On the convergence of PINNs

NATHAN DOUMÈCHE[1,a], GÉRARD BIAU[1,b] and CLAIRE BOYER[1,c]

[1]*Sorbonne Université, CNRS, LPSM, F-75005 Paris, France* , [a]*nathan.doumeche@sorbonne-universite.fr,* [b]*gerard.biau@sorbonne-universite.fr,* [c]*claire.boyer@sorbonne-universite.fr*

Physics-informed neural networks (PINNs) are a promising approach that combines the power of neural networks with the interpretability of physical modeling. PINNs have shown good practical performance in solving partial differential equations (PDEs) and in hybrid modeling scenarios, where physical models enhance data-driven approaches. However, it is essential to establish their theoretical properties in order to fully understand their capabilities and limitations. In this study, we highlight that classical training of PINNs can suffer from systematic overfitting. This problem can be addressed by adding a ridge regularization to the empirical risk, which ensures that the resulting estimator is risk-consistent for both linear and nonlinear PDE systems. However, the strong convergence of PINNs to a solution satisfying the physical constraints requires a more involved analysis using tools from functional analysis and calculus of variations. In particular, for linear PDE systems, an implementable Sobolev-type regularization allows to reconstruct a solution that not only achieves statistical accuracy but also maintains consistency with the underlying physics.

MSC2020 subject classifications: Primary 62G08; secondary 68T07
Keywords: consistency; hybrid modeling; PDE solver; physics-informed neural networks

1. Introduction

Physics-informed machine learning Advances in machine learning and deep learning have led to significant breakthroughs in almost all areas of science and technology. However, despite remarkable achievements, modern machine learning models are difficult to interpret and do not necessarily obey the fundamental governing laws of physical systems (Linardatos, Papastefanopoulos and Kotsiantis, 2021). Moreover, they often fail to extrapolate scenarios beyond those on which they were trained (Xu et al., 2021). On the contrary, numerical or pure physical methods struggle to capture nonlinear relationships in complex and high-dimensional systems, while lacking flexibility and being prone to computational problems. This state of affairs has led to a growing consensus that data-driven machine learning methods need to be coupled with prior scientific knowledge based on physics. This emerging field, often called physics-informed machine learning (Raissi, Perdikaris and Karniadakis, 2019), seeks to combine the predictive power of machine learning techniques with the interpretability and robustness of physical modeling. The literature in this field is still disorganized, with a somewhat unstable nomenclature. In particular, the terms physics-informed, physics-based, physics-guided, and theory-guided are used interchangeably. For a comprehensive account, we refer to the reviews by Rai and Sahu (2020), Karniadakis et al. (2021), Cuomo et al. (2022), and Hao et al. (2022), which survey some of the prevailing trends in embedding physical knowledge in machine learning, present some of the current challenges, and discuss various applications.

Vocabulary and use cases Depending on the nature of the interaction between machine learning and physics, physics-informed machine learning is usually achieved by preprocessing the features (Rai and Sahu, 2020), by designing innovative network architectures that incorporate the physics of the problem (Karniadakis et al., 2021), or by forcing physics infusion into the loss function (Cuomo et al., 2022). It is this latter approach, which is most often referred to as physics regularization (Rai and Sahu, 2020), to which our article is devoted. Note that other names are possible, including physics consistency penalty

(Wang et al., 2020a), knowledge-based loss term (von Rueden et al., 2023), and physics-guided neural networks (Cunha et al., 2023). In the following, we will focus more specifically on neural networks incorporating a physical regularization, called PINNs (for physics-informed neural networks, Raissi, Perdikaris and Karniadakis 2019). Such models have been successfully applied to (i) model hybrid learning tasks, where the data-driven loss is regularized to satisfy a physical prior, and (ii) design efficient solvers of partial differential equations (PDEs). A significant advantage of PINNs is that they are easy to implement compared to other PDE solvers, and that they rely on the backpropagation algorithm, resulting in reasonable computational cost. Although (i) and (ii) are different facets of the same mathematical problem, they differ in their geometry and the nature of the data on which they are based, as we will see later.

Related work and contributions Despite a rapidly growing literature highlighting the capabilities of PINNs in various real-world applications, there are still few theoretical guarantees regarding the overfitting, consistency, and error analysis of the approach. Most existing theoretical work focuses either on intractable modifications of PINNs (Cuomo et al., 2022) or on negative results, such as in Krishnapriyan et al. (2021) and Wang, Yu and Perdikaris (2022).

Our goal in the present article is to provide a comprehensive theoretical analysis of the mathematical forces driving PINNs, in both the hybrid modeling and PDE solver settings, with the constant concern to provide approaches that can be implemented in practice. Our results complement those of Shin (2020), Shin, Zhang and Karniadakis (2023), Mishra and Molinaro (2023), De Ryck and Mishra (2022), Wu et al. (2022), and Qian et al. (2023) for the PDE solver problem. Shin (2020) and Wu et al. (2022) focus on modifications of PINNs using the Hölder norm of the neural network in the loss function, which is unfortunately intractable in practice. In the context of linear PDEs, Shin, Zhang and Karniadakis (2023) analyze the expected generalization error of PINNs using the Rademacher complexity of the image of the neural network class by a differential operator. However, this Rademacher complexity does not obviously vanish with increasing sample size. Similarly, Mishra and Molinaro (2023) bound the generalization error by a quadrature rule depending on the Hölder norm of the neural network, which does not necessarily tend to zero as the number of training points tends to infinity. De Ryck and Mishra (2022) derive bounds on the expectation of the L^2 error, provided that the weights of the neural networks are bounded. In contrast to this series of works, we consider models and assumptions that can be practically verified or implemented. Moreover, our approach includes hybrid modeling, for which, as pointed out by Karniadakis et al. (2021), no theoretical guarantees have been given so far. Preliminary interesting results on the statistical consistency of a regression function penalized by a PDE are reported in Arnone et al. (2022). The original point of our approach lies in the use of a mix of statistical and functional analysis arguments (Evans, 2010) to characterize the PINN problem.

Overview After correctly defining the PINN problem in Section 2, we show in Section 3 that an additional regularization term is needed in the loss, otherwise PINNs can overfit. This first important result is consistent with the approach of Shin (2020), which penalizes PINNs by Hölder norms to ensure their convergence, and with the experiments of Nabian and Meidani (2020), which improve performance by adding an extra-regularization term. In Section 4, we establish the consistency of ridge PINNs by proving in Theorem 4.6 that a slowly vanishing ridge penalty is sufficient to prevent overfitting. Finally, in Section 5, we show that an additional level of regularization is sufficient in order to guarantee the strong convergence of PINNs (Theorem 5.7). We also prove that an adapted tuning of the hyperparameters allows to reconstruct the solution in the PDE solver setting (Theorem 5.8), as well as to ensure both statistical and physics consistency in the hybrid modeling setting (Theorem 5.13). All proofs are postponed to the Supplementary Material (Doumèche, Biau and Boyer, 2023). The code of all the numerical experiments can be found at https://github.com/NathanDoumeche/Convergence_and_error_analysis_of_PINNs.

2. The PINN framework

In its most general formulation, the PINN method can be described as an empirical risk minimization problem, penalized by a PDE system.

Notation Throughout this article, the symbol $\mathbb{E}$ denotes expectation and $\|\cdot\|_2$ (resp., $\langle\cdot,\cdot\rangle$) denotes the Euclidean norm (resp., scalar product) in $\mathbb{R}^d$, where d may vary depending on the context. Let $\Omega \subset \mathbb{R}^{d_1}$ be a bounded Lipschitz domain with boundary $\partial\Omega$ and closure $\bar{\Omega}$, and let $(\mathbf{X}, Y) \in \Omega \times \mathbb{R}^{d_2}$ be a pair of random variables. Recall that Lipschitz domains are a general category of open sets that includes bounded convex domains (such as $]0,1[^{d_1}$) and usual manifolds with C^1 boundaries (see the Appendix). This level of generality with respect to the domain Ω is necessary to encompass most of the physical problems, such as those presented in Arzani, Wang and D'Souza (2021), which use non-trivial (but Lipschitz) geometries. For $K \in \mathbb{N}$, the space of functions from Ω to $\mathbb{R}^{d_2}$ that are K times continuously differentiable is denoted by $C^K(\Omega, \mathbb{R}^{d_2})$.

Let $C^\infty(\Omega, \mathbb{R}^{d_2}) = \cap_{K\geqslant 0} C^K(\Omega, \mathbb{R}^{d_2})$ be the space of infinitely differentiable functions. The space $C^K(\Omega, \mathbb{R}^{d_2})$ is endowed with the Hölder norm $\|\cdot\|_{C^K(\Omega)}$, defined for any u by $\|u\|_{C^K(\Omega)} = \max_{|\alpha|\leqslant K} \|\partial^\alpha u\|_{\infty,\Omega}$. The space $C^\infty(\bar{\Omega}, \mathbb{R}^{d_2})$ of smooth functions is defined as the subspace of continuous functions $u : \bar{\Omega} \to \mathbb{R}^{d_2}$ satisfying $u|_\Omega \in C^\infty(\Omega, \mathbb{R}^{d_2})$ and, for all $K \in \mathbb{N}$, $\|u\|_{C^K(\Omega)} < \infty$. A differential operator $\mathscr{F} : C^\infty(\Omega, \mathbb{R}^{d_2}) \times \Omega \to \mathbb{R}$ is said to be of order K if it can be expressed as a function over the partial derivatives of order less than or equal to K. For example, the operator $\mathscr{F}(u, \mathbf{x}) = \partial_1 u(\mathbf{x}) \partial^2_{1,2} u(\mathbf{x}) + u(\mathbf{x}) \sin(\mathbf{x})$ has order 2. A summary of the mathematical notation used in this paper is to be found in the Appendix.

Hybrid modeling As in classical regression analysis, we are interested in estimating the unknown regression function $u^\star$ such that $Y = u^\star(\mathbf{X}) + \varepsilon$, for some random noise ε that satisfies $\mathbb{E}(\varepsilon|\mathbf{X}) = 0$. What makes the problem original is that the function $u^\star$ is assumed to satisfy (at least approximately) a collection of $M \geqslant 1$ PDE-type constraints of order at most K, denoted in a standard form by $\mathscr{F}_k(u^\star, \mathbf{x}) \simeq 0$ for $1 \leqslant k \leqslant M$. It is therefore assumed that $u^\star$ can be derived K times. Moreover, there exists some subset $E \subseteq \partial\Omega$ and an boundary/initial condition function $h : E \to \mathbb{R}^{d_2}$ such that, for all $\mathbf{x} \in E$, $u^\star(\mathbf{x}) \simeq h(\mathbf{x})$. We stress that E can be strictly included in Ω, as shown in Example 2.2 for a spatio-temporal domain Ω. The specific case $E = \partial\Omega$ corresponds to Dirichlet boundary conditions.

These constraints model some a priori physical information about $u^\star$. However, this knowledge may be incomplete (e.g., the PDE system may be ill-posed and have no or multiple solutions) and/or imperfect (i.e., there is some modeling error, that is, $\mathscr{F}_k(u^\star, \mathbf{x}) \neq 0$ and $u^\star|_E \neq h$). This again emphasizes that $u^\star$ is not necessarily a solution of the system of differential equations.

Example 2.1 (Maxwell equations). Let $\mathbf{x} = (x, y, z, t) \in \mathbb{R}^3 \times \mathbb{R}_+$, and consider Maxwell equations describing the evolution of an electro-magnetic field $u^\star = (E^\star, B^\star)$ in vacuum, defined by

$$\begin{cases} \mathscr{F}_1(u^\star, \mathbf{x}) = \mathrm{div} E^\star(\mathbf{x}) \\ \mathscr{F}_2(u^\star, \mathbf{x}) = \mathrm{div} B^\star(\mathbf{x}) \\ (\mathscr{F}_3,\ \mathscr{F}_4,\ \mathscr{F}_5)(u^\star, \mathbf{x}) = \partial_t E^\star(\mathbf{x}) - \mathrm{curl} B^\star(\mathbf{x}) \\ (\mathscr{F}_6,\ \mathscr{F}_7,\ \mathscr{F}_8)(u^\star, \mathbf{x}) = \partial_t B^\star(\mathbf{x}) + \mathrm{curl} E^\star(\mathbf{x}), \end{cases}$$

where $E^\star \in C^1(\mathbb{R}^4, \mathbb{R}^3)$ is the electric field, $B^\star \in C^1(\mathbb{R}^4, \mathbb{R}^3)$ the magnetic field, and the div and curl operators are respectively defined for $F = (F_x, F_y, F_z) \in C^1(\mathbb{R}^4, \mathbb{R}^3)$ by

$$\mathrm{div} F = \partial_x F_x + \partial_y F_y + \partial_z F_z \quad \text{and} \quad \mathrm{curl} F = (\partial_y F_z - \partial_z F_y,\ \partial_z F_x - \partial_x F_z,\ \partial_x F_y - \partial_y F_x).$$

In this case, $d_1 = 4$, $d_2 = 6$, and $M = 8$. □

Example 2.2 (Spatio-temporal condition function). Assume that the domain $\Omega \subseteq \mathbb{R}^{d_1}$ is of the form $\Omega = \Omega_1 \times]0,T[$, where $\Omega_1 \subseteq \mathbb{R}^{d_1-1}$ is a bounded Lipschitz domain and $T \geqslant 0$ is a finite time horizon. The spatio-temporal PDE system admits (spatial) boundary conditions specified by a function $f : \partial\Omega_1 \to \mathbb{R}^{d_2}$, i.e.,

$$\forall x \in \partial\Omega_1,\ \forall t \in [0,T], \quad u^\star(x,t) = f(x),$$

and a (temporal) initial condition specified by a function $g : \Omega_1 \to \mathbb{R}^{d_2}$, that is

$$\forall x \in \Omega_1, \quad u^\star(x,0) = g(x).$$

The set on which the boundary and initial conditions are defined is $E = (\Omega_1 \times \{0\}) \cup (\partial\Omega_1 \times [0,T])$, and the associated condition function $h : E \to \mathbb{R}^{d_2}$ is

$$h(\mathbf{x}) = \begin{cases} f(x) \text{ if } \mathbf{x} = (x,t) \in \partial\Omega_1 \times [0,T] \\ g(x) \text{ if } \mathbf{x} = (x,t) \in \Omega_1 \times \{0\}. \end{cases}$$

Notice that $E \subsetneq \partial\Omega$. □

In order to estimate $u^\star$, we assume to have at hand three sets of data:

- (i) A collection of i.i.d. random variables $(\mathbf{X}_1, Y_1), \ldots, (\mathbf{X}_n, Y_n)$ distributed as $(\mathbf{X}, Y) \in \Omega \times \mathbb{R}^{d_2}$, the distribution of which is *unknown*;
- (ii) A collection of i.i.d. random variables $\mathbf{X}_1^{(e)}, \ldots, \mathbf{X}_{n_e}^{(e)}$ distributed according to some *known* distribution μ_E on E;
- (iii) A sample of i.i.d. random variables $\mathbf{X}_1^{(r)}, \ldots, \mathbf{X}_{n_r}^{(r)}$ *uniformly distributed* on Ω.

The function $u^\star$ is then estimated by minimizing the empirical risk function

$$R_{n,n_e,n_r}(u_\theta) = \frac{\lambda_d}{n} \sum_{i=1}^{n} \|u_\theta(\mathbf{X}_i) - Y_i\|_2^2 + \frac{\lambda_e}{n_e} \sum_{j=1}^{n_e} \|u_\theta(\mathbf{X}_j^{(e)}) - h(\mathbf{X}_j^{(e)})\|_2^2$$
$$+ \frac{1}{n_r} \sum_{k=1}^{M} \sum_{\ell=1}^{n_r} \mathscr{F}_k(u_\theta, \mathbf{X}_\ell^{(r)})^2 \tag{1}$$

over the class $\mathrm{NN}_H(D) := \{u_\theta, \theta \in \Theta_{H,D}\}$ of feedforward neural networks with H hidden layers of common width D (see below for a precise definition), where $(\lambda_d, \lambda_e) \in \mathbb{R}_+^2 \backslash (0,0)$ are hyperparameters that establish a tradeoff between the three terms. In practice, one often encounters the case where $\lambda_e = 0$ (data + PDEs). Another situation of interest is when $\lambda_d = 0$ (PDEs + boundary/initial conditions), which corresponds to the special case of a PDE solver. Setting (1) is more general as it includes all the combinations data + PDEs + boundary/initial conditions. Since a minimizer of the empirical risk function (1) does not necessarily exist, we denote by $(\hat{\theta}(p, n_e, n_r, D))_{p \in \mathbb{N}} \in \Theta_{H,D}^{\mathbb{N}}$ any minimizing sequence, i.e.,

$$\lim_{p \to \infty} R_{n,n_e,n_r}(u_{\hat{\theta}(p,n_e,n_r,D)}) = \inf_{\theta \in \Theta_{H,D}} R_{n,n_e,n_r}(u_\theta).$$

In practice, such a sequence is usually obtained by implementing some optimization procedure, the exact description of which is not important for our purpose.

On the practical side, simulations using hybrid modeling have been successfully applied to model image denoising (Wang et al., 2020a), turbulence (Wang et al., 2020b), blood streams (Arzani, Wang

and D'Souza, 2021), wave propagation (Davini et al., 2021), and ocean streams (de Wolff et al., 2021). Experiments with real data have been performed to assess the sea temperature (de Bézenac, Pajot and Gallinari, 2019), subsurface transport (He et al., 2020), fused filament fabrication (Kapusuzoglu and Mahadevan, 2020), seismic response (Zhang, Liu and Sun, 2020), glacier dynamic (Riel, Minchew and Bischoff, 2021), lake temperature (Daw et al., 2022), thermal modeling of buildings (Gokhale, Claessens and Develder, 2022), blasts (Pannell, Rigby and Panoutsos, 2022), and heat transfers (Ramezankhani et al., 2022). The generality and flexibility of the empirical risk function (1) allows it to encompass most PINN-like problems. For example, the case $M \geqslant 2$ is considered in de Bézenac, Pajot and Gallinari (2019) and Riel, Minchew and Bischoff (2021), while Zhang, Liu and Sun (2020) and Wang et al. (2020b) assume that $d_1 = d_2 = 3$. Importantly, the situation where $\lambda_d > 0$ *and* $\lambda_e > 0$ (data + boundary conditions + PDEs) is also interesting from a physical point of view. This is, for example, the approach advocated by Arzani, Wang and D'Souza (2021), which uses both data and boundary conditions (see also Cuomo et al., 2022, and Hao et al., 2022).

The PDE solver case The particular case $\lambda_d = 0$ deserves a special comment. In this setting, without physical measures $(\mathbf{X}_i, Y_i)$, the function $u^\star$ is viewed as the unknown solution of the system of PDEs $\mathscr{F}_1, \dots, \mathscr{F}_M$ with boundary/initial conditions h. The goal is to estimate the solution $u^\star$ of the PDE problem

$$\begin{cases} \forall k,\ \forall \mathbf{x} \in \Omega,\ \mathscr{F}_k(u^\star, \mathbf{x}) = 0 \\ \forall \mathbf{x} \in E, \qquad\qquad u^\star(\mathbf{x}) = h(\mathbf{x}), \end{cases}$$

with neural networks from $\mathrm{NN}_H(D)$. In this case, the empirical risk function (1) becomes

$$R_{n_e,n_r}(u_\theta) = \frac{\lambda_e}{n_e} \sum_{j=1}^{n_e} \|u_\theta(\mathbf{X}_j^{(e)}) - h(\mathbf{X}_j^{(e)})\|_2^2 + \frac{1}{n_r} \sum_{k=1}^{M} \sum_{\ell=1}^{n_r} \mathscr{F}_k(u_\theta, \mathbf{X}_\ell^{(r)})^2,$$

where the boundary and initial conditions $(\mathbf{X}_1^{(e)}, h(\mathbf{X}_1^{(e)})), \dots, (\mathbf{X}_{n_e}^{(e)}, h(\mathbf{X}_{n_e}^{(e)}))$ are sampled on $E \times \mathbb{R}^{d_2}$ according to some known distribution μ_E, and $(\mathbf{X}_1^{(r)}, \dots, \mathbf{X}_{n_r}^{(r)})$ are uniformly distributed on Ω. Note that, for simplicity, we write $R_{n_e,n_r}(u_\theta)$ instead of $R_{n,n_e,n_r}(u_\theta)$ because no $\mathbf{X}_i$ is involved in this context. Since no confusion is possible, the same convention is used for all subsequent risk functions throughout the paper. The first term of $R_{n_e,n_r}(u_\theta)$ measures the gap between the network u_θ and the condition function h on E, while the second term forces u_θ to obey the PDE in a discretized way. Since both the condition function h and the distribution μ_E are known, it is reasonable to think of n_e and n_r as large (up to the computational resources). In this scientific computing perspective, PINNs have been successfully applied to solve a wide variety of linear and nonlinear problems, including motion, advection, heat, Euler, high-frequency Helmholtz, Schrödinger, Blasius, Burgers, and Navier-Stokes equations, covering various fields ranging from classical (mechanics, fluid dynamics, thermodynamics, and electromagnetism) to quantum physics (e.g., Cuomo et al., 2022, Li et al., 2023).

The class of neural networks A fully-connected feedforward neural network with $H \in \mathbb{N}^\star$ hidden layers of sizes $(L_1, \dots, L_H) := (D, \dots, D) \in (\mathbb{N}^\star)^H$ and activation tanh, is a function from $\mathbb{R}^{d_1}$ to $\mathbb{R}^{d_2}$, defined by

$$u_\theta = \mathscr{A}_{H+1} \circ (\tanh \circ \mathscr{A}_H) \circ \cdots \circ (\tanh \circ \mathscr{A}_1),$$

where the hyperbolic tangent function tanh is applied element-wise. Each $\mathscr{A}_k : \mathbb{R}^{L_{k-1}} \to \mathbb{R}^{L_k}$ is an affine function of the form $\mathscr{A}_k(\mathbf{x}) = W_k \mathbf{x} + b_k$, with W_k a $(L_{k-1} \times L_k)$-matrix, $b_k \in \mathbb{R}^{L_k}$ a vector, $L_0 = d_1$, and $L_{H+1} = d_2$. The neural network u_θ is parameterized by $\theta = (W_1, b_1, \dots, W_{H+1}, b_{H+1}) \in \Theta_{H,D}$, where $\Theta_{H,D} = \mathbb{R}^{\sum_{i=0}^{H}(L_i+1)\times L_{i+1}}$. Throughout, we let $\mathrm{NN}_H(D) = \{u_\theta,\ \theta \in \Theta_{H,D}\}$. We emphasize that

the tanh function is the most common activation in PINNs (see, e.g., Cuomo et al., 2022). It is preferable to the classical ReLU $(x) = \max(x, 0)$ activation. In fact, since ReLU neural networks are a subset of piecewise linear functions, their high derivatives vanish and therefore cannot be captured by the penalty term $\frac{1}{n_r}\sum_{k=1}^{M}\sum_{\ell=1}^{n_r}\mathscr{F}_k(u_\theta, \mathbf{X}_\ell^{(r)})^2$.

The parameter space $\mathrm{NN}_H(D)$ must be chosen large enough to approximate both the solutions of the PDEs and their derivatives. This property is encapsulated in Proposition 2.3, which shows that for any number $H \geqslant 2$ of hidden layers, the set $\mathrm{NN}_H := \cup_D \mathrm{NN}_H(D)$ is dense in the space $(C^\infty(\bar{\Omega}, \mathbb{R}^{d_2}), \|\cdot\|_{C^K(\Omega)})$. This generalizes Theorem 5.1 in De Ryck, Lanthaler and Mishra (2021) which states that NN_2 is dense in $(C^\infty([0,1]^{d_1}, \mathbb{R}), \|\cdot\|_{C^K(]0,1[^{d_1})})$ for all $d_1 \geqslant 1$ and $K \in \mathbb{N}$.

Proposition 2.3 (Density of neural networks in Hölder spaces). *Let $K \in \mathbb{N}$, $H \geqslant 2$, and $\Omega \subseteq \mathbb{R}^{d_1}$ be a bounded Lipschitz domain. Then $\mathrm{NN}_H := \cup_D \mathrm{NN}_H(D)$ is dense in $(C^\infty(\bar{\Omega}, \mathbb{R}^{d_2}), \|\cdot\|_{C^K(\Omega)})$, i.e., for any function $u \in C^\infty(\bar{\Omega}, \mathbb{R}^{d_2})$, there exists a sequence $(u_p)_{p\in\mathbb{N}} \in \mathrm{NN}_H^{\mathbb{N}}$ such that $\lim_{p\to\infty}\|u - u_p\|_{C^K(\Omega)} = 0$.*

In the remainder of the article, the number H of hidden layers is considered to be fixed. Krishnapriyan et al. (2021) use $\mathrm{NN}_4(50)$, Xu et al. (2021) take $\mathrm{NN}_5(100)$, whereas Arzani, Wang and D'Souza (2021) employ $\mathrm{NN}_{10}(100)$. It is worth noting that in this series of papers the width D is much larger than H, as in Proposition 2.3.

3. PINNs can overfit

Our goal in this section is to show through two examples how learning with standard PINNs can lead to severe overfitting problems. This weakness has already been noted in Costabal et al. (2020), Nabian and Meidani (2020), Chandrajit et al. (2023), and Esfahani (2023), which propose to improve the performance of their models by resorting to an additional regularization strategy. The pathological cases that we highlight both rely on neural networks with exploding derivatives.

The theoretical risk function is defined by

$$\mathscr{R}_n(u) = \frac{\lambda_d}{n}\sum_{i=1}^{n}\|u(\mathbf{X}_i) - Y_i\|_2^2 + \lambda_e\mathbb{E}\|u(\mathbf{X}^{(e)}) - h(\mathbf{X}^{(e)})\|_2^2 + \frac{1}{|\Omega|}\sum_{k=1}^{M}\int_\Omega \mathscr{F}_k(u, \mathbf{x})^2 d\mathbf{x}. \tag{2}$$

Observe that in $\mathscr{R}_n(u)$ we take expectation with respect to μ_E (for the boundary/initial condition part) and integrate with respect to the uniform measure on Ω (for the PDE part), but keep the term $\sum_{i=1}^{n}\|u_\theta(\mathbf{X}_i) - Y_i\|_2^2$ intact. This regime corresponds to the limit of the empirical risk function (1), holding n fixed and letting $n_e, n_r \to \infty$. The rationale is that while the random samples $(\mathbf{X}_i, Y_i)$ may be limited in number (e.g., because their acquisition is more delicate and require physical measurements), this is not the case for $\mathbf{X}_j^{(e)}$ or $\mathbf{X}_j^{(r)}$, which can be freely sampled (up to computational resources). Note however that in the PDE solver setting, the first term is not included.

Given any minimizing sequence $(\hat{\theta}(p, n_e, n_r, D))_{p\in\mathbb{N}}$ of the empirical risk, satisfying

$$\lim_{p\to\infty} R_{n,n_e,n_r}(u_{\hat{\theta}(p,n_e,n_r,D)}) = \inf_{\theta\in\Theta_{H,D}} R_{n,n_e,n_r}(u_\theta),$$

a natural requirement, called risk-consistency, is that

$$\lim_{n_e,n_r\to\infty}\lim_{p\to\infty}\mathscr{R}_n(u_{\hat{\theta}(p,n_e,n_r,D)}) = \inf_{u\in\mathrm{NN}_H(D)}\mathscr{R}_n(u).$$

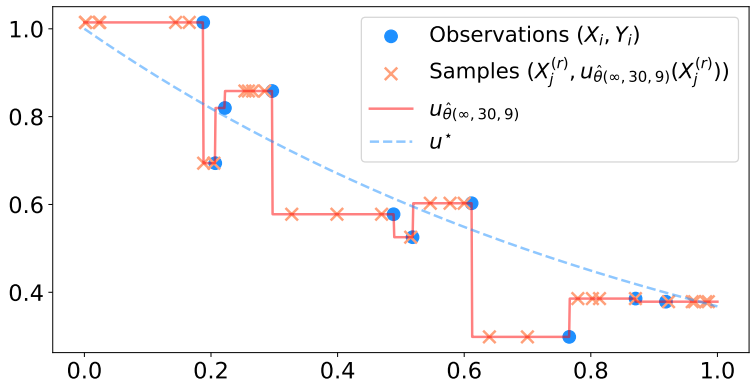

Figure 1. An inconsistent PINN estimator in hybrid modeling with $m = \gamma = 1$, $\varepsilon \sim \mathcal{N}(0, 10^{-2})$, and $n = 10$.

We show below that standard PINNs can dramatically fail to be risk-consistent, through two counterexamples, one in the hybrid modeling context and one in the specific PDE solver setting.

The case of dynamics with friction Consider the following ordinary differential constraint, defined on the domain $\Omega =]0, T[$ (with closure $\bar{\Omega} = [0, T]$) by

$$\forall u \in C^2(\bar{\Omega}, \mathbb{R}),\ \forall \mathbf{x} \in \Omega, \quad \mathscr{F}(u, \mathbf{x}) = mu''(\mathbf{x}) + \gamma u'(\mathbf{x}). \tag{3}$$

This models the dynamics of an object of mass $m > 0$, subjected to a fluid force of friction coefficient $\gamma > 0$. The goal is to reconstruct the real trajectory $u^\star$ by taking advantage of the model $\mathscr{F}$ and the noisy observations Y_i at the $\mathbf{X}_i$. This is an example where the modeling is perfect, i.e., $\mathscr{F}(u^\star, \cdot) = 0$, but the challenge is that the physical model is incomplete because the boundary conditions are unknown. Following the hybrid modeling framework, the trajectory $u^\star$ is estimated by minimizing over the space $\mathrm{NN}_H(D)$ the empirical risk function

$$R_{n,n_r}(u_\theta) = \frac{\lambda_d}{n} \sum_{i=1}^{n} |u_\theta(\mathbf{X}_i) - Y_i|^2 + \frac{1}{n_r} \sum_{\ell=1}^{n_r} \mathscr{F}(u_\theta, \mathbf{X}_\ell^{(r)})^2.$$

Proposition 3.1 (Overfitting). *Consider the dynamics with friction model (3), and assume that there are two observations such that $Y_i \neq Y_j$. Then, whenever $D \geqslant n - 1$, for any integer n_r, for all $\mathbf{X}_1^{(r)}, \ldots, \mathbf{X}_{n_r}^{(r)}$, there exists a minimizing sequence $(u_{\hat{\theta}(p,n_r,D)})_{p\in\mathbb{N}} \in \mathrm{NN}_H(D)^{\mathbb{N}}$ such that $\lim_{p\to\infty} R_{n,n_r}(u_{\hat{\theta}(p,n_r,D)}) = 0$ but $\lim_{p\to\infty} \mathscr{R}_n(u_{\hat{\theta}(p,n_r,D)}) = \infty$. So, this PINN estimator is not consistent.*

Proposition 3.1 illustrates how fitting a PINN by minimizing the empirical risk alone can lead to a catastrophic situation, where the empirical risk of the minimizing sequence is (close to) zero, while its theoretical risk is infinite. This phenomenon is explained by the existence of piecewise constant functions interpolating the observations $\mathbf{X}_1, \ldots, \mathbf{X}_n$, whose derivatives are null at the points $\mathbf{X}_1^{(r)}, \ldots, \mathbf{X}_{n_r}^{(r)}$, but diverge between these points (see Figure 1). These functions correspond to neural networks u_θ such that $\|\theta\|_2 \to \infty$.

PDE solver: The heat propagation case Consider the heat propagation differential operator defined on the domain $\Omega =]-1,1[\times]0,T[$ (with closure $\bar{\Omega} = [-1,1]\times[0,T]$) by

$$\forall u \in C^2(\bar{\Omega},\mathbb{R}),\ \forall \mathbf{x}\in\Omega, \quad \mathscr{F}(u,\mathbf{x}) = \partial_t u(\mathbf{x}) - \partial^2_{x,x} u(\mathbf{x}), \tag{4}$$

associated with the boundary conditions

$$\forall t \in [0,T], \quad u(-1,t) = u(1,t) = 0,$$

and the initial condition defined, for all $x \in [-1,1]$, by

$$u(x,0) = \tanh^{\circ H}(x+0.5) - \tanh^{\circ H}(x-0.5) + \tanh^{\circ H}(0.5) - \tanh^{\circ H}(1.5).$$

The notation $\tanh^{\circ k}$ stands for the function recursively defined by $\tanh^{\circ 1} = \tanh$ and $\tanh^{\circ(k+1)} = \tanh\circ\tanh^{\circ k}$. The unique solution $u^\star$ of the PDE is shown in Figure 2 (right). It models the time evolution of the temperature of a wire, whose extremities at $x=-1$ and $x=1$ are maintained at zero temperature. Note that the initial condition corresponds to a bell-shaped function, which belongs to $\mathrm{NN}_H(2)$. However, the setting can be extended to arbitrary initial conditions that take the form of a neural network function, given the boundary condition $u(\partial\Omega\times[0,T]) = \{0\}$.

To solve the PDE (4), we use n_e i.i.d. samples $\mathbf{X}^{(e)}_1,\dots,\mathbf{X}^{(e)}_{n_e}$ on $E = ([-1,1]\times\{0\})\cup(\{-1,1\}\times[0,T])$, distributed according to μ_E, together with n_r i.i.d. samples $\mathbf{X}^{(r)}_1,\dots,\mathbf{X}^{(r)}_{n_r}$, uniformly distributed on Ω. Let $(\hat{\theta}(p,n_e,n_r,D))_{p\in\mathbb{N}}$ be a sequence of parameters minimizing the empirical risk function

$$R_{n_e,n_r}(u_\theta) = \frac{\lambda_e}{n_e}\sum_{j=1}^{n_e} |u_\theta(\mathbf{X}^{(e)}_j) - h(\mathbf{X}^{(e)}_j)|^2 + \frac{1}{n_r}\sum_{\ell=1}^{n_r} \mathscr{F}(u_\theta,\mathbf{X}^{(r)}_\ell)^2,$$

over the space $\mathrm{NN}_H(D)$. The theoretical counterpart of this empirical risk is

$$\mathscr{R}(u) = \lambda_e \mathbb{E}|u(\mathbf{X}^{(e)}) - h(\mathbf{X}^{(e)})|^2 + \frac{1}{|\Omega|}\int_\Omega \mathscr{F}(u,\mathbf{x})^2 d\mathbf{x}.$$

Proposition 3.2 (PDE solver overfitting). *Consider the heat propagation model (4). Then, whenever $D \geqslant 4$, for any pair (n_e,n_r), for all $\mathbf{X}^{(e)}_1,\dots,\mathbf{X}^{(e)}_{n_e}$ and for all $\mathbf{X}^{(r)}_1,\dots,\mathbf{X}^{(r)}_{n_r}$, there exists a minimizing sequence $(u_{\hat{\theta}(p,n_e,n_r,D)})_{p\in\mathbb{N}} \in \mathrm{NN}_H(D)^{\mathbb{N}}$ such that $\lim_{p\to\infty} R_{n_e,n_r}(u_{\hat{\theta}(p,n_e,n_r,D)}) = 0$ but $\lim_{p\to\infty}\mathscr{R}(u_{\hat{\theta}(p,n_e,n_r,D)}) = \infty$. So, this PINN estimator is not consistent.*

Figure 2 (left) shows an example of an inconsistent PINN estimator. Such an estimator corresponds to a function that equals zero on Ω (and thus satisfies the linear PDE), while satisfying the initial condition on $\partial\Omega$. This function corresponds to a limit of neural networks u_θ such that $\|\theta\|_2 \to \infty$.

The proof strategy of Propositions 3.1 and 3.2 does not depend on the geometry of the points $\mathbf{X}^{(r)}$ and the points $\mathbf{X}^{(e)}$, which could therefore be sampled along a grid, or by any quasi Monte Carlo method. We emphasize that the two negative examples of Propositions 3.1 and 3.2 are no exceptions. In fact, their proofs can be easily generalized to differential operators $\mathscr{F}$ such that the following property holds: for all $\mathbf{x}\in\Omega$, for all $u\in C^\infty(\Omega,\mathbb{R}^{d_2})$, if ∇u vanishes on an open set containing $\mathbf{x}$, then $\mathscr{F}(u,\mathbf{x}) = 0$. This property is satisfied in the case of motion with friction, advection, heat, wave propagation, Schrödinger, Maxwell and Navier-Stokes equations, which are so as many cases that will suffer from overfitting.

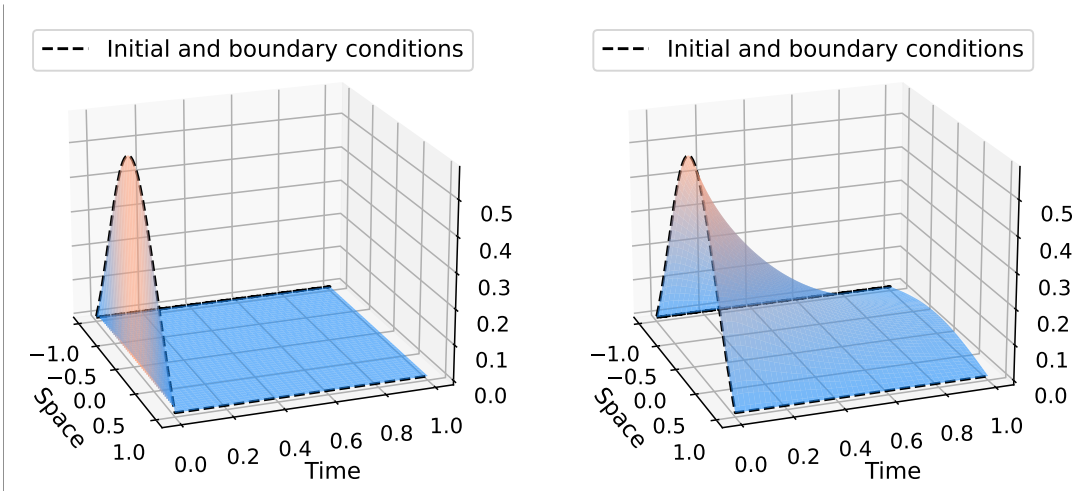

Figure 2. Inconsistent PINN (left) compared to the solution $u^\star$ of the PDE (right) for the heat propagation case.

4. Consistency of regularized PINNs for linear and nonlinear PDE systems

Training PINNs can be tricky because it can lead to the type of pathological situations highlighted in Section 3. To avoid such an overfitting behavior, a standard approach in machine learning is to resort to ridge regularization, where the empirical risk to be minimized is penalized by the L^2 norm of the parameters θ. This technique has been shown to improve not only the optimization convergence during the training phase, but also the generalization ability of the resulting predictor (Guo et al., 2017, Krogh and Hertz, 1991). Ridge regularization is available in most deep learning libraries (e.g., `pytorch` or `keras`), where it is implemented using the so-called weight decay (Loshchilov and Hutter, 2019). Interestingly, the ridge regularization of a slight modification of PINNs, using adaptive activation functions, has been studied in Jagtap, Kawaguchi and Karniadakis (2020), which shows that gradient descent algorithms manage to generate an effective minimizing sequence of the penalized empirical risk. In this section, we formalize ridge PINNs and study their risk-consistency.

Definition 4.1 (Ridge PINNs). The ridge risk function is defined by

$$R^{(\text{ridge})}_{n,n_e,n_r}(u_\theta) = R_{n,n_e,n_r}(u_\theta) + \lambda_{(\text{ridge})}\|\theta\|_2^2, \tag{5}$$

where $\lambda_{(\text{ridge})} > 0$ is the ridge hyperparameter. We denote by $(\hat{\theta}^{(\text{ridge})}_{(p,n_e,n_r,D)})_{p\in\mathbb{N}}$ a minimizing sequence of this risk, i.e.,

$$\lim_{p\to\infty} R^{(\text{ridge})}_{n,n_e,n_r}(u_{\hat{\theta}^{(\text{ridge})}_{(p,n_e,n_r,D)}}) = \inf_{\theta\in\Theta} R^{(\text{ridge})}_{n,n_e,n_r}(u_\theta).$$

Our next Proposition 4.2 states that the L^2 norm of the parameters θ bounds the Hölder norm of the neural network u_θ. This result is interesting in itself because it establishes a connection between the L^2 norm of a fully connected neural network and its regularity. (Note that, by equivalence of the norms, this result also holds if the ridge penalty is replaced by $\|\theta\|_p^p$.) In the present paper it plays a key role in the risk-consistency analysis.

Proposition 4.2 (Bounding the norm of a neural network by the norm of its parameter). *Consider the class* $\mathrm{NN}_H(D) = \{u_\theta, \theta \in \Theta_{H,D}\}$. *Let* $K \in \mathbb{N}$. *Then there exists a constant* $C_{K,H} > 0$, *depending only on* K *and* H, *such that, for all* $\theta \in \Theta_{H,D}$,

$$\|u_\theta\|_{C^K(\mathbb{R}^{d_1})} \leqslant C_{K,H}(D+1)^{HK+1}(1+\|\theta\|_2)^{HK}\|\theta\|_2.$$

Moreover, this bound is tight with respect to $\|\theta\|_2$, *in the sense that, for all* $H, D \geqslant 1$ *and all* $K \in \mathbb{N}$, *there exists a sequence* $(\theta_p)_{p\in\mathbb{N}} \in \mathrm{NN}_H(D)$ *and a constant* $\bar{C}_{K,H} > 0$ *such that* (i) $\lim_{p\to\infty}\|\theta_p\|_2 = \infty$ *and* (ii) $\|u_{\theta_p}\|_{C^K(\mathbb{R}^{d_1})} \geqslant \bar{C}_{K,H}\|\theta_p\|_2^{HK+1}$.

In order to study the generalization capabilities of regularized PINNs, we need to restrict the PDEs to a class of smooth differential operators, which we call polynomial operators (Definition 4.4 below). This class includes the most common PDE systems, as shown in the following example with the Navier-Stokes equations.

Example 4.3 (Navier-Stokes equations). Let $\Omega = \Omega_1 \times]0,T[$, where $\Omega_1 \subseteq \mathbb{R}^3$ is a bounded Lipschitz domain and $T \geqslant 0$ is a finite time horizon. The incompressible Navier-Stokes system of equations is defined for all $u = (u_x, u_y, u_z, p) \in C^2(\bar{\Omega}, \mathbb{R}^4)$ and for all $\mathbf{x} = (x, y, z, t) \in \Omega$, by

$$\begin{cases} \mathscr{F}_1(u,\mathbf{x}) = \partial_t u_x - (u_x\partial_x + u_y\partial_y + u_z\partial_z)u_x - \eta(\partial^2_{x,x} + \partial^2_{y,y} + \partial^2_{z,z})u_x + \rho^{-1}\partial_x p \\ \mathscr{F}_2(u,\mathbf{x}) = \partial_t u_y - (u_x\partial_x + u_y\partial_y + u_z\partial_z)u_y - \eta(\partial^2_{x,x} + \partial^2_{y,y} + \partial^2_{z,z})u_y + \rho^{-1}\partial_y p \\ \mathscr{F}_3(u,\mathbf{x}) = \partial_t u_z - (u_x\partial_x + u_y\partial_y + u_z\partial_z)u_z - \eta(\partial^2_{x,x} + \partial^2_{y,y} + \partial^2_{z,z})u_z + \rho^{-1}\partial_z p + g(\mathbf{x}) \\ \mathscr{F}_4(u,\mathbf{x}) = \partial_x u_x + \partial_y u_y + \partial_z u_z, \end{cases}$$

where $\eta, \rho > 0$ and $g \in C^\infty(\bar{\Omega}, \mathbb{R})$. Observe that $\mathscr{F}_1, \mathscr{F}_2, \mathscr{F}_3$, and $\mathscr{F}_4$ are polynomials in u and its derivatives, with coefficients in $C^\infty(\bar{\Omega}, \mathbb{R})$. For example, $\mathscr{F}_3(u,\mathbf{x}) = P_3(u_x, u_y, u_z, \partial_x u_z, \partial_y u_z, \partial_z u_z, \partial_t u_z, \partial^2_{x,x}u_z, \partial^2_{y,y}u_z, \partial^2_{z,z}u_z, \partial_z p)(\mathbf{x})$, where the polynomial $P_3 \in C^\infty(\bar{\Omega}, \mathbb{R})[Z_1, \ldots, Z_{11}]$ is defined by $P_3(Z_1, \ldots, Z_{11}) = Z_7 - Z_1Z_4 - Z_2Z_5 - Z_3Z_6 - \eta(Z_8 + Z_9 + Z_{10}) + \rho^{-1}Z_{11} + g$. □

The above example can be generalized with the following definition.

Definition 4.4 (Polynomial operator). An operator $\mathscr{F} : C^K(\bar{\Omega}, \mathbb{R}^{d_2}) \times \Omega \to \mathbb{R}$ is a polynomial operator of order $K \in \mathbb{N}$ if there exists an integer $s \in \mathbb{N}$ and multi-indexes $(\alpha_{i,j})_{1\leqslant i\leqslant d_2, 1\leqslant j\leqslant s} \in (\mathbb{N}^{d_1})^{sd_2}$ such that

$$\forall u = (u_1, \ldots, u_{d_2}) \in C^K(\bar{\Omega}, \mathbb{R}^{d_2}), \quad \mathscr{F}(u,\cdot) = P((\partial^{\alpha_{i,j}}u_i)_{1\leqslant i\leqslant d_2, 1\leqslant j\leqslant s}),$$

where $P \in C^\infty(\bar{\Omega}, \mathbb{R})[Z_{1,1}, \ldots, Z_{d_2,s}]$ is a polynomial with smooth coefficients.

In other words, $\mathscr{F}$ is a polynomial operator if it is of the form

$$\mathscr{F}(u,\mathbf{x}) = \sum_{k=1}^{N(P)} \phi_k \times \prod_{i=1}^{d_2}\prod_{j=1}^{s}(\partial^{\alpha_{i,j}}u_i(\mathbf{x}))^{I(i,j,k)},$$

where $N(P) \in \mathbb{N}^\star$, $\phi_k \in C^\infty(\bar{\Omega}, \mathbb{R})$, and $I(i,j,k) \in \mathbb{N}$. The associated polynomial is $P(Z_{1,1}, \ldots, Z_{d_2,s}) = \sum_{k=1}^{N(P)} \phi_k \times \prod_{i=1}^{d_2}\prod_{j=1}^{s} Z_{i,j}^{I(i,j,k)}$ (recall that $\partial^\alpha u_i = u_i$ when $\alpha = 0$).

Definition 4.5 (Degree). The degree of the polynomial operator $\mathscr{F}$ is

$$\deg(\mathscr{F}) = \max_{1 \leqslant k \leqslant N(P)} \sum_{i=1}^{d_2} \sum_{j=1}^{s} (1 + |\alpha_{i,j}|) I(i,j,k).$$

As an illustration, in Example 4.3, one has $\deg(\mathscr{F}_3) = 3$, and this degree is reached in both the terms $u_z \partial_z u_z$ and $\partial^2_{z,z} u_z$. Note that $\deg(P_3) = 2$ but $\deg(\mathscr{F}_3) = 3$. To compute $\deg(\mathscr{F}_3)$, we first count the number of terms in each monomial ($u_z \partial_z u_z$ has two terms while $\partial^2_{z,z} u_z$ has one term), which is $\sum_{i=1}^{d_2} \sum_{j=1}^{s} I(i,j,k)$ for the kth monomial, and add the number of derivatives involved in the product ($u_z \partial_z u_z$ contains a single ∂_z operator while $\partial^2_{z,z} u_z$ contains two derivatives in ∂_z), which corresponds to $\sum_{i=1}^{d_2} \sum_{j=1}^{s} |\alpha_{i,j}| I(i,j,k)$ for the kth monomial. Thus, for each monomial k, the total sum is $\sum_{i=1}^{d_2} \sum_{j=1}^{s} (1 + |\alpha_{i,j}|) I(i,j,k)$.

We emphasize that this class includes a large number of PDEs, such as linear PDEs (e.g., advection, heat, and Maxwell equations), as well as some nonlinear PDEs (e.g., Blasius, Burger's, and Navier-Stokes equations). Proposition 4.2 is a key ingredient to uniformly bound the risk of PINNs involving polynomial PDE operators (see Doumèche, Biau and Boyer, 2023, Supplementary Material, Section 5). This in turn can be used to establish the risk-consistency of these PINNs when n_e and n_r tend to ∞, as follows.

Theorem 4.6 (Risk-consistency of ridge PINNs). *Consider the ridge PINN problem (5), over the class* $\mathrm{NN}_H(D) = \{u_\theta, \theta \in \Theta_{H,D}\}$*, where* $H \geqslant 2$*. Assume that the condition function* h *is Lipschitz and that* $\mathscr{F}_1, \dots, \mathscr{F}_M$ *are polynomial operators. Assume, in addition, that the ridge parameter is of the form*

$$\lambda_{(\mathrm{ridge})} = \min(n_e, n_r)^{-\kappa}, \quad \textit{where} \quad \kappa = \frac{1}{12 + 4H(1 + (2+H)\max_k \deg(\mathscr{F}_k))}.$$

Then, almost surely,

$$\lim_{n_e, n_r \to \infty} \lim_{p \to \infty} \mathscr{R}_n(u_{\hat{\theta}^{(\mathrm{ridge})}(p, n_e, n_r, D)}) = \inf_{u \in \mathrm{NN}_H(D)} \mathscr{R}_n(u).$$

Thus, minimizing the ridge empirical risk (5) over $\Theta_{H,D}$ amounts to minimizing the theoretical risk (2) over $\Theta_{H,D}$ in the asymptotic regime $n_e, n_r \to \infty$. This fundamental result is complemented by the following one, which resorts to another asymptotics in the width D. This ensures that the choice of the neural architecture $\mathrm{NN}_H \subseteq C^\infty(\bar{\Omega}, \mathbb{R}^{d_2})$ does not introduce any asymptotic bias.

Theorem 4.7 (The ridge PINN is asymptotically unbiased). *Under the same assumptions as in Theorem 4.6, one has, almost surely,*

$$\lim_{D \to \infty} \lim_{n_e, n_r \to \infty} \lim_{p \to \infty} \mathscr{R}_n(u_{\hat{\theta}^{(\mathrm{ridge})}(p, n_e, n_r, D)}) = \inf_{u \in C^\infty(\bar{\Omega}, \mathbb{R}^{d_2})} \mathscr{R}_n(u).$$

In other words, minimizing the ridge empirical risk over $\Theta_{H,D}$ and letting $D, n_e, n_r \to \infty$ amounts to minimizing the theoretical risk (2) over the entire class $C^\infty(\bar{\Omega}, \mathbb{R}^{d_2})$. We emphasize that these two theorems hold independently of the values of the hyperparameters $\lambda_d, \lambda_e \geqslant 0$. Therefore, our results cover the general hybrid modeling framework (1), which includes the PDE solver. To the best of our knowledge, these are the first results that provide theoretical guarantees for PINNs regularized with a standard penalty. They complement the state-of-the-art approaches of Shin (2020), Shin, Zhang and

Karniadakis (2023), Mishra and Molinaro (2023), and Wu et al. (2022), which consider regularization strategies that are unfortunately not feasible in practice.

It is worth noting that Theorem 4.7 still holds by choosing D as a function of n_e and n_r. In fact, an easy modification of the proofs reveals that one can take $D(n_e, n_r) = \min(n_e, n_r)^{\xi}$, where ξ is a constant depending only on H and $\max_k \deg(\mathscr{F}_k)$. Thus, in this setting,

$$\lim_{n_e,n_r\to\infty} \lim_{p\to\infty} \mathscr{R}_n(u_{\hat{\theta}^{(\mathrm{ridge})}(p,n_e,n_r,D(n_e,n_r))}) = \inf_{u\in C^\infty(\bar{\Omega},\mathbb{R}^{d_2})} \mathscr{R}_n(u).$$

Remark 4.8 (Dirichlet boundary conditions). Theorems 4.6 and 4.7 can be easily adapted to PINNs with Von Neumann conditions instead of Dirichlet boundary conditions. This is achieved by substituting the term $n_e^{-1}\sum_{j=1}^{n_e} \|u_\theta(\mathbf{X}_j^{(e)}) - h(\mathbf{X}_j^{(e)})\|_2^2$ in the PINN definition (1) by $n_e^{-1}\sum_{j=1}^{n_e} \|\partial_{\overrightarrow{n}} u_\theta(\mathbf{X}_j^{(e)})\|_2^2$, where $\overrightarrow{n}$ is the normal to $\partial\Omega$.

Practical considerations The decay rate of $\lambda_{(\mathrm{ridge})} = \min(n_e, n_r)^{-\kappa}$ does not depend on the dimension d_1 of Ω. This is consistent with the results of Karniadakis et al. (2021) and De Ryck and Mishra (2022), which suggest that PINNs can overcome the curse of dimensionality, opening up interesting perspectives for efficient solvers of high-dimensional PDEs. We also emphasize that $\lambda_{(\mathrm{ridge})}$ depends only on the degree of the polynomial PDE operator, the depth H, and the sample sizes n_e and n_r. All these quantities are known, which makes this hyperparameter immediately useful for practical applications. For example, in Navier-Stokes equations of Example 4.3, one has $\max_k \deg(\mathscr{F}_k) = 3$. Thus, for a neural network of depth, say $H = 2$, the ridge hyperparameter $\lambda_{(\mathrm{ridge})} = \min(n_e, n_r)^{-1/116}$ is sufficient to ensure consistency. It is also interesting to note that the bound on $\lambda_{(\mathrm{ridge})}$ in the theorems deteriorates with increasing depth H. This confirms the preferential use of shallow neural networks in the experimental works of Arzani, Wang and D'Souza (2021), Karniadakis et al. (2021), and Xu et al. (2021). The bound also deteriorates as $\max_k \deg \mathscr{F}_k$ increases. This is in line with the empirical results of Davini et al. (2021), which was able to improve the performance of PINNs by reformulating their polynomial differential equation of degree 3 as a system of two polynomial differential equations of degree 2.

It is also interesting to note that Theorems 4.6 and 4.7 hold for any ridge hyperparameter $\lambda_{(\mathrm{ridge})} \geqslant \min(n_e, n_r)^{-\kappa}$ such that $\lim_{n_e,n_r\to\infty} \lambda_{(\mathrm{ridge})} = 0$. However, if n_e and n_r are fixed, choosing too large a $\lambda_{(\mathrm{ridge})}$ will lead to a bias toward parameters of $\Theta_{H,D}$ with a low L^2 norm. Therefore, there is a trade-off between taking $\lambda_{(\mathrm{ridge})}$ as small as possible to reduce this bias, but large enough to avoid overfitting, as illustrated in Section 3. Moreover, our choice of $\lambda_{(\mathrm{ridge})}$ may be suboptimal, since these results rely on inequalities involving a general class of polynomial operators. When studying a particular PDE, the consistency results of Theorems 4.6 and 4.7 should eventually hold with a smaller $\lambda_{(\mathrm{ridge})}$. To tune $\lambda_{(\mathrm{ridge})}$ in practice, one could, for example, monitor the overfitting gap $\mathrm{OG}_{n,n_e,n_r} = |R_{n,n_e,n_r} - \mathscr{R}_n|$ for a ridge estimator $\hat{\theta}^{(\mathrm{ridge})}(p, n_e, n_r, D)$, by standard validation strategy (e.g., by sampling $\tilde{n}_r$ and $\tilde{n}_e$ new points to estimate $\mathscr{R}_n(u_{\hat{\theta}^{(\mathrm{ridge})}(p,n_e,n_r,D)})$ at a $\min(\tilde{n}_r, \tilde{n}_e)^{-1/2}$-rate given by the central limit theorem), and then choose the smallest parameter $\lambda_{(\mathrm{ridge})}$ to introduce as little bias as possible. More information about the relevance of OG_{n,n_e,n_r} is given in Doumèche, Biau and Boyer (2023, Supplementary Material, Section 2).

5. Strong convergence of PINNs for linear PDE systems

Beyond risk-consistency concerns, the ultimate goal of PINNs is to learn a physics-informed regression function $u^\star$, or, in the PDE solver setting, to strongly approximate the unique solution $u^\star$ of a PDE

system. Thus, what we want is to have guarantees regarding the convergence of $u_{\hat{\theta}^{(\mathrm{ridge})}(p,n_e,n_r,D)}$ to $u^\star$ for an adapted norm. This requirement is called strong convergence in the functional analysis literature. This is however not guaranteed under the sole convergence of the theoretical risk $(\mathscr{R}_n(u_{\hat{\theta}^{(\mathrm{ridge})}(p,n_e,n_r,D)}))_{p,n_e,n_r,D\in\mathbb{N}}$, as shown in the following two examples.

Example 5.1 (Lack of data incorporation in the hybrid modeling setting). Suppose $M = 1$, $d_1 = 2$, $d_2 = 1$, $\Omega =]0,1[\times]0,T[$, $h(x,0) = 1$ and $h(0,t) = 1$, and let $\mathscr{F}(u,\mathbf{x}) = \partial_x u(\mathbf{x}) + \partial_t u(\mathbf{x})$. This corresponds to the assumption that the solution should approximately follow the advection equation and that it should be close to 1. For any $\delta > 0$, let the sequence $(u_{\delta,p})_{p\in\mathbb{N}} \in \mathrm{NN}_H(2n)^{\mathbb{N}}$ be defined by

$$u_{\delta,p}(x,t) = 1 + \sum_{i=1}^{n} \frac{Y_i}{2}\big(\tanh_p^{\circ H}(x - t - x_i + t_i + \delta) - \tanh_p^{\circ H}(x - t - x_i + t_i - \delta)\big),$$

where $\tanh_p := \tanh(p\,\cdot)$, and $\mathbf{X}_i = (x_i, t_i)$. Then, as soon as $\delta \leqslant \frac{1}{2}\min_{i\neq j}|x_i - x_j + t_j - t_i|$, we have that $\lim_{p\to\infty}\mathscr{R}_n(u_{\delta,p}) = 0$. Thus, as long as $D \geq 2n$, $\inf_{u\in\mathrm{NN}_H(D)}\mathscr{R}_n(u) = 0$. Therefore, Theorem 4.7 shows that $\lim_{D\to\infty}\lim_{n_e,n_r\to\infty}\lim_{p\to\infty}\mathscr{R}_n(u_{\hat{\theta}^{(\mathrm{ridge})}(p,n_e,n_r,D)}) = 0$. It is then easy to check that this implies that $u_{\hat{\theta}^{(\mathrm{ridge})}(p,n_e,n_r,D)}$ converges in $L^2(\Omega)$ to 1, independently of n and the function $u^\star$. This shows that the ridge PINNs fails to learn $u^\star$ whenever the model is inexact. □

In the PDE solver setting, one can consider the a priori favorable case where the PDE system admits a unique (strong) solution $u^\star$ in $C^K(\bar{\Omega},\mathbb{R}^{d_2})$ (where K is the maximum order of the differential operators $\mathscr{F}_1$, …, $\mathscr{F}_M$). Note that $u^\star$ is the unique minimizer of $\mathscr{R}$ over $C^K(\bar{\Omega},\mathbb{R}^{d_2})$, with $\mathscr{R}(u^\star) = 0$ (and $\mathscr{R}(u) = 0$ if and only if u satisfies the initial conditions, the boundary conditions, and the system of differential equations). However, we describe below a situation where a minimizing sequence of $\mathscr{R}$ does not converge to the unique strong solution $u^\star$ of the PDE in question.

Example 5.2 (Divergence in the PDE solver setting). Suppose $M = 1$, $d_1 = d_2 = 1$, $\Omega =]-1,1[$, $h(1) = 1$, $\lambda_e > 0$, and let the polynomial operator be $\mathscr{F}(u,\mathbf{x}) = \mathbf{x}u'(\mathbf{x})$. Clearly, $u^\star(\mathbf{x}) = 1$ is the only strong solution of the PDE $\mathbf{x}u'(\mathbf{x}) = 0$ with $u(1) = 1$. Let the sequence $(u_p)_{p\in\mathbb{N}} \in \mathrm{NN}_H(D)^{\mathbb{N}}$ be defined by $u_p = \tanh_p \circ \tanh^{\circ(H-1)}$. According to Doumèche, Biau and Boyer (2023, Supplementary Material, Section 2), $\lim_{p\to\infty}\mathscr{R}(u_p) = \mathscr{R}(u^\star) = 0$. However, the minimizing sequence $(u_p)_{p\in\mathbb{N}}$ does not converge to $u^\star$, since $u_\infty(\mathbf{x}) := \lim_{p\to\infty} u_p(\mathbf{x}) = \mathbf{1}_{\mathbf{x}>0} - \mathbf{1}_{\mathbf{x}<0}$. □

We have therefore exhibited a sequence $(u_p)_{p\in\mathbb{N}}$ of neural networks that minimizes $\mathscr{R}$ and such that $(u_p)_{p\in\mathbb{N}}$ converges pointwise. However, its limit u_∞ is not the unique strong solution of the PDE. In fact, u_∞ is not differentiable at 0, which is incompatible with the differential operators $\mathscr{F}$ used in $\mathscr{R}(u_\infty)$. Interestingly, the Cauchy-Schwarz inequality states that the pathological sequence $(u_p)_{p\in\mathbb{N}}$ satisfies $\lim_{p\to\infty}\|u_p'\|^2_{L^2(\Omega)} = \infty$, as in Example 5.1.

5.1. Sobolev regularization

The two examples above illustrate how the convergence of the theoretical risk $\mathscr{R}_n(u_{\hat{\theta}^{(\mathrm{ridge})}(p,n_e,n_r,D)})$ to $\inf_{u\in C^\infty(\bar{\Omega},\mathbb{R}^{d_2})}\mathscr{R}_n(u)$ (for any n) is not sufficient to guarantee the strong convergence to a PDE or hybrid modeling solution. To ensure such a convergence, a different analysis is needed, mobilizing tools from functional analysis. In the sequel, we build upon the regression estimation penalized by PDEs of Azzimonti et al. (2015), Sangalli (2021), Arnone et al. (2022), and Ferraccioli, Sangalli and

Finos (2022), and make use of the calculus of variations (e.g., Evans, 2010, Theorems 1-4, Chapter 8). In the former references, the minimizer of $\mathscr{R}_n$ does not satisfy the PDE system injected in the PINN penalty, but another PDE system, known as the Euler-Lagrange equations. Although interesting, the mathematical framework is different from ours. First, the authors do not study the convergence of neural networks, but rather methods in which the boundary conditions are hard-coded, such as the finite element method. Second, these frameworks are limited to special cases of theoretical risks. Indeed, only second-order PDEs with $\lambda_e = \infty$ are considered in Azzimonti et al. (2015), while Evans (2010) deal with first-order PDEs, echoing the case of $\lambda_d = 0$ and $\lambda_e = \infty$.

It is worthwhile mentioning that the results of Azzimonti et al. (2015) rely on an important property of the theoretical risk function $\mathscr{R}_n$, called coercivity. This is a common assumption of the calculus of variations (Evans, 2010). The operator $\mathscr{R}_n$ is said to be coercive if there exist $K \in \mathbb{N}$ and $\lambda_t > 0$ such that, for all $u \in H^K(\Omega, \mathbb{R}^{d_2})$, $\mathscr{R}_n(u) \geqslant \lambda_t \|u\|^2_{H^K(\Omega)}$ (the notation $H^K(\Omega, \mathbb{R}^{d_2})$ stands for the usual Sobolev space of order K—see the Appendix. It turns out that the failures of Examples 5.1 and 5.2 are due to a lack of coercivity, since, in both cases, $\lim_{p\to\infty} \|u_p\|_{H^1(\Omega)} = \infty$ but $\lim_{p\to\infty} \mathscr{R}_n(u_p) \leqslant \mathscr{R}_n(u^\star)$. There are two ways to correct this problem: either one can restrict the study to coercive operators only, or one can resort to an explicit regularization of the risk to enforce its coercivity. We choose the latter, since most PDEs used in the practice of PINNs are not coercive. Note however that our results could be easily adapted to the coercive case.

In the following, we restrict ourselves to affine operators, which exactly correspond to linear PDE systems, including the advection, heat, wave, and Maxwell equations.

Definition 5.3 (Affine operator). The operator $\mathscr{F}$ is affine of order K if there exists $A_\alpha \in C^\infty(\bar{\Omega}, \mathbb{R}^{d_2})$ and $B \in C^\infty(\bar{\Omega}, \mathbb{R})$ such that, for all $\mathbf{x} \in \Omega$ and all $u \in H^K(\Omega, \mathbb{R}^{d_2})$,

$$\mathscr{F}(u, \mathbf{x}) = \mathscr{F}^{(\mathrm{lin})}(u, \mathbf{x}) + B(\mathbf{x}),$$

where $\mathscr{F}^{(\mathrm{lin})}(u, \mathbf{x}) = \sum_{|\alpha| \leqslant K} \langle A_\alpha(\mathbf{x}), \partial^\alpha u(\mathbf{x}) \rangle$ is linear.

The source term B is important, as it makes it possible to model a large variety of applied physical problems, as illustrated in Song, Alkhalifah and Waheed (2021). Note also that affine operators of order K are in fact polynomial operators of degree $K + 1$ (Definitions 4.4 and 4.5) that are extended from smooth functions to the whole Sobolev space $H^K(\Omega, \mathbb{R}^{d_2})$.

Definition 5.4 (Regularized PINNs). The regularized theoretical risk function is

$$\mathscr{R}_n^{(\mathrm{reg})}(u) = \mathscr{R}_n(u) + \lambda_t \|u\|^2_{H^{m+1}(\Omega)}, \tag{6}$$

where $\mathscr{R}_n$ is the original theoretical risk as defined in (2), and $m \in \mathbb{N}$. The corresponding regularized empirical risk function is

$$R^{(\mathrm{reg})}_{n,n_e,n_r}(u_\theta) = R_{n,n_e,n_r}(u_\theta) + \lambda_{(\mathrm{ridge})} \|\theta\|_2^2 + \frac{\lambda_t}{n_\ell} \sum_{\ell=1}^{n_\ell} \sum_{|\alpha| \leqslant m+1} \|\partial^\alpha u_\theta(\mathbf{X}_\ell^{(r)})\|_2^2.$$

It is noteworthy that $R^{(\mathrm{reg})}_{n,n_e,n_r}$ can be straightforwardly implemented in the usual PINN framework and benefit from the computational scalability of the backpropagation algorithm, by encoding the regularization as supplementary PDE-type constraints $\mathscr{F}_\alpha(u, \mathbf{x}) = \partial^\alpha u(\mathbf{x}) = 0$. Since this discretized Sobolev penalty can be seen as additional physical priors $\mathscr{F}_\alpha$, the overfitting behavior observed for the

unregularized PINNs can be transferred to Sobolev-regularized PINNs trained without ridge regularization. This is why the ridge penalty is still included in the risk. Note also that the Sobolev regularization has been shown to avoid overfitting in machine learning, yet in different contexts (e.g., Fischer and Steinwart, 2020).

The following proposition shows that the unique minimizer of (6) can be interpreted as the unique minimizer of an optimization problem involving a weak formulation of the differential terms included in the risk. Its proof is based on the Lax-Milgram theorem (e.g., Brezis, 2010, Corollary 5.8).

Proposition 5.5 (Characterization of the unique minimizer of $\mathscr{R}_n^{(\mathrm{reg})}$). *Assume that $\mathscr{F}_1, \dots, \mathscr{F}_M$ are affine operators of order K. Assume, in addition, that $\lambda_t > 0$ and $m \geqslant \max(\lfloor d_1/2 \rfloor, K)$. Then the regularized theoretical risk $\mathscr{R}_n^{(\mathrm{reg})}$ has a unique minimizer $\hat{u}_n$ over $H^{m+1}(\Omega, \mathbb{R}^{d_2})$. This minimizer $\hat{u}_n$ is the unique element of $H^{m+1}(\Omega, \mathbb{R}^{d_2})$ that satisfies*

$$\forall v \in H^{m+1}(\Omega, \mathbb{R}^{d_2}), \quad \mathscr{A}_n(\hat{u}_n, v) = \mathscr{B}_n(v),$$

where

$$\begin{aligned}
\mathscr{A}_n(\hat{u}_n, v) = {} & \frac{\lambda_d}{n} \sum_{i=1}^{n} \langle \tilde{\Pi}(\hat{u}_n)(\mathbf{X}_i), \tilde{\Pi}(v)(\mathbf{X}_i) \rangle + \lambda_e \mathbb{E} \langle \tilde{\Pi}(\hat{u}_n)(\mathbf{X}^{(e)}), \tilde{\Pi}(v)(\mathbf{X}^{(e)}) \rangle \\
& + \frac{1}{|\Omega|} \sum_{k=1}^{M} \int_{\Omega} \mathscr{F}_k^{(\mathrm{lin})}(\hat{u}_n, \mathbf{x}) \mathscr{F}_k^{(\mathrm{lin})}(v, \mathbf{x}) d\mathbf{x} \\
& + \frac{\lambda_t}{|\Omega|} \sum_{|\alpha| \leqslant m+1} \int_{\Omega} \langle \partial^\alpha \hat{u}_n(\mathbf{x}), \partial^\alpha v(\mathbf{x}) \rangle d\mathbf{x}, \\
\mathscr{B}_n(v) = {} & \frac{\lambda_d}{n} \sum_{i=1}^{n} \langle Y_i, \tilde{\Pi}(v)(\mathbf{X}_i) \rangle + \lambda_e \mathbb{E} \langle \tilde{\Pi}(v)(\mathbf{X}^{(e)}), h(\mathbf{X}^{(e)}) \rangle \\
& - \frac{1}{|\Omega|} \sum_{k=1}^{M} \int_{\Omega} B_k(\mathbf{x}) \mathscr{F}_k^{(\mathrm{lin})}(v, \mathbf{x}) d\mathbf{x},
\end{aligned}$$

and where $\tilde{\Pi} : H^{m+1}(\Omega, \mathbb{R}^{d_2}) \to C^0(\Omega, \mathbb{R}^{d_2})$ is the so-called Sobolev embedding, such that $\tilde{\Pi}(u)$ is the unique continuous function that coincides with u almost everywhere.

The Sobolev embedding $\tilde{\Pi}$ is essential in order to give a precise meaning to the pointwise evaluation at the points $\mathbf{X}_i$ of a function $u \in H^{m+1}(\Omega, \mathbb{R}^{d_2}) \subseteq L^2(\Omega, \mathbb{R}^{d_2})$, which is defined only almost everywhere. The rationale behind Proposition 5.5 is that

$$\mathscr{R}_n^{(\mathrm{reg})}(u) = \mathscr{A}_n(u, u) - 2\mathscr{B}_n(u) + \frac{\lambda_d}{n} \sum_{i=1}^{n} \|Y_i\|^2 + \lambda_e \mathbb{E} \|h(\mathbf{X}^{(e)})\|_2^2 + \frac{1}{|\Omega|} \sum_{k=1}^{M} \int_{\Omega} B_k(\mathbf{x})^2 d\mathbf{x}.$$

Therefore, minimizing $\mathscr{R}_n^{(\mathrm{reg})}$ amounts to minimizing $\mathscr{A}_n - 2\mathscr{B}_n$. It is also interesting to note that the weak formulation $\mathscr{A}_n(\hat{u}, v) = \mathscr{B}_n(v)$ can be interpreted as a weak PDE on $H^{m+1}(\Omega, \mathbb{R}^{d_2})$. In particular, if $\hat{u}_n \in H^{2(m+1)}(\Omega, \mathbb{R}^{d_2})$, then one has, almost everywhere,

$$\sum_{k=1}^{M} (\mathscr{F}_k^{(\mathrm{lin})})^* \mathscr{F}_k(\hat{u}_n, \mathbf{x}) + \lambda_t \sum_{|\alpha| \leqslant m+1} (-1)^{|\alpha|} (\partial^\alpha)^2 \hat{u}_n(\mathbf{x}) = 0.$$

$(\mathscr{F}_k^{(\mathrm{lin})})^*$ is the adjoint operator of $\mathscr{F}_k^{(\mathrm{lin})}$ such that, for all $u, v \in C^\infty(\Omega, \mathbb{R})$ with $v|_{\partial\Omega} = 0$,

$$\int_\Omega u \mathscr{F}^{(\mathrm{lin})}(v, \mathbf{x}) d\mathbf{x} = \int_\Omega (\mathscr{F}_k^{(\mathrm{lin})})^*(u, \mathbf{x}) v d\mathbf{x}.$$

Thus, even in the regime $\lambda_t \to 0$ (i.e., when the regularization becomes negligible), the solution of the PINN problem does not satisfy the constraints $\mathscr{F}_k(u, \mathbf{x}) = 0$, but the following constraint $\sum_{k=1}^M (\mathscr{F}_k^{(\mathrm{lin})})^* \mathscr{F}_k(u, \mathbf{x}) = 0$. (Notice that, in the PDE solver setting, since $u^\star$ satisfies all the constraints, it satisfies in particular the constraint $\sum_{k=1}^M (\mathscr{F}_k^{(\mathrm{lin})})^* \mathscr{F}_k(u^\star, \mathbf{x}) = 0$.) For instance, the advection equation constraint $\mathscr{F}(u, \mathbf{x}) = (\partial_x + \partial_t)u(\mathbf{x})$ of Example 5.1 becomes $\mathscr{F}^* \mathscr{F}(u, \mathbf{x}) = -(\partial_x + \partial_t)^2 u(\mathbf{x})$, and the constraint $\mathscr{F}(u, \mathbf{x}) = \mathbf{x}u'(\mathbf{x})$ of Example 5.2 becomes $\mathscr{F}^* \mathscr{F}(u, \mathbf{x}) = -2\mathbf{x}u'(\mathbf{x}) - \mathbf{x}^2 u''(\mathbf{x})$.

Proposition 5.5 shows that the regularization in λ_t is sufficient to make the PINN problem well-posed, i.e., to ensure that the theoretical risk function (6) admits a unique minimizer. The next natural requirement is that the regularized PINN estimator obtained by minimizing the regularized empirical risk function converges to this unique minimizer $\hat{u}_n$. Proposition 5.6 and Theorem 5.7 show that this is true for linear PDE systems.

Proposition 5.6 (From risk-consistency to strong convergence). *Assume that $\lambda_t > 0$ and $m \geqslant \max(\lfloor d_1/2 \rfloor, K)$. Let $(u_p)_{p\in\mathbb{N}} \in C^\infty(\bar{\Omega}, \mathbb{R}^{d_2})$ be a sequence of smooth functions satisfying that $\lim_{p\to\infty} \mathscr{R}_n^{(\mathrm{reg})}(u_p) = \inf_{u\in C^\infty(\bar{\Omega},\mathbb{R}^{d_2})} \mathscr{R}_n^{(\mathrm{reg})}$. Then $\lim_{p\to\infty} \|u_p - \hat{u}_n\|_{H^m(\Omega)} = 0$, where $\hat{u}_n$ is the unique minimizer of $\mathscr{R}_n^{(\mathrm{reg})}$ over $H^{m+1}(\Omega, \mathbb{R}^{d_2})$.*

The next theorem follows from Theorem 4.7 and Proposition 5.6, by simply observing that the Sobolev regularization is just an ordinary PINN regularization, taking the form of a polynomial operator of degree $(m+2)$.

Theorem 5.7 (Strong convergence of regularized PINNs). *Assume that $\mathscr{F}_1, \ldots, \mathscr{F}_M$ are affine operators of order K. Assume, in addition, that $\lambda_t > 0$, $m \geqslant \max(\lfloor d_1/2 \rfloor, K)$, and the condition function h is Lipschitz. Let $(\hat{\theta}^{(\mathrm{reg})}(p, n_e, n_r, D))_{p\in\mathbb{N}}$ be a minimizing sequence of the regularized empirical risk function*

$$R_{n,n_e,n_r}^{(\mathrm{reg})}(u_\theta) = R_{n,n_e,n_r}(u_\theta) + \lambda_{(\mathrm{ridge})}\|\theta\|_2^2 + \frac{\lambda_t}{n_\ell} \sum_{\ell=1}^{n_\ell} \sum_{|\alpha|\leqslant m+1} \|\partial^\alpha u_\theta(\mathbf{X}_\ell^{(r)})\|_2^2$$

over the class $\mathrm{NN}_H(D) = \{u_\theta, \theta \in \Theta_{H,D}\}$, where $H \geqslant 2$. Then, with the choice

$$\lambda_{(\mathrm{ridge})} = \min(n_e, n_r)^{-\kappa}, \quad \text{where} \quad \kappa = \frac{1}{12 + 4H(1 + (2+H)(m+2))},$$

one has, almost surely,

$$\lim_{D\to\infty} \lim_{n_e,n_r\to\infty} \lim_{p\to\infty} \|u_{\hat{\theta}^{(\mathrm{reg})}(p,n_e,n_r,D)} - \hat{u}_n\|_{H^m(\Omega)} = 0,$$

where $\hat{u}_n$ is the unique minimizer of $\mathscr{R}_n^{(\mathrm{reg})}$ over $H^{m+1}(\Omega, \mathbb{R}^{d_2})$.

Theorem 5.7 ensures that the sequence $u_{\hat{\theta}^{(\mathrm{reg})}(p,n_e,n_r,D)}$ of PINNs converges to the unique minimizer $\hat{u}_n$ of the regularized theoretical risk function (6), provided that the ridge hyperparameter $\lambda_{(\mathrm{ridge})}$

vanishes slowly enough. However, it does not provide any information about the proximity between $u_{\hat{\theta}^{(\mathrm{reg})}(p,n_e,n_r,D)}$ and $u^\star$. On the other hand, since the regularized theoretical risk function is a small perturbation of the theoretical risk function (2), it is reasonable to think that its minimizer $\hat{u}_n$ should in some way converge to $u^\star$ as $\lambda_t \to 0$. This is encapsulated in Theorem 5.8 for the PDE solver setting and in Theorem 5.13 for the more general hybrid modeling setting.

5.2. The PDE solver case

Theorem 5.8 (Strong convergence of linear PDE solvers). *Assume that $\mathscr{F}_1, \dots, \mathscr{F}_M$ are affine operators of order K. Consider the PDE solver setting (i.e., $\lambda_e > 0$ and $\lambda_d = 0$) and assume that the condition function h is Lipschitz. Assume, in addition, that the PDE system admits a unique solution $u^\star$ in $H^{m+1}(\Omega, \mathbb{R}^{d_2})$ for some $m \geqslant \max(\lfloor d_1/2 \rfloor, K)$. Let $(\hat{\theta}^{(\mathrm{reg})}(p, n_e, n_r, D, \lambda_t))_{p\in\mathbb{N}}$ be a minimizing sequence of the regularized empirical risk function*

$$R^{(\mathrm{reg})}_{n_e,n_r}(u_\theta) = R_{n_e,n_r}(u_\theta) + \lambda_{(\mathrm{ridge})}\|\theta\|_2^2 + \frac{\lambda_t}{n_\ell}\sum_{\ell=1}^{n_\ell}\sum_{|\alpha|\leqslant m+1}\|\partial^\alpha u_\theta(\mathbf{X}_\ell^{(r)})\|_2^2$$

over the class $\mathrm{NN}_H(D) = \{u_\theta, \theta \in \Theta_{H,D}\}$, where $H \geqslant 2$. Then, with the choice

$$\lambda_{(\mathrm{ridge})} = \min(n_e, n_r)^{-\kappa}, \quad \textit{where} \quad \kappa = \frac{1}{12 + 4H(1 + (2+H)(m+2))},$$

one has, almost surely,

$$\lim_{\lambda_t\to 0}\lim_{D\to\infty}\lim_{n_e,n_r\to\infty}\lim_{p\to\infty}\|u_{\hat{\theta}^{(\mathrm{reg})}(p,n_e,n_r,D,\lambda_t)} - u^\star\|_{H^m(\Omega)} = 0.$$

Back to Example 5.2, one has $m = 1$. Recall that, in this setting, the unique minimizer of $\mathscr{R}$ over $C^0([-1,1],\mathbb{R})$ is $u^\star(\mathbf{x}) = 1$, satisfying $u^\star \in H^2(]-1,1[,\mathbb{R})$. Therefore, by letting $\lambda_t \to 0$, this theorem shows that any sequence minimizing the regularized empirical risk function converges, with respect to the $H^2(\Omega)$ norm, to the unique strong solution $u^\star$ of the PDE $\mathbf{x}u'(\mathbf{x}) = 0$ and $u(1) = 1$.

Remark 5.9 (Dimensionless hyperparameters and lower regularity assumptions on $u^\star$). The condition $m \geqslant \lfloor d_1/2 \rfloor$ in Theorem 5.7 is necessary to make the pointwise evaluations $\tilde{\Pi}(u)(\mathbf{X}_i)$ continuous. This condition does have an impact on $\lambda_{(\mathrm{ridge})}$, which grows exponentially fast with the dimension d_1. However, in the PDE solver setting, it is possible to get rid of this dimension problem, taking $m = \max_k \deg(\mathscr{F}_k)$. To see this, just note that there is no $\mathbf{X}_i$, and so there is no need to resort to the $\tilde{\Pi}(u)(\mathbf{X}_i)$. Indeed, the proof of Theorem 5.8 can be adapted by replacing the Sobolev inequalities in the proofs of Theorem 5.7 by the trace theorem for Lipschitz domains (e.g., Grisvard, 2011, Theorem 1.5.1.10). In this case, it is enough to assume that $u^\star \in H^{K+1}(\Omega, \mathbb{R}^{d_2})$, which is less restrictive than $u^\star \in H^{\max(\lfloor d_1/2 \rfloor, K)+1}(\Omega, \mathbb{R}^{d_2})$. However, this comes at the price of assuming that μ_E admits a density with respect to the hypersurface measure on $\partial\Omega$ (as it is often the case in practice).

5.3. The hybrid modeling case

To apply Theorem 5.7 to the general hybrid modeling setting, it is necessary to measure the gap between $u^\star$ and the model specified by the constraints $\mathscr{F}_1, \dots, \mathscr{F}_M$ and the condition function h. This is encapsulated in the next definition.

Definition 5.10 (Physics inconsistency). For any $u \in H^{m+1}(\Omega, \mathbb{R}^{d_2})$, the physics inconsistency of u is defined by

$$\mathrm{PI}(u) = \lambda_e \mathbb{E}\|\tilde{\Pi}(u)(\mathbf{X}^{(e)}) - h(\mathbf{X}^{(e)})\|_2^2 + \frac{1}{|\Omega|}\sum_{k=1}^{M}\int_\Omega \mathscr{F}_k(u,\mathbf{x})^2 d\mathbf{x}.$$

Observe that $\mathscr{R}_n(u) = \frac{\lambda_d}{n}\sum_{i=1}^n \|\tilde{\Pi}(u)(\mathbf{X}_i) - Y_i\|_2^2 + \mathrm{PI}(u)$. In short, the quantity $\mathrm{PI}(u)$ measures how well the boundary/initial conditions, encoded by h, and the PDE system, encoded by the $\mathscr{F}_k$, describe the function u (see also Willard et al., 2023). In particular, $\mathrm{PI}(u^\star)$ measures the modeling error—the better the model, the lower $\mathrm{PI}(u^\star)$.

Proposition 5.11 (Strong convergence of hybrid modeling). *Assume that the conditions of Theorem 5.7 are satisfied. Then $\hat{u}_n \equiv \hat{u}_n(\mathbf{X}_1, \ldots, \mathbf{X}_n, \varepsilon_1, \ldots, \varepsilon_n)$ is a random variable such that $\mathbb{E}\|\hat{u}_n\|^2_{H^{m+1}(\Omega)} < \infty$.*

Suppose, in addition, that $u^\star \in H^{m+1}(\Omega, \mathbb{R}^{d_2})$, that the noise ε is independent from $\mathbf{X}$, and that ε has the same distribution as $-\varepsilon$. Then there exists a constant $C_\Omega > 0$, depending only on Ω, such that

$$\begin{aligned}\mathbb{E}\int_\Omega \|\tilde{\Pi}(\hat{u}_n - u^\star)\|_2^2 d\mu_{\mathbf{X}} \leqslant \frac{1}{\lambda_d}\big(\mathrm{PI}(u^\star) + \lambda_t\|u^\star\|^2_{H^{m+1}(\Omega)}\big) \\ + \frac{C_\Omega d_2^{1/2}}{n^{1/2}}\Big(2\|u^\star\|^2_{H^{m+1}(\Omega)} + \frac{\mathrm{PI}(u^\star)}{\lambda_t}\Big) \\ + \frac{8\mathbb{E}\|\varepsilon\|_2^2}{n}\Big(1 + C_\Omega d_2^{3/2}\Big(\frac{\lambda_d}{\lambda_t} + \frac{\lambda_d^2}{\lambda_t^2 n^{1/2}}\Big)\Big).\end{aligned}$$

In particular, with the choice $\lambda_e = 1$, $\lambda_t = (\log n)^{-1}$, and $\lambda_d = n^{1/2}/(\log n)$, one has

$$\mathbb{E}\int_\Omega \|\tilde{\Pi}(\hat{u}_n - u^\star)\|_2^2 d\mu_{\mathbf{X}} \leqslant \frac{\Lambda \log^2(n)}{n^{1/2}},$$

where $\Lambda = 24 d_2^{3/2} C_\Omega(\mathrm{PI}(u^\star) + \|u^\star\|_{H^{m+1}(\Omega)} + \mathbb{E}\|\varepsilon\|_2^2)$.

This (nonasymptotic) proposition provides an insight into the scaling of the PINN hyperparameters. Indeed, the term $\frac{1}{\lambda_d}(\mathrm{PI}(u^\star) + \lambda_t\|u^\star\|_{H^{m+1}(\Omega)})$ encapsulates the modeling error, damped by the weight λ_d. However, λ_d cannot be arbitrarily large because of the term $\frac{8\mathbb{E}\|\varepsilon\|_2^2}{n}\big(1 + C_\Omega d_2^{3/2}\big(\frac{\lambda_d}{\lambda_t} + \frac{\lambda_d^2}{\lambda_t^2 n^{1/2}}\big)\big)$. So, there is a trade-off between the modeling error and the random variation in the data. Note also the other trade-off in the regularization hyperparameter λ_t, which should not converge to 0 too quickly because of the term $\frac{C_\Omega d_2^{1/2}}{n^{1/2}}\big(2\|u^\star\|^2_{H^{m+1}(\Omega)} + \frac{\mathrm{PI}(u^\star)}{\lambda_t}\big)$.

Proposition 5.12 (Physics consistency of hybrid modeling). *Under the conditions of Proposition 5.11, if $\lim_{n\to\infty}\frac{\lambda_d^2}{n\lambda_t} = 0$ and $\lim_{n\to\infty}\lambda_t = 0$, one has*

$$\mathbb{E}(\mathrm{PI}(\hat{u}_n)) \leqslant \mathrm{PI}(u^\star) + \underset{n\to\infty}{\mathrm{o}}(1).$$

(Note that the conditions are satisfied with $\lambda_e = 1$, $\lambda_t = (\log n)^{-1}$, and $\lambda_d = n^{1/2}/(\log n)$.)

As usual, we let $(u^{(n)}_{\hat{\theta}^{(\mathrm{reg})}(p,n_e,n_r,D)})_{p\in\mathbb{N}} \in \mathrm{NN}_H(D)^{\mathbb{N}}$ be a minimizing sequence of $R^{(\mathrm{reg})}_{n,n_e,n_r}$, where the exponent n indicates that the sample size n is kept fixed along the sequence. Since $u^{(n)}_{\hat{\theta}^{(\mathrm{reg})}(p,n_e,n_r,D)} \in C^\infty(\bar{\Omega},\mathbb{R}^{d_2})$, one has $\tilde{\Pi}(u^{(n)}_{\hat{\theta}^{(\mathrm{reg})}(p,n_e,n_r,D)}) = u^{(n)}_{\hat{\theta}^{(\mathrm{reg})}(p,n_e,n_r,D)}$. Thus, by combining Theorem 5.7 with Propositions 5.11 and 5.12, we obtain the following important theorem.

Theorem 5.13 (Strong convergence of regularized PINNs). *Under the same assumptions as in Theorem 5.7 and Proposition 5.11, with the choice $\lambda_e = 1$, $\lambda_t = (\log n)^{-1}$, and $\lambda_d = n^{1/2}/(\log n)$, one has*

$$\lim_{D\to\infty}\lim_{n_e,n_r\to\infty}\lim_{p\to\infty}\mathbb{E}\int_\Omega \|u^{(n)}_{\hat{\theta}^{(\mathrm{reg})}(p,n_e,n_r,D)} - u^\star\|_2^2 d\mu_{\mathbf{X}} \leqslant \frac{\Lambda\log^2(n)}{n^{1/2}}$$

and

$$\lim_{D\to\infty}\lim_{n_e,n_r\to\infty}\lim_{p\to\infty}\mathbb{E}(\mathrm{PI}(u^{(n)}_{\hat{\theta}^{(\mathrm{reg})}(p,n_e,n_r,D)})) \leqslant \mathrm{PI}(u^\star) + \underset{n\to\infty}{\mathrm{o}}(1).$$

The minimax regression rate over any bounded class of functions in $C^{(m+1)}(\Omega,\mathbb{R}^{d_2})$ is known to be $n^{-2(m+1)/(2(m+1)+d_1)}$ (Stone, 1982, Theorem 1). Theorem 5.13 shows that the regularized PINN estimator achieves the rate $\log(n)/n^{1/2}$ over any *larger* class bounded in $H^{(m+1)}(\Omega,\mathbb{R}^{d_2})$. Thus, the regularized PINN estimator has the nearly optimal rate, up to a log term, in the regime $d_1 \to \infty$ and $m = \lfloor d_1/2 \rfloor$.

Theorem 5.13 shows that a properly regularized PINN estimator is both statistically *and* physics consistent, in the sense that the error $\mathbb{E}\int_\Omega \|u^{(n)}_{\hat{\theta}^{(\mathrm{reg})}(p,n_e,n_r,D)} - u^\star\|_2^2 d\mu_{\mathbf{X}}$ converges to zero with a physics inconsistency $\mathbb{E}(\mathrm{PI}(u^{(n)}_{\hat{\theta}^{(\mathrm{reg})}(p,n_e,n_r,D)}))$ that is asymptotically no larger than $\mathrm{PI}(u^\star)$. It is also worth mentioning that in some applications, the physical measures $\mathbf{X}_1,\ldots,\mathbf{X}_n$ are forced to be sampled in certain subset of Ω. An important application is when Ω is spatio-temporal and one wishes to extrapolate/transfer a model from a training dataset collected on $\mathrm{supp}(\mu_{\mathbf{X}}) = \Omega_1\times]0,T_{\mathrm{train}}[$ to a test $\Omega_1\times]T_{\mathrm{train}},T_{\mathrm{test}}[$, using a temporal evolution PDE system to extrapolate (e.g., Cai et al., 2021). On the other hand, the physical restriction on the data measurement can be also strictly spatial. This is for example the case in some blood modeling problems, where the blood flow measures can only be taken in a specific region of a blood vessel, as illustrated in Arzani, Wang and D'Souza (2021). Thus, in both these contexts, the support $\mathrm{supp}(\mu_{\mathbf{X}})$ of the distribution $\mu_{\mathbf{X}}$ is strictly contained in Ω. Of course, this is compatible with Theorem 5.13, which shows that the regularized PINN estimator consistently interpolates the function $u^\star$ on $\mathrm{supp}(\mu_{\mathbf{X}})$. Furthermore, Theorem 5.13 shows that the estimator uses the physical model to extrapolate on $\Omega\backslash\mathrm{supp}(\mu_{\mathbf{X}})$. In summary, the better the model, the lower the modeling error $\mathrm{PI}(u^\star)$, and the better the domain adaptation capabilities. This provides an interesting mathematical insight into the relevance of combining data-driven statistical models with the interpretability and extrapolation capabilities of physical modeling.

Numerical illustration of imperfect modeling In the following experiments, we illustrate with a toy example the results of Theorem 5.13 and show how the Sobolev regularization can be implemented directly in the PINN framework, taking advantage of the automatic differentiation and backpropagation. Let $\Omega =]0,1[^2$ and assume that $Y = u^\star(\mathbf{X}) + \mathcal{N}(0,10^{-2})$, where $u^\star(x,t) = \exp(t-x) + 0.1\cos(2\pi x)$. In this hybrid modeling setting, the goal is to reconstruct $u^\star$. We consider an advection model of the form $\mathscr{F}(u,\mathbf{x}) = \partial_x u(\mathbf{x}) + \partial_t u(\mathbf{x})$, with $h(x,0) = \exp(-x)$ and $h(0,t) = \exp(t)$. The unique solution of this PDE is $u_{\mathrm{model}}(x,t) = \exp(t-x)$ (Figure 5, left). Note that the function u_{model} is different from $u^\star$ (Figure 5, middle), which casts our problem in the imperfect modeling setting. This PDE prior is relevant because $\|u_{\mathrm{model}} - u^\star\|^2_{L^2(\Omega)} \simeq \exp(-5.3)$ and $\mathrm{PI}(u^\star) \simeq \exp(-1.6)$, two quantities that are negligible

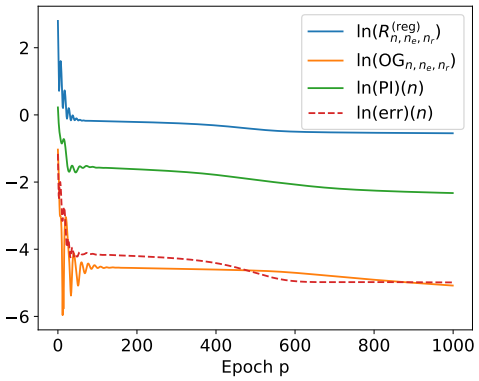

Figure 3. Regularized empirical risk (blue) and overfitting gap OG (orange) with respect to the number p of epochs for $n = 10$. The physics inconsistency $\mathrm{PI}(n)$ (green) and the L^2 error $\mathrm{err}(n)$ (red) are also depicted.

with respect to $\|u^\star\|^2_{L^2(\Omega)} \simeq \exp(0.3)$. We randomly sample n observations $\mathbf{X}_1, \dots, \mathbf{X}_n$ uniformly on the rectangle $\mathrm{supp}(\mu_{\mathbf{X}}) =]0, 0.5[\times]0, 1[\subsetneq \Omega$ (note that this is a strict inclusion), and let n vary from $n_{\min} = 10$ to $n_{\max} = 10^3$ (linearly in a log scale).

The architecture of the neural networks is set to $H = 2$ hidden layers with width $D = 100$, so that the total number of parameters is $10\,600 \gg n_{\max}$. We fix $n_e, n_r = 10^4 \gg n_{\max}$ and $\lambda_{(\mathrm{ridge})} = \min(n_e, n_r)^{-1/2}$. Figure 3 shows the evolution of the regularized risk $R^{(\mathrm{reg})}_{n,n_e,n_r}(u^{(n)}_{\hat\theta^{(\mathrm{reg})}(p,n_r,n_e,D)})$ in blue, with respect to the number p of epochs in the gradient descent (for $n = 10$). For a fixed number n of observations, the number $p_{\max}$ of epochs to stop training is determined by monitoring the evolution of the risk $R^{(\mathrm{reg})}_{n,n_e,n_r}(u^{(n)}_{\hat\theta^{(\mathrm{reg})}(p_{\max},n_r,n_e,D)})$ (blue curve) and the overfitting gap $\mathrm{OG}_{n,n_e,n_r} = |R^{(\mathrm{reg})}_{n,n_e,n_r} - \mathscr{R}^{(\mathrm{reg})}_n|$ (orange curve). Both are required to be stable around a minimal value, so that the minimum of the risk is approximately reached, i.e., $R^{(\mathrm{reg})}_{n,n_e,n_r}(u^{(n)}_{\hat\theta^{(\mathrm{reg})}(p_{\max},n_r,n_e,D)}) \simeq \inf_{u\in\mathrm{NN}_H(D)} R^{(\mathrm{reg})}_{n,n_e,n_r}(u)$ and $\mathscr{R}^{(\mathrm{reg})}_n(u^{(n)}_{\hat\theta^{(\mathrm{reg})}(p_{\max},n_r,n_e,D)}) \simeq \inf_{u\in\mathrm{NN}_H(D)} \mathscr{R}^{(\mathrm{reg})}_n(u)$. In this overparameterized regime (D is large), one can consider that $\mathscr{R}^{(\mathrm{reg})}_n(u^{(n)}_{\hat\theta^{(\mathrm{reg})}(p_{\max},n_r,n_e,D)}) \simeq \inf_{u\in C^\infty(\bar\Omega,\mathbb{R}^{d_2})} \mathscr{R}^{(\mathrm{reg})}_n(u)$ (Theorem 4.7). Keeping n_e, n_r, and λ_{ridge} fixed, the proximity between the PINN and $u^\star$ is measured by

$$\mathrm{err}(n) = 2\int_0^{0.5}\int_0^1 \|u^{(n)}_{\hat\theta^{(\mathrm{reg})}(p_{\max},n_r,n_e,D)}(x,t) - u^\star(x,t)\|_2^2 dxdt.$$

According to Theorem 5.13, there exists some constant $\Lambda > 0$ such that, approximately,

$$\ln\big(\mathbb{E}(\mathrm{err}(n))\big) \lesssim \ln(\Lambda) - \frac{\ln(n)}{2}.$$

This bound is validated numerically in Figure 4, attesting a linear rate in log-log scale between $\mathrm{err}(n)$ and n of $-0.69 \leqslant -0.5$. Furthermore, the second statement of Theorem 5.13 suggests that $\ln \mathrm{PI}(n) = \ln \mathrm{PI}(u^{(n)}_{\hat\theta^{(\mathrm{reg})}(p_{\max},n_r,n_e,D)}) \leqslant \ln \mathrm{PI}(u^\star) = -1.6$, which is also verified in Figure 4. Interestingly, the regularized PINN estimator quickly becomes more accurate than the initial model, since $\mathrm{err}(n)$ is less than $\int_\Omega \|u_{\mathrm{model}} - u^\star\|_2^2 d\mu_{\mathbf{X}} \simeq \exp(-5.3)$ as soon as $\ln(n) > 2.8$, i.e., $n \geqslant 17$.

The obtained regularized PINN estimator for $n = 10^3$ is shown in Figure 5 (right). This estimator looks globally similar to the model u_{model} (Figure 5, left) while managing to reconstruct the variation

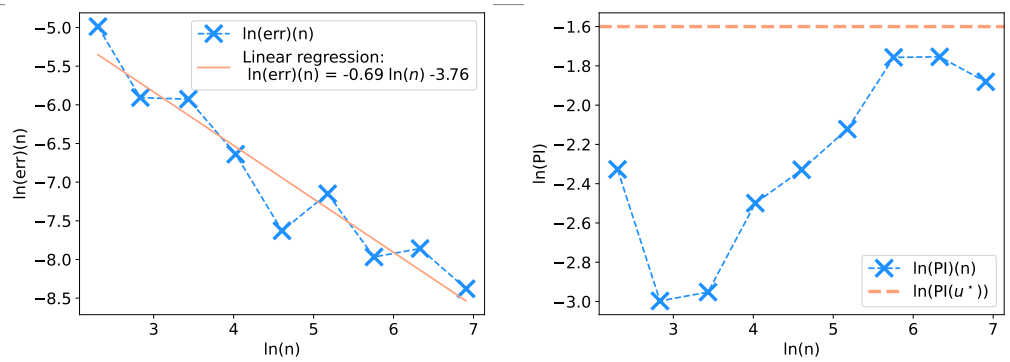

Figure 4. Distance $\mathrm{err}(n)$ to $u^\star$ (left) and physics inconsistency PI (right) of the regularized PINN estimator with respect to the number n of observations in log-log scale.

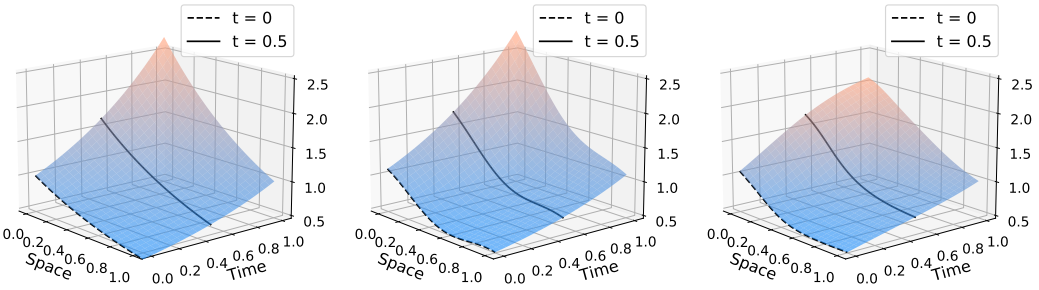

Figure 5. Functions u_{model} (left), $u^\star$ (middle), and regularized PINN estimator with $n = 10^3$ (right).

typical of the cosine perturbation of $u^\star$ (Figure 5, middle) at $t = 0$. Of course, for $t \geqslant 0.5$, the estimator cannot approximate $u^\star$ with an infinite precision, since the measurements $\mathbf{X}_i$ are only sampled for $t < 0.5$. However, the regularized PINN estimator succeeds to follow the advection equation dynamics, as it does not vary much along the lines $x - t = \text{cst}$— despite some flattening effect of the Sobolev regularization for $t \geqslant 0.5$.

Conclusion

We have shown that unregularized PINNs can overfit. To remedy this problem, we have proposed to add a ridge penalty to the empirical risk. This regularization ensures the consistency of the PINNs for both linear and nonlinear PDE systems. However, to enforce strong convergence to the target function, another layer of regularization is needed. For linear PDEs, we have proved that the addition of a Sobolev-type penalty is sufficient to ensure the strong convergence of the PINNs. Regarding future research, the next step would be to derive tighter bounds to better quantify the impact of the physical penalty on the convergence speed.

Appendix

Composition of functions Given two functions $u, v : \mathbb{R} \to \mathbb{R}$, we denote by $u \circ v$ the function $u \circ v(x) = u(v(x))$. For all $k \in \mathbb{N}$, the function $u^{\circ k}$ is defined by induction as $u^{\circ 0}(x) = x$ and $u^{\circ(k+1)} = u^{\circ k} \circ u = u \circ u^{\circ k}$. The composition symbol is placed before the derivative, so that the kth derivative of $u^{\circ H}$ is denoted by $(u^{\circ H})^{(k)}$.

Norms The p-norm $\|x\|_p$ of a vector $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ is defined by $\|x\|_p = (\frac{1}{d}\sum_{i=1}^d |x_i|^p)^{1/p}$. In addition, $\|x\|_\infty = \max_{1 \leqslant i \leqslant d} |x_i|$. For a function $u : \Omega \to \mathbb{R}^d$, we let $\|u\|_{L^p(\Omega)} = (\frac{1}{|\Omega|}\int_\Omega \|u\|_p^p)^{1/p}$. Similarly, $\|u\|_{\infty,\Omega} = \sup_{x \in \Omega} \|u(x)\|_\infty$. For the sake of clarity, we sometimes write $\|u\|_\infty$ instead of $\|u\|_{\infty,\Omega}$.

Multi-indices and partial derivatives For a multi-index $\alpha = (\alpha_1, \dots, \alpha_{d_1}) \in \mathbb{N}^d$ and a differentiable function $u : \mathbb{R}^{d_1} \to \mathbb{R}^{d_2}$, the α partial derivative of u is defined by

$$\partial^\alpha u = (\partial_1)^{\alpha_1} \dots (\partial_{d_1})^{\alpha_{d_1}} u.$$

The set of multi-indices of sum less than k is defined by

$$\{|\alpha| \leqslant k\} = \{(\alpha_1, \dots, \alpha_{d_1}) \in \mathbb{N}^d, \alpha_1 + \dots + \alpha_{d_1} \leqslant k\}.$$

If $\alpha = 0$, $\partial^\alpha u = u$. Given two multi-indices α and β, we write $\alpha \leqslant \beta$ when $\alpha_i \leqslant \beta_i$ for all $1 \leqslant i \leqslant d_1$. The set of multi-indices less than α is denoted by $\{\beta \leqslant \alpha\}$. For a multi-index α such that $|\alpha| \leqslant k$, both sets $\{|\beta| \leqslant k\}$ and $\{\beta \leqslant \alpha\}$ are contained in $\{0, \dots, k\}^{d_1}$ and are therefore finite.

Hölder norm For $K \in \mathbb{N}$, the Hölder norm of order K of a function $u \in C^K(\Omega, \mathbb{R}^d)$, is defined by $\|u\|_{C^K(\Omega)} = \max_{|\alpha| \leqslant K} \|\partial^\alpha u\|_{\infty,\Omega}$. This norm allows to bound a function as well as its derivatives. The space $C^K(\Omega, \mathbb{R}^d)$ endowed with the Hölder norm $\|\cdot\|_{C^K(\Omega)}$ is a Banach space. The space $C^\infty(\bar{\Omega}, \mathbb{R}^{d_2})$ is defined as the subspace of continuous functions $u : \bar{\Omega} \to \mathbb{R}^{d_2}$ satisfying $u|_\Omega \in C^\infty(\Omega, \mathbb{R}^{d_2})$ and, for all $K \in \mathbb{N}$, $\|u\|_{C^K(\Omega)} < \infty$.

Lipschitz function Given a normed space $(V, \|\cdot\|)$, the Lipschitz norm of a function $u : V \to \mathbb{R}^{d_1}$ is defined by

$$\|u\|_{\mathrm{Lip}} = \sup_{x,y \in V} \frac{\|u(x) - u(y)\|_2}{\|x - y\|}.$$

A function u is Lipschitz if $\|u\|_{\mathrm{Lip}} < \infty$. For all $u \in C^1(V, \mathbb{R})$, $\|u\|_{\mathrm{Lip}} \leqslant \|u\|_{C^1(V)}$.

Lipschitz surface and domain A surface $\Gamma \subseteq \mathbb{R}^{d_1}$ is said to be Lipschitz if locally, in a neighborhood $U(x)$ of any point $x \in \Gamma$, an appropriate rotation r_x of the coordinate system transforms Γ into the graph of a Lipschitz function ϕ_x, i.e.,

$$r_x(\Gamma \cap U(x)) = \{(x_1, \dots, x_{d-1}, \phi_x(x_1, \dots, x_{d-1})), \forall (x_1, \dots, x_d) \in r_x(\Gamma \cap U_x)\}.$$

A domain $\Omega \subseteq \mathbb{R}^{d_1}$ is said to be Lipschitz if its has Lipschitz boundary and lies on one side of it, i.e., $\phi_x < 0$ or $\phi_x > 0$ on all intersections $\Omega \cap U_x$. All manifolds with C^1 boundary and all convex domains are Lipschitz domains (e.g., Agranovich, 2015).

Sobolev spaces Let $\Omega \subseteq \mathbb{R}^{d_1}$ be an open set. A function $v \in L^2(\Omega, \mathbb{R}^{d_2})$ is said to be the αth weak derivative of $u \in L^2(\Omega, \mathbb{R}^{d_2})$ if, for any $\phi \in C^\infty(\bar{\Omega}, \mathbb{R}^{d_2})$ with compact support in Ω, one has $\int_\Omega \langle v, \phi \rangle = (-1)^{|\alpha|} \int_\Omega \langle u, \partial^\alpha \phi \rangle$. This is denoted by $v = \partial^\alpha u$. For $m \in \mathbb{N}$, the Sobolev space $H^m(\Omega, \mathbb{R}^{d_2})$ is the

space of all functions $u \in L^2(\Omega, \mathbb{R}^{d_2})$ such that $\partial^\alpha u$ exists for all $|\alpha| \leqslant m$. This space is naturally endowed with the norm $\|u\|_{H^m(\Omega)} = (\sum_{|\alpha| \leqslant m} |\Omega|^{-1} \|\partial^\alpha u\|^2_{L^2(\Omega)})^{1/2}$. For example, the function $u :]-1,1[\to \mathbb{R}$ such that $u(x) = |x|$ is not derivable on $]-1,1[$, but it admits $u'(x) = \mathbf{1}_{x>0} - \mathbf{1}_{x<0}$ as weak derivative. Since $u' \in L^2([-1,1], \mathbb{R})$, u belongs to the Sobolev space $H^1(]-1,1[, \mathbb{R})$. However, u' has no weak derivative, and so $u \notin H^2(]-1,1[, \mathbb{R})$. Of course, if a function u belongs to the Hölder space $C^K(\bar{\Omega}, \mathbb{R}^{d_2})$, then it belongs to the Sobolev space $H^K(\Omega, \mathbb{R}^{d_2})$, and its weak derivatives are the usual derivatives. For more on Sobolev spaces, we refer the reader to Evans (2010, Chapter 5).

Supplementary Material

Supplement to Convergence and error analysis of PINNs
The Supplementary Material contains all the proofs of the main text.

References

AGRANOVICH, M. S. (2015). *Sobolev Spaces, Their Generalizations and Elliptic Problems in Smooth and Lipschitz Domains*. Springer, Cham. https://doi.org/10.1007/978-3-319-14648-5_2

ARNONE, E., KNEIP, A., NOBILE, F. and SANGALLI, L. M. (2022). Some first results on the consistency of spatial regression with partial differential equation regularization. *Stat. Sinica* **32** 209–238. https://doi.org/10.5705/ss.202019.0346

ARZANI, A., WANG, J. X. and D'SOUZA, R. M. (2021). Uncovering near-wall blood flow from sparse data with physics-informed neural networks. *Phys. Fluids* **33** 071905. https://doi.org/10.1063/5.0055600

AZZIMONTI, L., SANGALLI, L. M., SECCHI, P., DOMANIN, M. and NOBILE, F. (2015). Blood flow velocity field estimation via spatial regression with PDE penalization. *J. Am. Stat. Assoc.* **110** 1057–1071. https://doi.org/10.1080/01621459.2014.946036

BREZIS, H. (2010). *Functional Analysis, Sobolev Spaces and Partial Differential Equations*. Springer, New York. https://doi.org/10.1007/978-0-387-70914-7

CAI, S., WANG, Z., WANG, S., PERDIKARIS, P. and KARNIADAKIS, G. E. (2021). Physics-informed neural networks for heat transfer problems. *J. Heat. Transf.* **143**. https://doi.org/10.1115/1.4050542

CHANDRAJIT, B., MCLENNAN, L., ANDEEN, T. and ROY, A. (2023). Recipes for when physics fails: Recovering robust learning of physics informed neural networks. *Mach. Learn.: Sci. Technol.* **4** 015013. https://doi.org/10.1088/2632-2153/acb416

COSTABAL, F. S., YANG, Y., PERDIKARIS, P., HURTADO, D. E. and KUHL, E. (2020). Physics-informed neural networks for cardiac activation mapping. *AIP Conf. Proc.* **8** 42. https://doi.org/10.3389/fphy.2020.00042

CUNHA, B., DROZ, C., ZINE, A., FOULARD, S. and ICHCHOU, M. (2023). A review of machine learning methods applied to structural dynamics and vibroacoustic. *Mech. Syst. Signal. Pr.* 110535. https://doi.org/10.1016/j.ymssp.2023.110535

CUOMO, S., COLA, V. S. D., GIAMPAOLO, F., ROZZA, G., RAISSI, M. and PICCIALLI, F. (2022). Scientific machine learning through physics-informed neural networks: Where we are and what's next. *J. Sci. Comput.* **92** 88. https://doi.org/10.1007/s10915-022-01939-z

DAVINI, D., SAMINENI, B., THOMAS, B., TRAN, A. H., ZHU, C., HA, K., DASIKA, G. and WHITE, L. (2021). Using physics-informed regularization to improve extrapolation capabilities of neural networks. In *Fourth Workshop on Machine Learning and the Physical Sciences (NeurIPS 2021)*.

DAW, A., KARPATNE, A., WATKINS, W. D., READ, J. S. and KUMAR, V. (2022). Physics-guided neural networks (PGNN): An application in lake temperature modeling. In *Knowledge guided machine learning: Accelerating discovery using scientific knowledge and data* (A. KARPATNE, R. KANNAN and V. KUMAR, eds.) 352–372. Chapman and Hall/CRC, New York. https://doi.org/10.1201/9781003143376-15

DE BÉZENAC, E., PAJOT, A. and GALLINARI, P. (2019). Deep learning for physical processes: Incorporating prior scientific knowledge. *J. Stat. Mech.-Theory E.* 124009. https://doi.org/10.1088/1742-5468/ab3195

DE WOLFF, T., CARRILLO, H., MARTÍ, L. and SANCHEZ-PI, N. (2021). Towards optimally weighted physics-informed neural networks in ocean modelling. *arXiv:2106.08747*. https://doi.org/10.48550/arXiv.2106.08747

DOUMÈCHE, N., BIAU, G. and BOYER, C. (2023). Supplement to "On the convergences of PINNs".

ESFAHANI, I. C. (2023). A data-driven physics-informed neural network for predicting the viscosity of nanofluids. *AIP Adv.* **13** 025206. https://doi.org/10.1063/5.0132846

EVANS, L. C. (2010). *Partial Differential Equations*, 2nd ed. *Graduate Studies in Mathematics* **19**. American Mathematical Society, Providence. https://doi.org/10.1090/gsm/019

FERRACCIOLI, F., SANGALLI, L. M. and FINOS, L. (2022). Some first inferential tools for spatial regression with differential regularization. *J. Multivariate Anal.* **189** 104866. https://doi.org/10.1016/j.jmva.2021.104866

FISCHER, S. and STEINWART, I. (2020). Sobolev norm learning rates for regularized least-squares algorithm. *J. Mach. Learn. Res.* **21** 8464–8501. https://doi.org/10.48550/arXiv.1702.07254

GOKHALE, G., CLAESSENS, B. and DEVELDER, C. (2022). Physics informed neural networks for control oriented thermal modeling of buildings. *Appl. Energ.* **314** 118852. https://doi.org/10.1016/j.apenergy.2022.118852

GRISVARD, P. (2011). *Elliptic Problems in Nonsmooth Domains*. *Classics in Applied Mathematics* **69**. SIAM, Philadelphia. https://doi.org/10.1137/1.9781611972030

GUO, C., PLEISS, G., SUN, Y. and WEINBERGER, K. Q. (2017). On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning* (D. PRECUP and Y. W. TEH, eds.). *Proceedings of Machine Learning Research* **70** 1321–1330. PMLR. https://doi.org/10.48550/arXiv.1706.04599

HAO, Z., LIU, S., ZHANG, Y., YING, C., FENG, Y., SU, H. and ZHU, J. (2022). Physics-informed machine learning: A survey on problems, methods and applications. *arXiv:2211.08064*. https://doi.org/10.48550/arXiv.2211.08064

HE, Q., BARAJAS-SOLANO, D., TARTAKOVSKY, G. and TARTAKOVSKY, A. M. (2020). Physics-informed neural networks for multiphysics data assimilation with application to subsurface transport. *Adv. Water. Resourc.* **141** 103610. https://doi.org/10.1016/j.advwatres.2020.103610

JAGTAP, A. D., KAWAGUCHI, K. and KARNIADAKIS, G. E. (2020). Adaptive activation functions accelerate convergence in deep and physics-informed neural networks. *J. Comput. Phys.* **404** 109136. https://doi.org/10.1016/j.jcp.2019.109136

KAPUSUZOGLU, B. and MAHADEVAN, S. (2020). Physics-informed and hybrid machine learning in additive manufacturing: Application to fused filament fabrication. *JOM-US* **72** 4695–4705. https://doi.org/10.1007/s11837-020-04438-4

KARNIADAKIS, G. E., KEVREKIDIS, I. G., LU, L., PERDIKARIS, P., WANG, S. and YANG, L. (2021). Physics-informed machine learning. *Nat. Rev. Phys.* **3** 422–440. https://doi.org/10.1038/s42254-021-00314-5

KRISHNAPRIYAN, A., GHOLAMI, A., ZHE, S., KIRBY, R. and MAHONEY, M. W. (2021). Characterizing possible failure modes in physics-informed neural networks. In *Advances in Neural Information Processing Systems* (M. RANZATO, A. BEYGELZIMER, Y. DAUPHIN, P. S. LIANG and J. W. VAUGHAN, eds.) **34** 26548–26560. Curran Associates, Inc. https://doi.org/10.48550/arXiv.2109.01050

KROGH, A. and HERTZ, J. (1991). A simple weight decay can improve generalization. In *Advances in Neural Information Processing Systems* (J. MOODY, S. HANSON and R. P. LIPPMANN, eds.) **4** 950–957. Morgan-Kaufmann.

LI, S., WANG, G., DI, Y., WANG, L., WANG, H. and ZHOU, Q. (2023). A physics-informed neural network framework to predict 3D temperature field without labeled data in process of laser metal deposition. *Eng. Appl. Artif. Intel.* **120** 105908. https://doi.org/10.1016/j.engappai.2023.105908

LINARDATOS, P., PAPASTEFANOPOULOS, V. and KOTSIANTIS, S. (2021). Explainable AI: A review of machine learning interpretability methods. *Entropy* **23** 18. https://doi.org/10.3390/e23010018

LOSHCHILOV, I. and HUTTER, F. (2019). Decoupled weight decay regularization. In *7th International Conference on Learning Representations*. https://doi.org/10.48550/arXiv.1711.05101

MISHRA, S. and MOLINARO, R. (2023). Estimates on the generalization error of physics-informed neural networks for approximating PDEs. *IMA J. Numer. Anal.* **43** 1–43. https://doi.org/10.1093/imanum/drab093

NABIAN, M. A. and MEIDANI, H. (2020). Physics-driven regularization of deep neural networks for enhanced engineering design and analysis. *J. Comput. Inf. Sci. Eng.* **20** 011006. https://doi.org/10.1115/1.4044507

PANNELL, J. J., RIGBY, S. E. and PANOUTSOS, G. (2022). Physics-informed regularisation procedure in neural networks: An application in blast protection engineering. *Int. J. Prot. Struct.* **13** 555–578. https://doi.org/10.1177/20414196211073501

QIAN, Y., ZHANG, Y., HUANG, Y. and DONG, S. (2023). Physics-informed neural networks for approximating dynamic (hyperbolic) PDEs of second order in time: Error analysis and algorithms. *J. Comput. Phys.* **495** 112527. https://doi.org/10.1016/j.jcp.2023.112527

RAI, R. and SAHU, C. K. (2020). Driven by data or derived through physics? A review of hybrid physics guided machine learning techniques with cyber-physical system (CPS) focus. *IEEE Access* **8** 71050–71073. https://doi.org/10.1109/ACCESS.2020.2987324

RAISSI, M., PERDIKARIS, P. and KARNIADAKIS, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. Phys.* **378** 686–707. https://doi.org/10.1016/j.jcp.2018.10.045

RAMEZANKHANI, M., NAZEMI, A., NARAYAN, A., VOGGENREITER, H., HARANDI, M., SEETHALER, R. and MILANI, A. S. (2022). A data-driven multi-fidelity physics-informed learning framework for smart manufacturing: A composites processing case study. In *2022 IEEE 5th International Conference on Industrial Cyber-Physical Systems (ICPS)* 01–07. IEEE. https://doi.org/10.1109/ICPS51978.2022.9816983

RIEL, B., MINCHEW, B. and BISCHOFF, T. (2021). Data-driven inference of the mechanics of slip along glacier beds using physics-informed neural networks: Case Study on Rutford Ice Stream, Antarctica. *J. Adv. Model. Earth Syst.* **13** e2021MS002621. https://doi.org/10.1029/2021MS002621

DE RYCK, T., LANTHALER, S. and MISHRA, S. (2021). On the approximation of functions by tanh neural networks. *Neural Netw.* **143** 732–750. https://doi.org/10.1016/j.neunet.2021.08.015

DE RYCK, T. and MISHRA, S. (2022). Error analysis for physics informed neural networks (PINNs) approximating Kolmogorov PDEs. *Adv. Comput. Math.* **48** 79. https://doi.org/10.1007/s10444-022-09985-9

SANGALLI, L. M. (2021). Spatial regression with partial differential equation regularisation. *Int. Stat. Rev.* **89** 505–531. https://doi.org/10.1111/insr.12444

SHIN, Y. (2020). On the convergence of physics informed neural networks for linear second-order elliptic and parabolic type PDEs. *Commun. Comput. Phys.* **28** 2042–2074. https://doi.org/10.4208/cicp.OA-2020-0193

SHIN, Y., ZHANG, Z. and KARNIADAKIS, G. E. (2023). Error estimates of residual minimization using neural networks for linear PDEs. *JMLMC* **4** 73–101. https://doi.org/10.1615/JMachLearnModelComput.2023050411

SONG, C., ALKHALIFAH, T. and WAHEED, U. B. (2021). Solving the frequency-domain acoustic VTI wave equation using physics-informed neural networks. *Geophys. J. Int.* **225** 846–859. https://doi.org/10.1093/gji/ggab010

STONE, C. J. (1982). Optimal global rates of convergence for nonparametric regression. *Ann. Stat.* **10** 1040–1053. https://doi.org/10.1214/aos/1176345969

VON RUEDEN, L., MAYER, S., BECKH, K., GEORGIEV, B., GIESSELBACH, S., HEESE, R., KIRSCH, B., WALCZAK, M., PFROMMER, J., PICK, A., RAMAMURTHY, R., GARCKE, J., BAUCKHAGE, C. and SCHUECKER, J. (2023). Informed machine learning – A taxonomy and survey of integrating prior knowledge into learning systems. *IEEE T. Knowl. Data. En.* **35** 614–633. https://doi.org/10.1109/TKDE.2021.3079836

WANG, S., YU, X. and PERDIKARIS, P. (2022). When and why PINNs fail to train: A neural tangent kernel perspective. *J. Comput. Phys.* **449** 110768. https://doi.org/10.1016/j.jcp.2021.110768

WANG, C., BENTIVEGNA, E., ZHOU, W., KLEIN, L. and ELMEGREEN, B. (2020a). Physics-informed neural network super resolution for advection-diffusion models. In *Third Workshop on Machine Learning and the Physical Sciences (NeurIPS 2020)*.

WANG, R., KASHINATH, K., MUSTAFA, M., ALBERT, A. and YU, R. (2020b). Towards physics-informed deep learning for turbulent flow prediction. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* 1457–1466. Association for Computing Machinery. https://doi.org/10.48550/arXiv.1911.08655

WILLARD, J., JIA, X., XU, S., STEINBACH, M. and KUMAR, V. (2023). Integrating scientific knowledge with machine learning for engineering and environmental systems. *ACM Comput. Surv.* **55** 66. https://doi.org/10.1145/3514228

WU, S., ZHU, A., TANG, Y. and LU, B. (2022). Convergence of physics-informed neural networks applied to linear second-order elliptic interface problems. *arXiv:2203.03407*. https://doi.org/10.48550/arXiv.2203.03407

XU, K., ZHANG, M., LI, J., DU, S. S., KAWARABAYASHI, K. I. and JEGELKA, S. (2021). How neural networks extrapolate: From feedforward to graph neural networks. In *International Conference on Learning Representations*. https://doi.org/10.48550/arXiv.2009.11848

ZHANG, R., LIU, Y. and SUN, H. (2020). Physics-guided convolutional neural network (PhyCNN) for data-driven seismic response modeling. *Eng. Struct.* **215** 110704. https://doi.org/10.1016/j.engstruct.2020.110704