Frequency-aware multigrid training of Physics-Informed Neural Networks

Elisa Riccietti

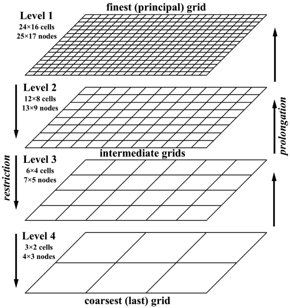
Collaboration with
S. Gratton and V. Mercier (INP Toulouse), P. Toint (Unamur)

LIP-ENS Lyon

Journée Représentations Neuronales Implicites : des NeRF aux PINN
04 November 2025

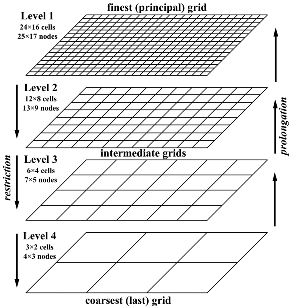
The context: multigrid methods and PINNS

Classical multigrid (MG) : accelerate the solution of PDEs by exploiting the structure to build a hierarchy of subproblems



The context: multigrid methods and PINNS

Classical multigrid (MG) : accelerate the solution of PDEs by exploiting the structure to build a hierarchy of subproblems



Question: can we extend this concept to PINNs training?

Outline

Multigrid (MG) methods

Multigrid training of PINNs

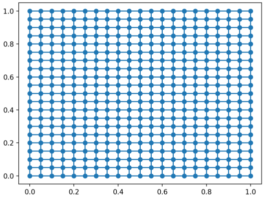
Outline

Multigrid (MG) methods

Multigrid training of PINNs

The numerical solution of PDEs

- Classically PDEs are discretized on a grid
- The resulting linear system $Au = f$ is solved using a fixed point method
- The size of the grids impacts:
  - the size of the system
  - the accuracy of the solution approximation



Fixed point schemes

Fixed point scheme:

$$u^* = Bu^* + g$$
$$u^{(m+1)} = Bu^{(m)} + g$$

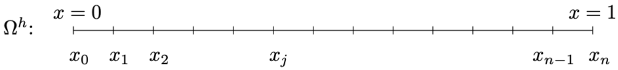
Fixed point schemes

Fixed point scheme:

$$u^* = Bu^* + g$$
$$u^{(m+1)} = Bu^{(m)} + g$$

Limitations: What are the limitations of such schemes?

$$\begin{cases} -u''(x) = f(x), \text{ for } 0 < x < 1 \\ u(0) = u(1) = 0 \end{cases}$$



- Jacobi method

Reduction of the error

After M iterations:

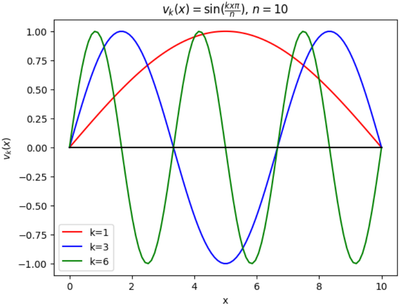
$$e^{(M)} = u^{(M)} - u^* = \sum_{k=1}^{n-1} c_k \lambda_k^M(B) v_k$$

Fourier modes:

$$v_k(j) = \sin\left(\frac{kj\pi}{n}\right), \quad k \text{ frequency component}$$

On a n-point grid:

- $1 \leq k < \frac{n}{2}$ low
- $\frac{n}{2} \leq k < n-1$ high



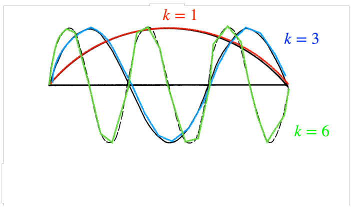
Limitation of iterative schemes: the smoothing property

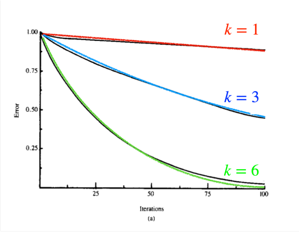
$$e^{(M)} = u^{(M)} - u^* = \sum_{k=1}^{n-1} c_k \lambda_k^M(B) v_k$$

- $\lambda_1(B) \approx 1$
- $|\lambda_k(B)| < 1/3$ for $n/2 \leq k \leq n-1$

  Hard to reduce the low frequency components of the error

$$v_k(x) = sin\left(\frac{xk\pi}{n}\right), 0 \leq x \leq n$$



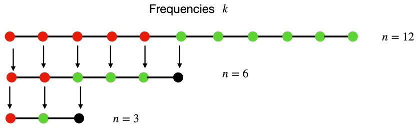


How to make the methods efficient on all frequencies?

Frequency shift!

- **Fine** grid Ω^h with n points: $1 \leq k \leq n-1$
- **Coarse** grid Ω^{2h} with $n/2$ points: $1 \leq k \leq n/2$

Property: $v_k^h(2j) = v_k^{2h}(j) \quad k < n/2$



Two-level multigrid methods

Consider a PDE:

$$\mathcal{A}(u) = f.$$

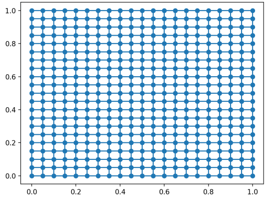
Consider two discretizations:

- Fine grid: $\mathcal{A}_h(u_h) = f_h$
- Coarse grid: $\mathcal{A}_H(u_H) = f_H$

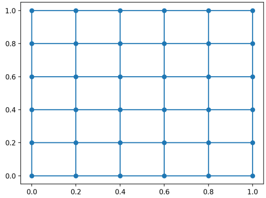
Idea: write the solution u as the sum of a fine and a coarse term:

$$u \sim \underbrace{v_h}_{\in\mathbb{R}^h} + P(\underbrace{e_H}_{\in\mathbb{R}^H}), \; H < h.$$

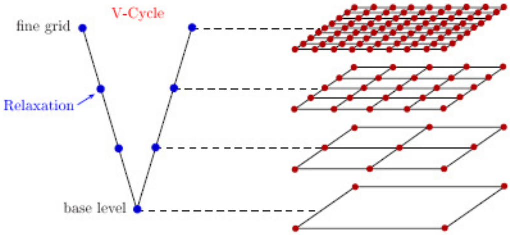
and update the two components in an alternate fashion.



$R \Downarrow \qquad P \Uparrow$



General multigrid methods



W. Briggs, V. Henson, S. McCormick. A Multigrid Tutorial, SIAM, 2000.

Two beneficial effects:

- dimensionality reduction
- efficiency in attacking selected frequencies

Outline

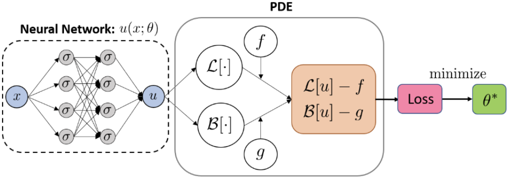
Multigrid (MG) methods

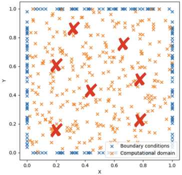
Multigrid training of PINNs

Physics Informed Neural Networks (PINNs)

Neural network $NN(\theta, x) \approx u(x)$

Training data

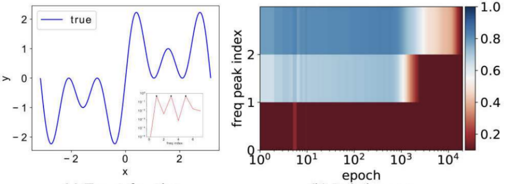




Training problem: $\min_{\theta \in \Theta} L(\theta) = L_{OBS}(\theta) + L_{PDE}(\theta)$

Aim: We want to accelerate the solution of this problem

The F-principle



(a) Target function (b) Relative error

Figure 1: 1d input. (a) $f(x)$. Inset : $|\hat{f}(k)|$. (b) $\Delta_F(k)$ of three important frequencies (indicated by black dots in the inset of (a)) against different training epochs. The parameters of the DNN is initialized by a Gaussian distribution with mean 0 and standard deviation 0.1. We use a tanh-DNN with widths 1-8000-1 with full batch training. The learning rate is 0.0002. The DNN is trained by Adam optimizer [20] with the MSE loss function.

$\Rightarrow$ PINNs are not effective in approximating highly oscillatory solutions

On the Spectral Bias of Neural Networks

Nasim Rahaman [*1 2] Aristide Baratin [*1] Devansh Arpit [1] Felix Draxler [2] Min Lin [1] Fred A. Hamprecht [2] Yoshua Bengio [1] Aaron Courville [1]

WHEN AND WHY PINNS FAIL TO TRAIN: A NEURAL TANGENT KERNEL PERSPECTIVE

A PREPRINT

Sifan Wang
Graduate Group in Applied Mathematics and Computational Science
University of Pennsylvania
Philadelphia, PA 19104
sifanw@sas.upenn.edu

Xinling Yu
Graduate Group in Applied Mathematics and Computational Science
University of Pennsylvania
Philadelphia, PA 19104
xlyu@sas.upenn.edu

Paris Perdikaris
Department of Mechanichal Engineering and Applied Mechanics
University of Pennsylvania
Philadelphia, PA 19104
pgp@seas.upenn.edu

How to transpose the ingredients of success of MG

Basic idea of MG: Exploiting "complementarity" between problems involved

Classical MG vs Neural networks

- Consider a minimization method and a class of problems for which this method is efficient

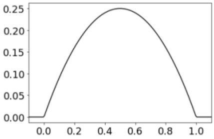
Classical MG	Neural networks
smoothing (GS or J)	first-order (GD, SGD)
high-frequency	low-frequency

- Split the problem depending of its frequency content
- Shift the frequencies

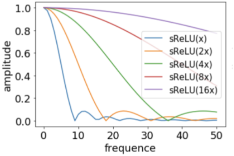
Classical MG	Neural networks
Coarser discretizations	Specialized architectures (Mscale networks)

Specialized architectures

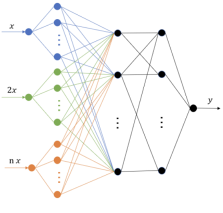
- **Mscale networks**: [Liu, Cai and Xu, (2020)]
  frequency-selective subnetworks + wavelet-inspired and frequency-located activation functions



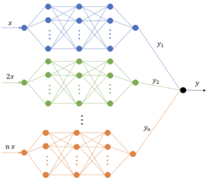
(b) sReLU



(a) sReLU

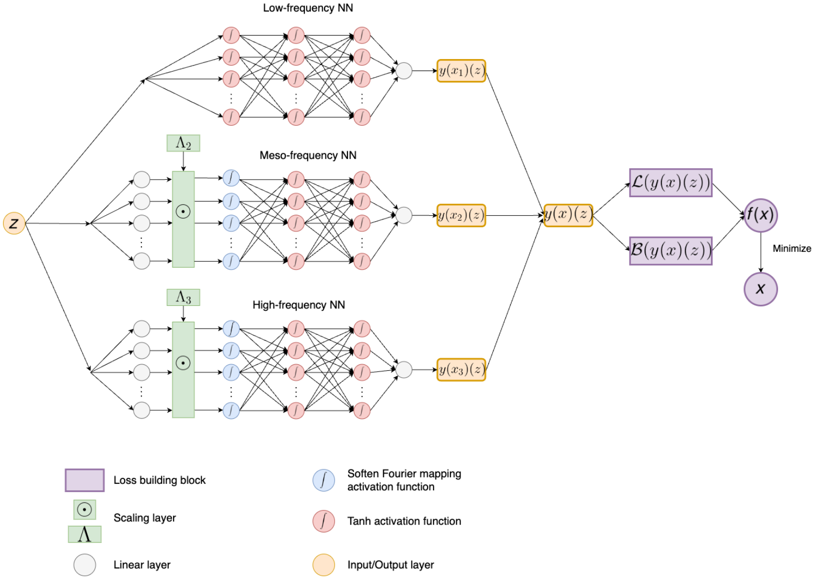


(a) MscaleDNN-1



(b) MscaleDNN-2

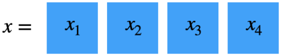
Our architecture



A BCD-MG algorithm: an iteration

$$\min_x f(x)$$

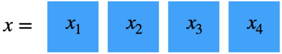
1 Partition x in blocks:



A BCD-MG algorithm: an iteration

$$\min_x f(x)$$

1 Partition x in blocks:



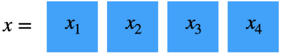
2 Select a block i $(x_1, \ldots, x_i, \ldots, x_n)$

- Criterion: $\|\nabla_i f(x)\| \geq \tau \|\nabla f(x)\|$, $\tau \in (0, 1)$

A BCD-MG algorithm: an iteration

$$\min_x f(x)$$

1 Partition x in blocks:



2 Select a block i $(x_1, \ldots, x_i, \ldots, x_n)$

- Criterion: $\|\nabla_i f(x)\| \geq \tau \|\nabla f(x)\|$, $\tau \in (0, 1)$

3 Update the block:

- p_k iterations of a first-order method (possibly *stochastic*)

$$\min_{x_i} f(x_1, \ldots, x_i, \ldots, x_n) \rightarrow x_i^{new}$$

- $x_i \leftarrow x_i^{new}$

A BCD-MG algorithm: convergence theory

Theorem (Gratton, Mercier, R., Toint, 2023)

If f has L-Lipschitz continuos gradient and step-size $\alpha_k = \alpha < 1/L$

- **Deterministic**

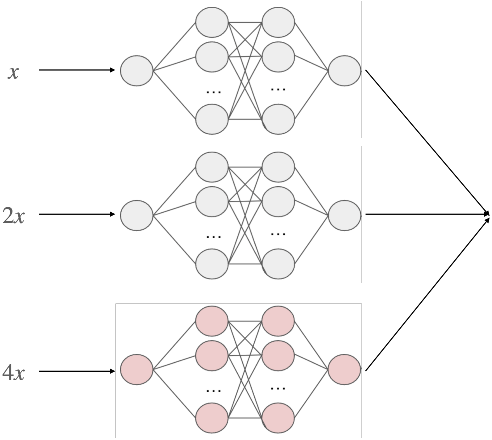
$$\|\nabla f(x^{(K)})\| \leq \epsilon \to K = O\left(\frac{1}{\epsilon^2 p}\right)$$

- **Stochastic**

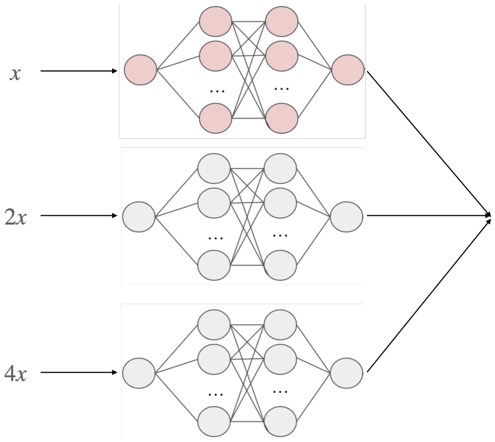
$$\mathbb{E}\left(\sum_{k=1}^{K} \|\nabla f(x^{(k)})\|^2\right) \leq C_1(\sigma^2) + O\left(\frac{1}{K}\right) - C_2(\sigma^2)p$$

p coarse iterations, σ^2 variance of stochastic gradient

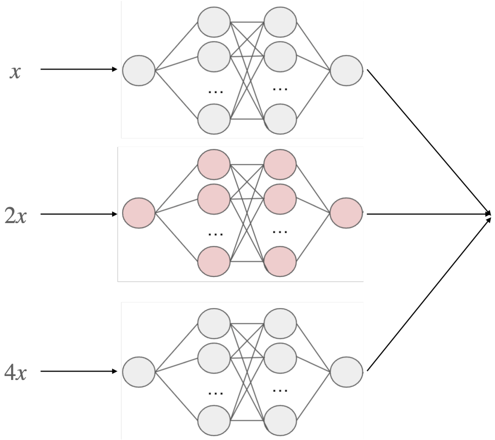
BCD-MG training of PINNs



BCD-MG training of PINNs

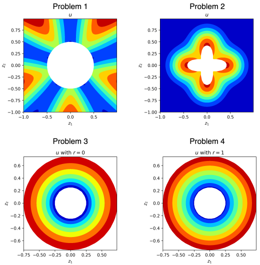


BCD-MG training of PINNs

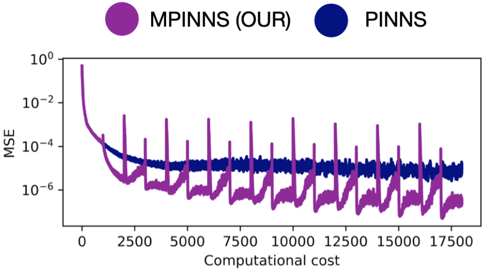


Numerical results: MSE vs iterations

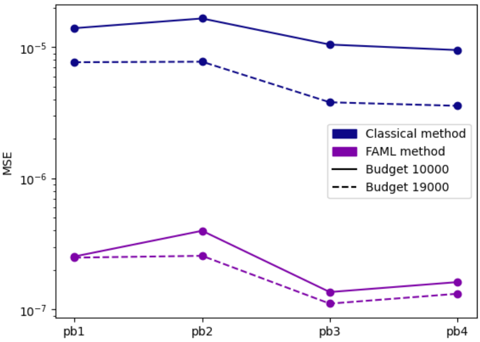
Problem: Poisson problem with Dirichlet/Neumann BC on Ω



Ω



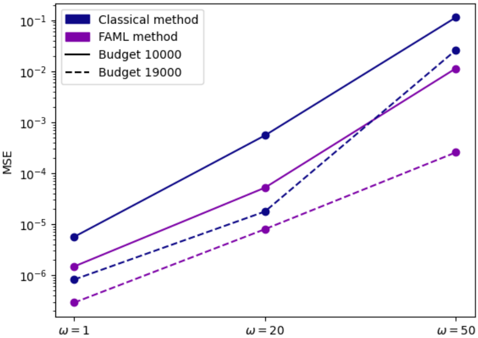
Numerical results: final MSE on average (10 runs)



Numerical results: MSE vs iterations

Problem: Heat equation

$$\begin{cases} \dfrac{\partial u(z,t)}{\partial t} = \dfrac{1}{(\omega\pi)^2} \dfrac{\partial^2 u(z,t)}{\partial z^2}, \\ u(z,0) = \sin(\omega\pi z), \ z \in [0,1], \\ u(0,t) = u(1,t) = 0, \ t \in [0,1], \end{cases}$$



Conclusions

- We have presented a new MG training method for PINNs
- We have proved its convergence as a BCD method
- We have demonstrated that exploiting multiple scales provides significant computational benefits (faster convergence) and more accurate solution

Thank you for your attention !

S. Gratton, V. Mercier, E. Riccietti & Ph. L. Toint,
A block-coordinate approach of multi-level optimization with an application to physics-informed neural networks, *COAP*, 2024.