160.
- Mohapatra, J., Govindarao, L., & Priyadarshana, S. (2025). A splitting based higher-order numerical scheme for 2D time-dependent singularly perturbed reaction-diffusion problems. *The Journal of Supercomputing*, 81(1), 203. <https://doi.org/10.1007/s11227-024-06479-x>.
- Munyakazi, J.B., & Patidar, K.C. (2008). On Richardson extrapolation for fitted operator finite difference methods. *Applied Mathematics and Computation*, 201(1-2), 465-480.
- Rahman, M. (2007). *Integral equations and their applications*. WIT Press, Southampton, UK.
- Sekar, E. (2023). Second order singularly perturbed delay differential equations with nonlocal boundary condition. *Journal of Computational and Applied Mathematics*, 417, 114498. <https://doi.org/10.1016/j.cam.2022.114498>.
- Sekar, E., Govindarao, L., & Vadivel, R. (2025). A comparative study on numerical methods for Fredholm integro-differential equations of convection-diffusion problem with integral boundary conditions. *Applied Numerical Mathematics*, 207, 323-338. <https://doi.org/10.1016/j.apnum.2024.04.003>.
- Sekar, E., Govindarao, L., Mohapatra, J., Vadivel, R., & Hu, N.T. (2024). Numerical analysis for second order differential equation of reaction-diffusion problems in viscoelasticity. *Alexandria Engineering Journal*, 92, 92-101. <https://doi.org/10.1016/j.aej.2023.11.055>.
- Udupa, M., Saha, S., & Das, A. (2022). Computational analysis of blood flow through stenosed artery under the influence of body acceleration using Shishkin mesh. *International Journal of Applied and Computational Mathematics*, 8(3), 107. <https://doi.org/10.1007/s40819-022-01440-3>.
- Zlatev, Z., Dimov, I., Faragó, I., & Havasi, Á. (2017). *Richardson extrapolation: practical aspects and applications* (Vol. 2). Walter de Gruyter GmbH & Co KG., Berlin. <https://doi.org/10.1515/9783110465210>.



Original content of this work is copyright © Ram Arti Publishers. Uses under the Creative Commons Attribution 4.0 International (CC BY 4.0) license at <https://creativecommons.org/licenses/by/4.0/>

Publisher's Note- Ram Arti Publishers remains neutral regarding jurisdictional claims in published maps and institutional affiliations.