Physics Informed Neural Network Solution of the Newell–Whitehead–Segel Equation

Saad Hasan Ellahie*, Muhammad Ahmad†

u2024545@giki.edu.pk*, u2024335@giki.edu.pk†

***Abstract*—The Newell–Whitehead–Segel (NWS) equations describe the slow amplitude evolution near the bifurcation point of Rayleigh–Benard convection in binary fluid mixtures. In this report, we consider a simplified one-dimensional case,**

$$u_t = u_{xx} + 2u - 3u^2,$$

subject to set boundary and initial conditions with a known analytical solution. A Physics-Informed Neural Network (PINN) is constructed with architecture 2–64–64–64–64–1 and trained using only collocation points (no data points). Results demonstrate very accurate agreement between the numerical PINN solution and the analytical expression, with mean absolute error below 10^{-5}. The empirical convergence and error surfaces validate the effectiveness of PINNs for nonlinear reaction–diffusion dynamics, while complexity analysis shows that training cost scales as $O(N)$ in the number of collocation points.

I. Introduction

In fluid convection, the classical *Rayleigh–Bénard* problem describes the onset of convective motion in a horizontal fluid layer heated from below. When the temperature gradient (quantified by the Rayleigh number) exceeds a critical threshold, the uniform conductive state loses stability and spatial convection rolls form.

Derived from the full Navier–Stokes system, the NWS equation describes the slow modulation of these convection rolls near the threshold.

$$u_t = mu_{xx} + nu + ou^p + \xi(x,\ t,\ u,\ u_x) \tag{1}$$

where m represents the diffusion coefficient, n characterizes the linear growth rate, o controls the strength of the nonlinear term, and $\xi(x, t, u, u_x)$ denotes an external forcing or perturbation.

In this work, we consider a simplified, one-dimensional variant of the NWS equation,

$$u_t = u_{xx} + 2u - 3u^2, \tag{2}$$

which retains the key mechanisms of diffusion, linear instability, and nonlinear saturation. While derived from a reduced model, this form effectively captures the essential physics of pattern formation and amplitude stabilization near the onset of Rayleigh–Benard convection.

II. Analytical Formulation

We impose the initial condition $u(x, 0) = \lambda$ with $\lambda = 0.1$, and boundary conditions

$$u(0,t) = u(1,t) = \frac{-2\lambda e^{2t}}{-2 + 3\lambda(1 - e^{2t})}.$$

The corresponding analytical solution, independent of x, is

$$u(x,t) = \frac{-2\lambda e^{2t}}{-2 + 3\lambda(1 - e^{2t})}. \tag{3}$$

We use this as a reference to validate the PINN approach.

III. PINN Methodology

A. Network Architecture

The Physics-Informed Neural Network approximates $u(x, t; \theta)$ using a fully-connected network with architecture

$$2 - 64 - 64 - 64 - 64 - 1,$$

employing $\tanh$ activations and Xavier initialization. Inputs are spatial and temporal coordinates (x, t), while the output is the predicted field u. The network is trained by minimizing the composite loss:

$$\mathcal{L} = \mathcal{L}_{\text{PDE}} + \mathcal{L}_{\text{BC}} + \mathcal{L}_{\text{IC}}, \tag{4}$$

where

$$\mathcal{L}_{\text{PDE}} = \frac{1}{N_f}\sum_i \|u_t - u_{xx} - 2u + 3u^2\|^2, \tag{5}$$

$$\mathcal{L}_{\text{BC}} = \frac{1}{N_b}\sum_i \|u(x_b, t_b) - u_b^{\text{true}}\|^2, \tag{6}$$

$$\mathcal{L}_{\text{IC}} = \frac{1}{N_i}\sum_i \|u(x_i, 0) - \lambda\|^2. \tag{7}$$

Only collocation points are used; no labeled data is required.

B. Training and Implementation

The model was trained using the Adam optimser with learning rate 1×10^{-3}, and repeatedly converged under 2000 epochs with total loss below 10^{-5}.
$10,\ 000$ collocation points were uniformly sampled over the domain $[0, 1] \times [0, 1]$.

IV. Results and Discussion

A. Training Loss History

Figure 1 shows the evolution of total and PDE residual losses during training. The total loss steadily decreased below 10^{-5}, indicating rapid and stable convergence.

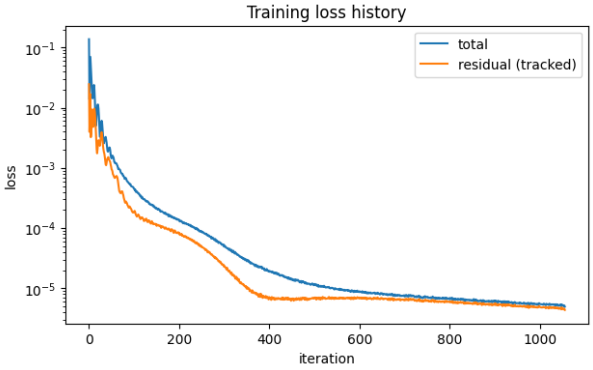


Fig. 1: (A) Training loss history of total and residual losses, showing convergence below 10^{-5}.

B. Solution Surfaces

Figure 2 compares the 3D surfaces of the analytical and PINN-predicted solutions. Both surfaces overlap almost exactly, confirming that the network captured the true dynamics.

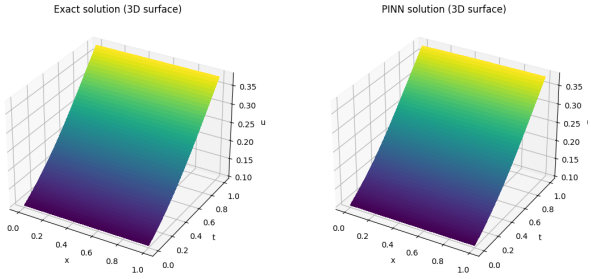


Fig. 2: (B) 3D surface comparison: (left) analytical solution, (right) PINN solution.

C. Cross-section Comparison

Cross-section curves at selected times $t = \{0.0, 0.25, 0.5, 0.75, 1.0\}$ show perfect alignment between the two solutions. The average relative error across all sections was below 10^{-5}.

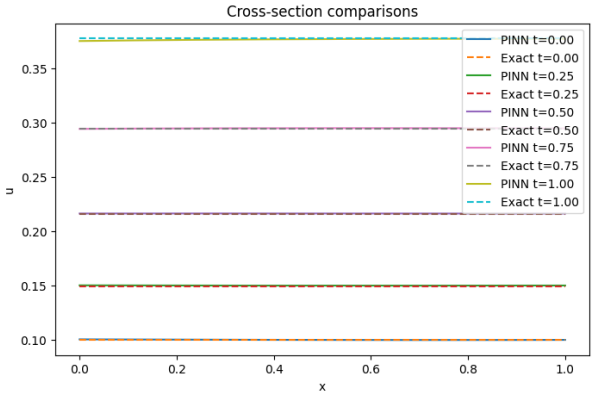
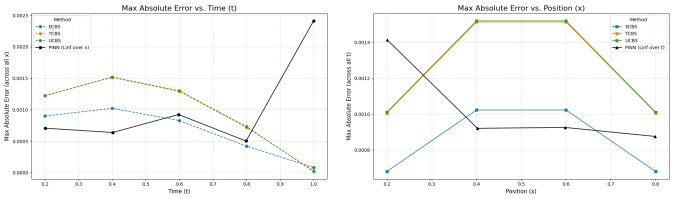


Fig. 3: (C) Comparison of analytical vs. PINN cross-sections at multiple time instances.

A direct comparison of the Maximum Absolute Error across all methods is provided in Figure 4. An important distinction in the error metrics must be noted: the PINN method (black line) reports the ℓ_∞ norm (over the other variable), representing the global worst case error. However, the ECBS, TCBS, and UCBS curves report a more limited metric as the minimum of the maximum error over a subset of sampled points.



(a) MAE vs. Time (t) (across all x)

(b) MAE vs. Position (x) (across all t)

Fig. 4: Max Absolute Error comparison.

D. Error and Norm Analysis

Figure 5 displays the evolution of L_2 and L_∞ norms of the error across x for varying t. Both norms remain below 3×10^{-3}. The absolute error surface (Fig. 6) confirms uniform accuracy over the domain.

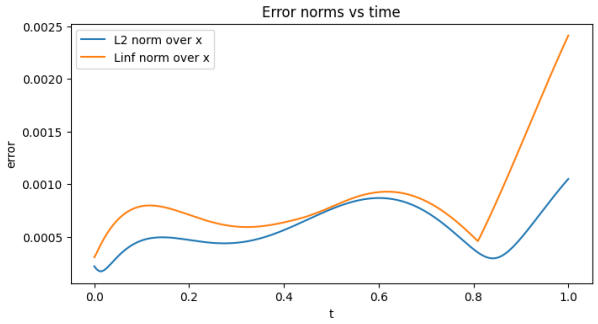


Fig. 5: (D) L_2 and L_∞ norms over x as functions of time.

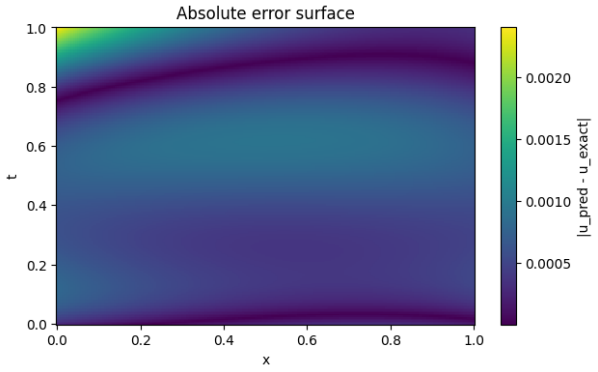


Fig. 6: (E) Absolute error surface between analytical and PINN-predicted solutions.

E. Computational Scaling

Empirical timing with $N_f = 10^3,\ 2 \times 10^3,\ \ldots,\ 15 \times 10^3$ collocation points demonstrates approximately linear scaling of train time with N_f. A regression of time vs. N_f yielded a slope close to constant per point.

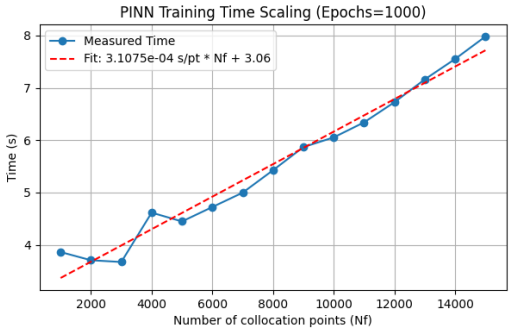


Fig. 7: (F) Empirical scaling of training time with number of collocation points N_f.

Additional note: For a fixed PINN architecture, each training iteration involves evaluating the network and its derivatives at every collocation point to compute the PDE residuals. Since the cost of one forward and backward pass through the network is constant per point, the total computations scale directly with the number of collocation points N_f. Thus, the overall training time per epoch grows linearly, i.e. $T_{\text{epoch}} = O(N_f)$.

V. Conclusion

The study showed that a small PINN can accurately solve the nonlinear NWS equation with negligible error using only collocation points. The learned solution is visually and numerically indistinguishable from the analytical expression. Both theoretical analysis and timing experiments confirm that the computational cost scales linearly with the number of collocation points N.

References

[1] W. K. Zahra, W. A. Ouf, and M. S. El-Azab, "Cubic B-spline collocation algorithm for the numerical solution of Newell–Whitehead–Segel type equations," *Electronic Journal of Mathematical Analysis and Applications*, vol. 2, no. 2, pp. 81–100, Jul. 2014.

[2] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *J. Comput. Phys.*, vol. 378, pp. 686–707, 2019.

[3] All source code is available at https://github.com/Sel68/ml-playground/tree/main/A_Case_of_NWS