

Natural gradients and kernel methods for Physics Informed Neural Networks (PINNs)

PhD defense of Nilo Schwencke

Reviewers: • Emmanuel Frank • Olga Mula

Examiners: • Victor Michel-Dansac • Claire Boyer • Francis Bach

Invited members: • Alena Shilova • Roland Maier

Supervisor: • Cyril Furtlehner

• TAU Team–INRIA Saclay • A&O–LISN–Paris-Saclay University • CNRS

December 8, 2025

- 1 General Introduction
- 2 First contribution : ANaGRAM
- 3 Second contribution : AMStraMGRAM
- 4 Ongoing work : Implicit Curriculum Learning
- 5 Conclusion and Perspectives

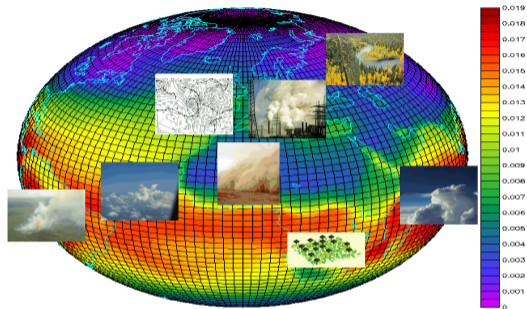


Figure: Climate modeling.

Credits: Thomas Hartmann / Hartmann photo design, Peter Pulkowski and Stefan F. Sämmer. Johannes Gutenberg University Mainz ©.

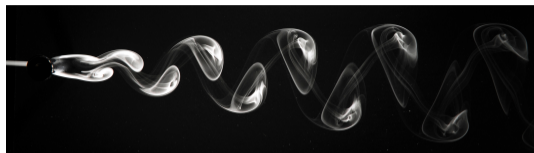


Figure: von Kármán vortex street.
(Jürgen Wagner – CC-BY-SA)

$\partial\Omega$ Ω $\begin{cases} D[u] = f & \text{in } \Omega \\ B[u] = g & \text{on } \partial\Omega \end{cases}$

Figure: A PDE specifies a physical law in the domain Ω and constraints on the boundary $\partial\Omega$.

Classical solvers

- Finite Difference Methods (FDMs; (Strikwerda2004))
- Finite Element Methods (FEMs; (Zienkiewicz and Taylor2013))
- Finite Volume Methods (FVMs; (Moukalled et al.2016))

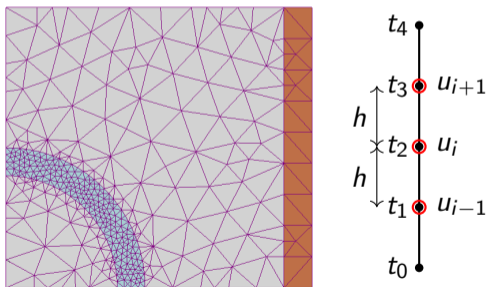


Figure: Example of a 2D mesh (Zureks – CC BY-SA) and a time discretization.

Classical PDE solvers and their limitations

Limitations

- **Curse of dimensionality:** cost grows exponentially with dimension.
- **Complex geometries:** meshing becomes difficult or unstable.
- **Nonlinearities and stiffness:** require extremely fine discretizations.
- **Equation-specific methods:** limited transfer across problems or regimes.
- **Limited data assimilation:** hard to combine physics with partial/noisy data.

- **Data-driven modeling:** learn governing equations directly from observed trajectories (e.g., SINDy (Brunton et al.2016), Neural ODEs (Chen et al.2018)).
- **Operator learning:** approximate full solution operators for families of PDEs (e.g., DeepONet (Lu et al.2021), FNO (Li et al.2020)).
- **Physics-informed ML :** embed physical laws as constraints in the learning process (e.g., PINNs and extensions).

PIML aims to minimize:

$$\mathcal{L}(u) := \int_{\Omega} \|D[u] - f\|_{\mathbb{R}^n}^2 + \int_{\partial\Omega} \|B[u] - g\|_{\mathbb{R}^m}^2.$$

PINNs key idea

- model u with a neural network (Lagaris et al.1998)

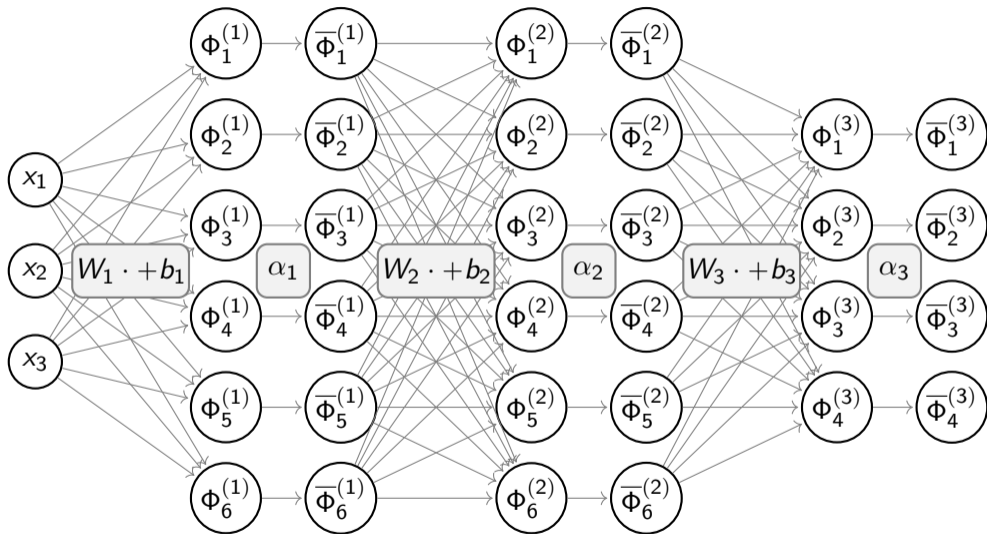


Figure: Illustration of a MLP of depth 3 with widths $(3, 6, 6, 4)$ and activations $(\alpha_i)_{i=1}^3$
 (adapted from Berner et al.2021)

- **Data-driven modeling:** learn governing equations directly from observed trajectories (e.g., SINDy (Brunton et al.2016), Neural ODEs (Chen et al.2018)).
- **Operator learning:** approximate full solution operators for families of PDEs (e.g., DeepONet (Lu et al.2021), FNO (Li et al.2020)).
- **Physics-informed ML :** embed physical laws as constraints in the learning process (e.g., PINNs and extensions).

PIML aims to minimize:

$$\mathcal{L}(u) := \int_{\Omega} \|D[u] - f\|_{\mathbb{R}^n}^2 + \int_{\partial\Omega} \|B[u] - g\|_{\mathbb{R}^m}^2.$$

PINNs key idea

- model u with a neural network (Lagaris et al.1998)
- use autodiff to compute D and B (Raissi et al.2019)

Sampling yields the loss:

$$\hat{\ell}_{D,B}(\theta) := \frac{1}{2S_D} \sum_{i=1}^{S_D} \left(D[u_{\theta}](x_i^D) - f(x_i^D) \right)^2 + \frac{1}{2S_B} \sum_{i=1}^{S_B} \left(B[u_{\theta}](x_i^B) - g(x_i^B) \right)^2.$$

Problem: This leads to low accuracy when using usual optimizers.

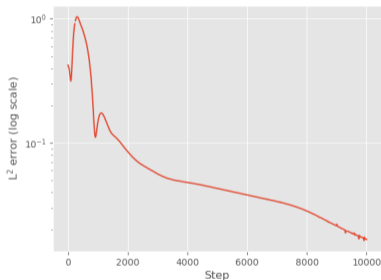


Figure: L^2 -error of a PINN optimized with Adam on the 2D Laplace equation.

PINNs main drawback : optimization error

- Highly nonconvex & ill-conditioned optimization
- Gradient imbalance between loss terms
- Spectral bias of neural nets : Low frequencies learned first; high-frequency or localized features missed.

Intuition from Fourier

$$S_N : (\alpha_k) \in \mathbb{C}^{\llbracket -N, N \rrbracket} \mapsto \sum_{k=-N}^N \alpha_k e^{2i\pi kx}.$$

S_N singular values are all 1. **BUT:**

$$\Delta[S_N] \text{ spectrum is } \{4\pi^2 k^2 : 1 \leq k \leq N\}$$

Δ strongly impact the spectral conditioning.

My approach : functional geometric perspective on machine-learning

Parametric model

$$u : \begin{cases} \mathbb{R}^P & \longrightarrow \mathcal{H} \\ \theta & \longmapsto u_\theta \end{cases}$$

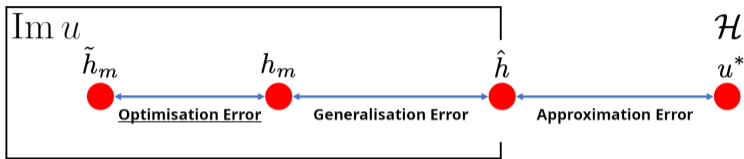


Figure: Schematic decomposition of the total approximation error in ML (adapted from Shin et al.2020))

Examples

- Neural Networks
- Partial Fourier Series
- Kernel Methods
- Finite Elements

Main tools

- Natural Gradient
- Neural Tangent Kernel (NTK)
- Reproducing Kernel Hilbert spaces (RKHS)

Natural Gradient

Classical quadratic regression problem, with batch (x_i) :

$$\hat{\ell}(\theta) := \frac{1}{2S} \sum_{i=1}^S (u_\theta(x_i) - f(x_i))^2.$$

In the population limit:

$$\hat{\ell}(\theta) \xrightarrow{S \rightarrow \infty} \mathcal{L}(u_\theta); \quad \mathcal{L}(u) := \frac{1}{2} \|u - f\|_{L^2(\Omega)}^2$$

This yields the Fréchet derivative:

$$d\mathcal{L}|_u(h) = \underbrace{\langle u - f, h \rangle}_{\nabla \mathcal{L}|_u} \Big|_{L^2(\Omega)},$$

and thus the gradient flow:

$$\begin{cases} u_0 \in L^2(\Omega) \\ \dot{u}_t = -\nabla \mathcal{L}|_{u_t} = f - u_t \end{cases}.$$

Solution: $u_t = f - e^{-t}(f - u_0)$.

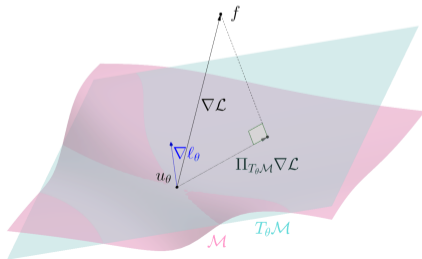
Natural gradient in functional space

The functional space is constrained to:

- $\mathcal{M} := \text{Im } u = \{u_\theta : \theta \in \mathbb{R}^P\}$
- $T_\theta \mathcal{M} := \text{Im } du_\theta = \text{Span}(\partial_p u_\theta)$

The Natural Gradient is then (Amari and Douglas1998):

$$\theta_{t+1} \leftarrow \theta_t - \eta du_{\theta_t}^\dagger \left(\Pi_{T_{\theta_t} \mathcal{M}}^\perp \nabla \mathcal{L}|_{u_{\theta_t}} \right),$$



Intuition

$$\hat{\ell}(u) := \frac{1}{2S} \sum_{i=1}^S (u(x_i) - f(x_i))^2.$$

$$\frac{du_{\theta(t)}}{dt}(x) = -\frac{1}{S} \sum_{i=1}^S (u_{\theta}(x_i) - f(x_i)) \delta_{x_i}$$

Gradient descent (Jacot et al.2018)

$$\frac{du_{\theta(t)}}{dt}(x) = -\frac{1}{S} \sum_{i=1}^S \text{NTK}_{\theta}(x, x_i) (u_{\theta}(x_i) - f(x_i))$$

$$\text{with: } \text{NTK}_{\theta}(x, y) := \sum_{p=1}^P \partial_p u_{\theta}(x) \partial_p u_{\theta}(y)^{\top}.$$

Natural Gradient (Rudner et al.2019)

$$\frac{du_{\theta(t)}}{dt}(x) = -\frac{1}{S} \sum_{i=1}^S \text{NNTK}_{\theta}(x, x_i) (u_{\theta}(x_i) - f(x_i))$$

$$\text{with: } \text{NNTK}_{\theta}(x, y) := \sum_{1 \leq p, q \leq P} \partial_p u_{\theta}(x) G_{\theta, p, q}^{\dagger} \partial_q u_{\theta}(y)^{\top},$$

$$G_{\theta, p, q} = \langle \partial_p u_{\theta}, \partial_q u_{\theta} \rangle.$$

Neural Tangent Kernel (NTK)

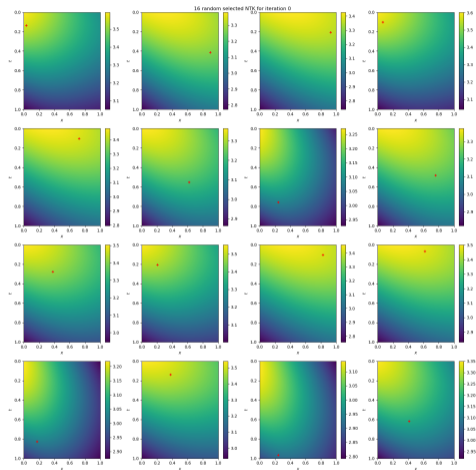


Figure: NTK for the Heat equation.

Reading: the plot represents the function

$\text{NTK}_{\theta_0}(\cdot, x_i)$; the red cross marks the point x_i .

First contribution : ANaGRAM

- Schwencke, N. and C. Furtlehner (2025): “ANaGRAM: A Natural Gradient Relative to Adapted Model for Efficient PINNs Learning,” in The Thirteenth International Conference on Learning Representations (ICLR).

Computational perspective on Natural Gradient

Definition-Proposition

The **Natural Neural Tangent Kernel (NNTK)** is the kernel of the projection $\Pi_{T_\theta \mathcal{M}} : L^2(\Omega) \rightarrow L^2(\Omega)$ onto $T_\theta \mathcal{M}$. It is given by the formula:

$$NNTK_\theta(x, y) := \sum_{1 \leq p, q \leq P} (\partial_p u_\theta(x)) G_{\theta, pq}^\dagger (\partial_q u_\theta(y))^t; \quad G_{\theta, p, q} := \langle \partial_p u_\theta, \partial_q u_\theta \rangle_{L^2(\Omega)}.$$

Corollary

The Natural Gradient update rewrites: $\theta_{t+1} \leftarrow \theta_t - \eta G_{\theta_t}^\dagger \nabla \ell(\theta_t)$; $\ell(\theta) := \mathcal{L}(u_\theta)$.

Shortcomings

- Computation of the Gram matrix G_{θ_t} is quadratic in the number of parameters.
- Inversion of G_{θ_t} is cubic

We introduce a the empirical Natural Gradient that scales linearly with the number of parameters.

Recall the the functional dynamics of (N)GD on the empirical loss $\hat{\ell}$:

$$\frac{d\mathbf{u}_{\theta_t}}{dt}(\mathbf{x}) = - \sum_{i=1}^S (N)NTK_{\theta_t}(\mathbf{x}, \mathbf{x}_i)(\mathbf{u}_{\theta_t}(\mathbf{x}_i) - y_i),$$

Key Observation

The empirical dynamics takes place in:

$$\hat{T}_{\theta_t}\mathcal{M} := \text{Span}((N)NTK_{\theta_t}(\mathbf{x}_i, \cdot) : (\mathbf{x}_i)_{1 \leq i \leq N}).$$

We can define the empirical Natural Gradient:

$$\theta_{t+1} = \theta_t - \eta d\mathbf{u}_{\theta_t}^\dagger \left(\Pi_{\hat{T}_{\theta_t}\mathcal{M}}^\perp \nabla \mathcal{L}|_{\mathbf{u}_{\theta_t}} \right).$$

Theorem (ANaGRAM)

Under mild assumptions:

$$d\mathbf{u}_{\theta_t}^\dagger \left(\Pi_{\hat{T}_{\theta_t}\mathcal{M}}^\perp \nabla \mathcal{L}|_{\mathbf{u}_{\theta_t}} \right) \simeq \hat{\phi}_{\theta_t}^\dagger \widehat{\nabla \mathcal{L}}_{\theta_t},$$

with: for all $1 \leq p \leq P, 1 \leq i \leq S$

- $\hat{\phi}_{\theta_t i, p} := \partial_p \mathbf{u}_{\theta_t}(\mathbf{x}_i)$
- $\widehat{\nabla \mathcal{L}}_{\theta_t i} := \nabla \mathcal{L}|_{\mathbf{u}_{\theta_t}}(\mathbf{x}_i)$

Key fact

$\hat{\phi}_{\theta_t}^\dagger$ can be computed with a SVD, with complexity $O(\min(P^2S, P^2S))$.

Corollary

There exists P points $(\hat{\mathbf{x}}_i)$ such that:

$$\Pi_{\hat{T}_{\theta_t}\mathcal{M}}^\perp \nabla \mathcal{L}|_{\mathbf{u}_{\theta_t}} = \Pi_{T_{\theta_t}\mathcal{M}}^\perp \nabla \mathcal{L}|_{\mathbf{u}_{\theta_t}}.$$

Recall the the functional dynamics of (N)GD on the empirical loss $\hat{\ell}$:

$$\frac{du_{\theta_t}}{dt}(x) = - \sum_{i=1}^S (N)NTK_{\theta_t}(x, x_i)(u_{\theta_t}(x_i) - y_i),$$

Key Observation

The empirical dynamics takes place in:

$$\hat{T}_{\theta} \mathcal{M} := \text{Span} \left((N)NTK_{\theta}(x_i, \cdot) : (x_i)_{1 \leq i \leq N} \right).$$

We can define the empirical Natural Gradient:

$$\theta_{t+1} = \theta_t - \eta du_{\theta_t}^{\dagger} \left(\Pi_{\hat{T}_{\theta_t} \mathcal{M}}^{\perp} \nabla \mathcal{L}|_{u_{\theta_t}} \right).$$

Byproduct

Yields an optimal criterion for (x_i) choice:

$$(x_i)^{\star} = \underset{(x_i) \in \Omega^S}{\operatorname{argmin}} \left\| \Pi_{\hat{T}_{\theta, K}^{(x_i)} \mathcal{M}}^{\perp} \nabla \mathcal{L}|_{u_{\theta_t}} - \nabla \mathcal{L}|_{u_{\theta_t}} \right\|_{L^2(\Omega)}$$

Theorem (ANaGRAM)

Under mild assumptions:

$$du_{\theta_t}^{\dagger} \left(\Pi_{\hat{T}_{\theta_t} \mathcal{M}}^{\perp} \nabla \mathcal{L}|_{u_{\theta_t}} \right) \simeq \hat{\phi}_{\theta_t}^{\dagger} \widehat{\nabla \mathcal{L}}_{\theta_t},$$

with: for all $1 \leq p \leq P, 1 \leq i \leq S$

- $\hat{\phi}_{\theta_t i, p} := \partial_p u_{\theta_t}(x_i)$
- $\widehat{\nabla \mathcal{L}}_{\theta_t i} := \nabla \mathcal{L}|_{u_{\theta_t}}(x_i)$

Key fact

$\hat{\phi}_{\theta_t}^{\dagger}$ can be computed with a SVD, with complexity $O(\min(P^2 S, S^2 P))$.

Corollary

There exists P points (\hat{x}_i) such that:

$$\Pi_{\hat{T}_{\theta} \mathcal{M}}^{\perp} \nabla \mathcal{L}|_{u_{\theta}} = \Pi_{T_{\theta} \mathcal{M}}^{\perp} \nabla \mathcal{L}|_{u_{\theta}}.$$

Key remark

The only difference between the losses:

$$\hat{\ell}_{D,B}(\theta) := \frac{1}{2S_D} \sum_{i=1}^{S_D} \left(D[u_\theta](x_i^D) - f(x_i^D) \right)^2 + \frac{1}{2S_B} \sum_{i=1}^{S_B} \left(B[u_\theta](x_i^B) - g(x_i^B) \right)^2,$$

and $\hat{\ell}(u) := \frac{1}{2S} \sum_{i=1}^S (u(x_i) - f(x_i))^2$ is the use of the operators D and B .

Proposition

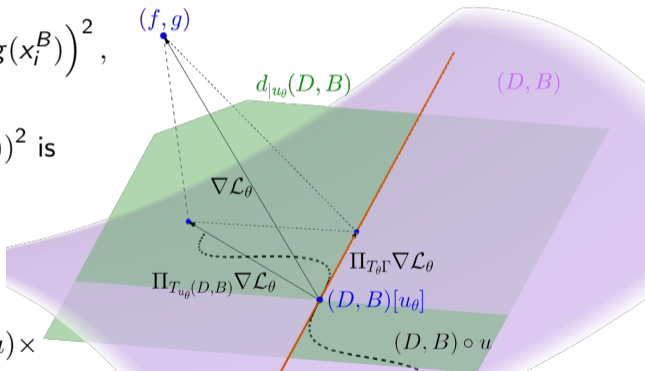
PINNs are a quadratic regression problem with model: $(D, B) \circ u$:

$$\begin{cases} \mathbb{R}^P & \rightarrow \mathcal{H} & \rightarrow L^2(\Omega \rightarrow \mathbb{R}, \mu) \times L^2(\partial\Omega \rightarrow \mathbb{R}, \sigma) \\ \theta & \mapsto u_\theta & \mapsto (D[u_\theta], B[u_\theta]) \end{cases}$$

Application to PINNs

Natural Gradient of PINNs

Figure: Illustration of PINNs Natural Gradient



Natural gradient of PINNs is a Green's function

Theorem

Let $D : \mathcal{H} \rightarrow L^2(\Omega \rightarrow \mathbb{R}, \mu)$ be a linear differential operator and $u : \mathbb{R}^P \rightarrow \mathcal{H}$ a parametric model. Then for all $\theta \in \mathbb{R}^P$, the generalized Green's function of D on $T_\theta \mathcal{M} = \text{Im } du_\theta$ is given by: for all $x, y \in \Omega$

$$g_{T_\theta \mathcal{M}}(x, y) := \sum_{1 \leq p, q \leq P} \partial_p u_\theta(x) G_{p,q}^\dagger \partial_q D[u_\theta](y),$$

with: for all $1 \leq p, q \leq P$

$$G_{pq} := \langle \partial_p D[u_\theta], \partial_q D[u_\theta] \rangle_{L^2(\Omega \rightarrow \mathbb{R}, \mu)}.$$

In particular, the natural gradient of PINNs can be rewritten:

$$\theta_{t+1} \leftarrow \theta_t - \eta du_{\theta_t}^\dagger \left(x \in \Omega \mapsto \int_{\Omega} g_{T_{\theta_t} \mathcal{M}}(x, y) \nabla \mathcal{L}_{\theta_t}(y) \mu(dy) \right).$$

1+1 D Heat equation

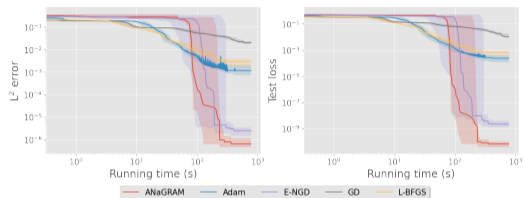


Figure: Performance comparison w.r.t running time for Heat equation in 1+1 D:

$$\left\{ \begin{array}{l} \partial_t u - \frac{1}{4} \partial_{xx} u = 0 \\ u = 0 \\ u = \sin(\pi x) \end{array} \right. \quad \text{in } [0, 1]^2 \quad \text{on } [0, 1] \times \{0, 1\} \quad \text{on } \{0\} \times [0, 1]$$

$$\left\{ \begin{array}{l} \partial_t u - 10^{-3} \partial_{xx} u = 5(u - u^3) \\ u = -1 \\ u(0, x) = x^2 \cos(\pi x) \end{array} \right. \quad \text{in } \Omega = [0, 1] \times [-1, 1] \quad \text{on } \partial\Omega_b = [0, 1] \times \{-1, 1\} \quad \text{on } \partial\Omega_0 = \{0\} \times [-1, 1]$$

1+1 D Allen-Cahn equation

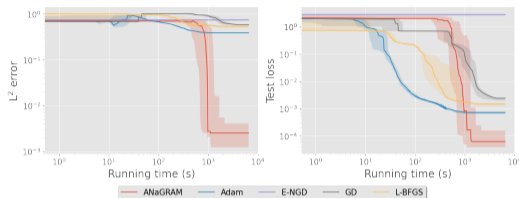
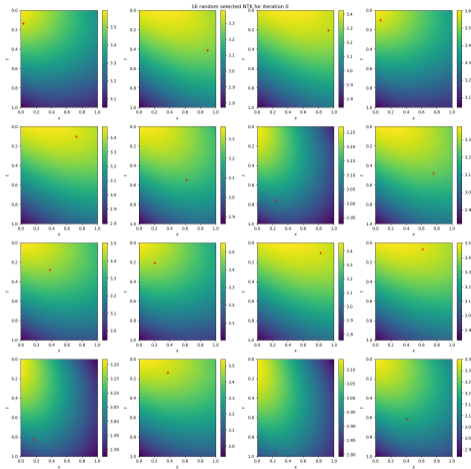


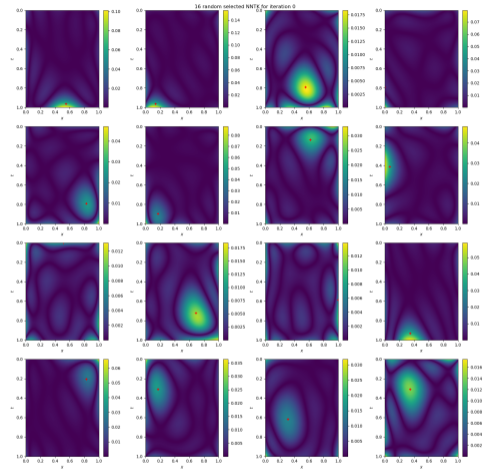
Figure: Performance comparison w.r.t running time for Allen-Cahn equation in 1+1 D:

Note: E-NGD refers to Müller and Zeinhofer2023).

NTK vs NNTK of PINNs



(a) NTK

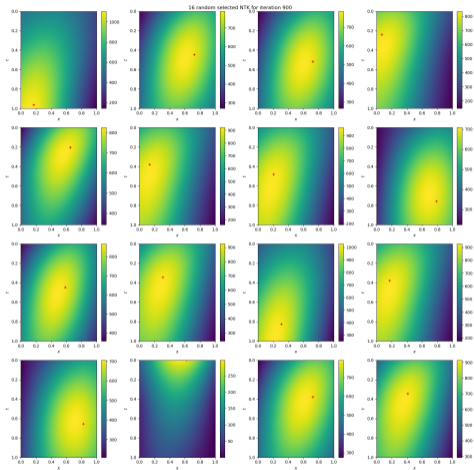


(b) NNTK

Figure: Comparison of NTK and NNTK at **initialization** for Heat equation.

Reading: the plot represents the function $(N)NTK_{\theta_0}(\cdot, x_i)$; the red cross marks the point x_i .

NTK vs NNTK of PINNs



(a) NTK



(b) NNTK

Figure: Comparison of NTK and NNTK at the **end of optimization** for Heat equation.

Reading: the plot represents the function $(N)NTK_{\theta_{\text{final}}}(\cdot, x_i)$; the red cross marks the point x_i .

Second contribution : AMStraMGRAM

- Schwencke, N., C. Rousselot, A. Shilova, and C. Furtlehner (2025): “AMStramGRAM: Adaptive Multi-Cutoff Strategy Modification for ANaGRAM,” arXiv Preprint.

In-Depth Empirical Analysis of *Cutoff* Regularization in ANaGRAM

SVD pseudoinverse details

$$\hat{\phi}_\theta = \hat{V} \hat{\Delta} \hat{U}^\top; \quad \hat{\phi}_{\theta_t} = \hat{U} \hat{\Delta}^\dagger \hat{V}^\top.$$

Singular values

In practice, we apply a *cutoff*:

$$\hat{\Delta}^{\dagger\alpha} := \begin{cases} \hat{\Delta}_i^{-1} & \text{if } \hat{\Delta}_i \geq \alpha \\ 0 & \text{otherwise} \end{cases},$$

with $\alpha > 0$ the cutoff level. Thus:

$$\hat{\phi}_\theta^{\dagger\alpha} \widehat{\nabla \mathcal{L}_\theta} = \sum_{i=1}^{r_\alpha} \hat{U}_i \hat{\Delta}_i^{-1} \hat{V}_i^\top \widehat{\nabla \mathcal{L}_\theta},$$

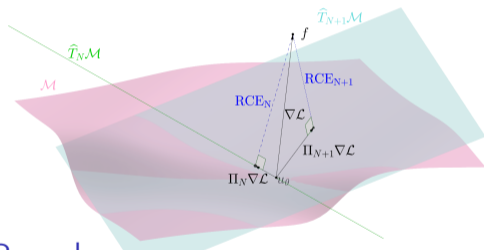
with $r_\alpha := \#\{i : \hat{\Delta}_i \geq \alpha\} \leq \min(P, S)$.

Reconstruction Error (RCE)

$$\text{RCE}_n = \frac{1}{\sqrt{S}} \left\| \widehat{\nabla \mathcal{L}_\theta} - \sum_{i=1}^n \hat{V}_i \hat{V}_i^\top \widehat{\nabla \mathcal{L}_\theta} \right\|_{\mathbb{R}^S}$$

Intuition on RCE

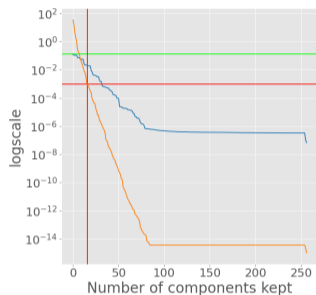
RCE_N quantifies the part of $\nabla \mathcal{L}_\theta$ orthogonal to the top N components of $\hat{T}_\theta \mathcal{M}$.



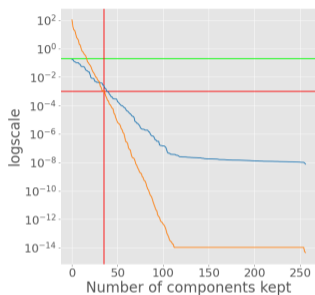
Remark

- $\lim_{S \rightarrow \infty} \hat{T}_\theta \mathcal{M} = T_\theta \mathcal{M}$
- $\text{RCE}_0^2 = \frac{1}{S} \left\| \widehat{\nabla \mathcal{L}_\theta} \right\|_{\mathbb{R}^S}^2 = \hat{\ell}(\theta)$

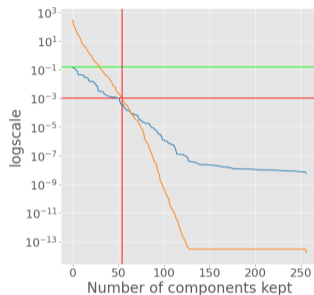
Empirical insights on the *cutoff* impact in ANaGRAM



(a) Iteration 0: intersection point between singular values and RCE lies before cutoff.



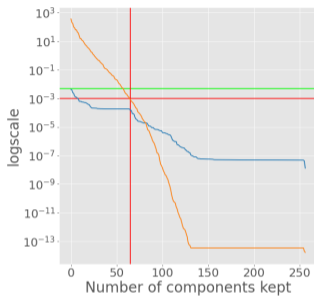
(b) Iteration 40: intersection point shifts rightward toward cutoff.



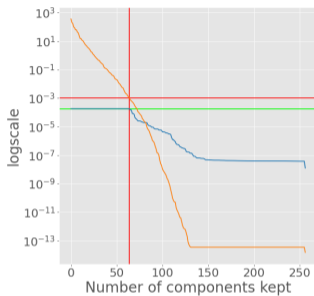
(c) Iteration 90: intersection point passes the cutoff threshold.



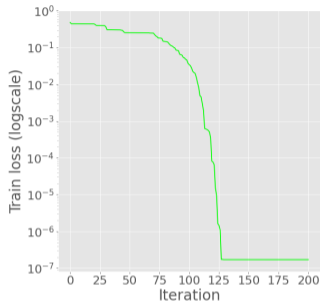
Empirical insights on the *cutoff* impact in ANaGRAM



(d) Iteration 120. Beginning of *flattening*: RCE stabilizes at constant level before cutoff.



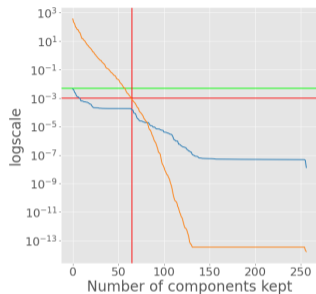
(e) Iteration 150: End of the *flattening phenomenon*. The train loss reaches the flattened part of the RCE.



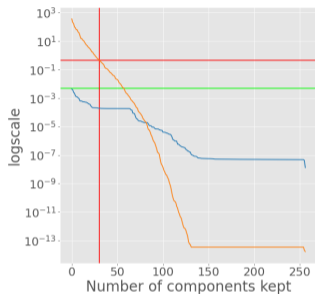
(f) Train loss dynamics.



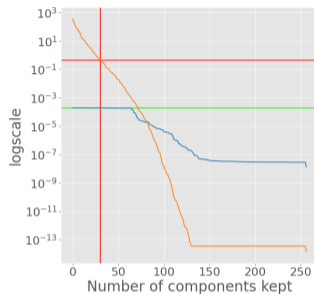
Incomplete and instant *flattening*



(g) Incomplete flattening of the RCE with a fixed cutoff at 10^{-3} .



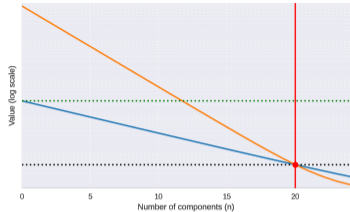
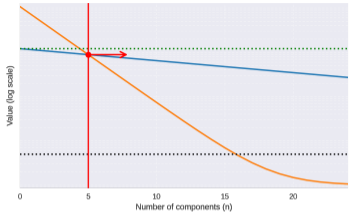
(h) New cutoff located roughly at the location of the "elbow" in the RCE curve.



(i) Complete flattening after one natural gradient step with the new cutoff.



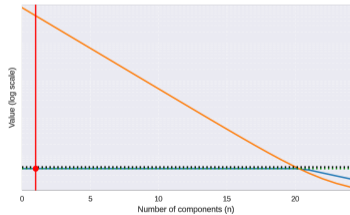
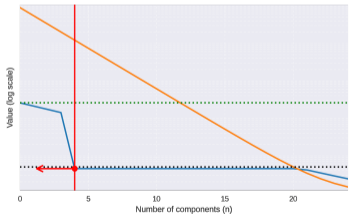
Outline of the algorithm



(j) Early iterations: cutoff set at the RCE–singular values intersection.

(k) The RCE–singular values intersection drops below precision.

- RCE
- Singular Values
- Precision
- $\sqrt{\text{Train Loss}}$
- Cutoff rank



(l) Flattening phase: Cutoff is set at the RCE–precision intersection.

(m) End of flattening: The RCE and loss stands at precision level.

Experiment	Train Loss		L_2 Error	
	AMStraMGRAM	ANaGRAM	AMStraMGRAM	ANaGRAM
Heat Equation	6.29e-29 \pm 6.78e-30	8.56e-11 \pm 7.05e-11	2.32e-14 \pm 1.14e-14	1.28e-06 \pm 1.75e-06
Laplace 2D	1.46e-28 \pm 1.87e-29	4.27e-13 \pm 4.66e-13	2.24e-15 \pm 2.52e-16	3.49e-09 \pm 3.58e-09
Laplace 5D	2.04e-08 \pm 1.16e-08	6.37e-08 \pm 7.01e-08	2.12e-05 \pm 8.15e-06	4.00e-05 \pm 2.93e-05
Allen–Cahn	3.19e-11 \pm 2.37e-11	2.19e-04 \pm 4.16e-04	5.87e-05 \pm 6.25e-06	4.32e-03 \pm 5.93e-03

Experiment	Train Loss		L_2 Error	
	AMStraMGRAM	SSBroyden *	AMStraMGRAM	SSBroyden *
Burgers (1+1 D)	2.99e-12 \pm 9.26e-13	2.92e-10 \pm 1.45e-10	1.5e-06 \pm 9.43e-7	1.59e-06 \pm 1.02e-6
Non-Linear Poisson	8.51e-24 \pm 2.24e-24	3.03e-16 \pm 3.82e-16	6.81e-10 \pm 1.41e-09	9.29e-12 \pm 5.85e-12
Allen–Cahn	3.19e-11 \pm 2.37e-11	6.42e-12 \pm 5.52e-12	5.87e-05 \pm 6.25e-06	3.94e-06 \pm 1.72e-06

* refers to the order two method of Urbán et al.2025), with adaptive sampling and hard constraint enforcement on boundary conditions.

Overfitting

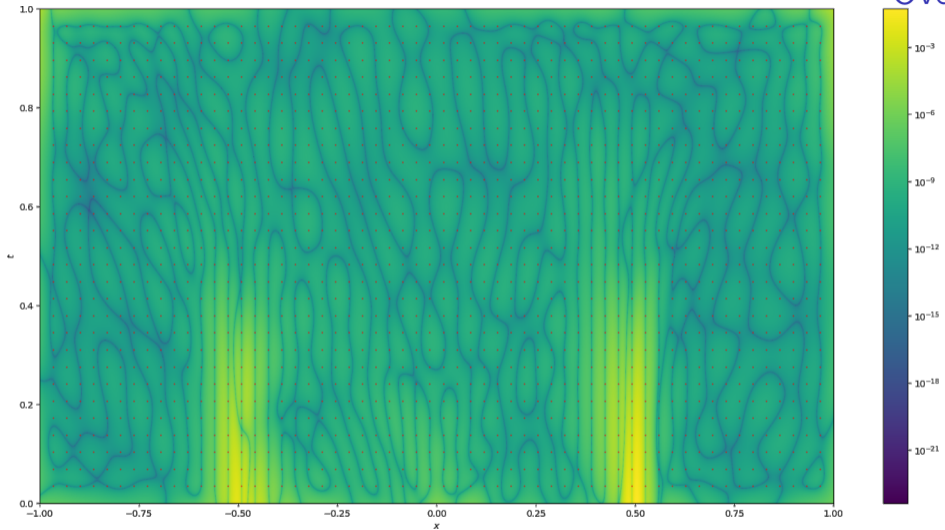


Figure: Overfitting on Allen–Cahn: residual lines align with sampling lines.

Overfitting

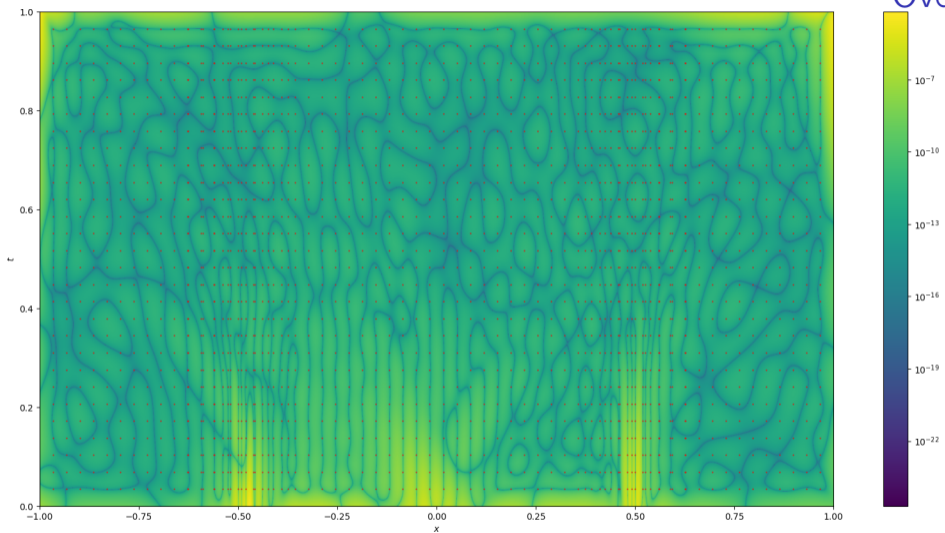


Figure: Overfitting on Allen–Cahn: densifying the sampling mitigates overfitting.

Ongoing work : Implicit Curriculum Learning

- Marie-Anne, J., C. Rousselot, N. Schwencke, and A. Shilova (2025): “Implicit Function Theorem in Physics-Informed Neural Networks to Solve Parameterized Differential Equations,” in EurIPS 2025 Workshop: Differentiable Systems and Scientific Machine Learning.

Implicit Curriculum Learning

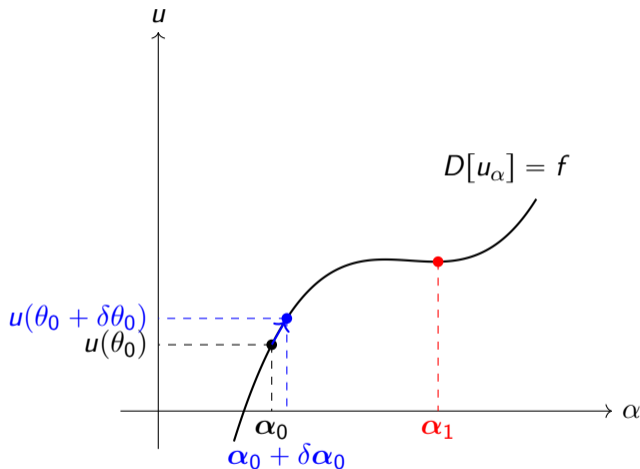


Figure: Illustration of implicit function theorem.

Credits : Julien Marie-Anne

Parameterized PDE

Let $\alpha \in \mathcal{A} \subset \mathbb{R}$ and consider

$$D[\alpha, u] = f \quad \text{in } \Omega.$$

Implicit function theorem

If $dD_{\alpha, u} \neq 0$, there is $[a, b] \ni \alpha$ and v such that for all $\beta \in [a, b]$

$$D[\beta, v(\beta)] = f.$$

with v solution to the ODE:

$$\frac{d}{d\beta} v = \partial_u D_{\beta, v(\beta)}^{-1} \left[\frac{d}{d\beta} D_{\beta, v(\beta)} \right]$$

Implicit curriculum learning

Follow the update:

$$\frac{d}{d\beta} \theta(\beta) = \partial_{\theta} (D \circ u_{\theta})_{\beta, u_{\theta(\beta)}}^{\dagger} \left[\frac{d}{d\beta} D_{\beta, u_{\theta(\beta)}} \right]$$

Table: Comparison of the method with Adam on different PDEs

Equation	Method	Metric
Hamilton–Jacobi–Bellman (Relative Error)	Adam	$5.56e-01 \pm 1.99e-01$
	Implicit Curriculum Learning	$3.44e-01 \pm 2.05e-01$
Eikonal (Relative Error)	Adam	$8.02e-01 \pm 9.78e-01$
	Implicit Curriculum Learning	$2.05e-02 \pm 1.09e-02$
Burgers (Evaluation Loss)	Adam	$1.32e-02$
	Implicit Curriculum Learning	$7.21e-03$

Conclusions

- Anagram: lowers natural-gradient cost from $O(P^3)$ to $O(\min(PN^2, P^2N))$ (overfactor $\min(P, N)$ w.r.t SGD).
- AMStramGRAM gives a principled way to adapt cutoff reaching machine-level error.
- We prove that PINNs natural gradient corresponds to an optimal linear update following the Green's function.
- Empirical results are competitive with state-of-the-art PINNs optimizers.

Perspectives

- Connection with FEMs through the notion of kernel and localization.
- Implicit Curriculum Learning.
- Design of an optimal collocation points procedure, coupled with AMStramGRAM's cutoff adaptation strategy.
- Include data assimilation in this theoretical setting, and understand its regularizing effect.
- Include common optimization techniques (e.g. Momentum)
- Extension to Operator learning

Thank you for your attention !

Publications

- Schwencke, N. and C. Furtlehner (2025): “ANaGRAM: A Natural Gradient Relative to Adapted Model for Efficient PINNs Learning,” in The Thirteenth International Conference on Learning Representations.
- Marie-Anne, J., C. Rousselot, N. Schwencke, and A. Shilova (2025): “Implicit Function Theorem in Physics-Informed Neural Networks to Solve Parameterized Differential Equations,” in EurIPS 2025 Workshop: Differentiable Systems and Scientific Machine Learning.

Preprint

- Schwencke, N., C. Rousselot, A. Shilova, and C. Furtlehner (2025): “AMStramGRAM: Adaptive Multi-Cutoff Strategy Modification for ANaGRAM,” arXiv Preprint.

- AMARI, S.-I. AND S. C. DOUGLAS (1998): “Why Natural Gradient?” in *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP’98 (Cat. No. 98CH36181)*, IEEE, vol. 2, 1213–1216.
- BEREZANSKY, Y. M., Z. G. SHEFTEL, AND G. F. US (1996): *Functional Analysis. Vol. II*, vol. 86 of *Operator Theory Advances and Applications*, Birkhäuser.
- BERNER, J., P. GROHS, G. KUTYNIOK, AND P. PETERSEN (2021): “The Modern Mathematics of Deep Learning,” *arXiv preprint arXiv:2105.04026*, 86–114.
- BRUNTON, S. L., J. L. PROCTOR, AND J. N. KUTZ (2016): “Discovering Governing Equations from Data by Sparse Identification of Nonlinear Dynamical Systems,” *Proceedings of the National Academy of Sciences*, 113, 3932–3937.
- CHEN, R. T., Y. RUBANOVA, J. BETTENCOURT, AND D. K. DUVENAUD (2018): “Neural Ordinary Differential Equations,” *Advances in neural information processing systems*, 31.

- JACOT, A., F. GABRIEL, AND C. HONGLER (2018): “Neural Tangent Kernel: Convergence and Generalization in Neural Networks,” *Advances in neural information processing systems*, 31.
- LAGARIS, I. E., A. LIKAS, AND D. I. FOTIADIS (1998): “Artificial Neural Networks for Solving Ordinary and Partial Differential Equations,” *IEEE transactions on neural networks*, 9, 987–1000.
- LI, Z., N. KOVACHKI, K. AZIZZADENESHELI, B. LIU, K. BHATTACHARYA, A. STUART, AND A. ANANDKUMAR (2020): “Fourier Neural Operator for Parametric Partial Differential Equations,” *arXiv preprint arXiv:2010.08895*.
- LU, L., P. JIN, AND G. E. KARNIADAKIS (2021): “DeepONet: Learning Nonlinear Operators for Identifying Differential Equations Based on the Universal Approximation Theorem of Operators,” *Nature Machine Intelligence*, 3, 218–229.

- MOUKALLED, F., L. MANGANI, AND M. DARWISH (2016): *The Finite Volume Method in Computational Fluid Dynamics: An Advanced Introduction with OpenFOAM® and Matlab*, vol. 113 of *Fluid Mechanics and Its Applications*, Cham: Springer International Publishing.
- MÜLLER, J. AND M. ZEINHOFER (2023): “Achieving High Accuracy with PINNs via Energy Natural Gradient Descent,” in *International Conference on Machine Learning*, PMLR, 25471–25485.
- RAISSI, M., P. PERDIKARIS, AND G. KARNIADAKIS (2019): “Physics-Informed Neural Networks: A Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear Partial Differential Equations,” *Journal of Computational Physics*, 378, 686–707.
- RUDNER, T. G., F. WENZEL, Y. W. TEH, AND Y. GAL (2019): “The Natural Neural Tangent Kernel: Neural Network Training Dynamics under Natural Gradient Descent,” in *4th Workshop on Bayesian Deep Learning (NeurIPS 2019)*.

- SHIN, Y., J. DARBON, AND G. E. KARNIADAKIS (2020): “On the Convergence of Physics Informed Neural Networks for Linear Second-Order Elliptic and Parabolic Type PDEs,” *Communications in Computational Physics*, 28, 2042–2074.
- STRIKWERDA, J. C. (2004): *Finite Difference Schemes and Partial Differential Equations, Second Edition*, Society for Industrial and Applied Mathematics.
- URBÁN, J. F., P. STEFANOU, AND J. A. PONS (2025): “Unveiling the Optimization Process of Physics Informed Neural Networks: How Accurate and Competitive Can PINNs Be?” *Journal of Computational Physics*, 523, 113656.
- ZIENKIEWICZ, O. C. AND R. L. TAYLOR (2013): *The Finite Element Method: Its Basis and Fundamentals*, Butterworth-Heinemann.

Definition (Green's function of D)

A Green's function is any kernel function $g : \Omega \times \Omega \rightarrow \mathbb{R}$ such that the operator:

$$R : f \in D[\mathcal{H}] \mapsto \left(x \in \Omega \mapsto \int_{\Omega} g(x, s) f(s) \mu(ds) \right) \in \mathcal{H}$$

verifies the equation: $D \circ R = I_{D[\mathcal{H}]}$

Definition (generalized Green's function of D on $\mathcal{H}_0 \subset \mathcal{H}$)

A generalized Green's function is any kernel function $g : \Omega \times \Omega \rightarrow \mathbb{R}$ such that the operator:

$$R : f \in L^2(\Omega \rightarrow \mathbb{R}, \mu) \mapsto \left(x \in \Omega \mapsto \int_{\Omega} g(x, s) f(s) \mu(ds) \right) \in \mathcal{H}$$

verifies the equation: $D \circ R = \Pi_{D[\mathcal{H}_0]}^{\perp}$

Theorem

Let $D : \mathcal{H} \rightarrow L^2(\Omega \rightarrow \mathbb{R}, \mu)$ be a linear differential operator and $\mathcal{H}_0 \subset \mathcal{H}$ an RKHS with kernel k_0 . Given the spectral decomposition:

$$\Pi_{\mathcal{H}_0} D^* D \Pi_{\mathcal{H}_0} = \int_0^{+\infty} \lambda \pi_{D, \mathcal{H}_0}(d\lambda) \quad I_{\mathcal{H}_0} = \int_0^{+\infty} \pi_{D, \mathcal{H}_0}(d\lambda)$$

Then the generalized Green's function at regularization level $\alpha > 0$ is given by:
for all $x, y \in \Omega$

$$g_{\mathcal{H}_0, \alpha}(x, y) := D \left[\int_{\alpha^2}^{+\infty} \lambda^{-1} \pi_{D, \mathcal{H}_0}(d\lambda) [k(x, \cdot)] \right] (y)$$

Strong formulation

Find $u \in H^2(\Omega)$ such that

$$\begin{cases} \Delta u = f \in L^2(\Omega) & \text{in } \Omega \\ u = 0 \in L^2(\partial\Omega) & \text{on } \partial\Omega \end{cases}.$$

Weak formulation

Find $u \in H_0^1(\Omega)$ such that: $\forall v \in H_0^1(\Omega)$,

$$\langle \nabla u, \nabla v \rangle_{L^2(\Omega \rightarrow \mathbb{R}^d)} = \langle v, -f \rangle_{L^2(\Omega)}.$$

Galerkin method

Fixing a finite dimensional space

$$H_n \subset H_0^1(\Omega).$$

Find $u \in H_n$ such that: $\forall v \in H_n$,

$$\langle \nabla u, \nabla v \rangle_{L^2(\Omega \rightarrow \mathbb{R}^d)} = \langle v, -f \rangle_{L^2(\Omega)}.$$

Kernelization of Galerkin's Method

Kernelization of Galerkin

Let $(v_i^n)_{i=1}^n$ be a basis of H_n .

$$\mathcal{T}_{H_n} : \theta \in \mathbb{R}^n \mapsto \sum_{i=1}^n \theta_i v_i^n \in H_n.$$

$$\text{NNTK}_s(x, y) = \sum_{1 \leq p, q \leq n} \partial_p \mathcal{T}_{H_n} G_{sp,q}^\dagger \partial_q \mathcal{T}_{H_n}.$$

Strong formulation

$$G_{2p,q} = \langle \Delta \partial_p \mathcal{T}_{H_n}, \Delta \partial_q \mathcal{T}_{H_n} \rangle_{L^2(\Omega)}.$$

$$u_n = \langle \Delta \text{NNTK}_2(x, \cdot), f \rangle_{L^2(\Omega)}.$$

Weak formulation

$$G_{1p,q} = \langle \nabla \partial_p \mathcal{T}_{H_n}, \nabla \partial_q \mathcal{T}_{H_n} \rangle_{L^2(\Omega \rightarrow \mathbb{R}^d)}.$$

$$u_n = \langle \text{NNTK}_1(x, \cdot), -f \rangle_{L^2(\Omega)}.$$

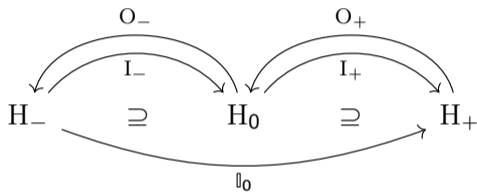


Figure: Schematic diagram of the basic structure of a Hilbert Rigging (a.k.a. Gelfand triple), adapted from Berezansky et al.1996).

O_+ : embedding of H_+ into H_0

$$I_+ := O_+^* : H_0 \rightarrow H_+$$

$$\langle \cdot, \cdot \rangle_- : (u, v) \in H_0^2 \mapsto \langle I_+ u, I_+ v \rangle_{H_+}$$

$$H_- := \overline{H_0}^{\|\cdot\|_-} \simeq H'_+$$

$$\mathbb{I}_0 := \overline{I_+} : H_- \rightarrow H_+$$

O_- : embedding of H_0 into H_-

$$I_- := O_-^* : H_- \rightarrow H_0$$

$$\mathbb{I}_0 O_- = I_+; \quad O_+ \mathbb{I}_0 = I_-$$

Weak-Strong equivalence

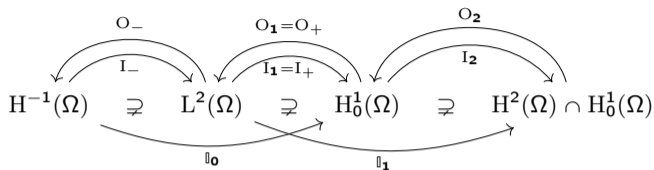


Figure: Schematic representation of the Hilbert rigging chain unifying the weak and strong formulations. Adapted from Berezansky et al.1996).

Proposition (Green's Identity)

$$\langle \Delta u, O_1 v \rangle_{L^2(\Omega)} = - \int_{\Omega} \langle \nabla u, \nabla v \rangle_{\mathbb{R}^d} = - \langle O_2 u, v \rangle_{H_0^1(\Omega)}.$$

Proposition

$$\Delta_2 := \Delta = -\mathbb{I}_1^* : H^2(\Omega) \cap H_0^1(\Omega) \rightarrow L^2(\Omega)$$

$$\Delta_1 := \overline{\Delta}^{H_0^1(\Omega)} = -\mathbb{I}_0^* : H_0^1(\Omega) \rightarrow H^{-1}(\Omega)$$

Test form of the Strong formulation

Find $u \in H^2(\Omega) \cap H_0^1(\Omega)$ s.t.,
 $\langle \Delta u, v \rangle_{H_0^1(\Omega)} = - \langle f, v \rangle_{H_0^1(\Omega)},$
 $\forall v \in H^2(\Omega) \cap H_0^1(\Omega).$

Least-squares form of the Weak formulation

Find $u \in H_0^1(\Omega)$ s.t $\Delta_1[u] = O_- f$
 or even $\Delta_1[u] = \alpha \in H^{-1}(\Omega).$

Proposition

$$u_{\text{weak}} = -I_1 f = O_2 u_{\text{strong}}$$

Form \ Formulation	Weak formulation	Strong formulation
Test form	$ \begin{aligned} & -O_1[\text{NNTK}_{1,n}(\mathbf{x}, \cdot)] \\ & \simeq -\text{NNTK}_{1,n}(\mathbf{x}, \cdot) \\ & \in L^2(\Omega) \end{aligned} $	$ \begin{aligned} & -O_2[\text{NNTK}_{2,n}(\mathbf{x}, \cdot)] \\ & \simeq -\text{NNTK}_{2,n}(\mathbf{x}, \cdot) \\ & \in H_0^1(\Omega) \end{aligned} $
Least-squares form	$ \begin{aligned} & -\mathbb{I}_0^*[\text{NNTK}_{1,n}(\mathbf{x}, \cdot)] \\ & = \Delta_1[\text{NNTK}_{1,n}(\mathbf{x}, \cdot)] \\ & \in H^{-1}(\Omega) \end{aligned} $	$ \begin{aligned} & -\mathbb{I}_1^*[\text{NNTK}_{2,n}(\mathbf{x}, \cdot)] \\ & = \Delta_2[\text{NNTK}_{2,n}(\mathbf{x}, \cdot)] \\ & \in L^2(\Omega) \end{aligned} $

N.B: While those approximations yield the same asymptotic solution, the approximations (*i.e.* solutions on H_n) differ.

Table: Summary of the Green's functions associated with each combination of formulation and form, computed over a finite-dimensional subspace H_n of functions, with $\mathbf{x} \in \Omega$.

Illustration of natural gradient dynamics

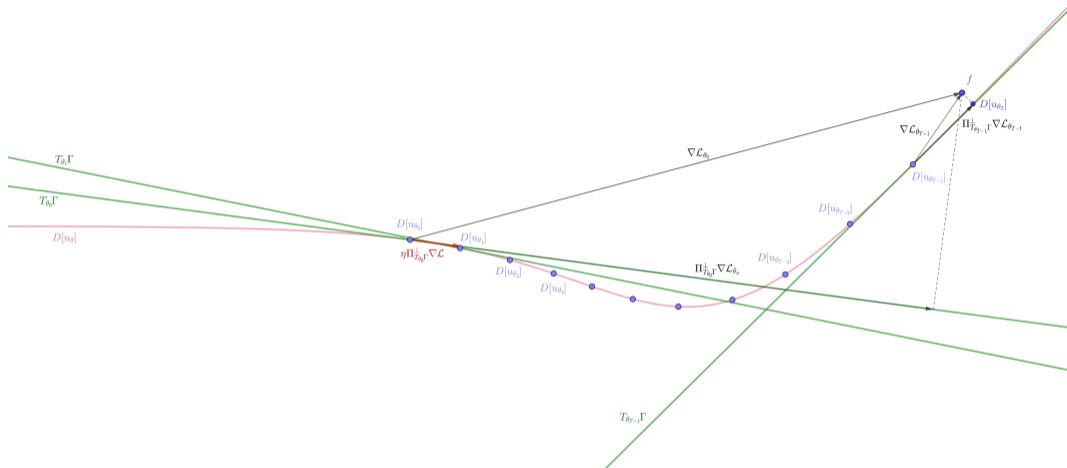


Figure: Illustration of PINNs learning process under natural gradient, as successive applications of Green's function

Remark

$$(x_i)^* = \operatorname{argmin}_{(x_i) \in \Omega^S} \left\| \Pi_{\hat{T}_{\theta, K}^{\perp}(x_i) \mathcal{M}} \nabla \mathcal{L}|_{u_{\theta t}} - \nabla \mathcal{L}|_{u_{\theta t}} \right\|_{L^2} = \operatorname{argmin}_{(x_i) \in \Omega^S} \inf_{\alpha \in \mathbb{R}^S} \left\| \sum_{i=1}^S \alpha_i K_{\theta}(x_i, \cdot) - \nabla \mathcal{L}|_{u_{\theta t}} \right\|_{L^2}^2$$

Consequence

$(x_i)^*$ can be “learned” by the minimization through natural gradient descent of

$$u : \begin{cases} \Omega^S \times \mathbb{R}^S & \rightarrow L^2(\Omega \rightarrow \mathbb{R}, \mu) \\ ((x_i), \alpha) & \mapsto \sum_{i=1}^S \alpha_i K_{\theta}(x_i, \cdot) \end{cases}$$

Even better: closed form formulas exist !

Proposition

- $\langle \partial_{\alpha_i} u_{\theta}, \partial_{\alpha_j} u_{\theta} \rangle = K_{\theta}(x_i, x_j)$
- $\langle \partial_{x_i} u_{\theta}, \partial_{\alpha_j} u_{\theta} \rangle = \alpha_j \partial_1 K_{\theta}(x_i, x_j)$
- $\langle \partial_{x_i} u_{\theta}, \partial_{x_j} u_{\theta} \rangle = \alpha_i \partial_2 \partial_1 K_{\theta}(x_i, x_j) \alpha_j$
- $\langle \partial_{\alpha_i} u_{\theta}, \nabla \mathcal{L} \rangle = \Pi_{T_{\theta} \mathcal{M}} \nabla \mathcal{L}(x_i) \simeq \nabla \mathcal{L}(x_i)$
- $\langle \partial_{x_i} u_{\theta}, \nabla \mathcal{L} \rangle = \alpha_i \Pi_{T_{\theta} \mathcal{M}} \nabla \mathcal{L}'(x_i) \simeq \alpha_i \nabla \mathcal{L}'(x_i)$

First results for collocation learning in Fourier space

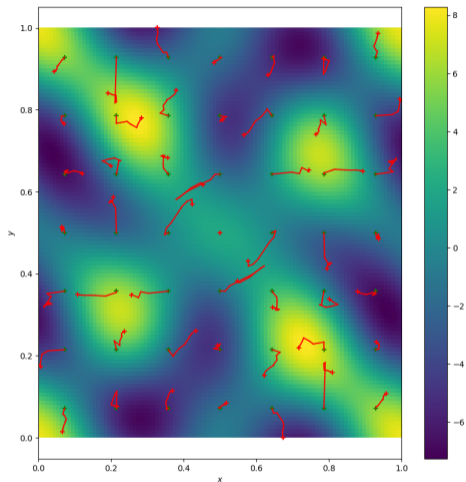


Figure: Points learning dynamic

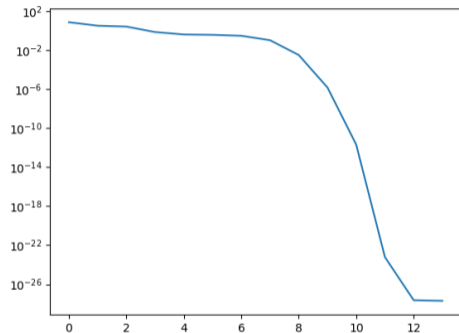


Figure: $\left\| \Pi_{\hat{\mathcal{T}}_{\theta, K}^{(x_i)} \mathcal{M}}^\perp \nabla \mathcal{L}|_{u_{\theta_t}} - \nabla \mathcal{L}|_{u_{\theta_t}} \right\|_{L^2}$ wrt (x_i)
learning steps

In the population limit, the natural gradient of PINNs is the update:

$$\theta_{t+1} \leftarrow \theta_t - \eta d((D, B) \circ u)_{\theta_t}^{\dagger} \left(\Pi_{T_{\theta_t} \Gamma}^{\perp} \nabla \mathcal{L}|_{u_{\theta_t}} \right)$$

Corollary

The kernel of $\Pi_{T_{\theta} \Gamma}$ is: for all $x, y \in (\Omega \times \partial\Omega)^2$

$$\begin{aligned} NNTK_{\theta}(x, y) &= \sum_{1 \leq p, q \leq P} \partial_p(D, B)[u_{\theta}](x) G_{\theta_{p,q}}^{\dagger} \partial_q(D, B)[u_{\theta}](y) \\ &= \sum_{1 \leq p, q \leq P} (\partial_p D[u_{\theta}](x_1), \partial_p B[u_{\theta}](x_2)) G_{\theta_{p,q}}^{\dagger} (\partial_q D[u_{\theta}](y_1), \partial_q B[u_{\theta}](y_2)), \end{aligned}$$

where for all $1 \leq p, q \leq P$

$$\begin{aligned} G_{\theta_{p,q}} &:= \langle \partial_p(D, B)[u_{\theta}], \partial_q(D, B)[u_{\theta}] \rangle_{L^2(\Omega \rightarrow \mathbb{R}, \mu) \times L^2(\partial\Omega \rightarrow \mathbb{R}, \sigma)} \\ &= \langle \partial_p D[u_{\theta}], \partial_q D[u_{\theta}] \rangle_{L^2(\Omega \rightarrow \mathbb{R}, \mu)} + \langle \partial_p B[u_{\theta}], \partial_q B[u_{\theta}] \rangle_{L^2(\partial\Omega \rightarrow \mathbb{R}, \sigma)}. \end{aligned}$$

ANaGRAM empirical validation complements 5 D Laplace equation

2 D Laplace equation

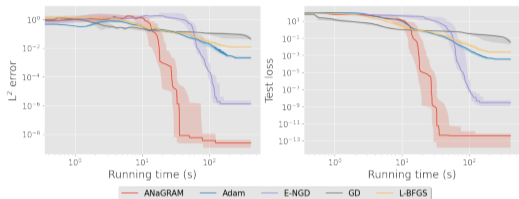


Figure: Performance comparison w.r.t running time for Laplace equation in 2 D:

$$\begin{cases} \Delta u = -2\pi^2 \sin(\pi x_1) \sin(\pi x_2) & \text{in } [0, 1]^2 \\ u = 0 & \text{on } \partial[0, 1]^2 \end{cases}$$

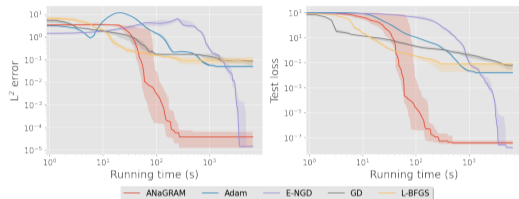


Figure: Performance comparison w.r.t running time for Laplace equation in 5 D:

$$\begin{cases} \Delta u = \pi^2 \sum_{k=1}^5 \sin(\pi x_k) & \text{in } \Omega = [0, 1]^5 \\ u = \sum_{k=1}^5 \sin(\pi x_k) & \text{on } \partial\Omega \end{cases}$$

Burgers equation with ANaGRAM

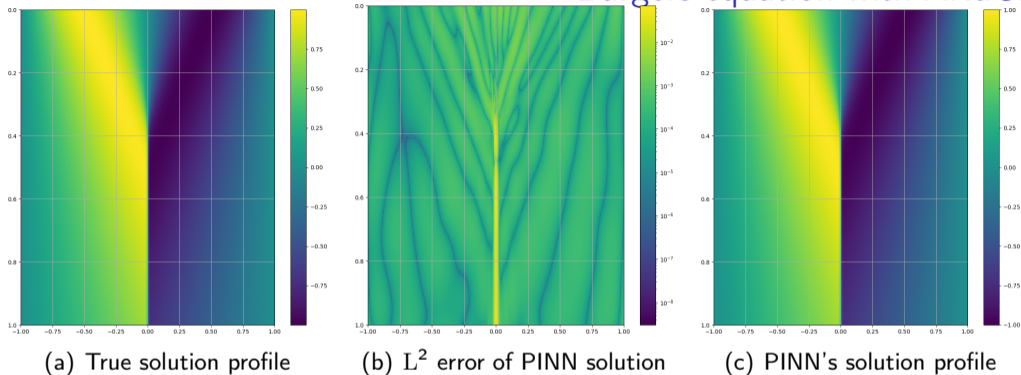


Figure: PINN's solution under Adam (15k steps) + L-BFGS (15k) steps for Burgers equation in 1+1 D:

$$\begin{cases} \partial_t u + u \partial_x u = \nu \partial_{xx} u & \text{in } [0, 1] \times [-1, 1] \\ u = 0 & \text{on } [0, 1] \times \{-1, 1\} \\ u = -\sin(\pi x) & \text{on } \{0\} \times [-1, 1] \end{cases}$$