

SEMI-ANALYTIC PHYSICS INFORMED NEURAL NETWORK FOR CONVECTION-DOMINATED BOUNDARY LAYER PROBLEMS IN 2D*

GUNG-MIN GIE[†], YOUNGJOON HONG[‡], CHANG-YEOL JUNG[§], AND DONGSEOK LEE[¶]

Abstract. This research investigates the numerical approximation of the two-dimensional convection-dominated singularly perturbed problem on square, circular, and elliptic domains. Singularly perturbed boundary value problems present a significant challenge due to the presence of sharp boundary layers in their solutions. Additionally, the considered domain exhibits characteristic points, giving rise to a degenerate boundary layer problem. The stiffness of the problem is attributed to the sharp singular layers, which can result in substantial computational errors if not appropriately addressed. Traditional numerical methods typically require extensive mesh refinements near the boundary to achieve accurate solutions, which can be computationally expensive. To address the challenges posed by singularly perturbed problems, we employ physics-informed neural networks (PINNs). However, PINNs may struggle with rapidly varying singularly perturbed solutions over a small domain region, leading to inadequate resolution and potentially inaccurate or unstable results. To overcome this limitation, we introduce a semi-analytic method that augments PINNs with singular layers or corrector functions. Through our numerical experiments, we demonstrate significant improvements in both accuracy and stability, thus demonstrating the effectiveness of our proposed approach.

Key words. boundary layer, physics-informed neural networks, singular perturbation, characteristic points

MSC codes. 68T07, 76M45, 65L11

1. Introduction. The use of neural networks for approximating solutions to differential equations has gained significant attention in research [9, 15, 19, 5]. Various unsupervised neural network methods have been developed within this domain, including physics-informed neural networks (PINNs) [23], deep Ritz method (DRM) [29], and Galerkin Neural Network (GNN) [1]. These methods share a common characteristic of defining the loss function based on the residual of the differential equation being considered. PINNs, in particular, utilize collocation points in the space-time domain as inputs, making them well-suited for solving complex, time-dependent, multi-dimensional equations with intricate domain geometries [21, 25, 28, 24, 30, 18, 26]. They have become popular in scientific machine learning, facilitating the integration of physics-based and data-driven modeling in a deep learning framework. Nevertheless, the robustness of PINNs in certain problem types remains an ongoing concern. Notably, PINNs exhibit limitations in accurately capturing complex and highly nonlinear flow patterns such as turbulence, vortical structures, and boundary layers [7, 3, 8, 11]. Addressing this challenge holds great significance in scientific machine learning research, as the development of robust and reliable models is crucial for advancing the field.

The robustness of PINNs is frequently tested when it comes to approximating boundary layers. Convergence issues for PINNs can arise in the context of singularly perturbed differential equations within bounded regions, mainly when the perturbation parameter is sufficiently small. These challenges also pose a hurdle for operator learning approaches like DeepONet and FNO [22, 20]. Overcoming these limitations is crucial for achieving more accurate and reliable predictions across a wide range of PINN applications. Various techniques have recently been proposed to address the approximation of singular problems. Fourier feature networks [27], cPINN [14], XPINN [6], and similar approaches have emerged to tackle the spectral bias of deep neural networks, which limits their capacity to learn high-frequency functions. However, none

*Submitted to the editors 07/31/2023.

Funding: Gie was partially supported by Ascending Star Fellowship, Office of EVPRI, University of Louisville; Simons Foundation Collaboration Grant for Mathematicians; Research R-II Grant, Office of EVPRI, University of Louisville; Brain Pool Program through the National Research Foundation of Korea (NRF) (2020H1D3A2A01110658). Hong was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2021R1A2C1093579) and the Korea government(MSIT)(No. 2022R1A4A3033571). Jung was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. 2023R1A2C1003120).

[†] Department of Mathematics, University of Louisville, Louisville, KY (gungmin.gie@louisville.edu).

[‡]Department of Mathematical Sciences, KAIST, Korea (hongyj@kaist.ac.kr).

[§]Department of Mathematical Sciences, Ulsan National Institute of Science and Technology, Ulsan, Korea (cjung@unist.ac.kr).

[¶]Department of Mathematics, Sungkyunkwan University, Korea (lorakorea@g.skku.edu).

of these techniques mentioned above have been specifically designed to handle thin-layer problems. In a more recent study conducted by the authors of [2], a theory-guided neural network was proposed to address boundary layer problems. This approach introduces a network architecture consisting of two interconnected networks: the inner and outer networks. The inner network focuses on capturing the asymptotic expansion approximation of the solution within the boundary layer region, while the outer network handles the approximation outside of the boundary layer. Both the inner and outer networks consist of multiple parallel PINN networks, with each network representing a specific order approximation of the solution. The coupling between the inner and outer networks is enforced through the matching boundary condition loss.

In this article, we introduce a novel semi-analytic machine learning approach for studying singular boundary layer behavior. Our framework draws inspiration from singular perturbation analysis and asymptotic expansion methods employed in solving singularly perturbed differential equations; for general references, see, for example, [10] and the numerous cited articles therein. It is worth noting that boundary layer analysis for singular perturbation problems is extensively studied and firmly established in applied mathematics, with much of its development driven by the fluid dynamics community. The novelty of our approach lies in establishing and introducing a version of PINNs that exploit boundary layer analysis to approximate sharp transitions. Expanding on the techniques adopted from boundary layer analysis, our work probes multi-dimensional problems with complex domains.

A singularly perturbed boundary value problem, such as the one presented in equations (1.1) below, is widely recognized for inducing a thin layer in proximity to the boundary, commonly referred to as a boundary layer. Within this boundary layer, the solution undergoes a rapid transition. Extensive scholarly research has been conducted on the mathematical theory of singular perturbations and boundary layers. Regarding the numerical approximation of singular perturbation problems, a significant computational error arises near the boundary due to the stiffness of the solution within the boundary layer. Consequently, achieving a sufficiently accurate approximation of the solution in the boundary vicinity typically necessitates a substantial refinement of the mesh, particularly for conventional numerical schemes. However, instead of relying on extensive mesh refinements, novel semi-analytic methods have been proposed. These methods primarily involve enriching the basis of traditional numerical techniques, such as finite elements or finite volumes, by incorporating a global basis function known as the corrector. The corrector effectively captures the singular behavior of the solution within the boundary layers. These semi-analytic methods have demonstrated remarkable efficiency without requiring mesh refinement near the boundary.

In this paper, we address the challenge posed by boundary layers by performing a comprehensive boundary layer analysis for each singular perturbation problem. Through this analysis, we identify the corrector function, which accurately captures the singular behavior within the boundary layer. Subsequently, we develop a new semi-analytic approach by constructing Neural Networks enriched with the corrector function, embedded within the structure of a two-layer PINN with hard constraints, referred to as a *singular layer PINN (SL-PINN)*. To validate the effectiveness of our proposed method, we conduct numerical simulations for each example presented below. The results confirm that our SL-PINNs naturally capture the singular behavior of boundary layers and provide highly accurate approximations for the singularly perturbed boundary value problems discussed in this article. This paper primarily focuses on studying the two-dimensional convection-diffusion equations in various domains such as a unit square, circle, or ellipse denoted by Ω :

$$(1.1) \quad \begin{aligned} L_\varepsilon u^\varepsilon &:= -\varepsilon \Delta u^\varepsilon - u_y^\varepsilon = f, & \text{in } \Omega, \\ u^\varepsilon &= 0, & \text{at } \partial\Omega. \end{aligned}$$

The main focus of this study is to develop semi-analytical neural networks that can effectively incorporate physics-based information and improve their performance by integrating corrector functions. Our approach to constructing a two-layer neural network shares similarities with PINNs, but emphasizes the inclusion of hard constraints in enforcing boundary conditions. We make use of a straightforward neural network, \hat{u} ,

multiplied by $g(x, y)$ to satisfy the boundary condition as follows:

$$(1.2) \quad \bar{u}(x, y; \boldsymbol{\theta}) = g(x, y) \hat{u}(x, y; \boldsymbol{\theta}),$$

where \hat{u} is defined by the two-layer NN, with $g(x, y) = 0$ on $\partial\Omega$. We define a two-layer neural network

$$(1.3) \quad \hat{u}(x, y; \boldsymbol{\theta}) = \sum_{j=1}^n c_j \sigma(w_{1j}x + w_{2j}y + b_j),$$

where n is the number of neurons. The network parameters are denoted by

$$(1.4) \quad \boldsymbol{\theta} = (w_{11}, \dots, w_{1n}, w_{21}, \dots, w_{2n}, b_1, \dots, b_n, c_1, \dots, c_n),$$

and we choose the logistic sigmoid as an activation function, $\sigma(z) = 1/(1 + e^{-z})$. The PINNs with hard constraints described in (1.3) rely on a simplified structure, which enables us to calculate the loss function using explicit derivatives of \hat{u} instead of relying on automatic differentiation (AD). This approach is particularly beneficial when dealing with boundary layers, as it helps avoid potential computational errors that may arise. It's important to note that our approach utilizes the two-layer neural network, which is not only computationally cost-effective but also sufficient for achieving accurate numerical approximations. Furthermore, by incorporating symbolic computation, our methodology can conveniently extend to an M -layer neural network. For comparison, while traditional PINNs leverage an M -layer architecture, our SL-PINNs deploy a two-layer structure. Nonetheless, SL-PINNs significantly surpass the performance of conventional PINNs.

2. Convection-diffusion equations in a square domain. To begin, we examine two-dimensional convection-diffusion equations in square domains. To emphasize the presence of the boundary layer, we employ periodic boundary conditions in the x direction and zero boundary conditions in the y direction:

$$(2.1) \quad \begin{aligned} -\varepsilon \Delta u^\varepsilon - u_y^\varepsilon &= f, & \text{in } \Omega = (0, 1) \times (0, 1), \\ u^\varepsilon &= 0, & \text{at } y = 0 \text{ and } y = 1, \\ u^\varepsilon &\text{ is periodic in } x. \end{aligned}$$

We construct an SL-PINN for the singular perturbation problem and compare its performance with conventional PINNs. To obtain the corrector function, we initiate the process by examining the limit problem, where we set ε to 0 in the governing equation:

$$(2.2) \quad \begin{aligned} -u_y^0 &= f & \text{in } \Omega, \\ u^0 &= 0 & \text{at } y = 1, \\ u^0 &\text{ is periodic in } x. \end{aligned}$$

Along the asymptotic analysis, the boundary layers are defined by the inner expansion $u^\varepsilon \sim \sum_{j=0}^{\infty} \varepsilon^j \theta^j$. Let $\bar{y} = y/\varepsilon$ be considered as an $\mathcal{O}(1)$ quantity in the boundary layer.

$$(2.3) \quad -\sum_{j=0}^{\infty} \{\varepsilon^{j+1} \theta_{xx}^j + \varepsilon^{j-1} \theta_{\bar{y}\bar{y}}^j\} - \sum_{j=0}^{\infty} \varepsilon^{j-1} \theta_y^j = 0.$$

By formally identifying each power of ε , we can derive the corrector equation by selecting the dominant terms,

$$(2.4) \quad \begin{aligned} -\varepsilon \varphi_{yy}^0 - \varphi_y^0 &= 0 & \text{in } \Omega, \\ \varphi^0 &= -u^0(x, 0) & \text{at } y = 0, \\ \varphi^0 &\text{ is periodic in } x. \end{aligned}$$

One can easily find an explicit solution of (2.4)

$$(2.5) \quad \varphi^0 = -u^0(x, 0) \frac{e^{-y/\varepsilon} - e^{-1/\varepsilon}}{1 - e^{-1/\varepsilon}} = -u^0(x, 0)e^{-y/\varepsilon} + e.s.t.,$$

where *e.s.t.* stands for an exponentially small term. Here, we omit further boundary layer analysis, such as convergence to the limit problem. For more detailed theoretical results, one can refer to the work by [10],[16].

We modify the 2-layer PINN in (1.2) based on boundary layer analysis, incorporating the profile of the corrector described in (2.5). Our new *SL-PINN* is proposed of the form,

$$(2.6) \quad \tilde{u}(x, y; \boldsymbol{\theta}) = x(x-1)((y-1)\hat{u}(x, y, \boldsymbol{\theta}) + \hat{u}(x, 0, \boldsymbol{\theta})\bar{\varphi}^0),$$

where the corrector function is given by $\bar{\varphi}^0 = e^{-\frac{y}{\varepsilon}}$. The residual loss function is defined by

$$(2.7) \quad Loss = \frac{1}{N} \sum_{i=0}^N \|L_\varepsilon(\tilde{u}(x_i, y_i; \boldsymbol{\theta})) - f(x_i, y_i)\|_p \quad \text{where } (x_i, y_i) \in \Omega,$$

where $p = 1, 2$. Using the fact that the exponentially decaying function $\bar{\varphi}^0$ satisfies the corrector equation (2.4), we can deduce the residual loss:

$$(2.8) \quad \begin{aligned} -\varepsilon\Delta\tilde{u} - \tilde{u}_y - f &= -2\varepsilon((y-1)\hat{u}(x, y, \boldsymbol{\theta}) + \hat{u}(x, 0, \boldsymbol{\theta})\bar{\varphi}^0) - \varepsilon(x^2 - x)\hat{u}_{xx}(x, 0, \boldsymbol{\theta})\bar{\varphi}^0 \\ &\quad - 2\varepsilon((2x-1)(y-1)\hat{u}_x(x, y, \boldsymbol{\theta}) + (x^2 - x)\hat{u}_y(x, y, \boldsymbol{\theta})) \\ &\quad - 2\varepsilon(2x-1)\hat{u}_x(x, 0, \boldsymbol{\theta})\bar{\varphi}^0 \\ &\quad - \varepsilon(x^2 - x)(y-1)(\hat{u}_{xx}(x, y, \boldsymbol{\theta}) + \hat{u}_{yy}(x, y, \boldsymbol{\theta})) - (x^2 - x)(\hat{u}(x, y, \boldsymbol{\theta}) \\ &\quad + (y-1)\hat{u}_y(x, y, \boldsymbol{\theta})) - f. \end{aligned}$$

Hence, the stiff terms of the residual loss are effectively eliminated. It is worth noting that our SL-PINN provides accurate approximations, as all terms in (2.8) remain bounded as ε approaches 0, regardless of the small parameter ε .

We compare the performance of the standard five-layer PINN approximation with our new SL-PINN approximation. Figure 2.1 displays the numerical solutions of (2.1) with $\varepsilon = 10^{-6}$ when $f = \sin 2\pi x$. In the conventional PINN approximation, we make use of an M -layer neural network, where $M \geq 2$, to enhance the performance. However, Figure 2.1 demonstrates that the conventional PINN method fails to approximate the solution of the singular perturbation problem. On the other hand, our new scheme, which utilizes only a 2-layer neural network with a small number of neurons, outperforms the conventional PINN. The numerical results shown in Figure 2.1 and Table 1 provide strong evidence that the SL-PINN significantly outperforms the conventional PINN method, thanks to the corrector function embedded in the scheme. It is worth noting that our SL-PINN, enriched with the corrector, produces stable and accurate approximate solutions, regardless of the small parameter ε . A closer examination can be made by referring to Figure 2.2. The one-dimensional profile of predicted solutions at $x = 0.25$ allows for a clear comparison. It is evident from Figure 2.2 that the L^1 and L^2 training approaches provide accurate approximations. However, the conventional PINN falls short in capturing the sharp transition near the boundary layer.

3. Convection-diffusion equations on a circular domain. We shift our focus to the convection-diffusion equation within a circular domain. We begin by examining the time-independent problem and subsequently delve into the computation of the time-dependent equations.

3.1. Time-independent problem. We examine the convection-diffusion equation in a circular domain such that

$$(3.1) \quad \begin{aligned} L_\varepsilon u^\varepsilon := -\varepsilon\Delta u^\varepsilon - u_y^\varepsilon &= f, & \text{in } \Omega = \{(x, y) | x^2 + y^2 < 1\} \\ u^\varepsilon &= 0, & \text{at } \partial\Omega. \end{aligned}$$

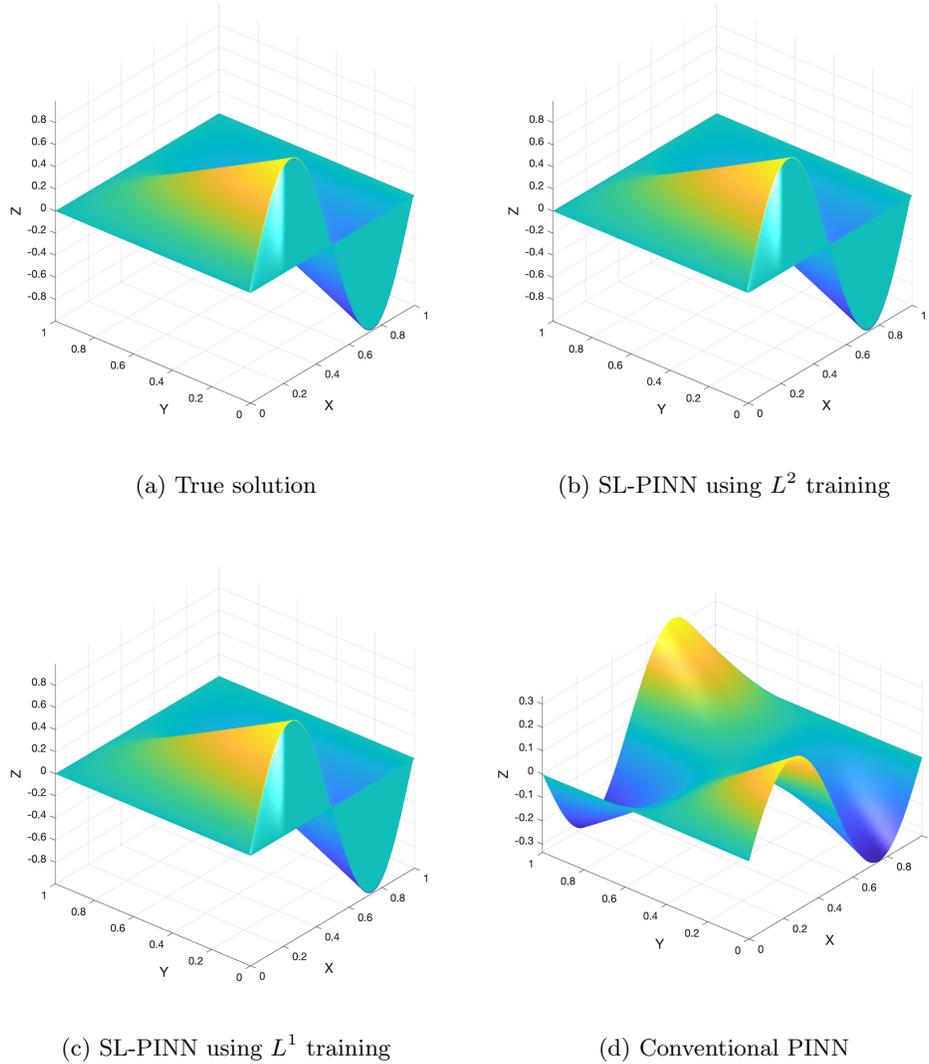


Fig. 2.1: Numerical prediction of (2.1) with $\varepsilon = 10^{-6}$ and $f = \sin(2\pi x)$. For our simulations, we select a uniform grid of 50 discretized points in each of the x and y directions.

The singularly perturbed problem described in equation (3.1) requires careful treatment when conducting a boundary layer analysis due to the presence of a degenerate boundary layer near the characteristic points at $(\pm 1, 0)$ [12]. To develop the boundary layer analysis, we utilize the polar coordinates such that $x = r \cos \xi, y = r \sin \xi$. By setting $u^\varepsilon(x, y) = v^\varepsilon(r, \xi)$, we derive that

$$(3.2) \quad \begin{aligned} P_\varepsilon v^\varepsilon := \frac{1}{r^2} (-\varepsilon(v_{\xi\xi}^\varepsilon + r v_r^\varepsilon + r^2 v_{rr}^\varepsilon) - r^2 \sin \xi v_r^\varepsilon - r \cos \xi v_\xi^\varepsilon) &= f \quad \text{in } D, \\ v^\varepsilon(1, \xi) &= 0 \quad \text{at } 0 \leq \xi \leq 2\pi, \end{aligned}$$

where $D := \{(r, \xi) | 0 < r < 1, 0 \leq \xi \leq 2\pi\}$. For the sake of simplicity, we define the upper semi-circle as $\Gamma_+ := \{(x, y) | x^2 + y^2 = 1, y > 0\}$ and the lower semi-circle as $\Gamma_- := \{(x, y) | x^2 + y^2 = 1, y < 0\}$. Performing

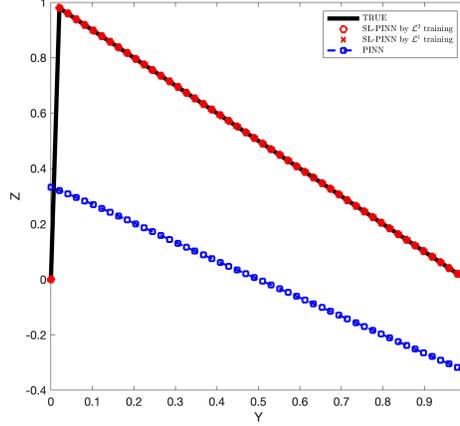


Fig. 2.2: The one-dimensional profile of predicted solutions along the line $x = 0.25$.

the boundary layer analysis as in [12], we are led to the following equation for the corrector equation

$$(3.3) \quad \begin{aligned} -\varepsilon \varphi_{rr}^0 - (\sin \xi) \varphi_r^0 &= 0, & \text{for } 0 < r < 1, \quad 0 < \xi \leq 2\pi, \\ \varphi^0 &= -v^0(\cos \xi, \sin \xi), & \text{on } \Gamma_-, \\ \varphi^0 &\rightarrow 0 & \text{as } r \rightarrow 0. \end{aligned}$$

Hence, we are able to find an explicit solution of the correction equation (3.3):

$$(3.4) \quad \varphi^0 = -v^0(\cos \xi \sin \xi) \exp\left(\frac{\sin \xi}{\varepsilon}(1-r)\right) \chi_{\pi, 2\pi}(\xi),$$

where χ stands for the characteristic function. To match the boundary condition in our numerical scheme, we introduce a cut-off function to derive an approximate form

$$(3.5) \quad \bar{\varphi}^0 = \exp\left(\frac{\sin \xi}{\varepsilon}(1-r)\right) \chi_{\pi, 2\pi}(\xi) \delta(r),$$

where $\delta(r)$ is a smooth cut-off function such that $\delta(r) = 1$ for $r \in [3/4, 1]$ and $= 0$ for $r \in [0, 1/2]$. We now establish the semi-analytic SL-PINN method as

$$(3.6) \quad \tilde{v}(r, \xi; \boldsymbol{\theta}) = (\hat{v}(r, \xi, \boldsymbol{\theta}) - \hat{v}(1, \xi, \boldsymbol{\theta}) \bar{\varphi}^0) C(r, \xi),$$

where $C(r, \xi)$ is given by

$$(3.7) \quad C(r, \xi) = \begin{cases} 1 - r^3, & \text{if } 0 \leq \xi \leq \pi, \\ 1 - r^3 - (r \sin \xi)^3, & \text{if } \pi < \xi < 2\pi, \end{cases}$$

and $\hat{v}(r, \xi) = \hat{u}(x, y)$. Then, the residual loss is defined by

$$(3.8) \quad Loss = \frac{1}{N} \sum_{i=0}^N \|P_\varepsilon \tilde{v}((r_i, \xi_i; \boldsymbol{\theta})) - f\|_p \quad \text{for } (r_i, \xi_i) \in D,$$

where $p = 1, 2$. Since the boundary layer behavior occurs near the lower semi-circle, we divide the residual loss (3.8) into two parts: $0 \leq \xi \leq \pi$ and $\pi \leq \xi \leq 2\pi$. For the case of $0 \leq \xi \leq \pi$, the calculation of the residual loss is straightforward:

$$\begin{aligned}
(3.9) \quad & P_\varepsilon \tilde{v}((r, \xi, \boldsymbol{\theta})) - f \\
&= -\varepsilon \left(\frac{1}{r^2} - r \right) \hat{v}_{\xi\xi}(r, \xi, \boldsymbol{\theta}) + \left(r^2 - \frac{1}{r} \right) \cos(\xi) \hat{v}_\xi(r, \xi, \boldsymbol{\theta}) - \varepsilon(1 - r^3) \hat{v}_{rr}(r, \xi, \boldsymbol{\theta}) \\
&\quad + \left(6\varepsilon + \varepsilon r^2 - \frac{\varepsilon}{r} - \sin(\xi)(1 - r^3) \right) \hat{v}_r(r, \xi, \boldsymbol{\theta}) + \left(3\varepsilon + \frac{6\varepsilon}{r} + 3r^2 \sin(\xi) \right) \hat{v}(r, \xi, \boldsymbol{\theta}) - f,
\end{aligned}$$

where $0 < r \leq 1$ and $0 \leq \xi \leq \pi$. However, in the case of $\pi \leq \xi \leq 2\pi$, the introduction of the boundary layer element in (3.6) and the inclusion of the regularizing term in (3.7) complicate the residual loss as follows:

$$\begin{aligned}
(3.10) \quad & P_\varepsilon \tilde{v}((r, \xi, \boldsymbol{\theta})) - f \\
&= -\varepsilon \left(\frac{1}{r^2} - r - r \sin^3(\xi) \right) \hat{v}_{\xi\xi}(r, \xi, \boldsymbol{\theta}) \\
&\quad + \left(6\varepsilon r \sin^2(\xi) \cos(\xi) + \left(r^2 - \frac{1}{r} + r^2 \sin^3(\xi) \right) \cos(\xi) \right) \hat{v}_\xi(r, \xi, \boldsymbol{\theta}) \\
&\quad - \varepsilon \left(1 - r^3 - r^3 \sin^3(\xi) \right) \hat{v}_{rr}(r, \xi, \boldsymbol{\theta}) \\
&\quad + \left(6\varepsilon + \varepsilon r^2 - \frac{\varepsilon}{r} + 7\varepsilon r^2 \sin^3(\xi) + r^3 \sin^3(\xi) - (1 - r^3) \sin(\xi) \right) \hat{v}_r(r, \xi, \boldsymbol{\theta}) \\
&\quad + \left(3\varepsilon + \frac{6\varepsilon}{r} + 6\varepsilon r \sin(\xi) + 6r^2 \sin(\xi) \right) \hat{v}(r, \xi, \boldsymbol{\theta}) \\
&\quad + \hat{v}(1, \xi, \boldsymbol{\theta}) \left[\varepsilon (C(r, \xi) \delta_{rr} + (rC(r, \xi) + 2C_r(r, \xi)) \delta_r + (rC_r(r, \xi) + C_{rr}(r, \xi)) \delta) \right. \\
&\quad \left. + C(r, \xi) \sin(\xi) \delta_r - C_r(r, \xi) \sin(\xi) \delta \right] \exp \left(\frac{\sin \xi}{\varepsilon} (1 - r) \right) \\
&\quad + \frac{\delta}{r^2} \left[(\varepsilon \hat{v}_{\xi\xi}(1, \xi, \boldsymbol{\theta}) + (2 - r) \cos(\xi) \hat{v}_\xi(1, \xi, \boldsymbol{\theta}) - (1 - r) \sin(\xi) \hat{v}(1, \xi, \boldsymbol{\theta})) C(r, \xi) \right. \\
&\quad \left. + (2\varepsilon \hat{v}_\xi(1, \xi, \boldsymbol{\theta}) + (2 - r) \cos(\xi) \hat{v}(1, \xi, \boldsymbol{\theta})) C_\xi(r, \xi) + \varepsilon \hat{v}(1, \xi, \boldsymbol{\theta}) C_{\xi\xi}(r, \xi) \right] \exp \left(\frac{\sin \xi}{\varepsilon} (1 - r) \right) \\
&\quad + C(r, \xi) \hat{v}(1, \xi, \boldsymbol{\theta}) \frac{\delta(r)}{r^2} \left(\frac{(1 - r) \cos^2(\xi)}{\varepsilon} \right) \exp \left(\frac{\sin \xi}{\varepsilon} (1 - r) \right) - f,
\end{aligned}$$

where $0 < r \leq 1$ and $\pi < \xi < 2\pi$. To make the computation in (3.10) feasible, we extract the largest order term in ε , which includes $O(\varepsilon^{-1})$ such that

$$(3.11) \quad \psi(r, \xi, \boldsymbol{\theta}) := (1 - r^3 - (r \sin(\xi))^3) \hat{v}(1, \xi, \boldsymbol{\theta}) \frac{\delta(r)}{r^2} \left(\frac{1 - r}{\varepsilon} \cos^2 \xi \right) \exp \left(\frac{\sin \xi}{\varepsilon} (1 - r) \right).$$

By the triangular inequality, the loss in (3.8) becomes

$$(3.12) \quad \text{Loss} \leq \frac{1}{N} \sum_{i=0}^N |\psi(r_i, \xi_i, \boldsymbol{\theta})| + \frac{1}{N} \sum_{i=0}^N |P_\varepsilon \tilde{v}((r_i, \xi_i, \boldsymbol{\theta})) - \psi(r_i, \xi_i, \boldsymbol{\theta}) - f|,$$

The rightmost term does not include a large order term such as ε^α (where $\alpha < 0$) since ψ contains the large terms such as ε^{-1} . Hence, this part can be computed using conventional L^1 or L^2 loss. Since ψ contains $\mathcal{O}(\varepsilon^{-1})$, we handle ψ using L^1 loss. More precisely, let us suppose we choose the number of sampling points

N to be sufficiently large

$$\begin{aligned}
(3.13) \quad & \frac{1}{N} \sum_{i=0}^N |\psi(r_i, \xi_i, \boldsymbol{\theta})| \approx \frac{1}{\pi} \int_{\pi}^{2\pi} \int_0^1 |\psi(r, \xi, \boldsymbol{\theta})| dr d\xi \\
& = \frac{1}{\pi} \int_{\pi}^{2\pi} \int_0^1 |1 - r^3 - r^3 \sin^3 \xi| |\hat{v}(1, \xi, \boldsymbol{\theta})| \frac{|\delta(r)|}{r^2} \frac{1-r}{\varepsilon} \cos^2 \xi \exp\left(\frac{\sin \xi}{\varepsilon}(1-r)\right) dr d\xi \\
& \leq \frac{2}{\pi} \int_{\pi}^{2\pi} \int_0^1 |\hat{v}(1, \xi, \boldsymbol{\theta})| \frac{|\delta(r)|}{r^2} \frac{1-r}{\varepsilon} \cos^2 \xi \exp\left(\frac{\sin \xi}{\varepsilon}(1-r)\right) dr d\xi.
\end{aligned}$$

Note that $\delta(r)$ is a smooth cut-off function such that $\delta(r) = 1$ for $r \in [3/4, 1]$ and $= 0$ for $r \in [0, 1/2]$,

$$\begin{aligned}
(3.14) \quad & \frac{1}{\pi} \int_{\pi}^{2\pi} \int_0^1 |\psi(r, \xi, \boldsymbol{\theta})| dr d\xi \leq \frac{2}{\pi} \int_{\pi}^{2\pi} \int_{\frac{1}{2}}^1 |\hat{v}(1, \xi, \boldsymbol{\theta})| \frac{1-r}{r^2 \varepsilon} \cos^2 \xi \exp\left(\frac{\sin \xi}{\varepsilon}(1-r)\right) dr d\xi \\
& \leq \frac{8}{\pi} \int_{\pi}^{2\pi} \int_{\frac{1}{2}}^1 |\hat{v}(1, \xi, \boldsymbol{\theta})| \frac{1-r}{\varepsilon} \cos^2 \xi \exp\left(\frac{\sin \xi}{\varepsilon}(1-r)\right) dr d\xi \\
& \leq \frac{8 \max_{\pi \leq \xi \leq 2\pi} |\hat{v}(1, \xi, \boldsymbol{\theta})|}{\pi} \int_{\pi}^{2\pi} \int_{\frac{1}{2}}^1 \frac{1-r}{\varepsilon} \cos^2 \xi \exp\left(\frac{\sin \xi}{\varepsilon}(1-r)\right) dr d\xi \\
& \leq \frac{8 \max_{\pi \leq \xi \leq 2\pi} |\hat{v}(1, \xi, \boldsymbol{\theta})|}{\pi} \int_{\pi}^{2\pi} \int_{\frac{1}{2}}^1 \frac{1-r}{\varepsilon} |\cos \xi| \exp\left(\frac{\sin \xi}{\varepsilon}(1-r)\right) dr d\xi \\
& \leq \frac{16 \max_{\pi \leq \xi \leq 2\pi} |\hat{v}(1, \xi, \boldsymbol{\theta})|}{\pi} \int_{\frac{3\pi}{2}}^{2\pi} \int_{\frac{1}{2}}^1 \frac{1-r}{\varepsilon} \cos \xi \exp\left(\frac{\sin \xi}{\varepsilon}(1-r)\right) dr d\xi
\end{aligned}$$

By setting $\frac{1-r}{\varepsilon} = \bar{r}$, we have

$$\begin{aligned}
(3.15) \quad & \frac{1}{\pi} \int_{\pi}^{2\pi} \int_0^1 |\psi(r, \xi, \boldsymbol{\theta})| dr d\xi \leq K\varepsilon \int_{\frac{3\pi}{2}}^{2\pi} \int_0^{\frac{1}{2\varepsilon}} \bar{r} \cos \xi \exp(\bar{r} \sin \xi) d\bar{r} d\xi \\
& = K\varepsilon \int_0^{\frac{1}{2\varepsilon}} \left[e^{\bar{r} \sin \xi} \right]_{\xi=\frac{3\pi}{2}}^{2\pi} d\bar{r} \\
& = K\varepsilon \int_0^{\frac{1}{2\varepsilon}} 1 - e^{-\bar{r}} d\bar{r} \\
& = K\varepsilon \left[\bar{r} + e^{-\bar{r}} \right]_{\bar{r}=0}^{\frac{1}{2\varepsilon}} \\
& = \frac{K}{2} + K\varepsilon e^{-\frac{1}{2\varepsilon}} - K\varepsilon.
\end{aligned}$$

Thus, in (3.12), we focus on minimizing the rightmost term, as the first term on the right-hand side of (3.12) becomes nearly constant by (3.15) when ε is sufficiently small. This simplification enables feasible computations in our analysis.

The performance of our new approach is demonstrated through a series of numerical experiments in Figure (3.1)-(3.5). For numerical experiments, we choose 50 uniformly discretized grid points to be used in each ξ and r direction. It is important to note that if f does not vanish at the characteristic points $(\pm 1, 0)$, the function $f(x, y)$ becomes incompatible due to the emergence of singularities in its derivatives at these points. For the sake of simplicity, in this paper, we consider the first compatibility condition, that is,

$$(3.16) \quad \frac{\partial^{p_1+p_2} f}{\partial x^{p_1} \partial y^{p_2}} = 0, \quad \text{at } (\pm 1, 0),$$

for $0 \leq 2p_1 + p_2 \leq 2$ and $p_1, p_2 \geq 0$. For more details on compatibility conditions, see e.g. [12]. We compare the performance of the standard five-layer PINN approximation with our SL-PINN approximation. Figure

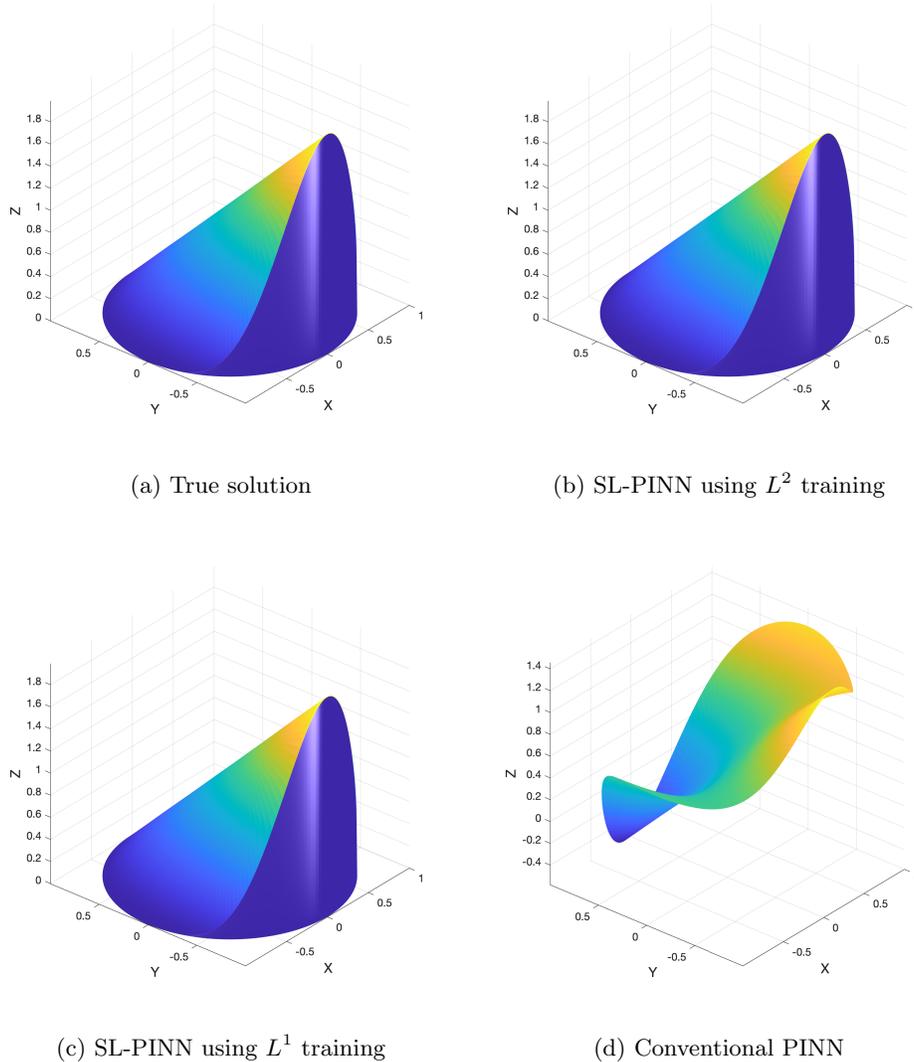


Fig. 3.1: Numerical prediction of (3.1) with $\varepsilon = 10^{-6}$ and $f = (1 - x^2)^2$. For our simulations, we select a uniform grid of 50 discretized points in each of the r and ξ directions.

3.1 displays the numerical solutions of (3.1) with $\varepsilon = 10^{-6}$ when $f = (1 - x^2)^2$. Figure 3.1 clearly illustrates the failure of the conventional PINN method in approximating the solution of the singular perturbation problem. On the contrary, our SL-PINN significantly outperforms the conventional PINN method. The numerical results depicted in Figure 3.1 and Table 1 offer compelling evidence that the semi-analytic SL-PINN outperforms the conventional PINN method significantly. This improvement is attributed to the incorporation of a corrector function within the scheme. Our SL-PINN, enriched with the corrector, consistently generates stable and accurate approximate solutions, irrespective of the small parameter ε . For a more detailed analysis, Figure 3.2 provides a closer examination of the one-dimensional profile of predicted solutions at $\xi = \pi/2$, enabling a clear and direct comparison. Figure 3.2 clearly demonstrates that both the L^1 and L^2 training approaches yield highly accurate approximations. However, the conventional

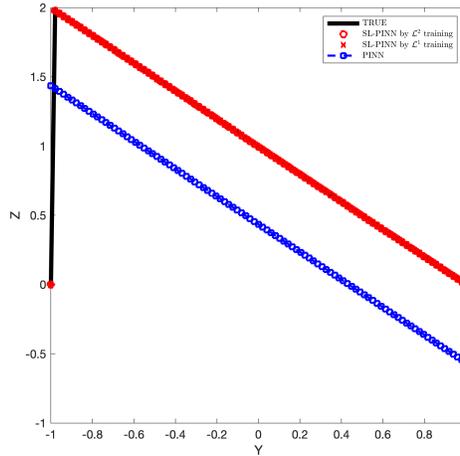


Fig. 3.2: The one-dimensional profile of predicted solutions along the line $\xi = \pi/2$.

PINN method falls short when it comes to capturing the sharp transition near the boundary layer.

To delve deeper into the subject, we consider a non-compatible case where $f = 1$, as shown in Figure 3.3. As mentioned earlier, this non-compatible case introduces an additional singularity when f does not vanish at the characteristic points $(\pm 1, 0)$. Analyzing the boundary layer in this scenario requires careful treatment, as discussed in [17]. However, our SL-PINN approach effectively resolves the boundary layer behavior, surpassing the limitations posed by theoretical singularities. We compare the performance of the standard five-layer PINN approximation with our novel SL-PINN approximation. Figure 3.3 clearly demonstrates the failure of the conventional PINN method in accurately approximating the solution of the singular perturbation problem. In contrast, our SL-PINN, utilizing a 2-layer neural network with a small number of neurons, significantly outperforms the conventional PINN method. The numerical results presented in Figure 3.3 and Table 1 provide compelling evidence of the superior performance of our SL-PINN approach over the conventional PINN method. For a closer examination, Figure 3.4 provides a detailed one-dimensional profile of the predicted solutions at $\xi = \pi/2$, enabling a clear and direct comparison. Moreover, Figure 3.4 shows the highly accurate approximations obtained through both the L^1 and L^2 training approaches. However, the conventional PINN method falls short in capturing the sharp transition near the boundary layer.

As an illustrative example, we utilize our SL-PINN method to tackle a complex solution profile:

$$(3.17) \quad u^\varepsilon = -A(x)(y + \sqrt{1-x^2})(1 - e^{\frac{-2A(x)\sqrt{1-x^2}}{\varepsilon}}) + 2A(x)(\sqrt{1-x^2})(1 - e^{\frac{-A(x)(y+\sqrt{1-x^2})}{\varepsilon}})$$

In this specific case, the corresponding forcing term can be obtained through direct computation. Due to the presence of multiple humps in the exact solution, the numerical solution displays oscillatory profiles, as depicted in Figure 3.5. However, our SL-PINN method effectively captures both the solution profile and the sharp transition near the boundary layer. We have excluded the numerical experiments with the conventional PINN approach since it consistently fails to produce an accurate solution profile.

3.2. Time-dependent problem. We extend our approach to a time dependent problem

$$(3.18) \quad \begin{aligned} L_\varepsilon u^\varepsilon : u_t^\varepsilon - \varepsilon \Delta u^\varepsilon - u_y^\varepsilon &= f, & \text{for } (x, y) \in \Omega, & \quad t \in (0, T), \\ u^\varepsilon(x, y, t) &= 0, & \text{for } (x, y) \in \partial\Omega, & \quad t \in (0, T), \\ u^\varepsilon(x, y, 0) &= 0, & \text{for } (x, y) \in \Omega, & \quad \text{at } t = 0. \end{aligned}$$

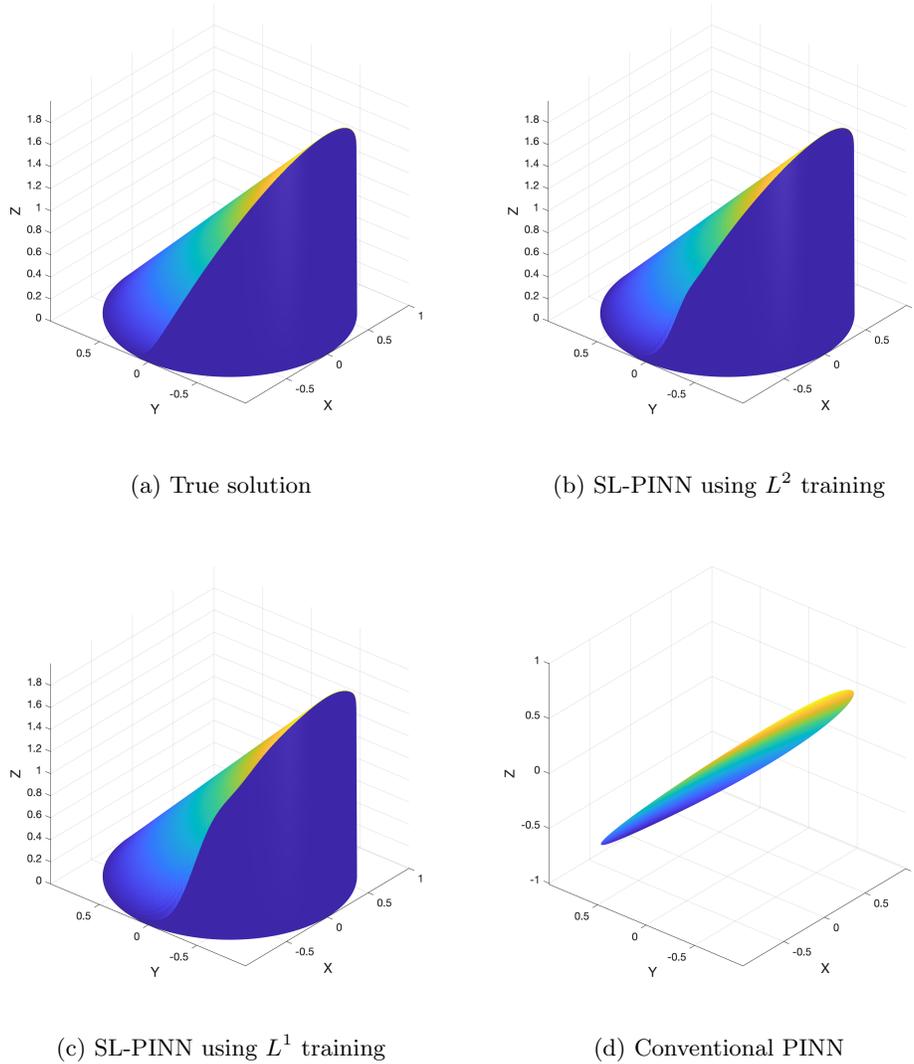


Fig. 3.3: Numerical prediction of (3.1) with $\varepsilon = 10^{-6}$ and $f = 1$ (non-compatible case). For our simulations, we select a uniform grid of 50 discretized points in each of the r and ξ directions.

We employ polar coordinates, expressed as $x = r \cos \xi$ and $y = r \sin \xi$, in order to facilitate the development of the boundary layer analysis. By introducing the transformation $u^\varepsilon(x, y, t) = v^\varepsilon(r, \xi, t)$, we obtain that

$$\begin{aligned}
 (3.19) \quad P_\varepsilon v^\varepsilon = v_t^\varepsilon + \frac{1}{r^2}(-\varepsilon(v_{\xi\xi}^\varepsilon + rv_r^\varepsilon + r^2v_{rr}^\varepsilon)) - r^2(\sin \xi)v_r^\varepsilon - r(\cos \xi)v_\xi^\varepsilon = f, \quad \text{in } (r, \xi) \in D, t \in (0, T), \\
 v^\varepsilon(1, \xi, t) = 0, \quad \text{at } 0 \leq \xi \leq 2\pi, t \in (0, T), \\
 v^\varepsilon(r, \xi, 0) = 0, \quad \text{in } (r, \xi) \in D.
 \end{aligned}$$

As stated in the boundary layer analysis conducted by [13], the sharp transition occurs solely in the spatial direction. Consequently, the solution does not exhibit a boundary layer in the temporal direction. Therefore, the time-independent problem yields the same form of the corrector equation as in the case

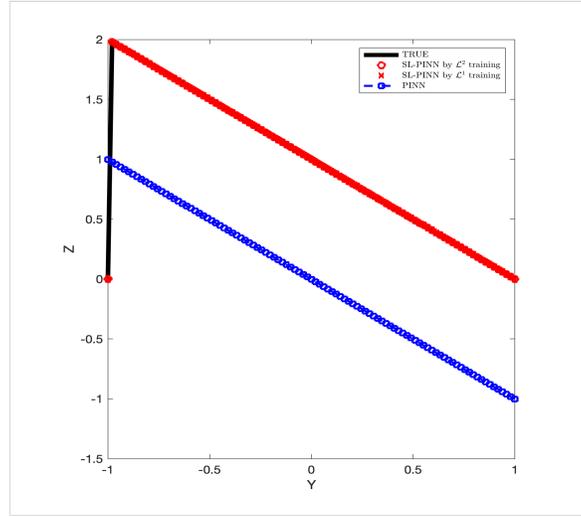


Fig. 3.4: True vs Predicted solution at $\xi = 0.5\pi$

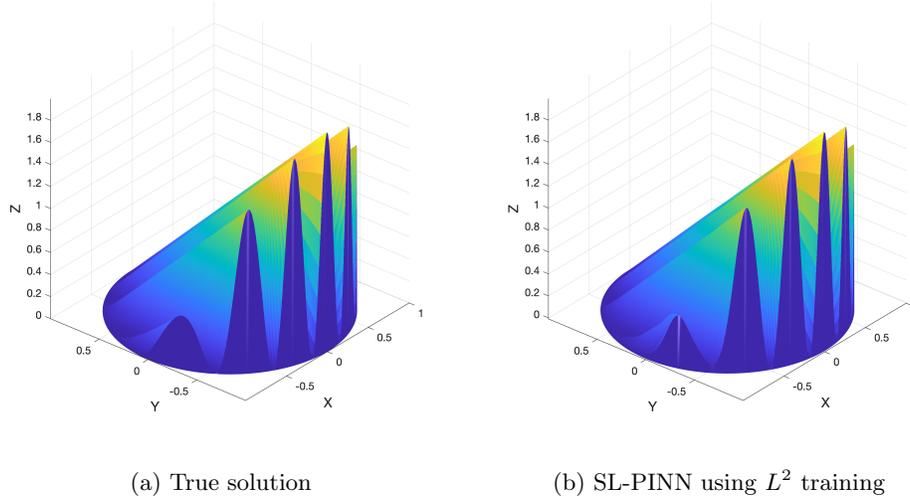


Fig. 3.5: Numerical prediction of (3.1) with $\varepsilon = 10^{-6}$ and f specified in (3.17). For our simulations, we select a uniform grid of 50 discretized points in each of the r and ξ directions.

without time dependence. Hence, we can derive the corrector equation as described [13]: For each $t > 0$,

$$\begin{aligned}
 (3.20) \quad & -\varepsilon\varphi_{rr}^0 - (\sin \xi)\varphi_r^0 = 0, \quad \text{for } (r, \xi) \in D \\
 & \varphi^0 = -v^0(\cos \xi, \sin \xi), \quad \text{for } (r, \xi) \in \Gamma_-, \\
 & \varphi^0 \rightarrow 0 \quad \text{as } r \rightarrow 0.
 \end{aligned}$$

Hence, we are able to find an explicit solution to the correction equation:

$$(3.21) \quad \varphi^0 = -v^0(t, \cos \xi, \sin \xi) \exp\left(\frac{\sin \xi}{\varepsilon}(1-r)\right) \chi_{[\pi, 2\pi]}(\xi),$$

where χ stands for the characteristic function. For our numerical scheme, we introduce an approximate form,

$$(3.22) \quad \tilde{\varphi}^0 = \exp\left(\frac{\sin \xi}{\varepsilon}(1-r)\right) \chi_{\pi, 2\pi}(\xi) \delta(r),$$

where $\delta(r)$ is a smooth cut-off function such that $\delta(r) = 1$ for $r \in [3/4, 1]$ and $= 0$ for $r \in [0, 1/2]$. We establish the *SL-PINN* based on boundary layer analysis, incorporating the profile of the corrector such that

$$(3.23) \quad \tilde{v}(t, r, \xi; \boldsymbol{\theta}) = (e^t - 1) (\hat{v}(t, r, \xi, \boldsymbol{\theta}) - \hat{v}(t, 1, \xi, \boldsymbol{\theta}) \tilde{\varphi}^0(r, \xi)) C(r, \xi),$$

where $C(r, \xi)$ is given by

$$(3.24) \quad C(r, \xi) = \begin{cases} 1 - r^3, & \text{if } 0 \leq \xi \leq \pi \\ 1 - r^3 - (r \sin \xi)^3, & \text{if } \pi < \xi < 2\pi, \end{cases}$$

and $\hat{v}(t, r, \xi, \boldsymbol{\theta}) = \hat{u}(t, x, y, \boldsymbol{\theta}) = W_4 \sigma(W_1 x + W_2 y + W_3 t + b)$. Then, the residual loss is defined by

$$(3.25) \quad Loss = \frac{1}{N} \sum_{i=0}^N \|P_\varepsilon \tilde{v}((t_i, r_i, \xi_i, \boldsymbol{\theta})) - f\|_p \quad \text{for } (t_i, r_i, \xi_i) \in [0, T] \times D,$$

where $p = 1, 2$. Due to the boundary layer behavior near the lower semi-circle, we split the residual loss (3.25) into two sections: $0 \leq \xi \leq \pi$ and $\pi \leq \xi \leq 2\pi$. For the $0 \leq \xi \leq \pi$, calculating the residual loss is simple:

$$(3.26) \quad \begin{aligned} & P_\varepsilon \tilde{v}((t, r, \xi, \boldsymbol{\theta})) - f \\ &= (e^t \hat{v} + (e^t - 1) \hat{v}_t(t, r, \xi, \boldsymbol{\theta}))(1 - r^3) + (e^t - 1) \left(-\varepsilon \left(\frac{1}{r^2} - r\right) \hat{v}_{\xi\xi}(t, r, \xi, \boldsymbol{\theta})\right. \\ &+ \left(r^2 - \frac{1}{r}\right) \cos(\xi) \hat{v}_\xi(t, r, \xi, \boldsymbol{\theta}) - \varepsilon(1 - r^3) \hat{v}_{rr}(t, r, \xi, \boldsymbol{\theta}) + \left(6\varepsilon + \varepsilon r^2 - \frac{\varepsilon}{r} - \sin(\xi)(1 - r^3)\right) \hat{v}_r(t, r, \xi, \boldsymbol{\theta}) \\ &+ \left(3\varepsilon + \frac{6\varepsilon}{r} + 3r^2 \sin(\xi)\right) \hat{v}(t, r, \xi, \boldsymbol{\theta})) - f \text{ for } 0 < r \leq 1, \quad 0 \leq \xi \leq \pi, \quad 0 \leq t \leq T, \end{aligned}$$

where $0 < r \leq 1$ and $0 \leq \xi \leq \pi$. Introducing the boundary layer element in equation (3.23) and adding the regularizing term in equation (3.24) makes the residual loss calculation more complex when ξ ranges from π to 2π such that

(3.27)

$$\begin{aligned}
& P_\varepsilon \tilde{v}((t, r, \xi, \boldsymbol{\theta})) - f \\
&= (e^t \hat{v}(t, r, \xi, \boldsymbol{\theta}) + (e^t - 1) \hat{v}_t - (e^t \hat{v}(t, 1, \xi, \boldsymbol{\theta}) + (e^t - 1) \hat{v}(t, 1, \xi, \boldsymbol{\theta})_t) \\
&\exp\left(\frac{\sin \xi}{\varepsilon}(1-r)\right)(1-r^3 - r^3 \sin^3(\xi) + (e^t - 1)(-\varepsilon(\frac{1}{r^2} - r - r \sin^3(\xi)) \hat{v}_{\xi\xi}(t, r, \xi, \boldsymbol{\theta}) \\
&+ (6\varepsilon r \sin^2(\xi) \cos(\xi) + (r^2 - \frac{1}{r} + r^2 \sin^3(\xi)) \cos(\xi)) \hat{v}_\xi(t, r, \xi, \boldsymbol{\theta}) - \varepsilon(1 - r^3 - r^3 \sin^3(\xi)) \hat{v}_{rr}(t, r, \xi, \boldsymbol{\theta}) \\
&+ (6\varepsilon + \varepsilon r^2 - \frac{\varepsilon}{r} + 7\varepsilon r^2 \sin^3(\xi) + r^3 \sin^3(\xi) - (1 - r^3) \sin(\xi)) \hat{v}_r(t, r, \xi, \boldsymbol{\theta}) \\
&+ (3\varepsilon + \frac{6\varepsilon}{r} + 6\varepsilon r \sin(\xi) + 6r^2 \sin(\xi)) \hat{v}(t, r, \xi, \boldsymbol{\theta}) \\
&+ \hat{v}(t, 1, \xi, \boldsymbol{\theta}) [\varepsilon(C(r, \xi) \delta_{rr} + (rC(r, \xi) + 2C_r(r, \xi)) \delta_r + (rC_r(r, \xi) + C_{rr})(r, \xi) \delta) \\
&+ C(r, \xi) \sin(\xi) \delta_r - C_r(r, \xi) \sin(\xi) \delta] \exp\left(\frac{\sin \xi}{\varepsilon}(1-r)\right) \\
&+ \frac{\delta}{r^2} [(\varepsilon \hat{v}_{\xi\xi}(t, 1, \xi, \boldsymbol{\theta}) + (2-r) \cos(\xi) \hat{v}_\xi(t, 1, \xi, \boldsymbol{\theta}) \\
&- (1-r) \sin(\xi) \hat{v}(t, 1, \xi, \boldsymbol{\theta})) C(r, \xi) + (2\varepsilon \hat{v}_\xi(t, 1, \xi, \boldsymbol{\theta}) + (2-r) \cos(\xi) \hat{v}(t, 1, \xi, \boldsymbol{\theta})) C_\xi(r, \xi) \\
&+ \varepsilon \hat{v}(t, 1, \xi, \boldsymbol{\theta}) C_{\xi\xi}(r, \xi)] \exp\left(\frac{\sin \xi}{\varepsilon}(1-r)\right) \\
&+ (e^t - 1) C(r, \xi) \hat{v}(t, 1, \xi, \boldsymbol{\theta}) \frac{\delta(r)}{r^2} \left(\frac{1-r}{\varepsilon} \cos^2(\xi)\right) \exp\left(\frac{\sin \xi}{\varepsilon}(1-r)\right) - f,
\end{aligned}$$

where $0 < r \leq 1$, $\pi < \xi < 2\pi$, $0 \leq t \leq T$. We make computation feasible in equation (3.27) by extracting the largest order term in ε , which includes $O(\varepsilon^{-1})$ such that

$$(3.28) \quad \psi(t, r, \xi, \boldsymbol{\theta}) = (e^t - 1)(1 - r^3 - (r \sin(\xi))^3) \hat{v}(t, 1, \xi, \boldsymbol{\theta}) \frac{\delta(r)}{r^2} \left(\frac{1-r}{\varepsilon} \cos^2 \xi\right) \exp\left(\frac{\sin \xi}{\varepsilon}(1-r)\right).$$

By the triangular inequality, we deduce that

$$(3.29) \quad Loss \leq \frac{1}{N} \sum_{i=0}^N |\psi(t_i, r_i, \xi_i, \boldsymbol{\theta})| + \frac{1}{N} \sum_{i=0}^N |P_\varepsilon \tilde{v}((t_i, r_i, \xi_i, \boldsymbol{\theta})) - \psi(t_i, r_i, \xi_i, \boldsymbol{\theta}) - f|,$$

The rightmost term does not include a large order term, ε^α where $\alpha < 0$, because ψ already accounts for the large terms such as ε^{-1} . Therefore, conventional L^1 or L^2 loss can be used to compute this part. As ψ contains $O(\varepsilon^{-1})$, we employ L^1 -loss to handle ψ . Specifically, we choose a sufficiently large number of sampling points N to derive

$$(3.30) \quad \frac{1}{N} \sum_{i=0}^N |\psi(t_i, r_i, \xi_i, \boldsymbol{\theta})| \approx \frac{1}{T\pi} \int_0^T \int_\pi^{2\pi} \int_0^1 |\psi(t, r, \xi, \boldsymbol{\theta})| dr d\xi dt.$$

By employing calculations similar to those presented in equations (3.13) to (3.15), we can deduce the following result:

$$(3.31) \quad \frac{1}{T\pi} \int_0^T \int_\pi^{2\pi} \int_0^1 |\psi(t, r, \xi, \boldsymbol{\theta})| dr d\xi dt \leq \frac{K}{2} + K\varepsilon e^{-\frac{1}{2\varepsilon}} - K\varepsilon.$$

Thus, in (3.29), we focus on minimizing the rightmost term, as the first term on the right-hand side of (3.29) becomes nearly constant by (3.31) when ε is sufficiently small. This simplification enables feasible computations in our analysis.

The effectiveness of our new approach is demonstrated through a series of numerical experiments, as shown in Figure 3.6. For these experiments, we employ a grid discretization consisting of 50 uniformly spaced grid points in each direction of ξ , r , and t . The function f used in these experiments satisfies the compatibility condition mentioned in [13]:

$$(3.32) \quad f = (1 - x^2)^2 t + (1 - x^2)^2 (-y + \sqrt{1 - x^2} + \frac{\varepsilon(y + \sqrt{1 - x^2})}{(1 - x^2)^{3/2}}) + O(\varepsilon).$$

Also, the corresponding u is given by

$$(3.33) \quad u(t, x, y) = \begin{cases} t(1 - x^2)^2 (-y + \sqrt{1 - x^2} + \varepsilon \frac{(y + \sqrt{1 - x^2})}{(1 - x^2)^{3/2}}), & \text{in } D \\ 0, & \text{on } \partial D. \end{cases}$$

Figure 3.6 presents the numerical solution obtained using SL-PINN with $\varepsilon = 10^{-6}$. In contrast, the conventional PINNs are not included in the figure due to their significant deviation from the true solution. The SL-PINN, enhanced with the corrector function, consistently produces stable and accurate approximate solutions, irrespective of the small parameter ε . Figure 3.6 (C) depicts the one-dimensional predicted solution profiles at $\xi = \pi/2$ for various time instances, including $t = 0.25, 0.5, 0.75$, and 1.0 , alongside the true solution. This example serves as evidence that our SL-PINN method delivers accurate predictions throughout the entire time span.

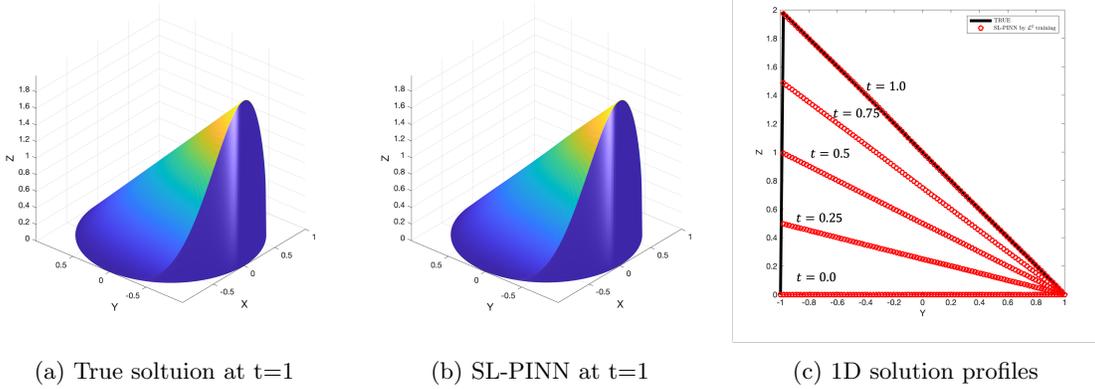


Fig. 3.6: Numerical prediction of (3.18) with $\varepsilon = 10^{-6}$ and f specified in (3.32). For our simulations, we select a uniform grid of 50 discretized points in each of the r and ξ directions. In the panel (c), the one-dimensional predicted solution profiles are displayed at $\xi = \pi/2$ for various time instances, including $t = 0.25, 0.5, 0.75$, and 1.0 .

4. Convection-diffusion equations on an elliptical domain. We explore the convection-diffusion equations on an elliptical domain, which represents a natural yet non-trivial extension of the problem:

$$(4.1) \quad \begin{aligned} L_\varepsilon u^\varepsilon &:= -\varepsilon \Delta u^\varepsilon - u_y^\varepsilon = f, & \text{in } \Omega = \left\{ (x, y) \mid \frac{x^2}{(R^x)^2} + \frac{y^2}{(R^y)^2} < 1 \right\} \\ u^\varepsilon &= 0, & \text{at } \partial\Omega. \end{aligned}$$

Within the elliptical domain, we utilize the elliptic coordinate system [4] and examine two separate scenarios; i) $R^x > R^y$; ii) $R^x < R^y$.

4.1. Major Axis Parallel to x -Axis. To start, we define the elliptic coordinates as (r, ξ) such that

$$(4.2) \quad x = a \cosh r \cos \xi, y = a \sinh r \sin \xi,$$

where $r > 0$ and $\xi \in [0, 2\pi]$. By setting $u^\varepsilon(x, y) = v^\varepsilon(r, \xi)$ with

$$a \cosh R = R^x, \quad a \sinh R = R^y,$$

we transform (4.1) into the following form

$$(4.3) \quad \begin{aligned} P_\varepsilon v^\varepsilon := -\varepsilon(v_{rr}^\varepsilon + v_{\xi\xi}^\varepsilon) - (a \cosh r \sin \xi)v_r^\varepsilon - (a \sinh r \cos \xi)v_\xi^\varepsilon = Hf \quad \text{in } D = [0, R] \times [0, 2\pi] \\ v^\varepsilon(R, \xi) = 0 \text{ at } 0 \leq \xi \leq 2\pi, \end{aligned}$$

where $H = (a \sinh r \cos \xi)^2 + (a \cosh r \sin \xi)^2$. To derive the corrector equation of φ , we use the following stretching variable,

$$(4.4) \quad \bar{r} = \frac{R - r}{\varepsilon^\alpha}.$$

We then obtain the corrector equation with $\alpha = 1$, which is the thickness of the boundary layer,

$$(4.5) \quad -\varphi_{\bar{r}\bar{r}}^0 + \left(a \left(\frac{e^{R-\varepsilon\bar{r}} + e^{\varepsilon\bar{r}-R}}{2} \right) \sin \xi \right) \varphi_{\bar{r}}^0 = 0.$$

Instead of seeking an explicit solution for the corrector equation, we employ the Taylor series expansion for computational convenience to obtain a solution profile for (4.5),

$$(4.6) \quad -\varphi_{\bar{r}\bar{r}}^0 + \left(a \left(\frac{e^R + e^{-R}}{2} + \varepsilon \left(\frac{e^{-R} - e^R}{2} \right) \bar{r} + \frac{\varepsilon^2}{2!} \left(\frac{e^R + e^{-R}}{2} \right) \bar{r}^2 + \dots \right) \sin \xi \right) \varphi_{\bar{r}}^0 = 0.$$

By identifying the dominant terms in equation (4.6), we derive the approximate corrector equations,

$$(4.7) \quad \begin{aligned} -\varphi_{\bar{r}\bar{r}}^0 + \left(a \left(\frac{e^R + e^{-R}}{2} \right) \sin \xi \right) \varphi_{\bar{r}}^0 = 0, \\ \varphi^0 = -v^0 \text{ at } \bar{r} = 0. \end{aligned}$$

where $v^0(r, \xi) = u^0(x, y)$. An explicit solution can be calculated as

$$(4.8) \quad \bar{\varphi}^0(\bar{r}, \xi) = -v^0(R, \xi)e^{(R^x \sin \xi)\bar{r}}.$$

For our numerical scheme, we introduce an approximate form,

$$(4.9) \quad \bar{\varphi}^0 = \exp\left(\frac{R^x(R-r)\sin \xi}{\varepsilon}\right) \chi_{(\pi, 2\pi)}\delta(r).$$

With (4.8), we now introduce our *SL-PINN* of the form,

$$(4.10) \quad \tilde{v}(r, \xi; \boldsymbol{\theta}) = (\hat{v}(r, \xi, \boldsymbol{\theta}) - \hat{v}(R, \xi, \boldsymbol{\theta})\bar{\varphi}^0(r, \xi)) C(r, \xi),$$

where $C(r, \xi)$ is the regularizing term given by

$$(4.11) \quad C(r, \xi) = \begin{cases} 1 - \left(\frac{r}{R}\right)^3, & \text{if } 0 \leq \xi \leq \pi \\ 1 - \left(\frac{r}{R}\right)^3 - \left(\frac{r}{R} \sin \xi\right)^3, & \text{if } \pi < \xi < 2\pi. \end{cases}$$

Then, the residual loss is defined by

$$(4.12) \quad Loss = \frac{1}{N} \sum_{i=0}^N \|P_\varepsilon \tilde{v}((r_i, \xi_i, \boldsymbol{\theta})) - f\|_p \quad \text{for } (r_i, \xi_i) \in D,$$

where $p = 1, 2$. To account for the boundary layer behavior occurring near the boundary where $\pi \leq \xi \leq 2\pi$, we divide the residual loss (4.12) into two sections: $0 \leq \xi \leq \pi$ and $\pi \leq \xi \leq 2\pi$. Calculating the residual loss is straightforward for the range of $0 \leq \xi \leq \pi$:

$$(4.13) \quad \begin{aligned} & P_\varepsilon \tilde{v}((r, \xi, \boldsymbol{\theta})) - f \\ &= -\varepsilon(\hat{v}_{rr}(r, \xi, \boldsymbol{\theta})(1 - (\frac{r}{R})^3) + 2\hat{v}_r(r, \xi, \boldsymbol{\theta})(-3\frac{r^2}{R^3}) + \hat{v}(r, \xi, \boldsymbol{\theta})(-6\frac{r}{R^3}) + \hat{v}_{\xi\xi}(r, \xi, \boldsymbol{\theta})(1 - (\frac{r}{R})^3)) \\ & - (a \cosh r \sin \xi)(\hat{v}_r(r, \xi, \boldsymbol{\theta})(1 - (\frac{r}{R})^3) + \hat{v}(r, \xi, \boldsymbol{\theta})(-3\frac{r^2}{R^3})) \\ & - (a \sinh r \cos \xi)\hat{v}_\xi(r, \xi, \boldsymbol{\theta})(1 - (\frac{r}{R})^3) - f \\ & \text{for } 0 < r \leq R, 0 \leq \xi \leq \pi. \end{aligned}$$

However, when considering $\pi \leq \xi \leq 2\pi$, incorporating the boundary layer element in (4.10) and including the regularizing term in (4.11) result in a more intricate form of the residual loss, as shown below:

$$(4.14) \quad \begin{aligned} & P_\varepsilon \tilde{v}((r, \xi, \boldsymbol{\theta})) - f = \\ & - \varepsilon(\hat{v}_{rr}(r, \xi, \boldsymbol{\theta})C + 2\hat{v}_r(r, \xi, \boldsymbol{\theta})C_r + \hat{v}(r, \xi, \boldsymbol{\theta})C_{rr} + \hat{v}_{\xi\xi}(r, \xi, \boldsymbol{\theta})C + 2\hat{v}_\xi(r, \xi, \boldsymbol{\theta})C_\xi + \hat{v}(r, \xi, \boldsymbol{\theta})C_{\xi\xi}) \\ & - (a \cosh r \sin \xi)(\hat{v}_r(r, \xi, \boldsymbol{\theta})C + \hat{v}(r, \xi, \boldsymbol{\theta})C_r) - (a \sinh r \cos \xi)(\hat{v}_\xi C + \hat{v}(r, \xi, \boldsymbol{\theta})C_\xi) \\ & + \varepsilon\hat{v}(R, \xi, \boldsymbol{\theta})[\delta_{rr}C + 2\delta_r C_r + \delta C_{rr}] \exp\left(\frac{R^x \sin \xi}{\varepsilon}(R - r)\right) \\ & - \hat{v}(R, \xi, \boldsymbol{\theta})[2R^x \sin \xi \delta_r C + 2R^x \sin \xi \delta C_r] \exp\left(\frac{R^x \sin \xi}{\varepsilon}(R - r)\right) \\ & + a(\cosh r \sin \xi)\hat{v}(R, \xi, \boldsymbol{\theta})[\delta_r C + \delta C_r] \exp\left(\frac{R^x \sin \xi}{\varepsilon}(R - r)\right) \\ & + \varepsilon\delta[\hat{v}_{\xi\xi}(r, \xi, \boldsymbol{\theta})C + 2\hat{v}_\xi(r, \xi, \boldsymbol{\theta})C_\xi + \hat{v}(r, \xi, \boldsymbol{\theta})C_{\xi\xi}] \exp\left(\frac{R^x \sin \xi}{\varepsilon}(R - r)\right) \\ & - \delta[R^x(R - r)((\sin \xi \hat{v}(r, \xi, \boldsymbol{\theta}) - 2 \cos \xi \hat{v}_\xi(r, \xi, \boldsymbol{\theta}))C - 2(\cos \xi \hat{v}(r, \xi, \boldsymbol{\theta}))C_\xi] \exp\left(\frac{R^x \sin \xi}{\varepsilon}(R - r)\right) \\ & + \frac{\hat{v}(R, \xi, \boldsymbol{\theta})}{\varepsilon} \delta C [(R^x \sin \xi)^2 - a \cosh r \sin^2 \xi R^x + \\ & (R^x(R - r) \cos \xi)^2 + a \sinh r \cos^2 \xi R^x(R - r)] \exp\left(\frac{R^x \sin \xi}{\varepsilon}(R - r)\right) - f \\ & \text{for } 0 < r \leq R, \pi \leq \xi \leq 2\pi \end{aligned}$$

As in the circular domain, calculating the residual loss in (4.14) involves a substantial term that can misguide our loss optimization process. In order to make the computation in equation (4.14) feasible, we extract the dominant term in ε , specifically the term of $\mathcal{O}(\varepsilon^{-1})$. This allows us to derive

$$(4.15) \quad \begin{aligned} \psi(r, \xi, \boldsymbol{\theta}) := & \frac{\hat{v}(R, \xi, \boldsymbol{\theta})}{\varepsilon} \delta(r) \left(1 - (\frac{r}{R})^3 - (\frac{r}{R} \sin \xi)^3\right) [(R^x \sin \xi)^2 - a \cosh r \sin^2 \xi R^x \\ & + (R^x(R - r) \cos \xi)^2 + a \sinh r \cos^2 \xi R^x(R - r)] \exp\left(\frac{R^x \sin \xi}{\varepsilon}(R - r)\right). \end{aligned}$$

By the triangular inequality, the loss in (4.12) bounds

$$(4.16) \quad \text{Loss} \leq \frac{1}{N} \sum_{i=0}^N |\psi(r_i, \xi_i, \boldsymbol{\theta})| + \frac{1}{N} \sum_{i=0}^N |P_\varepsilon \tilde{v}((r_i, \xi_i, \boldsymbol{\theta})) - \psi(r_i, \xi_i, \boldsymbol{\theta}) - f|,$$

The rightmost term does not involve a high-order term, such as ε^α where $\alpha < 0$, as ψ already contains large terms like ε^{-1} . Therefore, this part can be computed using a conventional L^1 or L^2 loss. Considering that ψ contains terms of $\mathcal{O}(\varepsilon^{-1})$, we utilize an L^1 loss for handling ψ . To ensure accuracy, we select a sufficiently large number of sampling points, denoted as N :

$$(4.17) \quad \begin{aligned} & \frac{1}{N} \sum_{i=0}^N |\psi(r_i, \xi_i, \boldsymbol{\theta})| \approx \frac{1}{R\pi} \int_\pi^{2\pi} \int_0^R |\psi(r, \xi, \boldsymbol{\theta})| dr d\xi \\ &= \frac{1}{R\pi} \int_\pi^{2\pi} \int_0^R \left| \frac{\hat{v}(R, \xi, \boldsymbol{\theta})}{\varepsilon} \delta(r) \left(1 - \left(\frac{r}{R}\right)^3 - \left(\frac{r}{R} \sin \xi\right)^3\right) [(R^x \sin \xi)^2 - a \cosh r \sin^2 \xi R^x] \right. \\ &+ \left. (R^x (R-r) \cos \xi)^2 + a \sinh r \cos^2 \xi R^x (R-r) \right] \exp\left(\frac{R^x \sin \xi}{\varepsilon} (R-r)\right) \Big| dr d\xi \\ &\leq \frac{2}{R\pi} \int_\pi^{2\pi} \int_0^R \left| \frac{\hat{v}(R, \xi, \boldsymbol{\theta})}{\varepsilon} \delta(r) [(R^x \sin \xi)^2 - a \cosh r \sin^2 \xi R^x] \right. \\ &+ \left. (R^x (R-r) \cos \xi)^2 + a \sinh r \cos^2 \xi R^x (R-r) \right] \exp\left(\frac{R^x \sin \xi}{\varepsilon} (R-r)\right) \Big| dr d\xi. \end{aligned}$$

Note that, $\delta(r)$ is a smooth cut-off function such that $\delta(r) = 1$ for $r \in [3R/4, R]$ and $= 0$ for $r \in [0, R/2]$.

$$(4.18) \quad \begin{aligned} & \frac{1}{R\pi} \int_\pi^{2\pi} \int_0^R |\psi(r, \xi, \boldsymbol{\theta})| dr d\xi \leq \frac{K}{\varepsilon} \int_\pi^{2\pi} \int_{\frac{R}{2}}^R |[(R^x \sin \xi)^2 - a \cosh r \sin^2 \xi R^x] \\ &+ (R^x (R-r) \cos \xi)^2 + a \sinh r \cos^2 \xi R^x (R-r)| \exp\left(\frac{R^x \sin \xi}{\varepsilon} (R-r)\right) dr d\xi \\ &\leq \frac{K}{\varepsilon} \int_\pi^{2\pi} \int_{\frac{R}{2}}^R [(-R_x^2 \sin \xi) - a \cosh R \sin \xi R^x \\ &+ (R^x (R-r))^2 |\cos \xi| + a \sinh r |\cos \xi| R^x (R-r)] \exp\left(\frac{R^x \sin \xi}{\varepsilon} (R-r)\right) dr d\xi \\ &\leq \frac{K}{\varepsilon} \int_\pi^{2\pi} \varepsilon (R^x + a \cosh R) \left[e^{\frac{R^x \sin \xi (R-r)}{\varepsilon}} \right]_{r=\frac{R}{2}}^{r=R} d\xi \\ &+ 2 \frac{K}{\varepsilon} \int_{\frac{R}{2}}^R \varepsilon (R^x (R-r) + a \sinh r) \left[e^{\frac{R^x \sin \xi (R-r)}{\varepsilon}} \right]_{\xi=\frac{3\pi}{2}}^{\xi=2\pi} dr \\ &\leq K\pi (R^x + a \cosh R) (1 - e^{-R^x R/2\varepsilon}) + \frac{1}{4} K R^x R^2 + 2Ka (\cosh R - \cosh(R/2)). \end{aligned}$$

Thus, in (4.16), we focus on minimizing the rightmost term, as the first term on the right-hand side of (4.16) becomes nearly constant by (4.18) when ε is sufficiently small. This simplification enables feasible computations in our analysis.

The effectiveness of our new approach is demonstrated through a series of numerical experiments presented in Figures (4.1) and (4.2). In these experiments, we utilize a grid discretization consisting of 50 uniformly spaced points in both the ξ and r directions. Similar to the circular domain, when f does not vanish at the characteristic points $(\pm 4, 0)$ the function $f(x, y)$ becomes incompatible due to the appearance of singularities in its derivatives at these points. To maintain simplicity in this paper, we focus on the first compatibility condition. We compare the performance of the standard five-layer PINN approximation with

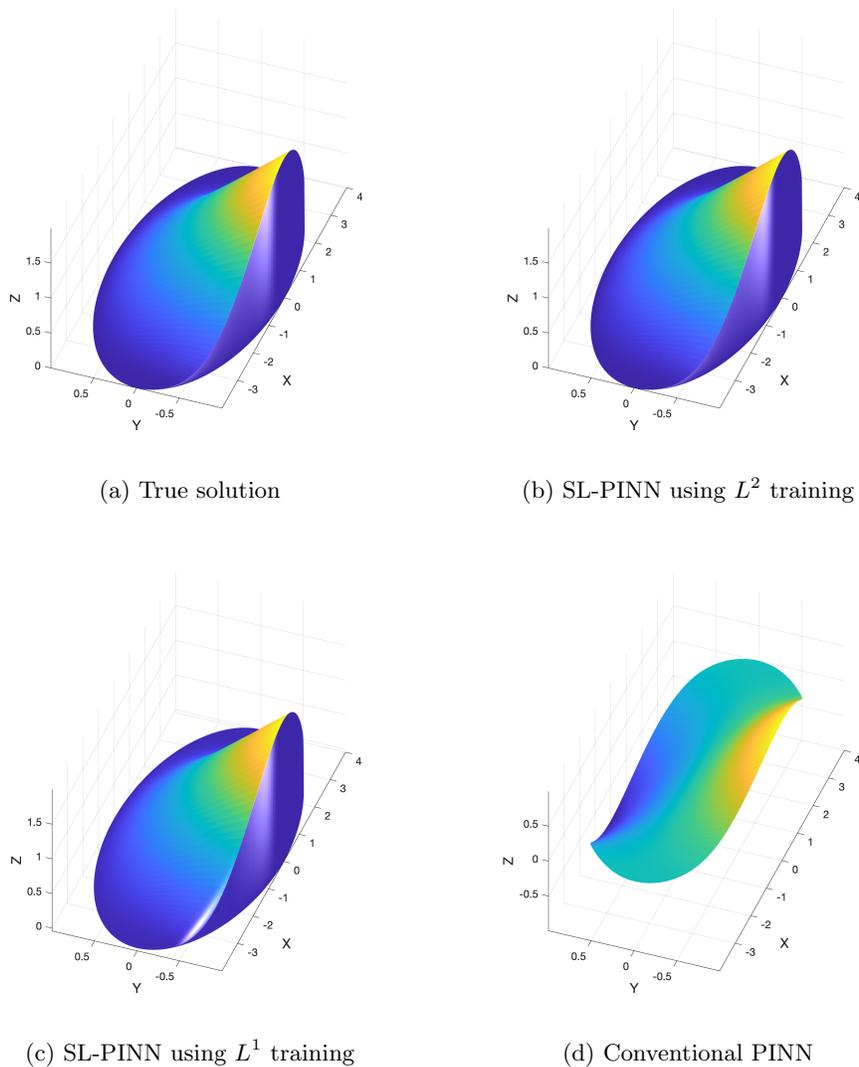


Fig. 4.1: Numerical prediction of (4.1) with $(R^x, R^y) = (4, 1)$ and $\varepsilon = 10^{-6}$ when $f = (1 - (x/R^x)^2)^2/R^y$. For our simulations, we select a uniform grid of 50 discretized points in each of the r and ξ directions.

our SL-PINN approximation. Figure 4.1 showcases the numerical solutions of (4.1) with $(R^x, R^y) = (4, 1)$ and $\varepsilon = 10^{-6}$ when $f = (1 - (x/R^x)^2)^2/R^y$. Figure 4.1 provides a clear illustration of the conventional PINN method's failure in accurately approximating the solution to the singular perturbation problem. In contrast, our new scheme exhibits significant superiority over the conventional PINN method. The numerical results presented in Figure 4.1 and Table 1 provide compelling evidence of the substantial performance improvement achieved by our SL-PINN approach. Our semi-analytic SL-PINN, enriched with the corrector, consistently produces stable and accurate approximate solutions regardless of the small parameter ε . Figure 4.2 offers a closer examination of the one-dimensional profile of predicted solutions at $\xi = \pi/2$, facilitating a clear and direct comparison. Figure 4.2 vividly demonstrates the high accuracy of both the L^1 and L^2 training approaches in generating approximations. However, the conventional PINN

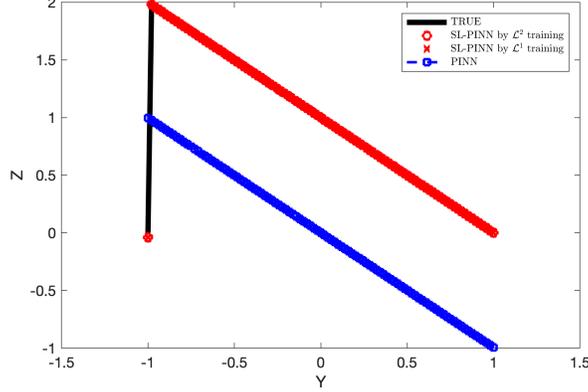


Fig. 4.2: The one-dimensional profile of predicted solutions in Figure 4.1 along the line $\xi = \pi/2$.

method falls short in capturing the sharp transition near the boundary layer.

4.2. Major Axis Parallel to y -Axis. We now define the elliptic coordinates as (r, ξ) such that

$$(4.19) \quad x = a \sinh r \cos \xi, y = a \cosh r \sin \xi,$$

where $r > 0$ and $\xi \in [0, 2\pi]$. Setting $u^\varepsilon(x, y) = v^\varepsilon(r, \xi)$ and $a \sinh R = R^x$, and $a \cosh R = R^y$, we transform (4.1) into the following form

$$(4.20) \quad \begin{aligned} P_\varepsilon v^\varepsilon := -\varepsilon(v_{rr}^\varepsilon + v_{\xi\xi}^\varepsilon) - (a \sinh r \sin \xi)v_r^\varepsilon - (a \cosh r \cos \xi)v_\xi^\varepsilon &= Hf \quad \text{in } D = [0, R] \times [0, 2\pi] \\ v^\varepsilon(R, \xi) &= 0 \text{ at } 0 \leq \xi \leq 2\pi, \end{aligned}$$

where $H = (a \sinh r \sin \xi)^2 + (a \cosh r \cos \xi)^2$. To derive the corrector equation of φ , we make use of the following stretching variable,

$$(4.21) \quad \bar{r} = \frac{R - r}{\varepsilon^\alpha}.$$

We then obtain the corrector equation with $\alpha = 1$, which is the thickness of the boundary layer,

$$(4.22) \quad -\varphi_{\bar{r}\bar{r}}^0 + \left(a \left(\frac{e^{R-\varepsilon\bar{r}} - e^{\varepsilon\bar{r}-R}}{2} \right) \sin \xi \right) \varphi_{\bar{r}}^0 = 0.$$

To find an explicit solution for the corrector, we simplify (4.22) by applying the Taylor series expansion,

$$(4.23) \quad -\varphi_{\bar{r}\bar{r}}^0 + \left(a \left(\frac{e^R - e^{-R}}{2} - \varepsilon \left(\frac{e^{-R} + e^R}{2} \right) \bar{r} + \frac{\varepsilon^2}{2!} \left(\frac{e^R - e^{-R}}{2} \right) \bar{r}^2 - \dots \right) \sin \xi \right) \varphi_{\bar{r}}^0 = 0.$$

By identifying the dominant terms in equation (4.23), we derive the approximate corrector equations,

$$(4.24) \quad \begin{aligned} -\varphi_{\bar{r}\bar{r}}^0 + \left(a \left(\frac{e^R - e^{-R}}{2} \right) \sin \xi \right) \varphi_{\bar{r}}^0 &= 0 \\ \varphi^0 &= -v^0 \text{ at } \bar{r} = 0, \end{aligned}$$

where $v^0(r, \xi) = u^0(x, y)$. An explicit solution can be calculated as

$$(4.25) \quad \bar{\varphi}^0(\bar{r}, \xi) = -v^0(R, \xi)e^{(R^x \sin \xi)\bar{r}}.$$

For our numerical scheme, we introduce an approximate form,

$$(4.26) \quad \bar{\varphi}^0 = \exp\left(\frac{R^x(R-r)\sin\xi}{\varepsilon}\right)\chi_{(\pi,2\pi)}\delta(r).$$

With (4.25), we now introduce our *SL-PINN* of the form,

$$(4.27) \quad \tilde{v}(r, \xi; \boldsymbol{\theta}) = (\hat{v}(r, \xi, \boldsymbol{\theta}) - \hat{v}(R, \xi, \boldsymbol{\theta})\bar{\varphi}^0)C(r, \xi),$$

where $C(r, \xi)$ is the boundary regularizing term given by

$$(4.28) \quad C(r, \xi) = \begin{cases} 1 - (\frac{r}{R})^3, & \text{if } 0 \leq \xi \leq \pi, \\ 1 - (\frac{r}{R})^3 - (\frac{r}{R}\sin\xi)^3, & \text{if } \pi < \xi < 2\pi. \end{cases}$$

Then, the residual loss is defined by

$$(4.29) \quad Loss = \frac{1}{N} \sum_{i=0}^N \|P_\varepsilon \tilde{v}((r_i, \xi_i, \boldsymbol{\theta})) - f\|_p \quad \text{for } (r_i, \xi_i) \in D,$$

where $p = 1, 2$. Considering the boundary layer behavior near $\pi \leq \xi \leq 2\pi$, we split the residual loss (4.29) into two sections: $0 \leq \xi \leq \pi$ and $\pi \leq \xi \leq 2\pi$. The residual loss calculation for $0 \leq \xi \leq \pi$ is relatively simple:

$$(4.30) \quad \begin{aligned} P_\varepsilon \tilde{v}((r, \xi, \boldsymbol{\theta})) - f = & \\ & - \varepsilon(\hat{v}_{rr}(r, \xi, \boldsymbol{\theta})(1 - (\frac{r}{R})^3) + 2\hat{v}_r(r, \xi, \boldsymbol{\theta})(-3\frac{r^2}{R^3}) + \hat{v}(r, \xi, \boldsymbol{\theta})(-6\frac{r}{R^3}) + \hat{v}_{\xi\xi}(r, \xi, \boldsymbol{\theta})(1 - (\frac{r}{R})^3)) \\ & - (a \sinh r \sin \xi)(\hat{v}_r(r, \xi, \boldsymbol{\theta})(1 - (\frac{r}{R})^3) + \hat{v}(r, \xi, \boldsymbol{\theta})(-3\frac{r^2}{R^3})) \\ & - (a \cosh r \cos \xi)\hat{v}_\xi(r, \xi, \boldsymbol{\theta})(1 - (\frac{r}{R})^3) - f, \end{aligned}$$

for $0 < r \leq R, 0 \leq \xi \leq \pi$. However, when considering $\pi \leq \xi \leq 2\pi$, incorporating the boundary layer element in (4.27) and including the regularizing term in (4.28) result in a more intricate form of the residual loss, as shown below:

$$(4.31) \quad \begin{aligned} P_\varepsilon \tilde{v}((r, \xi, \boldsymbol{\theta})) - f = & \\ & - \varepsilon(\hat{v}_{rr}(r, \xi, \boldsymbol{\theta})C + 2\hat{v}_r(r, \xi, \boldsymbol{\theta})C_r + \hat{v}(r, \xi, \boldsymbol{\theta})C_{rr} + \hat{v}_{\xi\xi}(r, \xi, \boldsymbol{\theta})C + 2\hat{v}_\xi(r, \xi, \boldsymbol{\theta})C_\xi + \hat{v}(r, \xi, \boldsymbol{\theta})C_{\xi\xi}) \\ & - (a \sinh r \sin \xi)(\hat{v}_r(r, \xi, \boldsymbol{\theta})C + \hat{v}(r, \xi, \boldsymbol{\theta})C_r) - (a \cosh r \cos \xi)(\hat{v}_\xi(r, \xi, \boldsymbol{\theta})C + \hat{v}(r, \xi, \boldsymbol{\theta})C_\xi) \\ & + \varepsilon\hat{v}(R, \xi, \boldsymbol{\theta})[\delta_{rr}C + 2\delta_r C_r + \delta C_{rr}] \exp\left(\frac{R^x \sin \xi}{\varepsilon}(R-r)\right) \\ & - \hat{v}(R, \xi, \boldsymbol{\theta})[2R^x \sin \xi \delta_r C + 2R^x \sin \xi \delta C_r] \exp\left(\frac{R^x \sin \xi}{\varepsilon}(R-r)\right) \\ & + a(\cosh r \sin \xi)\hat{v}(R, \xi, \boldsymbol{\theta})[\delta_r C + \delta C_r] \exp\left(\frac{R^x \sin \xi}{\varepsilon}(R-r)\right) \\ & + \varepsilon\delta[\hat{v}_{\xi\xi}(r, \xi, \boldsymbol{\theta})C + 2\hat{v}_\xi(r, \xi, \boldsymbol{\theta})C_\xi + \hat{v}(r, \xi, \boldsymbol{\theta})C_{\xi\xi}] \exp\left(\frac{R^x \sin \xi}{\varepsilon}(R-r)\right) \\ & - \delta[R^x(R-r)((\sin \xi \hat{v}(r, \xi, \boldsymbol{\theta}) - 2 \cos \xi \hat{v}_\xi(r, \xi, \boldsymbol{\theta}))C - 2(\cos \xi \hat{v}(r, \xi, \boldsymbol{\theta}))C_\xi] \exp\left(\frac{R^x \sin \xi}{\varepsilon}(R-r)\right) \\ & + \frac{\hat{v}(R, \xi, \boldsymbol{\theta})}{\varepsilon}\delta C[(R^x \sin \xi)^2 - a \sinh r \sin^2 \xi R^x + \\ & (R^x(R-r) \cos \xi)^2 + a \cosh r \cos^2 \xi R^x(R-r)] \exp\left(\frac{R^x \sin \xi}{\varepsilon}(R-r)\right) - f, \end{aligned}$$

for $0 < r \leq R$, $\pi \leq \xi \leq 2\pi$. Similar to the circular domain, the residual loss indicated in (4.14) encompasses a substantial term that may potentially misdirect our loss optimization process. In order to make the computation in (4.14) feasible, we extract the dominant term in ε , specifically the term of $\mathcal{O}(\varepsilon^{-1})$. Hence, we define

$$(4.32) \quad \begin{aligned} \psi(r, \xi, \boldsymbol{\theta}) := & \frac{\hat{v}(R, \xi, \boldsymbol{\theta})}{\varepsilon} \delta(r) \left(1 - \left(\frac{r}{R}\right)^3 - \left(\frac{r}{R} \sin \xi\right)^3\right) \left[(R^x \sin \xi)^2 - a \sinh r \sin^2 \xi R^x\right. \\ & \left. + (R^x(R-r) \cos \xi)^2 + a \cosh r \cos^2 \xi R^x(R-r)\right] \exp\left(\frac{R^x \sin \xi}{\varepsilon}(R-r)\right). \end{aligned}$$

By the triangular inequality, the loss in (4.29) bounds

$$(4.33) \quad \text{Loss} \leq \frac{1}{N} \sum_{i=0}^N |\psi(r_i, \xi_i, \boldsymbol{\theta})| + \frac{1}{N} \sum_{i=0}^N |P_\varepsilon \tilde{v}((r_i, \xi_i, \boldsymbol{\theta})) - \psi(r_i, \xi_i, \boldsymbol{\theta}) - f|.$$

Since ψ absorbs large terms such as ε^{-1} , the rightmost term does not involve high-order terms such as ε^α where $\alpha < 0$. Given that ψ includes terms of $\mathcal{O}(\varepsilon^{-1})$, we employ an L^1 loss function for managing ψ :

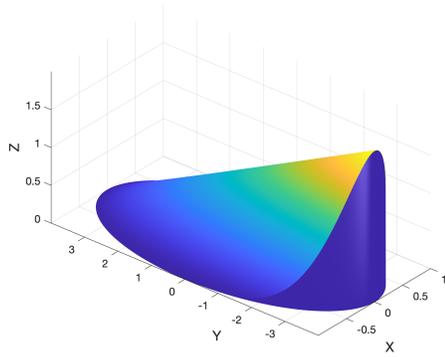
$$(4.34) \quad \frac{1}{N} \sum_{i=0}^N |\psi(r_i, \xi_i, \boldsymbol{\theta})| \approx \frac{1}{R\pi} \int_\pi^{2\pi} \int_0^R |\psi(r, \xi, \boldsymbol{\theta})| dr d\xi,$$

where N is the number of sampling points. By employing calculations similar to those presented in equations (4.16) to (4.18), we can deduce the following result:

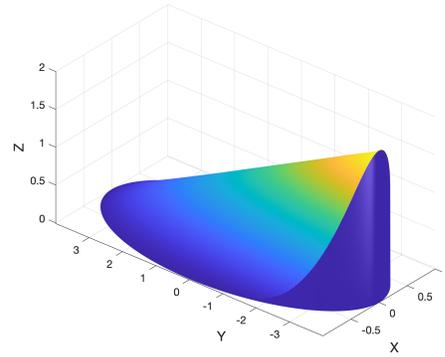
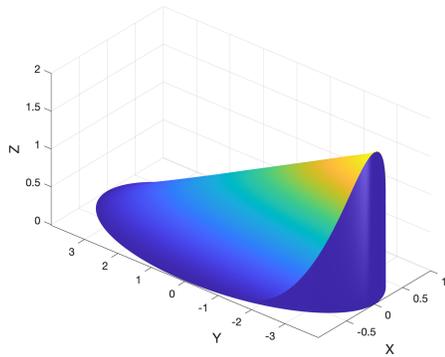
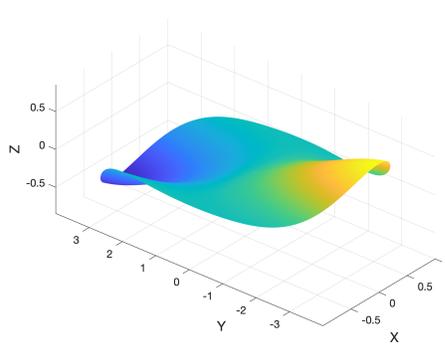
$$(4.35) \quad \begin{aligned} & \frac{1}{R\pi} \int_\pi^{2\pi} \int_0^R |\psi(r, \xi, \boldsymbol{\theta})| dr d\xi \\ & \leq \frac{K}{\varepsilon} \int_\pi^{2\pi} \varepsilon (R^x + a \sinh R) \left[e^{\frac{R^x \sin \xi (R-r)}{\varepsilon}} \right]_{r=\frac{R}{2}}^{r=R} d\xi \\ & + 2 \frac{K}{\varepsilon} \int_{\frac{R}{2}}^R \varepsilon (R^x (R-r) + a \cosh r) \left[e^{\frac{R^x \sin \xi (R-r)}{\varepsilon}} \right]_{\xi=\frac{3\pi}{2}}^{\xi=2\pi} dr \\ & \leq K\pi (R^x + a \sinh R) (1 - e^{-R^x R/2\varepsilon}) + \frac{1}{4} K R^x R^2 + 2Ka (\sinh R - \cosh(R/2)). \end{aligned}$$

Hence, in (4.33), we focus on minimizing the rightmost term, since the first term on the right-hand side of (4.33) becomes nearly constant by (4.35) when ε is sufficiently small. This simplification enables feasible computations in our computation. Figures (4.3) provide numerical evidence of the SL-PINN's effectiveness. For these experiments, we employed a grid comprising 50 uniformly spaced points in both the ξ and r directions. We compare the standard five-layer PINN with our SL-PINN. Figure 4.3 shows the numerical solutions of (4.1) with $(R^x, R^y) = (1, 4)$ and $\varepsilon = 10^{-6}$ when $f = (1 - (x/R^x)^2)^2/R^y$. The figure underscores the traditional PINN's inability to accurately approximate the solution to the singular perturbation problem. In contrast, our novel approach demonstrates superior performance. Figure 4.4 offers a detailed view of the one-dimensional profile of predicted solutions at $\xi = \pi/2$. This demonstrates the high accuracy of both the L^1 and L^2 training approaches in predicting solutions. However, the traditional PINN method falls short in capturing the sharp transition near the boundary layer.

5. Conclusion. In this study, we have introduced a semi-analytic approach to enhance the numerical performance of PINNs in tackling a range of singularly perturbed boundary value problems and convection-dominated equations on rectangular, circular, and elliptical domains. For each singular perturbation problem examined, we have derived an analytic approximation known as the *corrector* function, which captures the behavior of the fast (stiff) component of the solution within the boundary layer. By integrating these corrector functions into a 2-layer PINN framework with hard constraints, we successfully address the stiffness inherent in the approximate solutions, resulting in our novel semi-analytic SL-PINNs enriched with



(a) True solution

(b) SL-PINN using L^2 training(c) SL-PINN using L^1 training

(d) Conventional PINN

Fig. 4.3: Numerical prediction of (4.1) with $(R^x, R^y) = (1, 4)$ and $\varepsilon = 10^{-6}$ when $f = (1 - (x/R^x)^2)/R^y$. For our simulations, we select a uniform grid of 50 discretized points in each of the r and ξ directions.

the correctors. Through the incorporation of these corrector functions, we effectively overcome the stiffness challenge associated with approximate solutions, leading to the development of our improved SL-PINNs.

REFERENCES

- [1] M. AINSWORTH AND J. DONG, *Galerkin neural network approximation of singularly-perturbed elliptic systems*, Computer Methods in Applied Mechanics and Engineering, (2022), p. 115169.
- [2] A. ARZANI, K. W. CASSEL, AND R. M. D'SOUZA, *Theory-guided physics-informed neural networks for boundary layer problems with singular perturbation*, Journal of Computational Physics, 473 (2023), p. 111768, <https://doi.org/https://doi.org/10.1016/j.jcp.2022.111768>, <https://www.sciencedirect.com/science/article/pii/S0021999122008312>.
- [3] H. BARARNIA AND M. ESMAELPOUR, *On the application of physics informed neural networks (pinn) to solve*

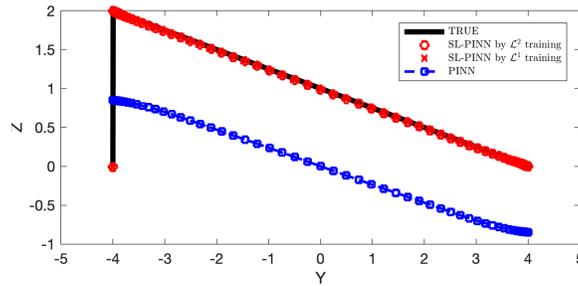


Fig. 4.4: The one-dimensional profile of predicted solutions in Figure 4.3 along the line $\xi = \pi/2$.

Relative L^2 error by the L^2 training (PINN)						
	square (Fig. 2.1)	circle with non com- patible (Fig. 3.3)	circle with compatible (Fig. 3.1)	ellipse 4:1 (Fig. 4.1)	ellipse 1:4 (Fig. 4.3)	oscillation (Fig. 3.5)
$\varepsilon = 10^{-4}$	9.55×10^{-1}	9.20×10^{-1}	9.27×10^{-1}	8.55×10^{-1}	7.84×10^{-1}	8.76×10^{-1}
$\varepsilon = 10^{-6}$	9.55×10^{-1}	9.21×10^{-1}	9.27×10^{-1}	8.65×10^{-1}	7.03×10^{-1}	9.12×10^{-1}
$\varepsilon = 10^{-8}$	9.55×10^{-1}	9.15×10^{-1}	9.20×10^{-1}	8.62×10^{-1}	7.85×10^{-1}	9.23×10^{-1}
Relative L^2 error by L^2 training (SL-PINN)						
	square (Fig. 2.1)	circle with non com- patible (Fig. 3.3)	circle with compatible (Fig. 3.1)	ellipse 4:1 (Fig. 4.1)	ellipse 1:4 (Fig. 4.3)	oscillation (Fig. 3.5)
$\varepsilon = 10^{-4}$	2.5×10^{-3}	3.2×10^{-3}	1.6×10^{-3}	5.1×10^{-3}	2.6×10^{-3}	6.5×10^{-3}
$\varepsilon = 10^{-6}$	2.0×10^{-3}	2.8×10^{-3}	1.5×10^{-3}	3.9×10^{-3}	1.7×10^{-3}	7.6×10^{-3}
$\varepsilon = 10^{-8}$	2.0×10^{-3}	4.7×10^{-3}	2.3×10^{-3}	5.8×10^{-3}	2.8×10^{-3}	9.5×10^{-3}
Relative L^2 error by L^1 training (SL-PINN)						
	square (Fig. 2.1)	circle with non com- patible (Fig. 3.3)	circle with compatible (Fig. 3.1)	ellipse 4:1 (Fig. 4.1)	ellipse 1:4 (Fig. 4.3)	oscillation (Fig. 3.5)
$\varepsilon = 10^{-4}$	9.5×10^{-5}	4.2×10^{-3}	2.6×10^{-3}	2.3×10^{-3}	3.4×10^{-3}	4.0×10^{-2}
$\varepsilon = 10^{-6}$	1.4×10^{-4}	3.6×10^{-3}	4.6×10^{-3}	5.6×10^{-3}	4.2×10^{-3}	5.2×10^{-2}
$\varepsilon = 10^{-8}$	3.9×10^{-4}	4.3×10^{-3}	1.8×10^{-3}	3.4×10^{-3}	4.8×10^{-3}	6.2×10^{-2}

Table 1: Comparison of relative L^2 errors for equations (2.1), (3.1), and (4.1) utilizing standard 5-layer PINN, SL-PINN employing L^2 training, and SL-PINN employing L^1 training.

boundary layer thermal-fluid problems, International Communications in Heat and Mass Transfer, 132 (2022), p. 105890, <https://doi.org/https://doi.org/10.1016/j.icheatmasstransfer.2022.105890>, <https://www.sciencedirect.com/science/article/pii/S0735193322000124>.

[4] H. S. CHURCHILL, *Mathematical handbook for scientists and engineers*, 1961.
 [5] S. CUOMO, V. S. DI COLA, F. GIAMPAOLO, G. ROZZA, M. RAISSI, AND F. PICCIALI, *Scientific machine learning through physics-informed neural networks: where we are and what's next*, Journal of Scientific Computing, 92 (2022), p. 88.
 [6] A. D. JAGTAP AND G. EM KARNIADAKIS, *Extended physics-informed neural networks (xpinns): A generalized space-time*

- domain decomposition based deep learning framework for nonlinear partial differential equations*, Communications in Computational Physics, 28 (2020), pp. 2002–2041, <https://doi.org/https://doi.org/10.4208/cicp.OA-2020-0164>, http://global-sci.org/intro/article_detail/cicp/18403.html.
- [7] F. FERNÁNDEZ DE LA MATA, A. GIJÓN, M. MOLINA-SOLANA, AND J. GÓMEZ-ROMERO, *Physics-informed neural networks for data-driven simulation: Advantages, limitations, and opportunities*, Physica A: Statistical Mechanics and its Applications, 610 (2023), p. 128415, <https://doi.org/https://doi.org/10.1016/j.physa.2022.128415>, <https://www.sciencedirect.com/science/article/pii/S0378437122009736>.
 - [8] A. T. A. GOMES, L. M. DA SILVA, AND F. VALENTIN, *Physics-aware neural networks for boundary layer linear problems*, 2022, <https://arxiv.org/abs/2208.12559>.
 - [9] S. GOSWAMI, C. ANITESCU, S. CHAKRABORTY, AND T. RABCZUK, *Transfer learning enhanced physics informed neural network for phase-field modeling of fracture*, Theoretical and Applied Fracture Mechanics, 106 (2020), p. 102447.
 - [10] M. HAMOUDA, G.-M. GIE, C.-Y. JUNG, AND R. TEMAM, *Singular Perturbations and Boundary Layers*, 12 2018, <https://doi.org/10.1007/978-3-030-00638-9>.
 - [11] J. HAN AND Y. LEE, *A neural network approach for homogenization of multiscale problems*, Multiscale Modeling & Simulation, 21 (2023), pp. 716–734, <https://doi.org/10.1137/22M1500903>, <https://doi.org/10.1137/22M1500903>, <https://arxiv.org/abs/https://doi.org/10.1137/22M1500903>.
 - [12] Y. HONG, C.-Y. JUNG, AND R. TEMAM, *On the numerical approximations of stiff convection–diffusion equations in a circle*, Numerische Mathematik, 127 (2014), pp. 291–313.
 - [13] Y. HONG, C.-Y. JUNG, AND R. TEMAM, *Singular perturbation analysis of time dependent convection–diffusion equations in a circle*, Nonlinear Analysis: Theory, Methods & Applications, 119 (2015), pp. 127–148.
 - [14] A. D. JAGTAP, E. KHARAZMI, AND G. E. KARNIADAKIS, *Conservative physics-informed neural networks on discrete domains for conservation laws: Applications to forward and inverse problems*, Computer Methods in Applied Mechanics and Engineering, 365 (2020), p. 113028, <https://doi.org/https://doi.org/10.1016/j.cma.2020.113028>, <https://www.sciencedirect.com/science/article/pii/S0045782520302127>.
 - [15] X. JIN, S. CAI, H. LI, AND G. E. KARNIADAKIS, *Nsfnets (navier-stokes flow nets): Physics-informed neural networks for the incompressible navier-stokes equations*, Journal of Computational Physics, 426 (2021), p. 109951.
 - [16] C.-Y. JUNG, *Numerical approximation of two-dimensional convection-diffusion equations with boundary layers*, Numerical Methods for Partial Differential Equations: An International Journal, 21 (2005), pp. 623–648.
 - [17] C.-Y. JUNG AND R. TEMAM, *Singular perturbations and boundary layer theory for convection-diffusion equations in a circle: The generic noncompatible case*, SIAM Journal on Mathematical Analysis, 44 (2012), pp. 4274–4296, <https://doi.org/10.1137/110839515>.
 - [18] G. E. KARNIADAKIS, I. G. KEVREKIDIS, L. LU, P. PERDIKARIS, S. WANG, AND L. YANG, *Physics-informed machine learning*, Nature Reviews Physics, 3 (2021), pp. 422–440.
 - [19] D. KOCHKOV, J. A. SMITH, A. ALIEVA, Q. WANG, M. P. BRENNER, AND S. HOYER, *Machine learning–accelerated computational fluid dynamics*, Proceedings of the National Academy of Sciences, 118 (2021), p. e2101784118.
 - [20] Z. LI, N. KOVACHKI, K. AZIZZADENESHELI, B. LIU, K. BHATTACHARYA, A. STUART, AND A. ANANDKUMAR, *Fourier neural operator for parametric partial differential equations*, arXiv preprint arXiv:2010.08895, (2020).
 - [21] L. LU, M. DAO, P. KUMAR, U. RAMAMURTY, G. E. KARNIADAKIS, AND S. SURESH, *Extraction of mechanical properties of materials through deep learning from instrumented indentation*, Proceedings of the National Academy of Sciences, 117 (2020), pp. 7052–7062.
 - [22] L. LU, P. JIN, G. PANG, Z. ZHANG, AND G. E. KARNIADAKIS, *Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators*, Nature Machine Intelligence, 3 (2021), pp. 218–229.
 - [23] L. LU, X. MENG, Z. MAO, AND G. E. KARNIADAKIS, *Deepxde: A deep learning library for solving differential equations*, SIAM Review, 63 (2021), pp. 208–228.
 - [24] X. MENG, Z. LI, D. ZHANG, AND G. E. KARNIADAKIS, *Ppinn: Parareal physics-informed neural network for time-dependent pdes*, Computer Methods in Applied Mechanics and Engineering, 370 (2020), p. 113250.
 - [25] K. SHUKLA, P. C. DI LEONI, J. BLACKSHIRE, D. SPARKMAN, AND G. E. KARNIADAKIS, *Physics-informed neural network for ultrasound nondestructive quantification of surface breaking cracks*, Journal of Nondestructive Evaluation, 39 (2020), pp. 1–20.
 - [26] S. WANG, S. SANKARAN, AND P. PERDIKARIS, *Respecting causality is all you need for training physics-informed neural networks*, arXiv preprint arXiv:2203.07404, (2022).
 - [27] S. WANG, H. WANG, AND P. PERDIKARIS, *On the eigenvector bias of fourier feature networks: From regression to solving multi-scale pdes with physics-informed neural networks*, Computer Methods in Applied Mechanics and Engineering, 384 (2021), p. 113938.
 - [28] L. YANG, X. MENG, AND G. E. KARNIADAKIS, *B-pinns: Bayesian physics-informed neural networks for forward and inverse pde problems with noisy data*, Journal of Computational Physics, 425 (2021), p. 109913.
 - [29] B. YU ET AL., *The deep ritz method: a deep learning-based numerical algorithm for solving variational problems*, Communications in Mathematics and Statistics, 6 (2018), pp. 1–12.
 - [30] L. YUAN, Y.-Q. NI, X.-Y. DENG, AND S. HAO, *A-pinn: Auxiliary physics informed neural networks for forward and inverse problems of nonlinear integro-differential equations*, Journal of Computational Physics, (2022), p. 111260.