Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Enhancing Neural Network Training For High Accuracy In Solving Singularly Perturbed 1D Reaction-Diffusion Equations

Lakshmi Sireesha Ch.,[1] SSVKSS Jyothiraditya[2], Shriya Vadavalli[3], Jayaditya Peddisetti[4]

[1] Assistant Professor, Department of Mathematics, CBIT Hyderabad, Telangana.
[2] IV B.Tech, Department of Electronics and Electrical Engineering, IIT Kharagpur, West Bengal.
[3] IV B.E., Department of Computer Science and Engineering, CBIT Hyderabad, Hyderabad, Telangana.
[4] IV B.E., Department of Artificial Intelligence & Machine Learning, CBIT Hyderabad, Hyderabad, Telangana.

In this paper we propose a supervised Neural Network architecture for solving Singularly Perturbed 1D Reaction-Diffusion Equations (SPRDE). By Taylor series expansion, first we transform SPRDE to a Singularly Perturbed Differential Equation (SPDE). Using the ordinary differential equation (ODE) framework, in the algorithm we formulate and train an NN architecture comprising five fully connected layers with a ReLU activation function and a Mean Squared Error (MSE) loss function. Next we employ the Adam optimizer on the model for optimization and convergence of the solutions to SPRDE. The optimizer is implemented using Tensor Flow's Neural Network architecture. This architecture helps for sequential stacking of layers thus enabling the construction of deep neural networks. The results presented in graphs, using the proposed ReLU based architecture demonstrate high accuracy of solutions.

Keywords: Singularly Perturbed 1D Reaction-Diffusion equations; Neural Networks; tf.keras.models.Sequential class; ReLU-based architecture.

1 INTRODUCTION

Singular Perturbation Problems (SPPs) frequently emerge in applied mathematics and engineering disciplines, including hydro- dynamics, chemical-reactor theory, fluid mechanics, elasticity, and reaction-diffusion processes. These problems typically involve a small parameter, ϵ, that is multiplied by the highest derivative term, leading to a solution with a multi-scale nature. This behavior arises due to the small parameter, creating distinct regions within the solution's behavior, a characteristic hallmark of singular perturbation problems. In neuroscience, Singular Perturbation Reaction-Diffusion Equations (SPRDEs) play a crucial role in modeling complex phenomena such as neural signal propagation, pattern formation in

neural tissues, and the dynamics of neural networks. The multi- scale nature of these equations allows for the accurate depiction of processes occurring at different spatial and temporal scales, providing deeper insights into the functioning of the brain and the emergence of neural patterns.
Consider the following equation:

$$v'' = \omega(u, v, \varepsilon);\ v(a) = v_a; v(b) = v_b \quad (1)$$

These are called self-adjoint singularly perturbed two-point boundary value problems or singularly perturbed 1D reaction- diffusion equations, due to the absence of the convection term. Equations of this nature possess analytical solutions that display localized regions of variation, primarily attributed to the presence of perturbation parameters. Specifically, the solution exhibits a thin transition layer(s), where it rapidly changes or experiences abrupt changes within a narrow region known as the boundary layer. Consequently, dealing with these problems is nontrivial due to the presence of the boundary layer phenomenon in their solutions. In contrast, outside the layer(s), the solution behaves irregularly, varying slowly in what is referred to as the outer region. As a result, addressing SPPs poses significant challenges that must be tackled to obtain accurate numerical solutions [5-7]. Traditional computational methods for SPRDEs prove to be inadequate as they necessitate an excessive number of mesh points to achieve satisfactory computed solutions [8-10]. Numerous researchers have made efforts to devise efficient numerical methodologies for addressing Singularly Perturbed Reaction-Diffusion Equations (SPRDEs) [11- 13]. One such approach involves employing finite difference methods with a variable mesh to overcome the challenges associated with solving singularly perturbed 1D reaction-diffusion equations. These methods utilize a mesh that is denser in the boundary layer region and coarser in the outer region, allowing for a more effective representation of the solution's behavior across different scales [5-13].

However, it is important to note that existing numerical techniques yield satisfactory outcomes only when the step size (denoted as h) is chosen to be less than or equal to the perturbation parameter ϵ. This requirement imposes a significant computational burden and prolongs the computation time. Although these numerical methods exhibit promising results, their efficiency is constrained by the necessity of adopting very small step sizes, resulting in escalated computational expenses. Consequently, this paper introduces a novel Neural Network (NN) approach tailored specifically for addressing singularly perturbed 1D reaction-diffusion equations.

Neural Networks (NNs) possess remarkable capabilities as universal approximators for continuous mappings and for approximating derivatives, as documented in [14-21]. This characteristic renders NNs highly suitable for approximating and solving various types of differential equations, including Singular Perturbation Problems (SPPs) and SPRDEs. Such approximations play a crucial role in constructing an appropriate trial solution [16].

To address the problem, we discretize the equation by trans- forming it into a minimization task, where the objective is to minimize a residual cost function. By employing an NN to approximate the solution, we ensure that it satisfies the equation within a given grid. Consequently, NNs can be trained to approximate solutions to Singularly Perturbed 1D

Reaction-Diffusion equations. The resulting solutions exhibit closed analytic forms and demonstrate excellent generalization properties. Furthermore, these NN solutions are differentiable, enabling accurate approximations of points located between grid steps without compromising precision or necessitaing additional computational expenses.
This paper extends and refines the concepts introduced in previous works [14-16] and introduces a novel neural network adaptive mesh training algorithm for test problems described in the literature [21]. The primary objective of our algorithm is to generate solutions that are uniformly accurate and continuous for the SPPs considered in this study. What distinguishes our approach is the direct integration of ϵ within the network architecture, significantly enhancing its ability to tackle these challenging problems.

1.1 Neural Networks and Fundamental Training Process

1.1.1 Selection of Hyperparameters

Achieving accurate and efficient solutions for singularly perturbed 1D reaction-diffusion equations using neural networks hinges greatly on selecting appropriate hyperparameters. Therefore, it is imperative to carefully consider parameters such as optimizers, the number of layers, and activation functions to refine the neural net- work training process. In this segment, we determine optimal hyperparameters through experimental findings. We fixed the number of neurons per layer to 5,in addition to fixing the initial weights and bases in different experiments, to ensure uniform observations.

1.1.2 Number of Layers

We carried out different experiments on wide varieties of layer configuration NNs to ensure that they fit well into the intricate regions of SPRDEs as well as to suit the challenges posed by vanishing/exploding gradients in NN architecture. Here the number of layers under consideration act as critical hyperparameters that influence the model's computational efficiency, in addition to representational capacity. In this paper our aim is to maintain a balance between model complexity and computational resources, allowing deeper networks to capture intricate patterns while maintaining training stability.

1.1.3 Optimizers

In training a NN, optimizers play a key role in dictating how the neural network updates its parameters to reduce the loss function. In our proposed study, we observed that the choice of an optimizer is very crucial, as it significantly influences both the speed of convergence and the model's ultimate accuracy. To compare the four optimizers, namely the Stochastic Gradient Descent (SGD), Adam, RMSprop, and LBFGS, we conducted empirical tests and observed that the Adam optimizer resulted in a relatively low loss and did not achieve the best possible outcome. Therefore, by combining the LBFGS and Adam optimizers, we implemented a hybrid strategy where one iteration of LBFGS was carried for every 100 iterations of Adam.

1.2 Modeling a Problem

Consider the self-adjoint second-order singularly perturbed 1D reaction-diffusion equations (SPRDEs) of the following type:

$$Lv(x) \equiv -\varepsilon v'' + \varphi(u);\ v = f(u); 0 \le u \le 1 \qquad (2)$$

subject to boundary conditions:

$$v(0) = \gamma_1;\ \ v(1) = \gamma_2 \qquad (3)$$

In this context, $f(u)$ is a smooth function and $\varphi(u) \geq \varphi > 0$ γ_1 and γ_2 and φ are constants, and ε is a small parameter within the interval $(0,1)$.

1.3 The torch. nn linear class for the proposed NN Technique:

In the proposed NN architecture, a linear class, namely the torch.nn from PyTorch is used to construct and train the NNs. The motive of anchoring torch.nn to the NN architecture is to optimize the parameters in NN. This optimization in turns aims to minimize composite loss function, encompassing the residual loss of Singularly Perturbed Ordinary Differential Equation(SPODE) with associated boundary conditions. This loss quantifies the disparity between the predicted and actual solution values, enabling the network to refine its predictions through iterative adjustments of the parameters. This process benefits significantly from automatic differentiation using intermediate symbols, circumventing the expression swell problem encountered with traditional differentiation methods.

Initially, random weights are assigned to the nodes of the NN. To properly compute the loss function, access to the output of higher-order derivatives with respect to the input is essential. For this purpose, we utilize a finite-difference approximation with an adaptive step size. As the step size h approaches zero, the slope of the secant line increasingly approximates the slope of the tangent line. For our experiments, we selected a step size of $h = e^{-100}$, which provided a balance of high accuracy and effective convergence rates.

During the training process, we employ a combination of LBFGS and Adam optimizer to optimize the parameters of the NN. The choice of the combined optimizer is based on the adaptive learning rate and efficient convergence properties of the Adam optimizer, which contribute to improved training performance, while the LBFGS optimizer is beneficial when an oscillating loss problem occurs. The loss function, is composed of the residual loss of the SPRDE and the loss of boundary conditions.

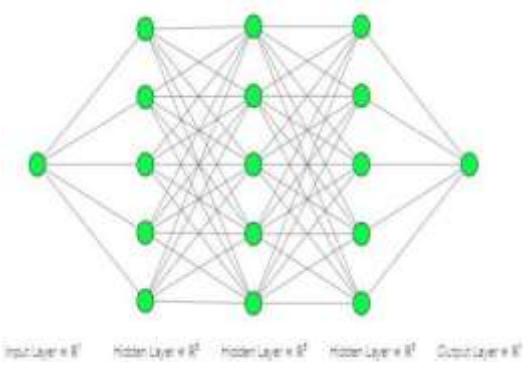

Figure 1. The proposed Neural Network Architecture.

Quantifies the disparity between the predicted solutions and the true solutions. This facilitates effective measurement of the deviation between the predicted values, guiding the network's parameter adjustments towards minimizing this discrepancy. To generate the training data, the independent variable 'u' is discretized, i.e., the interval [0, 1] is discretized. This involves creating a numpy array named 'utrain', comprising 1000 equally spaced points within the range of 0 to 1. The corresponding approximate values are obtained by evaluating the provided Equation(x) function using the 'utrain' values. For each iteration, the composite loss is calculated. If the loss exceeds a certain criterion, the weights of the model are adjusted using backpropagation. This process ensures convergence of the overall equation along with the boundary. Subsequently, the model is compiled, incorporating the Adam optimizer and the loss function. To commence the training process, we utilized the 'loss.backward()' method along with the 'optimizer.step()' method, which iteratively updated the NN weights based on the provided training data ('utrail', 'vtrail') for a specified number of epochs.

By adapting internal weights, the NN architecture approximates the solution of SPRDE. To ensure consistency and fair comparison between different experimental setups, we have chosen a mesh with exact number of data points, as the top-performing NN. Also we standardize the base line for NN by matching the primary mesh to the training set. This in turn allows precise performance evaluations and comparisons between different experiments and techniques. Figure 1 is the graphical representation of trial solution with the proposed NN architecture. From Fig 1 it is obvious the contribution of perturbation is almost nil, as the training set is bound to the solution grid points. We also observe that concerns regarding privacy and security of the data sets is insignificant.

2 NUMERICAL EXPERIMENTS

To validate the proposed NN algorithm, we conducted numerical experiments on two well-established test problems sourced from existing literature [14]. These test problems are widely recognized and serve as standard benchmarks for evaluating the performance and efficacy of various algorithms, including the proposed NN approach. By applying the NN algorithm to these test problems, we aim to assess its effectiveness and compare its results against existing approaches, thus demonstrating its potential and robustness in solving real-world problems.

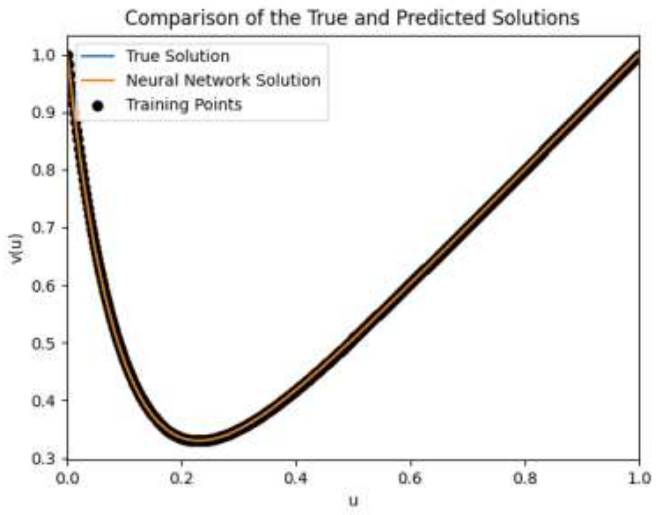

Figure 2. Comparison between True and Predicted

2.1 Test Problem-1:

Consider the SPRDE

$$\varepsilon v'' - v + u = 0$$

subject to the boundary conditions:

$$v(0) - v(0,\varepsilon) = 0;\ v(1) - \left(1 + e^{-\left(\frac{1}{\sqrt{\varepsilon}}\right)}\right) = 0$$

True Solution:

$$v(u,\varepsilon) = u + e^{-\varepsilon\sqrt{x}}.$$

In our approach, the initial mesh is intentionally chosen to have the same number of points as the training set of the best-performing neural network (NN) in each scenario. This selection ensures consistency and facilitates fair comparisons between different scenarios or experimental setups. By aligning the initial mesh with the training set, we establish a standardized baseline for the torch.nn linear class for the proposed NN Technique.

2.2 Test Problem-2:

$$\varepsilon \frac{d^2y}{dx^2} + y + \cos^2(\pi x) + 2\varepsilon\pi^2\cos(2\pi x) = 0$$

Boundary Conditions:

$$y(0) = 0;\ y(1) = 0$$

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

Fig. 3 illustrates the results obtained using the proposed NN technique for the problem 2.

3 DISCUSSIONS

Selecting the right neural network architecture is essential for accurately approximating the solution of singularly perturbed reaction-diffusion equations (SPRDEs). The number of layers and units in the network significantly influences its ability to capture complex patterns and variations in the input-output relationship. A deeper network with more layers enhances its capacity to model the intricate dynamics inherent in the SPRDE solution, leading to better accuracy and predictive performance.

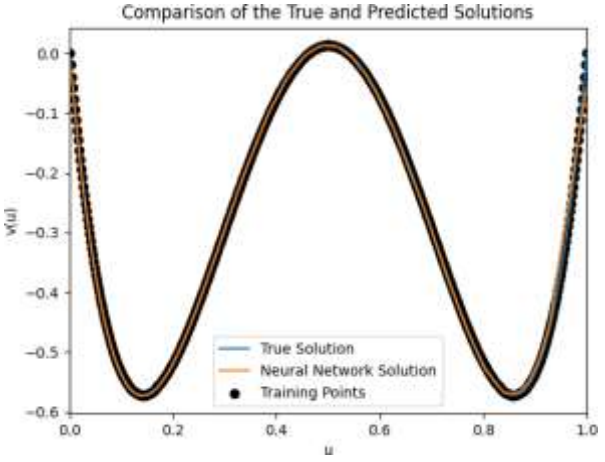

Fig-3

To introduce non-linearity, the hyperbolic tangent (tanh) activation function was applied in each layer. The tanh function allows the model to learn and represent complex functions effectively, helping to mitigate the vanishing gradient problem and improve the network's ability to capture the intricate dynamics of the reaction-diffusion problem's solution.

Additionally, the Adam optimizer was chosen for its adaptive learning rate and efficient optimization properties. By dynamically adjusting the step size for each weight update, the Adam optimizer mitigates challenges related to vanishing or exploding gradients, leading to improved convergence and training performance. Incorporating the tanh activation function and the Adam optimizer enables the network to handle non-linear relationships more effectively, enhancing its learning capacity and improving its ability to approximate the SPRDE solution accurately.

During the training process, the Adam optimizer continually minimizes the loss between predicted and true solution values. Through iterative weight adjustments, the network refines its internal representation, progressively enhancing its ability to predict the singular

perturbation problem solution accurately. This optimization objective ensures that the model effectively learns to approximate the true solution of the considered problem.

To evaluate the performance of the neural network, line plots were generated to visualize the actual and predicted solution curves. Comparing the predicted solution curve with the true solution curve allows for the assessment of approximation accuracy against existing literature references. The visual assessment provides insights into the effectiveness of the trained neural network in capturing underlying patterns and dynamics of the SPRDE solution.

The neural network employed in this study offers several ad- vantages, including its ability to capture nonlinear dynamics, universal function approximation capabilities, automatic feature ex- traction, parallel processing efficiency, generalization to unseen data, and end-to-end learning. These advantages contribute to the effectiveness of the neural network in accurately approximating the solution of the SPRDE, capturing intricate patterns and dynamics without requiring explicit knowledge of the mathematical form of the SPRDE.

4 CONCLUSION

This study introduces a physics-informed loss-based technique for solving singularly perturbed 1D reaction-diffusion equations (SPRDEs). The neural network, coupled with the training process, has proven to be a robust and effective approach for approximating the solution of the given SPRDE. Through careful design of the network architecture, utilization of non-linear activation functions, and application of an efficient optimization algorithm, the model successfully learns to approximate the SPRDE solution within the defined number of training epochs. The discretization of the domain and application of the physics-informed loss technique are pivotal factors contributing to the success of our proposed technique.

To validate the applicability of our approach, we considered two model examples from existing literature and plotted the graphical solutions. The results demonstrate that the proposed neural network technique accurately approximates the exact solution.

Compared to existing methods, our technique exhibits higher accuracy in approximating the SPRDE solution. Moreover, the technique shows potential for extension to handle SPRDEs with variable coefficients, enhancing its versatility and applicability in various scenarios. Overall, the physics-informed loss-based technique presented in this study offers a promising and effective approach for solving self-adjoint singularly perturbed 1D reaction-diffusion equations. It demonstrates the capability to accurately approximate solutions, outperforming existing methods, and holds potential for further advancements in solving SPRDEs with varying coefficients.

5 DECLARATION OF INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

6 DATA AVAILABILITY

This research doesn’t contain any data, figures, or tables from a third party. The data used in the manuscript are our own and the data will be made available on request.

7 REFERENCES

1. Derstine, M.W., Gibbs, F.A.H.H.M., Kaplan, D.L.: Bifurca- tion gap in a hybrid optical system. Phys. Rev. A 26, 3720–3722 (1982).
2. Longtin, A., Milton, J.: Complex oscillations in the human pupil light reflex with mixed and delayed feedback. Math. Biosci. 90, 183–199 (1988).
3. Mackey, M.C., Glass, L.: Oscillation and chaos in physiological control systems. Science 197, 287–289 (1977)
4. Mallet-Paret, J., Nussbaum, R.D.: A differential-delay equations arising in optics and physiology. SIAM J. Math. Anal. 20, 249–292 (1989).
5. Farell,P.A., Hegarty,A.F., Miller, J.J.H., O'Riordan, E. Shishkin, G. I.:Robust Computational Techniques for Boundary Layers, Chapman and Hall/CRC Press (2000)
6. Roos, H.G., Stynes, M., Tobiska L.: Robust Numerical Methods for Singularly Perturbed Differential Equations,SpringerVerlag, Berlin, Heidelberg. (2008)
7. Amiraliyev GM, Cimen E: Numerical method for a singularly perturbed convection–diffusion problem with delay. Appl Math Comput 216:2351–2359 (2010).
8. Doolan, E.P., Miller, J.J.H., Schilders, W.H.A.: Uniform Numerical Methods for Problems with Initial and Boundary Layers. Boole Press, Dublin. (1980).
9. Khuri, S.A., Sayfy, A.,: Self-adjoint singularly perturbed second order two – point boundary value problems: a patching approach. Appl. Math. Model. 38 (11), 2901–2914. (2014)
10. Kadalbajoo, M.K., Reddy, Y.N.: Asymptotic and numerical analysis of singular perturbation problems. a survey, Appl. Math. Comput. 30 223–259 (1989).
11. Mohanty, R.K., Jha, N.A.,: Class of variable mesh spline in compression methods for singularly perturbed two-point singular boundary-value problems. Appl. Math. Comput. 168 (1), 704–716 (2005).
12. Mishra, H. K., Kumar, M., Singh, P.: Initial-value technique for self-adjoint singular perturbation boundary value problems, Computational Mathematics and Modeling, 20(2) 336-347 (2009)
13. Veerasamy, S. Natesan Srinivasan:, Robust numerical method for space shift 2D singularly perturbed parabolic convection diffusion differential equations, https://doi.org/10. 1007/s40314-023-02289-8 (2023)
14. Cybenko G (1989) Approximation to superpositions of a sigmoidal function. Math Control Sig Syst 2:303–314
15. Lagaris IE, Fotiadis A, Likas DI, 1998. Approximation capabilities of multilayer feedforward networks. Neural Net 4(2):251–257 13.
16. L. S. Challa(2017) Non Standard Finite Difference Method for Singularly Perturbed Singular Two Point Boundary Value Problem using Non-Polynomial spline, WSEAS Transactions on Computer Research 5: 130-136.
17. Kurthornik, 1991:, Approximation Capabilities of Muitilayer Feedforward Networks Neural Networks, Vol. 4, pp. 251-257.
18. Kaloutsa V, Famelis ITh, 2020: On the neural network solution of stiff initial value problems. AIP Conf Proc 2293:420018. https://doi.org/10.1063/5.0026823
19. Famelis ITh, Kaloutsa V, 2021: Parameterized neural network training for the solution of a class of stiff initial value systems. Neural Comput Appl 33:3363–3370.
20. Simos T. E., Ioannis, Famelis, Th., 2022: A neural network training algorithm for singular perturbation boundary value problems, Neural Computing and Applications,607-615.
21. Fasika Wondimu Gelu, Gemechis File Duressa, Tesfaye Aga Bullo: Sixth-order compact finite difference method for singularly perturbed 1D reaction diffusion problems Journal of Taibah University for Science(11), 302-308. (2017)
22. Yang, Q., Sadeghi, A., Wang, G., & Sun, J., Learning two-layer ReLU networks is nearly as easy as learning linear classifiers on separable data IEEE Transactions on Signal Processing, 69, 4416-

4427. (2021)