Certifying Physics-Informed Neural Networks through Lower Error Bounds

Arzu Ahmadova* **Ismail Huseynov**† **Agamirza Bashirov**‡

Abstract

Physics-informed neural networks (PINNs) bring together machine learning and physical laws to solve differential equations. Although Hillebrecht and Unger (2022) provide rigorous upper error bounds for PINN prediction error under Lipschitz continuity conditions, certification requires complementary lower bounds to establish complete error enclosures. In this work, we obtain computable a posteriori lower bounds for PINN errors in ordinary differential equations (ODEs) under strong monotonicity conditions without prior knowledge on the true solution. This work gives fully certified a posteriori error bands for nonlinear ODEs and for linear ODEs satisfying structural assumptions, providing robust error enclosures computed after training.

1 Introduction

The certification of machine learning methods has emerged as a critical challenge in scientific computing, particularly for physics-informed neural networks (PINNs) where reliability guarantees are essential for deployment in safety-critical applications. PINNs incorporate physical constraints directly into the learning objective through differential operators, enabling them to leverage neural network expressivity while respecting physical consistency. This hybrid approach facilitates accurate predictions even with limited or noisy data by constructionally enforcing physical feasibility [10].

Certified error estimation for PINNs builds upon foundational developments in scientific machine learning [12]. Existing approaches encompass statistical uncertainty quantification and a deep confidence framework [4], dual-weighted residual methods [11], and goal-oriented error estimation [14], though these typically provide either statistical rather than rigorous bounds or require substantial training data. Drawing from certification frameworks in reduced-order modeling [7] and classical stability theory for ordinary differential equations [6], this work establishes rigorous a posteriori error bounds that remain valid for unseen data without requiring knowledge of the true solution. Theoretical analyses by [5] further advance PINN reliability by deriving rigorous, dimension-robust a priori approximation error bounds for both solution and operator-learning networks.

While upper error bounds estimate the proximity of the exact and approximate solutions, lower error bounds estimate the difference of them. Complete certification requires both upper and lower error enclosures to provide tight bounds on prediction quality. However, the complementary problem of establishing rigorous lower bounds remains largely unexplored. [8] has demonstrated rigorous a posteriori upper bounds for PINN errors in ODEs under Lipschitz continuity assumptions, however, the derivation of equally rigorous lower bounds remains an open challenge. Therefore, we present this problem to derive lower error bounds under the strong monotonicity assumptions for ODE-based PINNs. We complement *worst-case Lipschitz-based upper bounds* with *best-case strong-*

*Faculty of Mathematics, University of Duisburg-Essen, Essen, Germany, `arzu.ahmadova@uni-due.de`

†Mathematical Modelling&Data Analysis, National Metrology Institute of Germany, Berlin, Germany, `ismail.huseynov@ptb.de`

‡Department of Mathematics, Eastern Mediterranean University, Famagusta, T.R. Northern Cyprus, `agamirza.bashirov@emu.edu.tr`

Submitted to 1st Workshop on Differentiable Systems and Scientific Machine Learning @ EurIPS 2025.

monotonicity-based lower bounds, establishing *complete certified error intervals* for PINNs, while achieving sharper results for linear systems through matrix exponential eigenvalue analysis (e.g., exponential growth and decay rates).

These certified error intervals provide mathematically guaranteed enclosures for the approximation error, where a narrow interval indicates high confidence in the error estimation, while a wide interval primarily reflects limitations in the trained model's ability to satisfy the underlying physics. The interval width therefore serves as a direct quantitative measure of certification tightness, guiding both theoretical analysis and practical implementation decisions.

The paper is organized in the following way. Section 2 introduces the problem setup and strong monotonicity framework. Section 3 presents our main theoretical contributions: a posteriori lower bounds for nonlinear systems under strong monotonicity and improved bounds for linear systems. Section 4 discusses numerical implementation aspects, including Simpson's rule integration. Section 5 demonstrates different numerical examples.

We employ standard mathematical notation throughout, with $\mathbb{T} = [0, T]$ with $T > 0$: denoting the time domain, $\|\cdot\|$ the Euclidean 2-norm. We emphasize that in the theoretical part, any other norm can be used. If the symbol $\|\cdot\|$ is used on a set it describes the number of elements in this set.

2 Problem Description

We consider the initial value problem on the time interval $\mathbb{T}$:

$$\begin{cases} \dot{\mathbf{x}}(t) = f(t, \mathbf{x}(t)), \quad t \in \mathbb{T}, \\ \mathbf{x}(0) = \mathbf{x}_0, \end{cases} \tag{1}$$

where $f : \mathbb{T} \times \mathbb{R}^n \to \mathbb{R}^n$ is a continuous function and $\mathbf{x}_0 \in \mathbb{R}^n$. There are different theorems stating the existence and uniqueness of the solution of (1). The Picard–Lindelöf theorem [1] states the existence and uniqueness under additional Lipschitz continuity condition on f in its second variable with the Lipschitz constant independent on its first variable. The Brouwer's fixed-point theorem guarantees the existence and uniqueness under additional coercivity and strong monotonicity conditions [13]. Since the strong monotonicity condition plays a crucial role in derivation of the lower bound process, we present this condition as existence of a constant $\lambda > 0$ independently on t such that

$$(f(t, \mathbf{u}) - f(t, \mathbf{v})) \cdot (\mathbf{u} - \mathbf{v}) \geq \lambda \|\mathbf{u} - \mathbf{v}\|^2 \text{ for all } \mathbf{u}, \mathbf{v} \in \mathbb{R}^n. \tag{2}$$

Under the conditions of either of these theorems, a unique exact solution exists and, therefore, we can define the following flow map:

$$\varphi : \mathbb{T} \times \mathbb{R}^n \to \mathbb{R}^n \tag{3}$$

that maps the time $t \in \mathbb{T}$ and the initial value $\mathbf{x}_0 \in \mathbb{R}^n$ to the solution at time t, i.e., $\mathbf{x}(t) = \varphi(t, \mathbf{x}_0)$. In most applications, an explicit expression for φ is not available and evaluation of φ is only possible via suitable approximation techniques. We will learn approximate solution via physics-informed machine learning (ML) algorithms.

The learning process for initial value problem (1) is a two-step process. At first, a suitable ML candidate function:

$$\hat{\varphi} : \mathbb{T} \times \mathbb{R}^n \times \mathbb{R}^k \to \mathbb{R}^n, \quad (t, \mathbf{x}_0, \boldsymbol{\omega}) \mapsto \hat{\varphi}(t, \mathbf{x}_0, \boldsymbol{\omega}) \tag{4}$$

with parameter vector $\boldsymbol{\omega} \in \mathbb{R}^k$ is defined. In our case, $\hat{\varphi}$ is an activating function encoding the number of layers and neurons while $\boldsymbol{\omega}$ represents the weights and bias for the network. Our standing assumption throughout the paper is that the candidate function $\hat{\varphi}$ is sufficiently smooth. Secondly, we find $\boldsymbol{\omega}^*$ such that $\hat{\varphi}(\cdot, \cdot, \boldsymbol{\omega}^*)$ is a good approximation to flow map (3) by minimizing a suitable loss function $\mathcal{L} : \mathbb{R}^k \to \mathbb{R}$. Defining $\hat{\mathbf{x}} := \hat{\varphi}(\cdot, \mathbf{x}_0, \boldsymbol{\omega}^*)$, the ML prediction error is then given as:

$$\boldsymbol{\varepsilon}(t) := \mathbf{x}(t) - \hat{\mathbf{x}}(t) \tag{5}$$

for $t \in \mathbb{T}$. The second step typically relies on existing data, which in our case corresponds to evaluations of $\mathbf{x}$ on a discrete subset of $\mathbb{T}$.

Unlike the Lipschitz condition used for upper error bounds in [8], we employ the strong monotonicity for lower error bounds stated in (2). This condition implies a contractive structure sufficient to control the error dynamics from below. Thus, our main problem is as follows:

Main Problem: *Rigorously quantify the ML prediction error* (5) *from below for any $t \in \mathbb{T}$ at low computational cost without computing the true solution.*

3 A Posteriori Error Estimation: Lower Bounds

Assume that we have already trained our network and that for some initial value $\mathbf{x}_0 \in \mathbb{R}^n$ we have computed the machine learning approximation $\hat{\mathbf{x}}$ of the initial value problem (1). Since our candidate function $\hat{\varphi}$ is assumed to be smooth, we can define the residual[4]

$$\mathcal{R}_{\hat{\varphi}}(t) := \dot{\hat{\mathbf{x}}}(t) - f(t, \hat{\mathbf{x}}(t)). \tag{6}$$

Herein, the time derivative may be computed efficiently via *automatic differentiation*, see for instance [3]. With these preparations, we are now ready to formulate our first main result towards the solution of Main Problem.

Theorem 1. *Suppose f in* (1) *satisfies the strong monotonicity condition from* (2) *and the ML candidate function $\hat{\varphi}$ is sufficiently smooth. For any continuous function $\delta\colon \mathbb{T} \to \mathbb{R}_+$ with*

$$\|\mathcal{R}_{\hat{\varphi}}(t)\| \le \delta(t) \tag{7}$$

define

$$I(t, \delta) := \int_0^t \mathrm{e}^{\lambda(t-s)} \delta(s)\,\mathrm{d}s, \tag{8}$$

where λ is the strong monotonicity constant from (2)*. Then the ML prediction error* (5) *satisfies*

$$\|\boldsymbol{\varepsilon}(t)\| \ge \mathrm{e}^{\lambda t}\|\boldsymbol{\varepsilon}(0)\| - I(t, \delta). \tag{9}$$

3.1 Linear Systems and Improved Bounds

For linear time-invariant systems where the right-hand side of (1) is given by

$$f(t, \mathbf{x}) = A\mathbf{x} \tag{10}$$

for some matrix $A \in \mathbb{R}^{n \times n}$, we can derive improved bounds by leveraging the matrix exponential structure and matrix-measure inequalities.

Lemma 2. *For the linear map $f(\mathbf{x}) = A\mathbf{x}$, strong monotonicity condition* (2) *is equivalent to $\frac{A+A^\top}{2} \succeq \lambda I$. The largest admissible λ equals $\lambda_{\min}\left(\frac{A+A^\top}{2}\right)$.*

Theorem 3. *Suppose that the right-hand side of* (1) *is linear, i.e., given by* (10)*, and $\hat{\varphi}$ is sufficiently smooth. Assume there exists $\delta > 0$ such that*

$$\|\mathcal{R}_{\hat{\varphi}}(t)\| \le \delta \quad \textit{for all } t \in [0, T].$$

Then, the ML prediction error (5) *satisfies the following bounds:*

$$\|\boldsymbol{\varepsilon}(t)\| \ge \begin{cases} e^{m(A)t}\|\boldsymbol{\varepsilon}(0)\| - \frac{\delta}{\omega(A)}\left(e^{\omega(A)t} - 1\right), & \textit{if} \quad \omega(A) \neq 0, \\ e^{m(A)t}\|\boldsymbol{\varepsilon}(0)\| - \delta t, & \textit{if} \quad \omega(A) = 0. \end{cases} \tag{11}$$

where $m(A)$ and $\omega(A)$ are the smallest and largest eigenvalues of $\frac{A+A^\top}{2}$, respectively:

$$m(A) := \lambda_{\min}\left(\tfrac{A+A^\top}{2}\right), \qquad \omega(A) := \lambda_{\max}\left(\tfrac{A+A^\top}{2}\right).$$

4 Application to Physics-Informed Neural Networks

PINNs incorporate physical laws directly into machine learning [10]. They achieve this by adding terms to the loss function that require the network to satisfy the governing ODE (1). This physical loss component called the *physical loss* is used as an analytical tool in our error estimation approach.

[4]We use the notation $\mathcal{R}_{\hat{\varphi}}(t)$ to indicate that this is the residual for the ML candidate function $\hat{\varphi}$, which additionally depends on the parameter $\boldsymbol{\omega}$.

When experimental data is unavailable, the neural network $\hat{\varphi}(t, \mathbf{x}_0, \boldsymbol{\omega})$ must learn system dynamics purely from physical laws. The network approximates the system behavior by satisfying the differential equation throughout the domain.

We define collocation points $Y_{\text{coll}} \subset \mathbb{T} \times \mathbb{R}^n$, where each point $y = (t, \mathbf{x}_0)$ contains a time and initial condition. The physics loss measures how well the network satisfies the ODE:

$$\mathcal{L}_{\text{physics}}(\boldsymbol{\omega}) = \frac{1}{|Y_{\text{coll}}|} \sum_{y \in Y_{\text{coll}}} \left\| \frac{\partial \hat{\varphi}}{\partial t}(y.t, y.\mathbf{x}_0, \boldsymbol{\omega}) - f(y.t, \hat{\varphi}(y.t, y.\mathbf{x}_0, \boldsymbol{\omega})) \right\|^2 . \tag{12}$$

To ensure unique solutions, we enforce initial conditions through a separate loss term. For initial conditions $Y_{\text{ic}} = \{(0, \mathbf{x}_0^{(i)})\}_{i=1}^{M}$:

$$\mathcal{L}_{\text{initial}}(\boldsymbol{\omega}) = \frac{1}{|Y_{\text{ic}}|} \sum_{y \in Y_{\text{ic}}} \|\hat{\varphi}(0, y.\mathbf{x}_0, \boldsymbol{\omega}) - y.\mathbf{x}_0\|^2 . \tag{13}$$

The total loss function combines both components:

$$\mathcal{L}(\boldsymbol{\omega}) = \gamma_{\text{physics}} \cdot \mathcal{L}_{\text{physics}}(\boldsymbol{\omega}) + \gamma_{\text{initial}} \cdot \mathcal{L}_{\text{initial}}(\boldsymbol{\omega}), \tag{14}$$

where γ_{physics} and γ_{initial} balance equation satisfaction against initial condition accuracy.

This approach allows the network to learn the flow map $\varphi(t, \mathbf{x}_0)$ by simultaneously satisfying the system dynamics and initial conditions, without needing pre-computed solution data.

5 Numerical Examples

In this section, we present our framework on academic examples.

Example 1. Consider linear initial value problem:

$$\frac{\mathrm{d}u(t)}{\mathrm{d}t} = 0.8\, u(t), \qquad u(0) = 1, \qquad t \in [0, 1],$$

whose exact solution is $u(t) = \exp(0.8t)$. Since $f(u) = 0.8u$, $L = 0.8$. Further, for the linear form $A = [0.8]$, we have $\frac{A+A^\top}{2} = [0.8]$, hence $m(A) = \omega(A) = 0.8$. To separate the two lower-bound mechanisms, we choose $\lambda = 0.4 < m(A)$. Let $\mathcal{R}_{\hat{\varphi}}(t) = \frac{\mathrm{d}\hat{u}}{\mathrm{d}t} - 0.8\, \hat{u}(t)$ and $\boldsymbol{\varepsilon}(t) = \hat{u}(t) - u(t)$. Bounds are evaluated exactly as in Section 3.1 using a point-wise envelope $|\mathcal{R}_{\hat{\varphi}}(t)| \le \delta$ on $[0, 1]$.

Figure 1 shows that the PINN tracks the exact trajectory closely on $[0, 1]$. In Figure 2, the certified interval (shaded) contains the true error for all t. Because $L = \omega(A) = m(A) = 0.8$ for this scalar problem, the linear-sharp and nonlinear upper bounds coincide. The linear-sharp lower bound ($m(A) = \omega(A)$) tracks the error closely, while the nonlinear lower bound with $\lambda = 0.4$ is (deliberately) more conservative; both remain valid for all $t \in [0, 1]$.

Example 2. Consider nonlinear logistic equation:

$$\frac{\mathrm{d}u(t)}{\mathrm{d}t} = u(1-u), \quad u(0) = 0.1, \quad t \in [0, 1], \tag{15}$$

has exact solution $u(t) = 1/(1 + 9e^{-t})$ with $u(t) \in [0.1, 0.25]$. The nonlinearity $f(u) = u(1-u)$ yields for $u, v \in [0.1, 0.25]$:

$$\frac{f(v) - f(u)}{v - u} = 1 - (u + v) \in [0.5, 0.8],$$

giving strong monotonicity constant $\lambda = 0.5$ and Lipschitz constant $L = 0.8$.

A PINN (3×50 $\tanh$) was trained with 1000 collocation points and loss weights 100 (physics) : 1 (IC). Let $\boldsymbol{\varepsilon}(t) = \hat{u}(t) - u(t)$ and residual $\mathcal{R}_{\hat{\varphi}}(t) = \frac{\mathrm{d}\hat{u}(t)}{\mathrm{d}t} - \hat{u}(1 - \hat{u})$. Using smoothed residual envelope $\delta(t) = \sqrt{|\mathcal{R}_{\hat{\varphi}}(t)|^2 + \mu^2}$ with $\mu = 0.05 \max_t |\mathcal{R}_{\hat{\varphi}}(t)|$ and Simpson's rule (200 subintervals) for integral approximations.

Fig. 3 shows PINN and exact solutions are visually identical. Fig. 4 demonstrates rigorous certification: true error lies within the certified band, with lower bound (using $\lambda = 0.5$) closely tracking error and upper bound (using $L = 0.8$) being conservative. Band width depends on residual envelope and shrinks with better training of ODE-guided PINN.

Acknowledgments

This work has been partially funded by the German Research Foundation (Deutsche Forschungsgemeinschaft (DFG)) through the research grant HU1889/7-1.

References

[1] V. I. Arnold. *Ordinary Differential Equations*. The MIT Press, Cambridge, MA, 1978.

[2] K. E. Atkinson. *An Introduction to Numerical Analysis*. John Wiley & Sons, 2nd edition, 1989.

[3] A. G. Baydin, B. A. Pearlmutter, A. A. Radul, and J. M. Siskind. Automatic differentiation in machine learning: a survey. *Journal of Machine Learning Research*, 18, 2018.

[4] I. Cortés-Ciriano and A. Bender. Deep confidence: A computationally efficient framework for calculating reliable prediction errors for deep neural networks. *Journal of Chemical Information and Modeling*, 59(3): 1269–1281, 2019.

[5] T. De Ryck and S. Mishra. Generic bounds on the approximation error for physics-informed and operator learning. *In NeurIPS 2022, 14 pp*, 2022.

[6] E. Hairer, S. P. Norsett, and G. Wanner. *Solving Ordinary Differential Equations I: Nonstiff Problems*. Springer Science & Business Media, 2008.

[7] J. S. Hesthaven, G. Rozza, and B. Stamm. *Certified Reduced Basis Methods for Parametrized Partial Differential Equations*. Springer, 2016.

[8] B. Hillebrecht and B. Unger. Certified machine learning: A posteriori error estimation for physics-informed neural networks. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2022.

[9] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 2nd edition, 2012.

[10] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3:422–440, 2021.

[11] P. Minakowski and T. Richter. Error estimates for neural network solutions of partial differential equations. *arXiv preprint arXiv:2107.11035*, 2021.

[12] 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. *Journal of Computational Physics*, 378:686–707, 2019.

[13] P. V. Subrahmanyam. *Elementary Fixed Point Theorems*. Springer, Cham, 2018.

[14] R. van der Meer, C. W. Oosterlee, and A. Borzi. Goal-oriented error estimation for physics-informed neural networks. *arXiv preprint arXiv:2203.04247*, 2022.

A Technical Appendices and Supplementary Material

In this section, we provide all supplementary materials such as proofs of theorems stated in Section 2 and Section 3 and other numerical experiments presented in Section 5.

Proof of Theorem 1. Let $\hat{\mathbf{x}}_0 := \hat{\mathbf{x}}(0)$. The strong monotonicity condition from (2) together with the smoothness of $\hat{\varphi}$ implies that $\mathcal{R}_{\hat{\varphi}}$ is continuous. The error dynamics satisfy:

$$\dot{\boldsymbol{\varepsilon}}(t) = \dot{\mathbf{x}}(t) - \dot{\hat{\mathbf{x}}}(t) = f(t, \mathbf{x}(t)) - f(t, \hat{\mathbf{x}}(t)) - \mathcal{R}_{\hat{\varphi}}(t).$$

Consider the time-derivative of $\|\boldsymbol{\varepsilon}(t)\|^2$:

$$\frac{\mathrm{d}}{\mathrm{d}t}\|\boldsymbol{\varepsilon}(t)\|^2 = 2\boldsymbol{\varepsilon}(t) \cdot \dot{\boldsymbol{\varepsilon}}(t) = 2\boldsymbol{\varepsilon}(t) \cdot \big(f(t, \mathbf{x}(t)) - f(t, \hat{\mathbf{x}}(t)) - \mathcal{R}_{\hat{\varphi}}(t)\big).$$

Using the strong monotonicity condition from (2):

$$\boldsymbol{\varepsilon}(t) \cdot \big(f(t, \mathbf{x}(t)) - f(t, \hat{\mathbf{x}}(t))\big) \geq \lambda \|\boldsymbol{\varepsilon}(t)\|^2,$$

and using the upper bound from the Cauchy–Schwarz inequality:

$$\boldsymbol{\varepsilon}(t) \cdot \mathcal{R}_{\hat{\varphi}}(t) \leq \|\boldsymbol{\varepsilon}(t)\| \|\mathcal{R}_{\hat{\varphi}}(t)\|.$$

Combining these inequalities, we obtain:

$$\frac{\mathrm{d}}{\mathrm{d}t}\|\boldsymbol{\varepsilon}(t)\|^2 \geq 2\lambda \|\boldsymbol{\varepsilon}(t)\|^2 - 2\|\boldsymbol{\varepsilon}(t)\| \|\mathcal{R}_{\hat{\varphi}}(t)\|.$$

For $\|\boldsymbol{\varepsilon}(t)\| > 0$, dividing by $2\|\boldsymbol{\varepsilon}(t)\|$ yields:

$$\frac{\mathrm{d}}{\mathrm{d}t}\|\boldsymbol{\varepsilon}(t)\| \geq \lambda\|\boldsymbol{\varepsilon}(t)\| - \|\mathcal{R}_{\hat{\varphi}}(t)\|.$$

Rewriting as a differential inequality and multiplying by the integrating factor $\mathrm{e}^{-\lambda t}$:

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathrm{e}^{-\lambda t}\|\boldsymbol{\varepsilon}(t)\|\right) \geq -\mathrm{e}^{-\lambda t}\|\mathcal{R}_{\hat{\varphi}}(t)\|.$$

Integrating from 0 to t and rearranging and multiplying by $\mathrm{e}^{\lambda t}$:

$$\|\boldsymbol{\varepsilon}(t)\| \geq \mathrm{e}^{\lambda t}\|\boldsymbol{\varepsilon}(0)\| - \int_0^t \mathrm{e}^{\lambda(t-s)}\|\mathcal{R}_{\hat{\varphi}}(s)\|\,\mathrm{d}s.$$

Finally, applying the bound (7) gives:

$$\|\boldsymbol{\varepsilon}(t)\| \geq \mathrm{e}^{\lambda t}\|\boldsymbol{\varepsilon}(0)\| - \int_0^t \mathrm{e}^{\lambda(t-s)}\delta(s)\,\mathrm{d}s = \mathrm{e}^{\lambda t}\|\boldsymbol{\varepsilon}(0)\| - I(t,\delta).$$

This proves the theorem. □

Remark 4. Theorem 1 can be easily generalized for time-dependent strong monotonicity parameters

$$(f(t,\mathbf{u}) - f(t,\mathbf{v}))\cdot(\mathbf{u}-\mathbf{v}) \geq \lambda(t)\|\mathbf{u}-\mathbf{v}\|^2$$

by replacing the constant λ with the integrated quantity $\Lambda(t) = \int_0^t \lambda(s)\,\mathrm{d}s$, yielding

$$\|\boldsymbol{\varepsilon}(t)\| \geq \mathrm{e}^{\Lambda(t)}\|\boldsymbol{\varepsilon}(0)\| - \int_0^t \mathrm{e}^{\Lambda(t)-\Lambda(s)}\delta(s)\,\mathrm{d}s.$$

Proof of Lemma 2. Let $\mathbf{w} = \mathbf{u} - \mathbf{v}$. Then $(f(\mathbf{u}) - f(\mathbf{v}))\cdot(\mathbf{u}-\mathbf{v}) = \mathbf{w}^\top A\mathbf{w}$. Decompose A as $A = \frac{A+A^\top}{2} + \frac{A-A^\top}{2}$, where $\frac{A-A^\top}{2}$ is symmetric and $\frac{A-A^\top}{2}$ is skew-symmetric and $\mathbf{w}^\top\frac{A-A^\top}{2}\mathbf{w} = 0$ for all $\mathbf{w}$. Hence $\mathbf{w}^\top A\mathbf{w} = \mathbf{w}^\top\frac{A+A^\top}{2}\mathbf{w}$ and the claimed equivalence follows. The maximal λ is the smallest eigenvalue of the symmetric matrix $\frac{A+A^\top}{2}$. □

Proof of Theorem 3. For (10), the error satisfies

$$\dot{\boldsymbol{\varepsilon}}(t) = A\boldsymbol{\varepsilon}(t) - \mathcal{R}_{\hat{\varphi}}(t), \quad \boldsymbol{\varepsilon}(0) = \mathbf{x}(0) - \hat{\mathbf{x}}(0).$$

Using the variation of constants formula yields

$$\boldsymbol{\varepsilon}(t) = e^{At}\boldsymbol{\varepsilon}(0) - \int_0^t e^{A(t-s)}\mathcal{R}_{\hat{\varphi}}(s)\,\mathrm{d}s.$$

Taking norms and applying the triangle inequality produces

$$\|\boldsymbol{\varepsilon}(t)\| \geq \left\|e^{At}\boldsymbol{\varepsilon}(0)\right\| - \int_0^t \left\|e^{A(t-s)}\right\|\|\mathcal{R}_{\hat{\varphi}}(s)\|\,\mathrm{d}s.$$

From matrix analysis [9], it is known that for all $\mathbf{z} \in \mathbb{R}^n$:

$$e^{\,m(A)\,t}\,\|\mathbf{z}\| \;\leq\; \|e^{At}\mathbf{z}\| \;\leq\; e^{\,\omega(A)\,t}\,\|\mathbf{z}\|, \qquad t \geq 0,$$

and

$$\left\|e^{At}\right\| \leq e^{\omega(A)t}.$$

Application of these inequalities to the error bound produces:

$$\|\boldsymbol{\varepsilon}(t)\| \geq e^{m(A)t}\|\boldsymbol{\varepsilon}(0)\| - \delta\int_0^t e^{\omega(A)(t-s)}\,\mathrm{d}s. \tag{16}$$

Here, the integral can be evaluated as

$$\int_0^t e^{\omega(A)(t-s)}\,\mathrm{d}s = \left[-\frac{1}{\omega(A)}e^{\omega(A)(t-s)}\right]_0^t = \frac{e^{\omega(A)t}-1}{\omega(A)}.$$

if $\omega(A) \neq 0$, and

$$\int_0^t e^{\omega(A)(t-s)}\,\mathrm{d}s = \int_0^t \mathrm{d}s = t.$$

if $\omega(A) = 0$. Substitution these results in (16) completes the proof. □

Remark 5. For the upper bound case, using similar techniques one can obtain

$$\|\boldsymbol{\varepsilon}(t)\| \leq \begin{cases} e^{\omega(A)t}\|\boldsymbol{\varepsilon}(0)\| + \frac{\delta}{\omega(A)}\left(e^{\omega(A)t} - 1\right), & \text{if} \quad \omega(A) \neq 0, \\ \|\boldsymbol{\varepsilon}(0)\| + \delta t, & \text{if} \quad \omega(A) = 0. \end{cases} \tag{17}$$

This provides a worst-case error estimate. The lower bound in Theorem 3 establishes the best-case error persistence and together provide a complete error certification framework.

Remark 6. *Let $S := \frac{A+A^\top}{2}$. Then: (i)* Strong monotonicity: *the sharp constant is $\lambda_\star = m(A) := \lambda_{\min}(S)$; a positive λ exists iff $m(A) > 0$, and any $\lambda \in (0, m(A)]$ is valid. (ii)* (Global) Lipschitz: *The sharp Lipschitz constant $L_\star = \|A\|$. (iii)* Matrix–exponential growth: $\omega(A) := \lambda_{\max}(S)$ *and*

$$\lambda \leq m(A) \leq \omega(A) \leq \|A\| \leq L.$$

Remark 7. *For practical implementation, the lower bounds derived in Theorems 1 and 3 yield computable estimators $L_{\mathrm{LB}}(t)$. In both cases, the certified lower bound is given by $\|\boldsymbol{\varepsilon}(t)\| \geq \max\{0, L_{\mathrm{LB}}(t)\}$, ensuring non-negativity of the error estimate.*

Remark 8. *Our lower bounds complement the upper bounds derived in Hillebrecht and Unger [8]. While upper bounds use Lipschitz constants to quantify worst-case error growth, our lower bounds use strong monotonicity constants (or their linear equivalents $m(A)$ and $\omega(A)$) to establish best-case error persistence. For linear systems, the gap between $m(A)$ and $\omega(A)$ reflects the inherent uncertainty due to possible non-normal transient effects, providing a quantitative measure of the certification tightness.*

Proof of Lemma 7. We consider $J(t,\delta) = \int_0^t e^{-\lambda s}\delta(s)\mathrm{d}s$ so that $I(t,\delta) = e^{\lambda t}J(t,\delta)$. The composite Simpson's rule approximation is:

$$\hat{J}_n(t,\delta) = \frac{h}{3}\left[g(0) + g(t) + 4\sum_{j=1}^{n/2} g((2j-1)h) + 2\sum_{j=1}^{n/2-1} g(2jh)\right],$$

where $g(s) = e^{-\lambda s}\delta(s)$ and $h = t/n$. The standard error bound for Simpson's rule gives: [2]

$$|J(t,\delta) - \hat{J}_n(t,\delta)| \leq \frac{t^5}{180n^4}\max_{s\in[0,t]}|g^{(4)}(s)| \leq \frac{Kt^5}{180n^4}.$$

Multiplying by $e^{\lambda t}$ and applying Theorem 1 yields the result. □

A.1 Numerical Integration with Simpson's Rule

To compute the integral terms in the a posteriori lower bounds, we employ Simpson's rule which offers improved accuracy compared to the trapezoidal rule used in existing literature [8]. Under appropriate smoothness conditions, Simpson's rule achieves an error scaling of $O(n^{-4})$ compared to $O(n^{-2})$ for the trapezoidal rule [2].

Lemma 9. *Assume that f in* (1) *satisfies Assumption 2.1 and the ML candidate function $\hat{\varphi}$ is sufficiently smooth. Let $\delta \in C^4(\mathbb{T},\mathbb{R})$ with $\|\mathcal{R}_{\hat{\varphi}}(t)\| \leq \delta(t)$ and $\left|\frac{\mathrm{d}^4}{\mathrm{d}s^4}\left(e^{-\lambda s}\delta(s)\right)\right| \leq K$ for $s \in \mathbb{T}$. Then*

$$\|\boldsymbol{\varepsilon}(t)\| \geq e^{\lambda t}\|\boldsymbol{\varepsilon}(0)\| - \hat{I}_n(t,\delta) - E_{\mathrm{Int}},$$

where $\hat{I}_n(t,\delta)$ is the composite Simpson's rule approximation of $I(t,\delta) = \int_0^t e^{\lambda(t-s)}\delta(s)\mathrm{d}s$ using n subintervals (with n even), and

$$E_{\mathrm{Int}} = e^{\lambda t}\frac{Kt^5}{180n^4}.$$

Remark 10. *To ensure the required smoothness for Simpson's rule, we can construct a smooth upper bound for $\|\mathcal{R}_{\hat{\varphi}}(t)\|$ following [4]:*

$$\delta(t) := \sqrt{\|\mathcal{R}_{\hat{\varphi}}(t)\|^2 + \mu^2} \tag{18}$$

with $\mu \in \mathbb{R}_+$ chosen appropriately.

B Numerical Examples

We consider figures which are presented in Section 5.

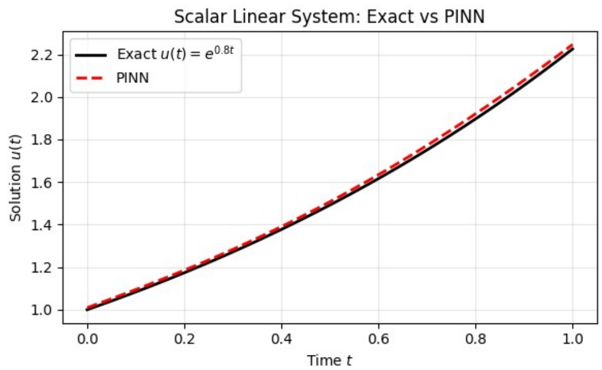


Figure 1: Exact solution $u(t) = \exp(0.8t)$ versus PINN on $[0, 1]$.

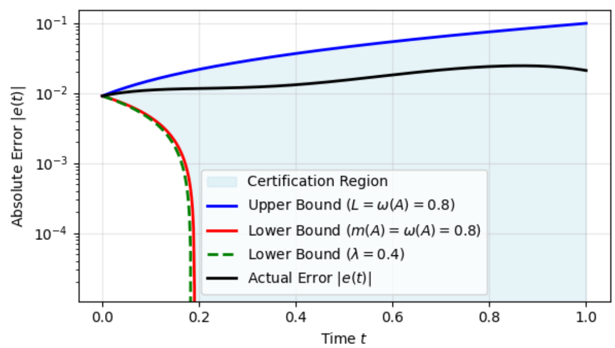


Figure 2: A posteriori certification. Shaded region: certified band. Curves: upper bound with $L = \omega(A) = 0.8$; lower bound (linear–sharp) with $m(A) = \omega(A) = 0.8$; lower bound (nonlinear) with $\lambda = 0.4$; and the actual error $|\varepsilon(t)|$.

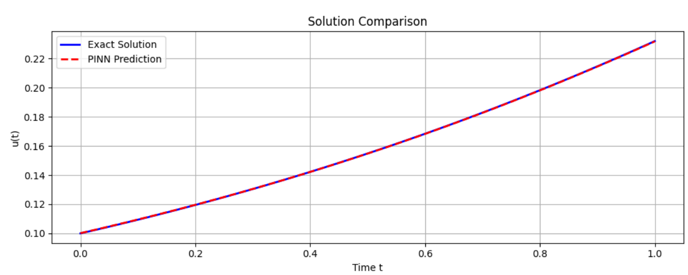
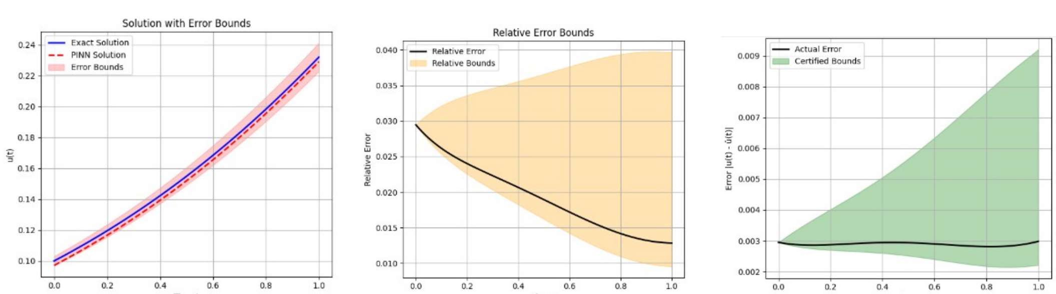


Figure 3: Exact solution vs. PINN on $[0, 1]$. The trajectories are visually indistinguishable at the plot scale.



(a) Solution with certified band using $\lambda = 0.5$, $L = 0.8$.

(b) Relative error and certified relative bounds.

(c) Absolute error with certified lower/upper bounds.

Figure 4: **A posteriori certification on the logistic ODE.** The lower envelope tracks the actual error closely, while the upper envelope is conservative.