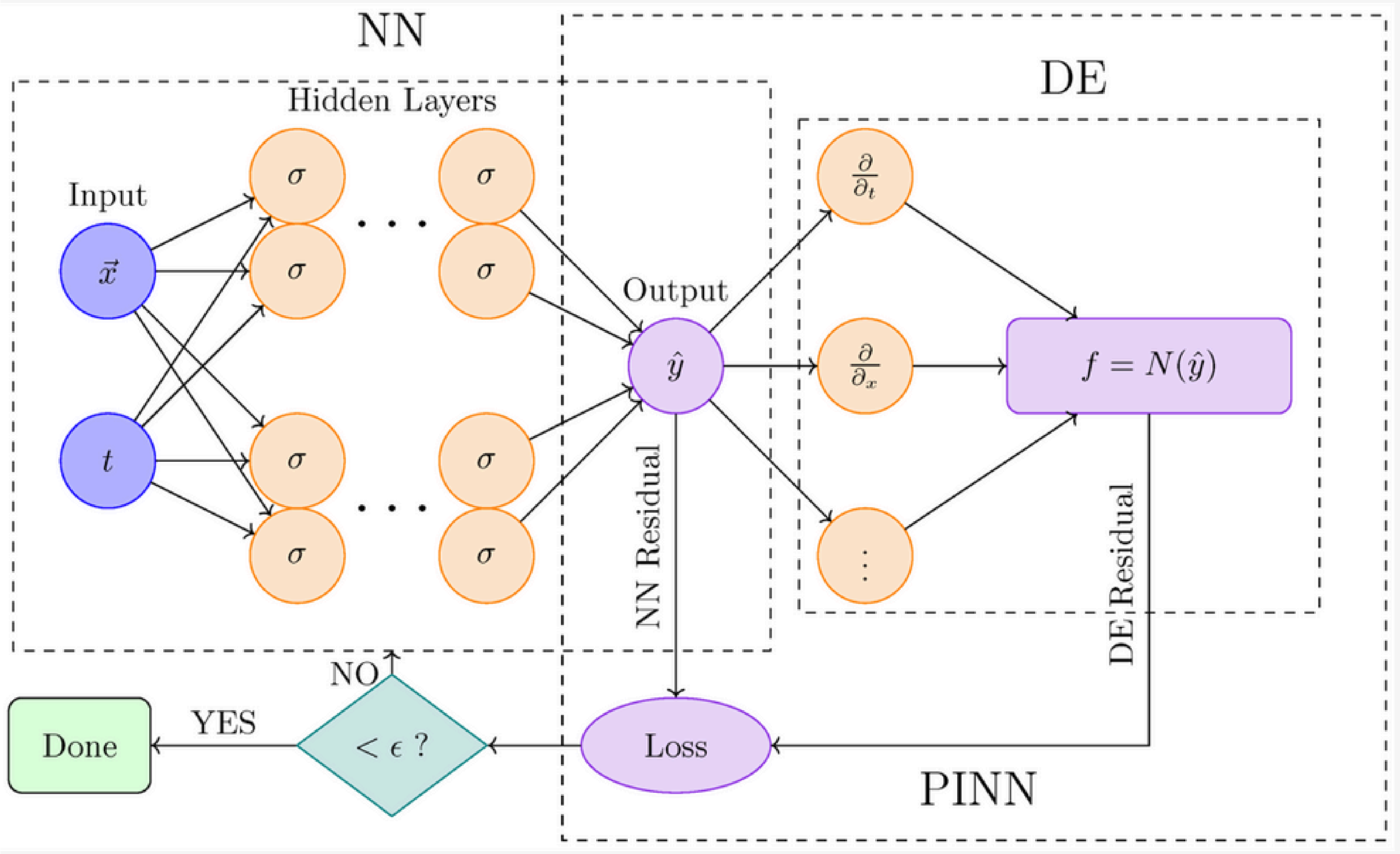
Physics-Informed Neural Networks and ODEs in vector population dynamics modelling

Climate-Sensitive Vector Dynamics Modelling Workshop, Nicosia, 2025

Mina Petrić
Avia-GIS

Physics Informed Nural Networks

Back to Agenda Page



Farea et al. (2024)

- Standard NNs learn a mapping from inputs to outputs purely from data.
- Inputs (e.g., space and time: x, y, z, t) pass through hidden layers of neurons.
- The network predicts outputs (u), optimized by minimizing a data loss function.

- PINNs extend NNs by embedding governing physical equations (ODEs/PDEs) into training.
- Automatic differentiation computes derivatives of the NN outputs (∂x, ∂^2x, ∂t, …)
- These derivatives are plugged into the physics equations (e.g., conservation laws)
- The loss function is calculated as a combination of data and physics loss

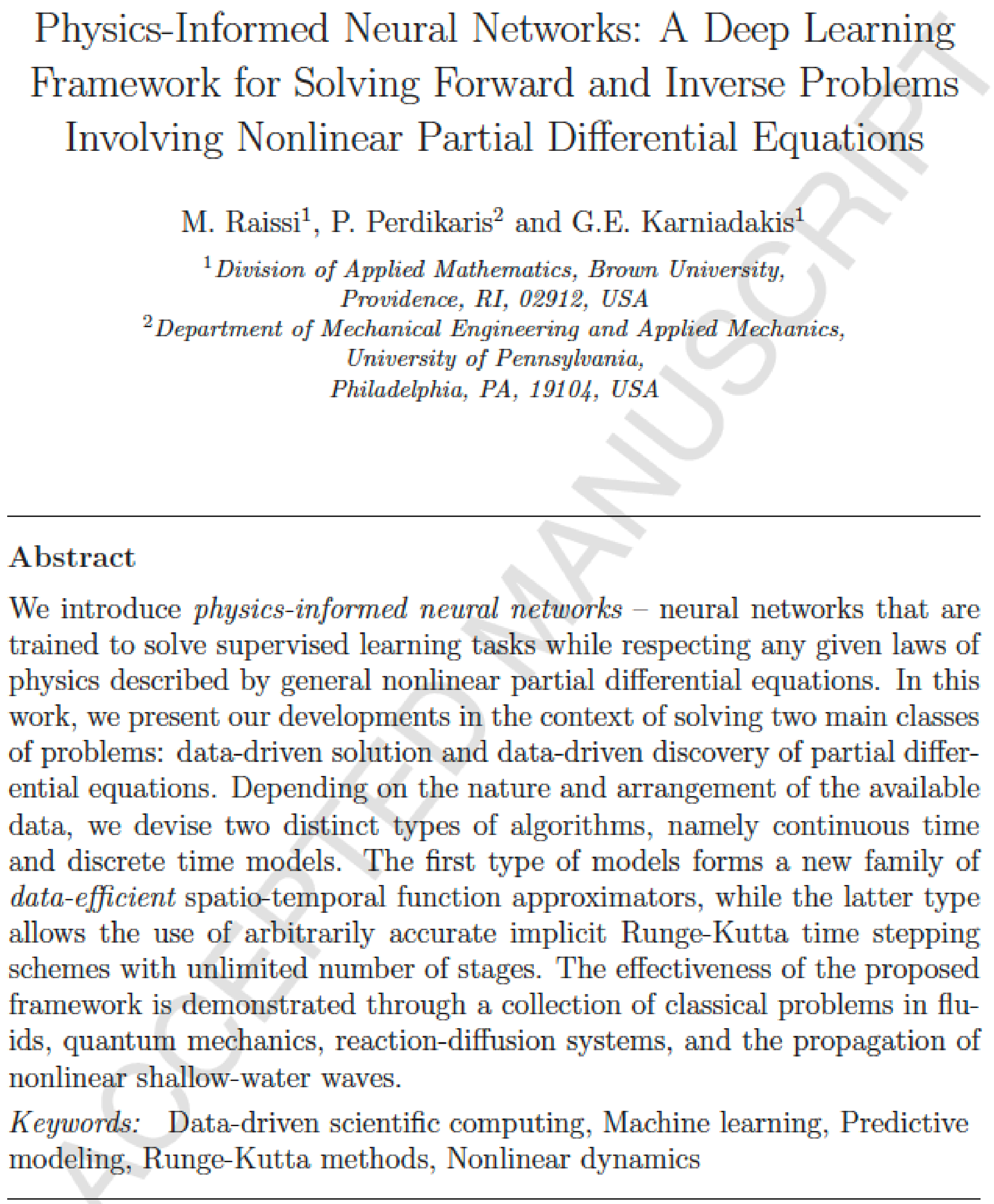
Back to Agenda Page

NN vs PINN framework

Baseline: Plain Neural Networks	PINNs: Add a physics loss
• Input: independent variables (r,t) • Output: scalar/vector field • Train with data loss	• Input and output remain the same • Compute PDE residual with automatic differentiation (AD) • Computes Physics loss at collocation points • Total loss = data loss + physics loss • In contrast to numerical solvers PINN learns a continuous function that satisfies the PDE and boundary/initial conditions. Once trained, you have a differentiable surrogate valid across the whole domain

Physics Informed Nural Networks

Back to Agenda Page



Physics-Informed Neural Networks: A Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear Partial Differential Equations

M. Raissi[1], P. Perdikaris[2] and G.E. Karniadakis[1]

[1] *Division of Applied Mathematics, Brown University, Providence, RI, 02912, USA*

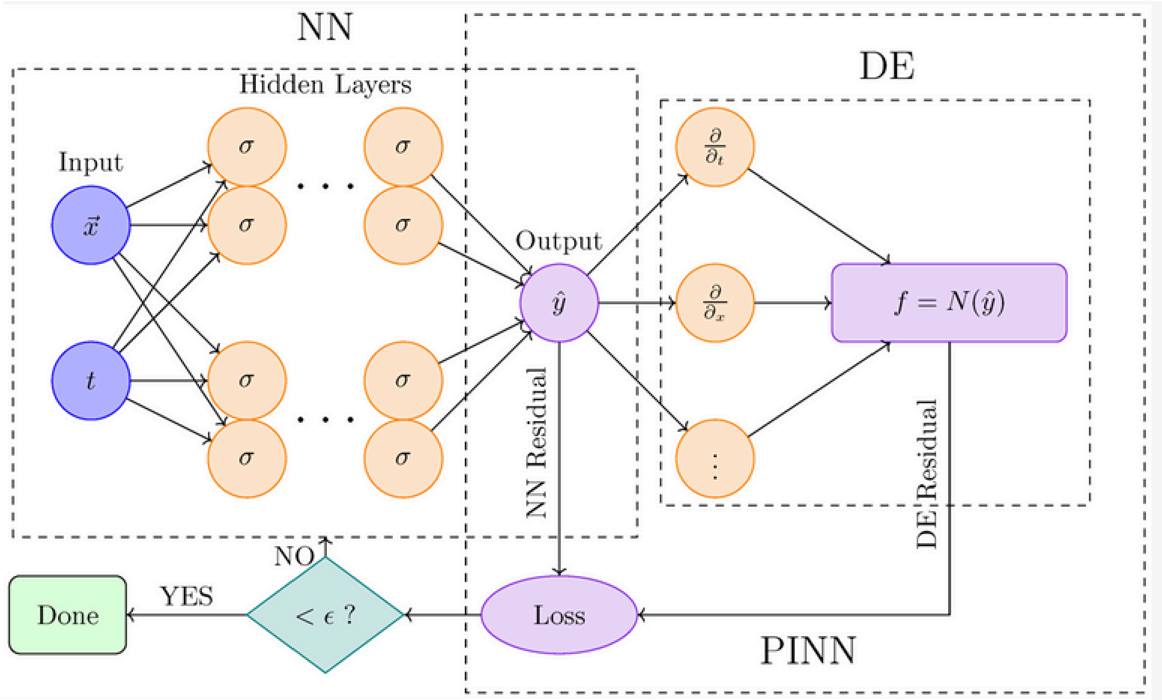
[2] *Department of Mechanical Engineering and Applied Mechanics, University of Pennsylvania, Philadelphia, PA, 19104, USA*

Abstract

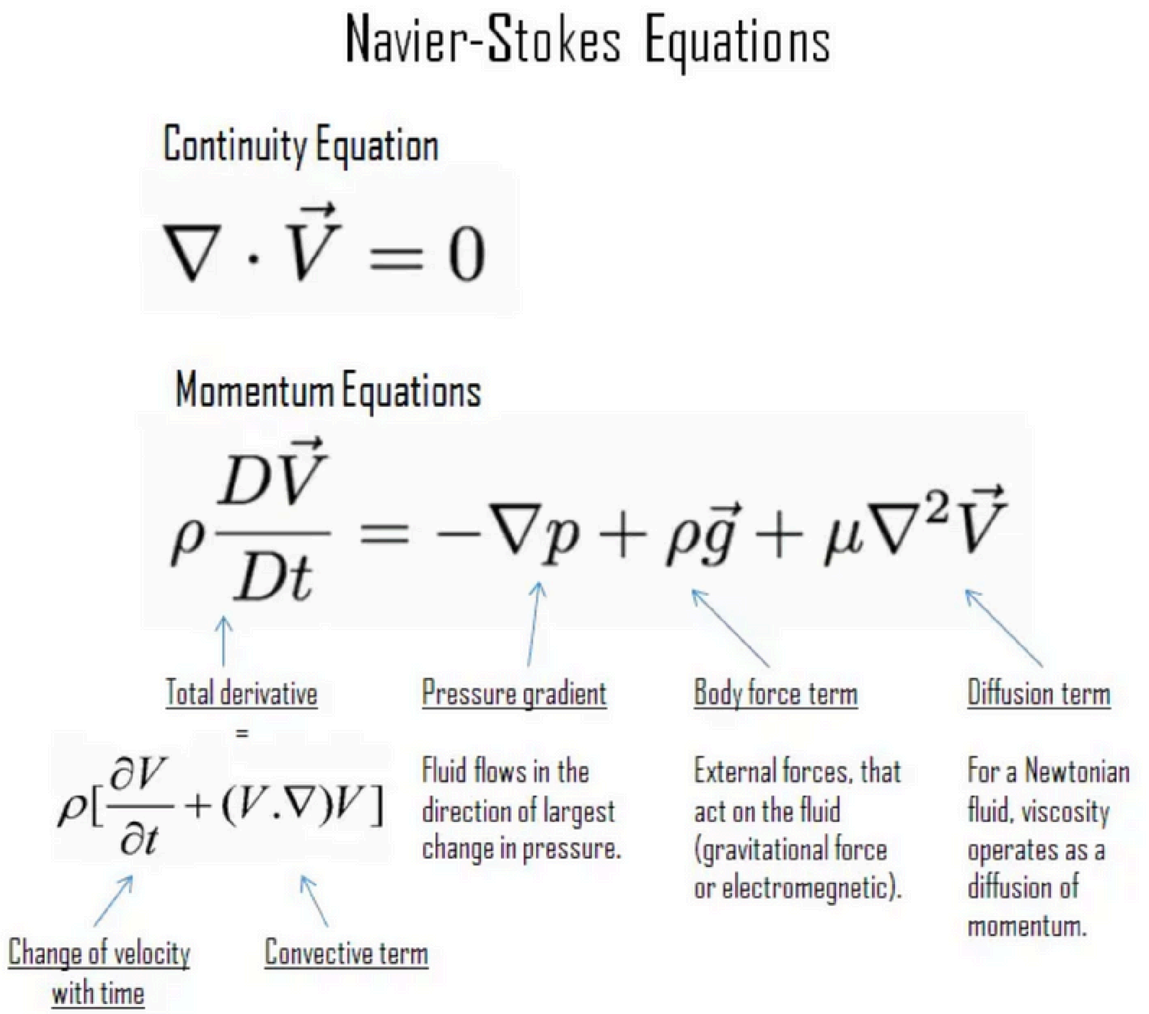
We introduce *physics-informed neural networks* – neural networks that are trained to solve supervised learning tasks while respecting any given laws of physics described by general nonlinear partial differential equations. In this work, we present our developments in the context of solving two main classes of problems: data-driven solution and data-driven discovery of partial differential equations. Depending on the nature and arrangement of the available data, we devise two distinct types of algorithms, namely continuous time and discrete time models. The first type of models forms a new family of *data-efficient* spatio-temporal function approximators, while the latter type allows the use of arbitrarily accurate implicit Runge-Kutta time stepping schemes with unlimited number of stages. The effectiveness of the proposed framework is demonstrated through a collection of classical problems in fluids, quantum mechanics, reaction-diffusion systems, and the propagation of nonlinear shallow-water waves.

Keywords: Data-driven scientific computing, Machine learning, Predictive modeling, Runge-Kutta methods, Nonlinear dynamics

Rassini et al. (2019)



Farea et al. (2024)

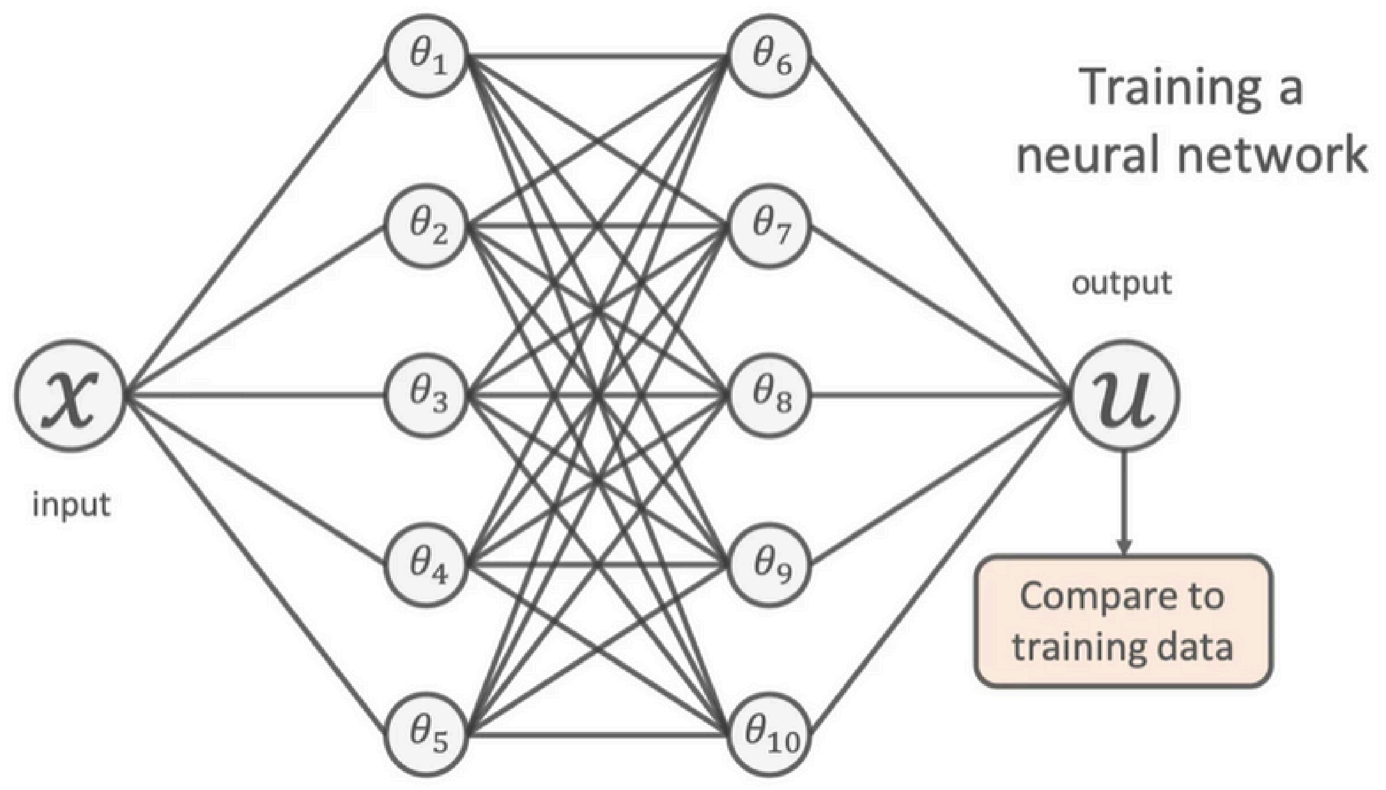


"Navier-Stokes Equations," Nuclear Power. [Online]

Back to Agenda Page

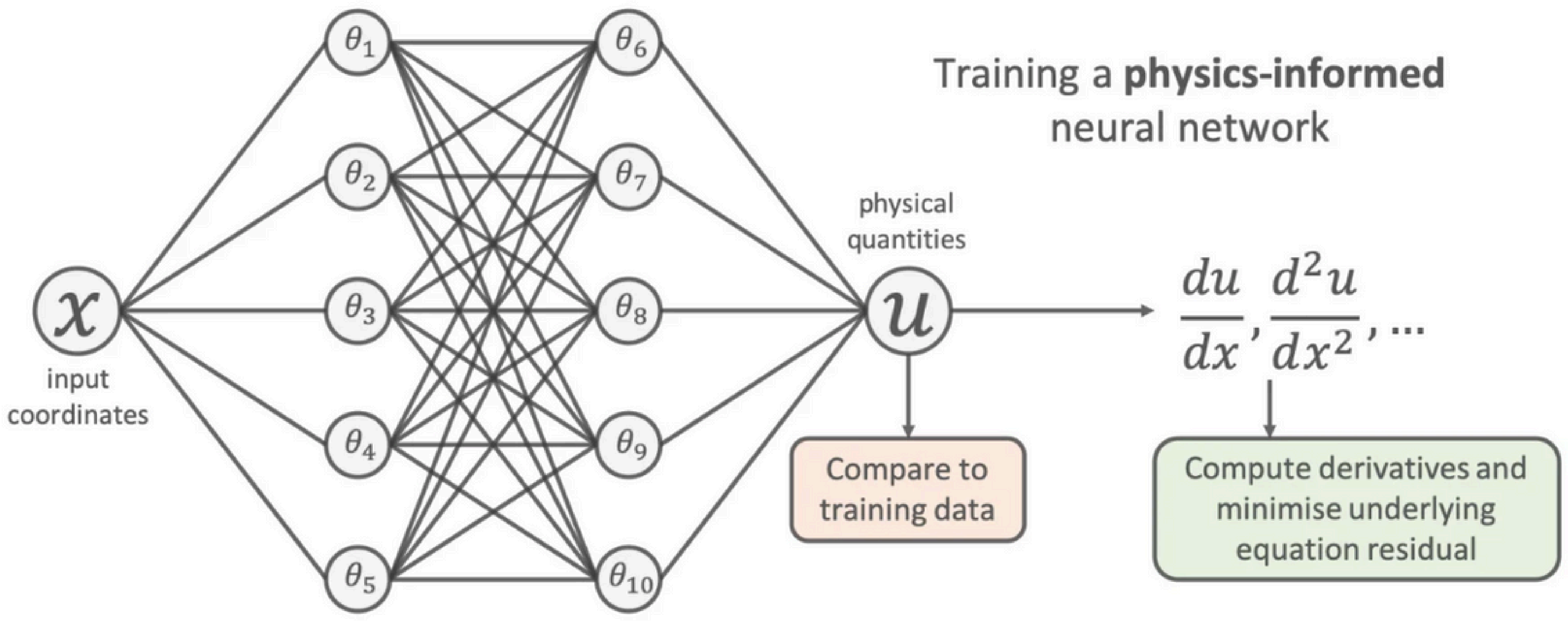
NN vs PINN framework

Baseline: Plain Neural Networks



Ben Moseley (2021)

PINNs: Add a Physics loss



Ben Moseley (2021)

Considerations

Back to Agenda Page

PINN Strengths	PINN Limitations
• Incorporates domain knowledge: physically consistent predictions and greatly improved generalisability compared to NN • Data efficiency (works with sparse/partial observations) • Mesh-free and continuous solution at any query point (scalar/vector field) • Unifies forward and inverse modelling • Once trained, inference is cheap at arbitrary spatio-temporal points	• Advection-dominated or sharp discontinuities, hard to learn • Multi-objective balance can stall training • Stiff/chaotic systems: optimization traps, gradient pathologies • Scaling to high-dimensional PDEs is challenging • Mitigations: rescaling, domain decomposition, better sampling, hard-constraints, pre-training • Way forward: Set realistic expectations; validate against trusted solvers.

Back to Agenda Page

Why PINNs

Vector populations: time-dependent dynamical systems described by differential equations, where state variables capture population abundances and additional functions represent environmental drivers or resources. The analysis of such models aims to elucidate key ecological mechanisms, assess stability and parameter sensitivity, and provide predictive insights under varying environmental conditions.

comment

Check for updates

The imperative of physics-based modeling and inverse theory in computational science

To best learn from data about large-scale complex systems, physics-based models representing the laws of nature must be integrated into the learning process. Inverse theory provides a crucial perspective for addressing the challenges of ill-posedness, uncertainty, nonlinearity and under-sampling.

Karen E. Willcox, Omar Ghattas and Patrick Heimbach

The notions of 'artificial intelligence (AI) for science' and 'scientific machine learning' (SciML) are gaining widespread attention in the scientific community. These initiatives target development and adoption of AI approaches in scientific and engineering fields with the goal of accelerating research

geological processes evolve. Physics-based models typically encode knowledge in the form of conservation and constitutive laws, often based on decades if not centuries of theoretical development and experimental validation. These laws often manifest as systems of differential equations that are solved numerically with high-performance

constraints, purely data-driven approaches are unlikely to be predictive, no matter how expressive the underlying representation. Even when physical models are not well-established (such as for many biological processes, in constitutive laws for complex materials, or in subgrid scale models for unresolved physics), we

Nature communication, 2021

- Many problems in ecological modelling have scarce/noisy data
- We don't really now the governing physics (ODEs/PDEs, boundary/initial conditions)
- Solution: include physics into training to improve generalisation, physical plausibility
- Useful for forward (solve ODEs) and inverse (infer parameters/fields) problems
- Incorporate multi-scale and multi-resolution drivers
- Hybrid models

AVIA GIS

PINNs for Mosquito Population Dynamics

From mechanistic ODEs to hybrid PINNs (Speaker notes: Quick overview of why and how we adapted PINNs to mosquito ODEs and what we learned)

Back to Agenda Page

AVIA GIS

PLOS ONE

Back to Agenda Page

RESEARCH ARTICLE

Adapting physics-informed neural networks to improve ODE optimization in mosquito population dynamics

Dinh Viet Cuong[1]*, Branislava Lalić[2], Mina Petrić[3], Nguyen Thanh Binh[4], Mark Roantree[5]

1 School of Computing, Dublin City University, Dublin, Ireland, **2** Faculty of Agriculture, University of Novi Sad, Novi Sad, Serbia, **3** Avia-GIS NV, Zoersel, Belgium, **4** University of Science, Ho Chi Minh City, Vietnam, **5** Insight Centre for Data Analytics, Dublin City University, Dublin, Ireland

* dinhviet.cuong@dcu.ie

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.XXXX.XXXXXXX

Modelling Mosquito Population Dynamics using PINN-derived Empirical Parameters

BRANISLAVA LALIC[1], DINH VIET CUONG[2], Mina Petric[3], Vladimir Pavlovic[4], Ana Firanj Sremac[1], Mark Roantree[5]

[1]Faculty of Agriculture, University of Novi Sad, Novi Sad, Serbia (e-mail: branislava.lalic@polj.edu.rs, ana.sremac@polj.edu.rs)
[2]School of Computing, Dublin City University, Dublin, Ireland (e-mail: dinhviet.cuong@dcu.ie)
[3]Avia-GIS, Zoersel, Belgium (e-mail: mpetric@avia-gis.com)
[4]Department of Computer Science, Rutgers University, New Brunswick, NJ, USA (e-mail: vladimir@pavlovic.net)
[5]Insight Centre for Data Analytics, Dublin City University, Dublin, Ireland (e-mail: mark.roantree@dcu.ie)

Corresponding author: Branislava Lalic (e-mail: branislava.lalic@polj.edu.rs).

Problem & Goal

- Problem: Sparse/noisy entomological data; highly coupled multi-stage ODEs; stiff, multi-scale dynamics
- Goal: Improve parameterisation & forecasting by embedding biology-informed ODEs into NN training (PINNs)
- Two tracks: (i) ODE-PINN framework for stiff systems; (ii) Hybrid model (Hy_PopMosq) with learned parameters from meteo drivers
- Forward (simulate) and inverse (infer parameters) problems.

AVIA GIS

Back to Agenda Page

The Team

- AI & ML: Dublin City University (Ireland) and Rutgers University (USA) / NSF (USA)
- Meteorology: University of Novi Sad (Serbia)
- Vector population dynamics: Avia-GIS (Belgium)

AI and ML | Meteorology | Vector dynamics

Dinh Viet Cuong (DCU)

Branislava Lalic (UNS)

Mina Petric (Avia-GIS)

Mark Roantree(DCU)

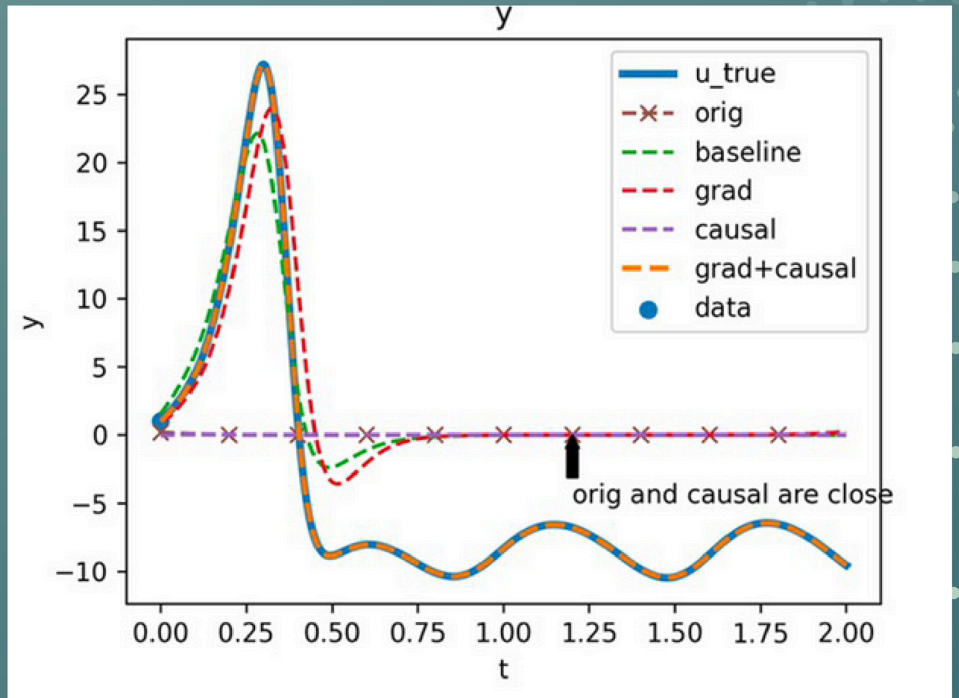
Ana Firanj Sremac (UNS)

Vladimir Pavlovic (RU, NSF)

Ablation Studies

- Ablation studies to improve convergence and stability of the coupled system (Cuong et. al, 2024)
- Naive PINNs are unstable on stiff, multi-scale ODEs with long horizons. Stabilisation strategies introduced:
    - Equation normalisation (loss scaling): rescales state variables and residuals → prevents dominance of large-magnitude terms, improves conditioning.
    - Adaptive gradient balancing (dynamic weighting): balances per-equation contributions in the loss → ensures stable multi-equation optimisation.

Back to Agenda Page



(Cuong et al., 2024)

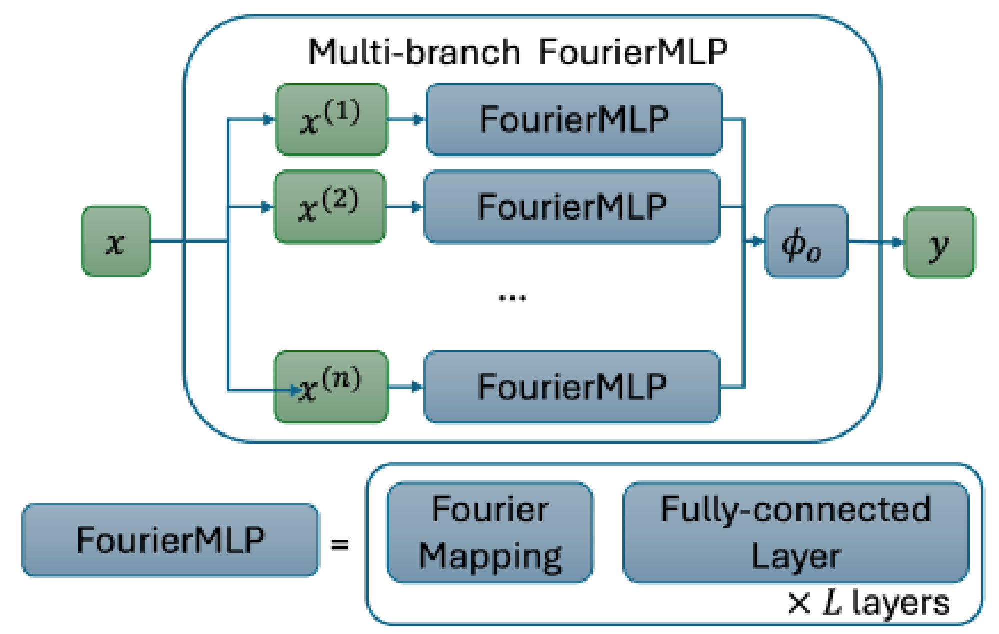


FIGURE 2. Multi-branch Fourier-MLP PINN Framework. Each branch receives a distinct group of inputs and passes them through a FourierMLP which is a Fourier-feature layer followed by fully-connected layers. The branch outputs are summed and passed through the *SoftAbs* activation function.

(Lalic et al., 2025)

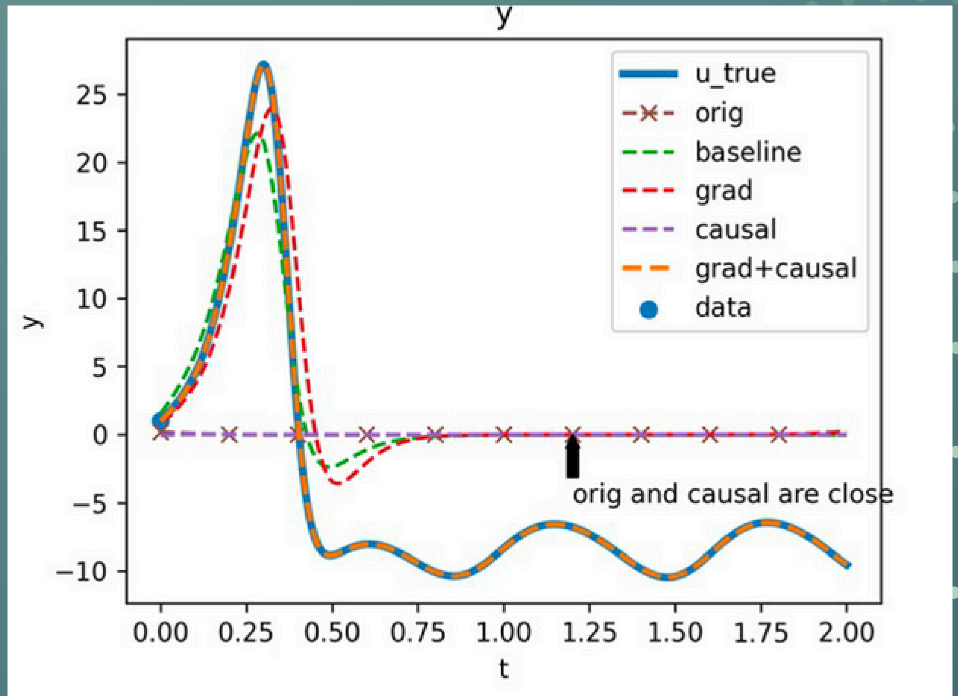
TABLE 2. Error Metrics from Parameters learned from PINNs comprising different network architectures. The best results for each metric are highlighted in bold, while the second best results are underlined.

	MLP	FourierMLP	Branched MLP	Branched FourierMLP
RMSE	0.193658	0.247858	0.197923	**0.179100**
Diff RMSE	0.187189	0.207561	0.191858	**0.172800**
2nd Diff RMSE	0.314825	0.348473	0.312093	**0.292700**
Recall Peak	0.145833	0.166667	0.270833	**0.562500**
Precision Peak	0.375000	0.085714	0.437500	**0.625000**
F1 Peak	0.208333	0.101010	0.291667	**0.566700**

(Lalic et al., 2025)

Ablation Study

- Demonstrated via ablation (Cuong et al., 2024): each strategy independently improves convergence, accuracy, and robustness; combined they enable successful mosquito ODE training.
- Hybrid framework (Lalic et al., 2025)
  - Fourier features accelerate convergence; branched Θ improves meteo→parameter mapping
  - SoftAbs enforces non-negative rates without kinks



(Cuong et al., 2024)

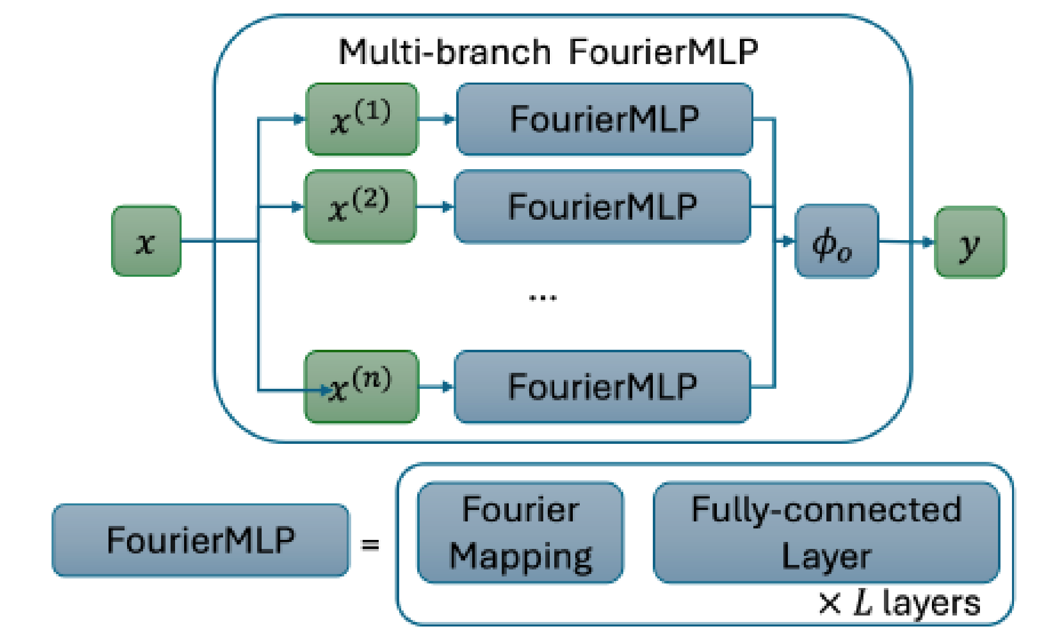


FIGURE 2. Multi-branch Fourier-MLP PINN Framework. Each branch receives a distinct group of inputs and passes them through a FourierMLP which is a Fourier-feature layer followed by fully-connected layers. The branch outputs are summed and passed through the *SoftAbs* activation function.

(Lalic et al., 2025)

TABLE 2. Error Metrics from Parameters learned from PINNs comprising different network architectures. The best results for each metric are highlighted in bold, while the second best results are underlined.

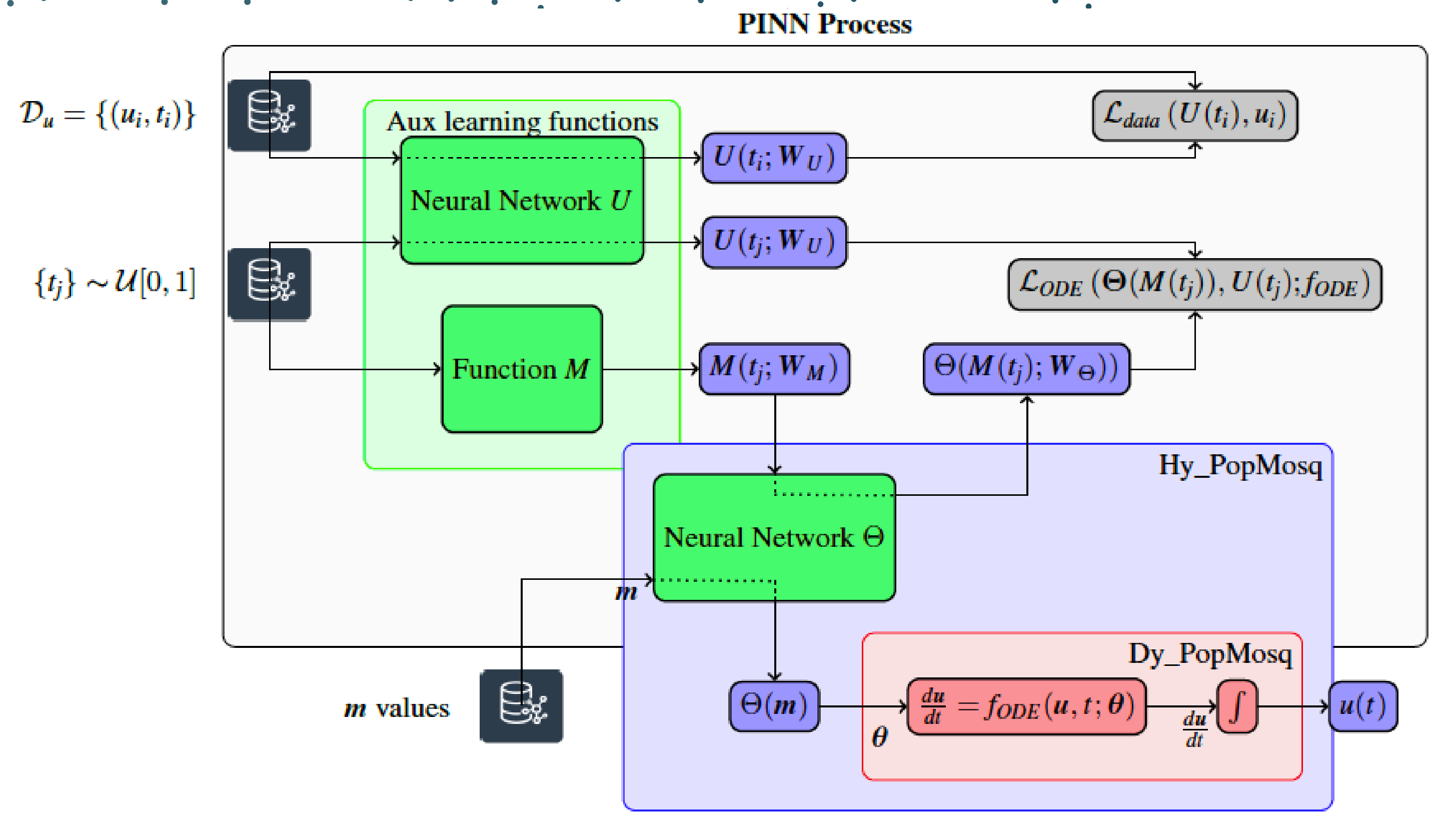
	MLP	FourierMLP	Branched MLP	Branched FourierMLP
RMSE	<u>0.193658</u>	0.247858	0.197923	**0.179100**
Diff RMSE	<u>0.187189</u>	0.207561	0.191858	**0.172800**
2nd Diff RMSE	0.314825	0.348473	<u>0.312093</u>	**0.292700**
Recall Peak	0.145833	0.166667	<u>0.270833</u>	**0.562500**
Precision Peak	0.375000	0.085714	<u>0.437500</u>	**0.625000**
F1 Peak	0.208333	0.101010	<u>0.291667</u>	**0.566700**

(Lalic et al., 2025)

Back to Agenda Page

Towards a hybrid PINN

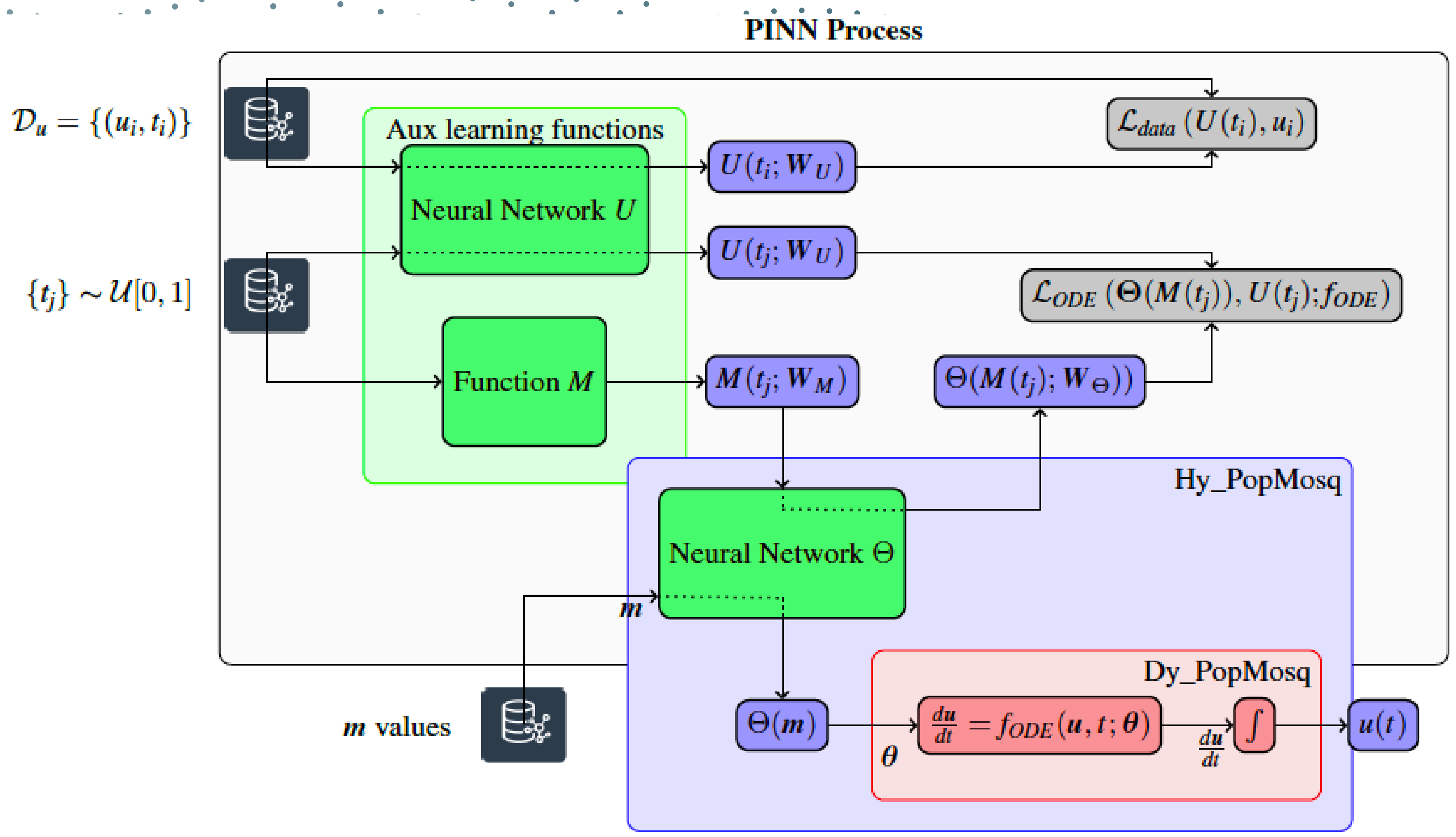
$$
\begin{cases}
\frac{\mathrm{dE}}{\mathrm{dt}} = \gamma_{\mathrm{Ao}}\left(\beta_1 A_{o1} + \beta_2 A_{o2}\right) - \left(\mu_E + f_E\right) E \\
\frac{\mathrm{dL}}{\mathrm{dt}} = f_E E - \left(m_L\left(1 + \frac{L}{\kappa_L}\right) + f_L\right) L \\
\frac{\mathrm{dP}}{\mathrm{dt}} = f_L L - \left(m_P + f_P\right) P \\
\frac{dA_{\mathrm{em}}}{\mathrm{dt}} = f_P \sigma e^{-\mu_{\mathrm{em}}\left(1 + \frac{1}{\kappa_P}\right)} P - \left(m_A + \gamma_{\mathrm{Aem}}\right) A_{\mathrm{em}} \\
\frac{dA_{b1}}{\mathrm{dt}} = \gamma_{\mathrm{Aem}} A_{\mathrm{em}} - \left(m_A + \mu_r + \gamma_{\mathrm{Ab}}\right) A_{b1} \\
\frac{dA_{g1}}{\mathrm{dt}} = \gamma_{\mathrm{Ab}} A_{b1} - \left(m_A + f_{\mathrm{Ag}}\right) A_{g1} \\
\frac{dA_{o1}}{\mathrm{dt}} = f_{\mathrm{Ag}} A_{g1} - \left(m_A + \mu_r + \gamma_{\mathrm{Ao}}\right) A_{o1} \\
\frac{dA_{b2}}{\mathrm{dt}} = \gamma_{\mathrm{Ao}}\left(A_{o1} + A_{o2}\right) - \left(m_A + \mu_r + \gamma_{\mathrm{Ab}}\right) A_{b2} \\
\frac{dA_{g2}}{\mathrm{dt}} = \gamma_{\mathrm{Ab}} A_{b2} - \left(m_A + f_{\mathrm{Ag}}\right) A_{g2} \\
\frac{dA_{o2}}{\mathrm{dt}} = f_{\mathrm{Ag}} A_{g2} - \left(m_A + \mu_r + \gamma_{\mathrm{Ao}}\right) A_{o2}
\end{cases}
\quad (1)
$$



(Lalic et al., 2025)

- Validated Coupled PINN + ODE system (Cuong et al., 2024): Loss scaling, Adaptive gradient balancing, Causal training, Domain decomposition. Each strategy independently improves convergence, accuracy, and robustness; combined they enable successful mosquito ODE training. Note: External meterologica forcing not yet integrated
- Parameter sensitivity analysis (fP, fL)
- Development of Hybrid PINN framework with parameter network that learns meteorology and parameter mapping (Lalic et al., 2025)
- Architectural improvements
    - Fourier features capture periodic drivers
    - Multi-branch MLPs separate short-term vs seasonal inputs
    - SoftAbs activation enforces biological constraints (>0)
- Results: Improved accuracy and peak detection (Lalic et al., 2025)

Towards a hybrid PINN



(Lalic et al., 2025)

- Training dataset
- Collocation points
- Learning function (Cuong et al., 2024)
- State network (U(t,W))
- M - network function that maps time t to meteorological inputs (m) [which need to be sampled at arbitrary intervals] to train the parameter function which in a next step maps the meteorological input to the selected parameters (fL, fP) and passes the parameter inputs to the mechanistic core
- Two networks trained jointly: U (state) and Θ (parameter network) driven by M(t); optimize data loss on observations and physics loss on collocation times; Θ yields parameters for forward ODE simulation.
- Objective: Improving the parameterisation of existing parameters and estimating unknown or latent variables.

Training and Validation

- Training: Petrovaradin (Serbia) 2016–2017; daily traps + daily T (mean/max/min), RH (mean/max/min), total daily precipitation
- Evaluation: 2000–2007 weekly catches for validation
- (Aknowledgement: Dušan Petrić, Aleksandra Ignjatović Ćupina, Mihaela Kavran, Nikola Nožinić and Dragan Dondur for their invaluable efforts in collecting the Culex pipiens data used in this study)

TABLE 1. **Performance metrics for Dy_PopMosq and Hy_PopMosq. Values in brackets refer to performance metrics when two worst-performing years are removed.**

	Dy_PopMosq	Hy_PopMosq	**Observed**
RMSE_ population	0.26 (0.25)	0.18 (0.14)	-
σ_population	0.16	0.12	0.13
RMSE_ rate	0.21 (0.17)	0.17 (0.14)	-
σ_rate	0.12	0.10	0.16
No. Peaks	3.63	1.38	1.88
Peak recall	0.13	0.56	
Precision Peak	0.09	0.63	
F1 Peak	0.11	0.57	

(Lalic et al., 2025)

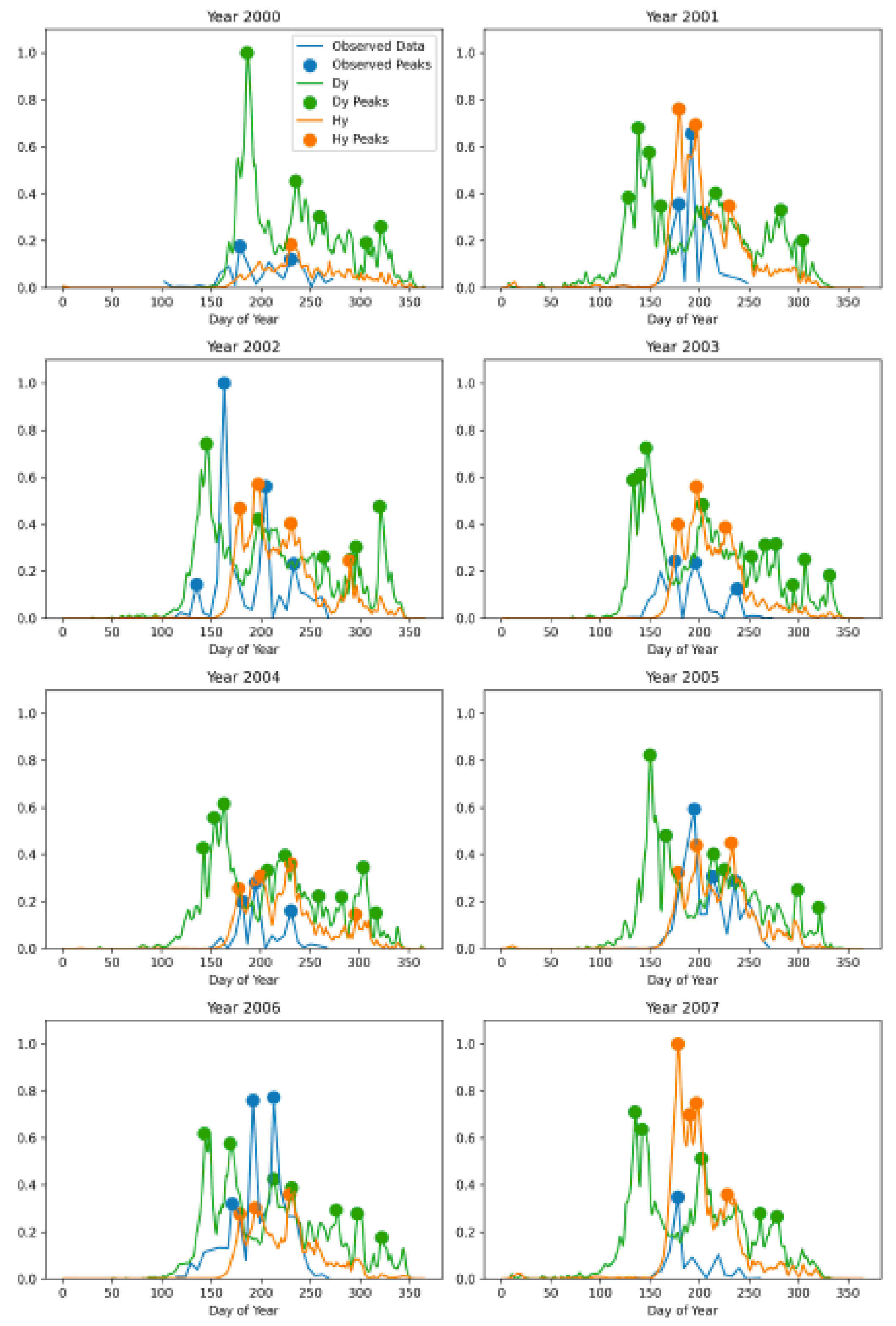


FIGURE 3. **Observed vs. simulated adult-mosquito abundance. Daily, normalised counts of blood-seeking adults ($A_{b1} + A_{b2}$) are plotted for each calendar year. Circles mark peaks identified with a 7-day, 0.2-prominence detector.**

Thank you!

Do you have any questions?

Useful links and references

Raissi M, Perdikaris P, Karniadakis GE. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational physics. 2019 Feb 1;378:686-707.

Farea A, Yli-Harja O, Emmert-Streib F. Understanding physics-informed neural networks: Techniques, applications, trends, and challenges. AI. 2024 Aug 29;5(3):1534-57.

Viet Cuong D, Lalić B, Petrić M, Thanh Binh N, Roantree M. Adapting physics-informed neural networks to improve ODE optimization in mosquito population dynamics. Plos one. 2024 Dec 23;19(12):e0315762.

Lalic B, Cuong DV, Petric M, Pavlovic V, Sremac AF, Roantree M. Physics-Based Dynamic Models Hybridisation Using Physics-Informed Neural Networks. arXiv preprint arXiv:2412.07514. 2024 Dec 10.

Willcox KE, Ghattas O, Heimbach P. The imperative of physics-based modeling and inverse theory in computational science. Nature Computational Science. 2021 Mar;1(3):166-8.

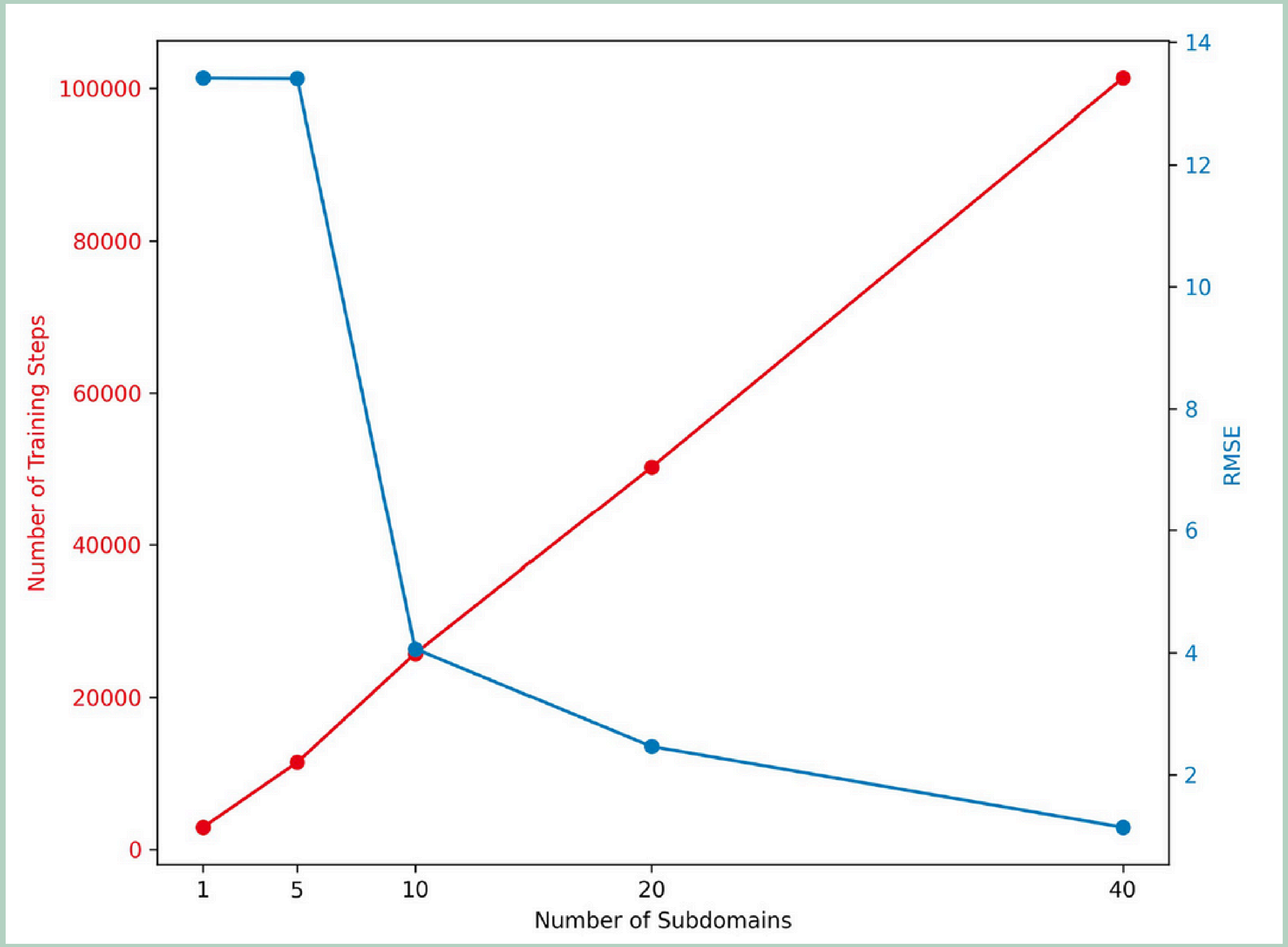
https://benmoseley.blog/blog/

Tutorial in R (CSVDMW, Nicosia 2025): https://avia-gis.quarto.pub/pinn-shiny-tutorial/

Resource page

Computation time (Cuong et al., 2024):

Figure (left) plots the relationship between the number of training steps and the RMSE across a range of subdomains. The training steps increase linearly with the number of subdomains as each subdomain requires approximately 200,000 to 300,000 training steps, equivalent to around 30 minutes on the GPU NVIDIA GeForce RTX 4090 hardware used for all experiments. The figure demonstrates a clear trade-off between the number of subdomains and. model accuracy: as the number of subdomains rise, the RMSE decreases exponentially, but at the cost of longer training times.



(Cuong et al., 2024)