Multilevel Physics Informed Neural Networks (MPINNs)

Elisa Riccietti

LIP-ENS Lyon

Joint work with:
S. Gratton - V. Mercier (ENSEEIHT, IRIT), P. Toint (UNamur)

Journées SMAI MODE 2022, Limoges, 30 May - 3 June 2022

The context

The problem: numerical approximation of PDE's solutions.

- *Classical approaches*: discretization and multigrid methods (MG)
- *New advances in machine learning*: Physics Informed Neural Networks (PINNs)

Our objective:
Transfer the advantages of the first approach to the second.

Outline

Outline

Multigrid methods (MG)

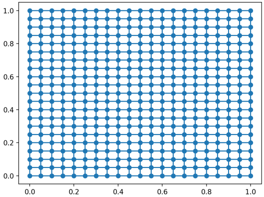
Physics Informed Neural Networks (PINNs)

Multilevel PINNs (MPINNs)

ENS DE LYON

The numerical solution of PDEs

- Classically PDEs are discretized on a grid using finite differences or finite elements
- The resulting linear system $Au = f$ is solved using a fixed point iterative method (Gauss-Siedel or Jacobi)
- The size of the grids impacts the size of the system and the accuracy of the solution approximation



UNIVERSITÉ DE LYON
ENS DE LYON

The intuition behind multigrid methods

- Example: $\Delta u = 0$, $v_k(j) = \sin(\frac{kj\pi}{n})$, k-th Fourier mode
- The smoothing property: hard for fixed point iterative methods to reduce the low frequency components of the error

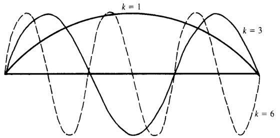
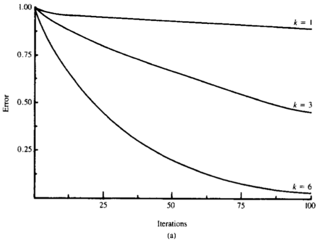


Figure 2.2: *The modes* $v_j = \sin\left(\frac{jk\pi}{n}\right)$, $0 \le j \le n$, *with wavenumbers* $k = 1, 3, 6$. *The* k*th mode consists of* $\frac{k}{2}$ *full sine waves on the interval.*



The intuition behind multigrid methods

- How does a smooth component look like on a coarser grid?

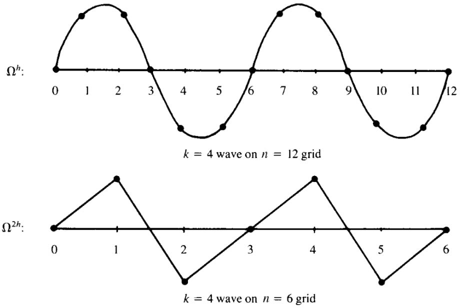
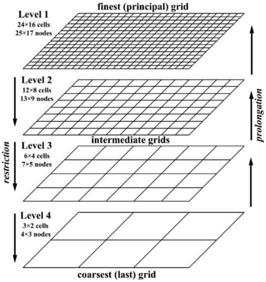


Figure 3.1: *Wave with wavenumber $k = 4$ on Ω^h ($n = 12$ points) projected onto Ω^{2h} ($n = 6$ points). The coarse grid "sees" a wave that is more oscillatory on the coarse grid than on the fine grid.*

Multigrid methods for PDEs

State-of-the-art methods for PDEs: exploit representation of the problem at different scales



- Fine scales: eliminate high frequency components of the error
- Coarse scales: eliminate low frequency components of the error

Two-level multigrid methods

Consider a linear PDE:

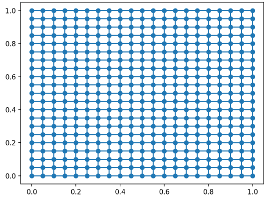
$$Au = f.$$

Consider two discretizations of the same system:

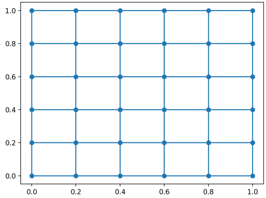
- Fine grid: $A_h u_h = f_h$
- Coarse grid: $A_H u_H = f_H$

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

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



$R \Downarrow \qquad P \Uparrow$



UNIVERSITÉ DE LYON

ENS DE LYON

Two-level multigrid methods

Update the two components in an alternate fashion:

$$u \sim v + e$$
$$r = f - Av$$
$$Ae = r \text{ residual equation}$$

- *Fine level*: get v_h by iterating on $A_h u = f_h$
- Compute $r_h = f - Av_h$ and project $r_H = Rr_h$
- *Coarse level*: compute correction: $A_H e_H = r_H$
- Correct: $v_h \leftarrow v_h + P(e_H)$

UNIVERSITÉ DE LYON ENS DE LYON

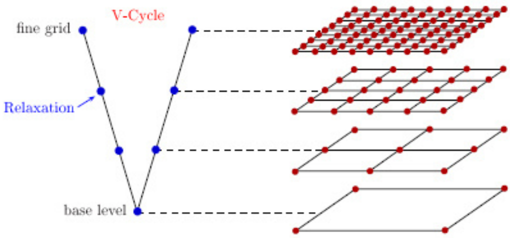
Two-level multigrid methods

Update the two components in an alternate fashion:

$$u \sim v + e$$
$$r = f - Av$$
$$Ae = r \text{ residual equation}$$

- *Fine level*: get v_h by iterating on $A_h u = f_h$
- Compute $r_h = f - Av_h$ and project $r_H = Rr_h$
- *Coarse level*: compute correction: $A_H e_H = r_H$
- Correct: $v_h \leftarrow v_h + P(e_H)$

General multigrid methods



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

Outline

Multigrid methods (MG)

Physics Informed Neural Networks (PINNs)

Multilevel PINNs (MPINNs)

UNIVERSITÉ DE LYON

ENS DE LYON

A new approach for PDEs

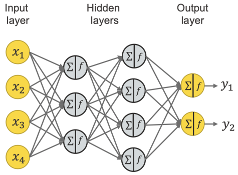
A recent development: use neural networks to approximate the solution of a PDE

M. Raissi, P. Perdikaris, G. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, 2019.

Why this approach ?

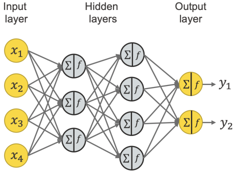
- Natural approach for nonlinear equations
- Provides analytic and continuously differentiable expression of the approximate solution
- The solution is meshless, well suited for problems with complex geometries
- The training is highly parallelizable on GPU
- Allows to alleviate the effect of the curse of dimensionality

General NN strategy for learning problems



- Dataset composed of input/output couples (x_i, y_i), $i = 1, \dots, m$.
- Loss function $L(\theta; x, y) = \frac{1}{m} \sum_{i=1}^{m} (NN(x_i, \theta) - y_i)^2 = MSE$
- The associated minimization problem : $\underset{\theta \in \Theta}{min} L(\theta; x, y)$
- Optimize by SGD

General NN strategy for learning problems

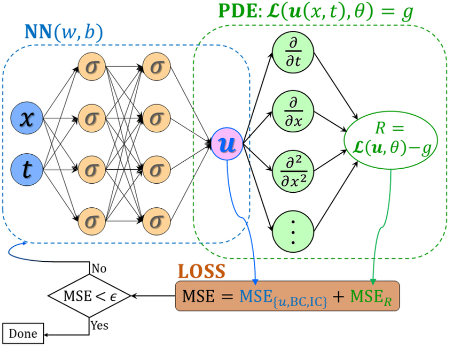


- Dataset composed of input/output couples (x_i, y_i), $i = 1, \dots, m$.
- Loss function $L(\theta; x, y) = \frac{1}{m}\sum_{i=1}^{m}(NN(x_i, \theta) - y_i)^2 = MSE$
- The associated minimization problem : $\min_{\theta \in \Theta} L(\theta; x, y)$
- Optimize by SGD

How to integrate the physical knowledge in the model?

Physics Informed Neural Networks (PINNs)

$\mathcal{L}(u(x,t),\theta) = g(x,t)$
$+$ BC $+$ IC
in $\Omega \times [0,T]$

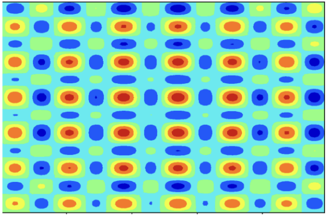


Given x_i, t_i sampled randomly, and $y_i = u^*(x_i,t_i)$,

$$MSE_{\{u,BC,IC\}} = \frac{1}{N_m} \sum_{x_i \in \Omega \cup \partial\Omega, t_i \in [0,T]} (NN(x_i,t_i) - y_i)^2,$$

$$MSE_R = \frac{1}{N_r} \sum_{x_i \in \Omega, t_i \in [0,T]} R(x_i,t_i)^2$$

On the spectral bias of neural networks



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

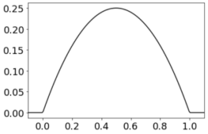
Paris Perdikaris
Department of Mechanichal Engineering
and Applied Mechanics
University of Pennsylvania
Philadelphia, PA 19104
pgp@seas.upenn.edu

⇒ PINNs are not effective in approximating highly oscillatory solutions

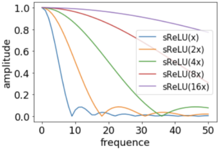
Mscale networks

Z.Q. Liu, W. Cai, and Z.Q. John Xu, Multi-scale Deep Neural Network (MscaleDNN) forSolving Poisson-Boltzmann Equation in Complex Domains, 2020

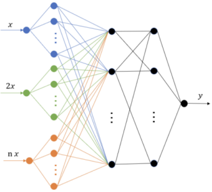
Idea: simultaneous training of frequency-selective subnetworks



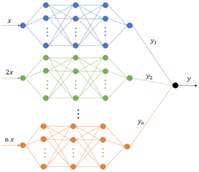
(b) sReLU



(a) sReLU



(a) MscaleDNN-1



(b) MscaleDNN-2

Outline

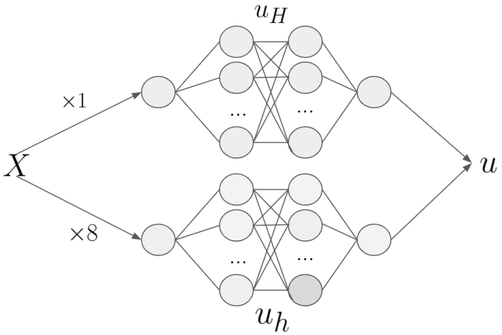
Multigrid methods (MG)

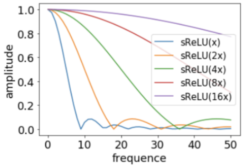
Physics Informed Neural Networks (PINNs)

Multilevel PINNs (MPINNs)

Multilevel PINNs: the architecture

Idea: $u_{sol}(x) \sim u_h(\theta_h, x) + u_H(\theta_H, x)$
Also exploit frequency-selective subnetworks





(a) sReLU

Multilevel PINNs: the loss

Problem definition

$$\mathcal{L}(u(x)) = g(x),\ x \in \Omega,$$
$$u_{sol}(x) \sim u_h(\theta_h, x) + u_H(\theta_H, x)$$

Fine problem	**Coarse problem**
$MSE_h(\theta_h) = MSE_{R,h}(\theta_h) + MSE_{B,h}(\theta_h)$	$MSE_H(\theta_H) = MSE_{R,H}(\theta_H) + MSE_{B,H}(\theta_H)$
$MSE_{R,h}(\theta_h) = \|\|\mathcal{L}(\hat{u}_h(\theta_h) + u_H) - g\|\|^2$	$MSE_{R,H}(\theta_H) = \|\|\mathcal{L}(\hat{u}_H(\theta_H) + u_h) - g\|\|^2$
$MSE_{B,h}(\theta_h) = \|\|\hat{u}_h(\theta_h) + u_H - u\|\|^2$	$MSE_{B,H}(\theta_H) = \|\|\hat{u}_H(\theta_H) + u_h - u\|\|^2$
Computed on z_h the fine sampling	Computed on z_H the coarse sampling

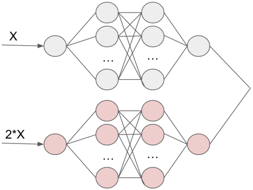
UNIVERSITÉ DE LYON

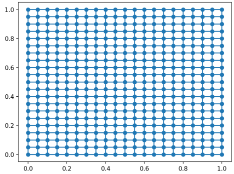
ENS DE LYON

Multilevel PINNs: the training

Algorithm 1 2-level training of PINNs

1: Freeze coarse-network parameters, unfreeze fine-network parameters
2: **for** i=1,2,... **do**
3: Perform ν_1 epochs for the minimization of the fine problem
4: Freeze fine-network parameters, unfreeze coarse-network parameters
5: Perform ν_2 epochs for the minimization of the coarse problem
6: **end for**
7: Return : $u_H + u_h$

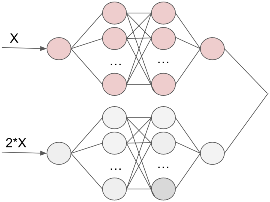


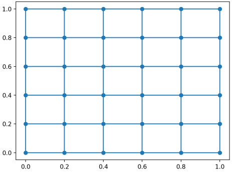


Multilevel PINNs: the training

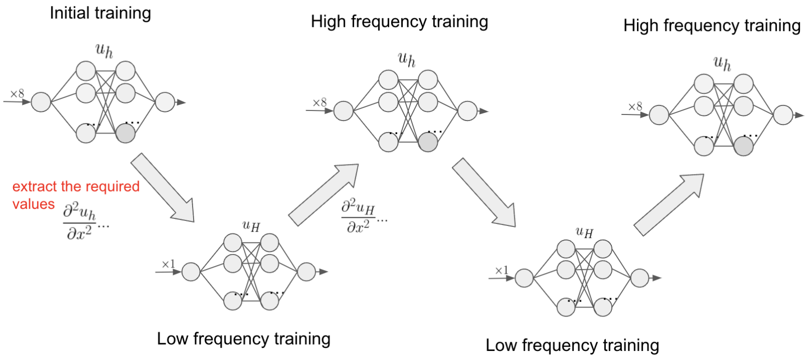
Algorithm 2 2-level training of PINNs

Freeze coarse-network parameters, unfreeze fine-network parameters
for i=1,2,... **do**
    Perform ν_1 epochs for the minimization of the fine problem
    Freeze fine-network parameters, unfreeze coarse-network parameters
    Perform ν_2 epochs for the minimization of the coarse problem
end for
Return : $u_H + u_h$



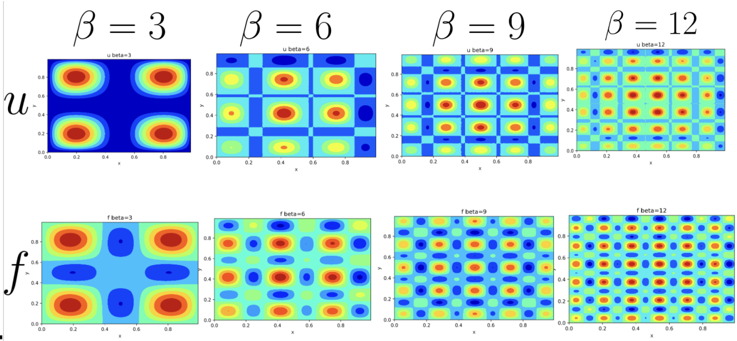


Multilevel PINNs: V-cycles



A simple Poisson problem

- $\Omega = [0, 1] \times [0, 1]$
- $\Delta u = f \ \forall x \in \Omega$
- $u = 0 \ \forall x \in \partial\Omega$
- $u(x, y) = (sin(\pi x) + sin(\beta\pi x)) * (sin(\pi y) + sin(\beta\pi y))$

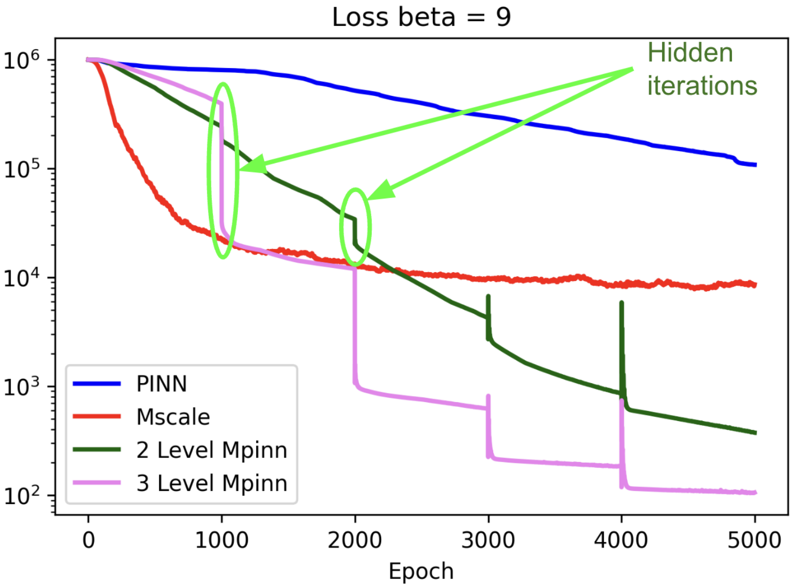


Experimental settings

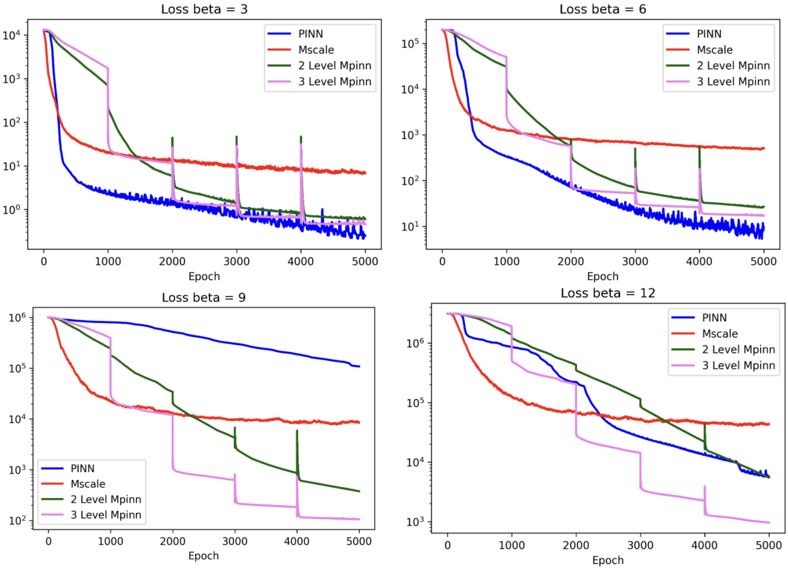
In what follows:

- The PINNs have two hidden layers of 300 neurons each.
- The Mscale have four subnetworks of two hidden layers of 150 neurons each, the input scaling used are 1,2,4 and 8.
- The two-level MPINN is composed of two networks of two hidden layers of 210 neurons each and trained in a V-cycle with 1 and 8 input scalings ($\nu_1 = \nu_2 = 1000$).
- The three-level MPINN is composed of three networks of two hidden layers of 150 neurons each and trained in a V-cycle with 1,4 and 8 input scalings ($\nu_1 = \nu_2 = \nu_3 = 1000$).
- The input of all networks is a regular grid sample of 80×80 points
- In all cases, we plot the median for ten random runs.

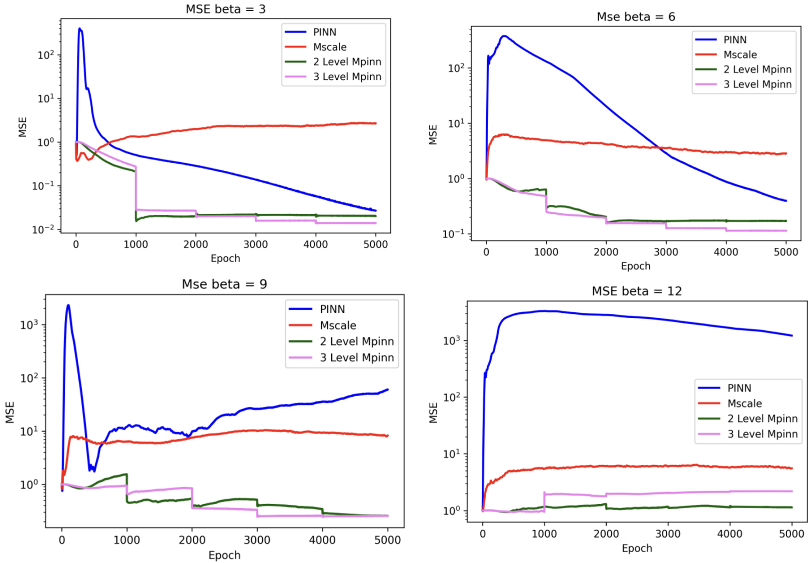
Experimental results



Varying β (the frequency content)

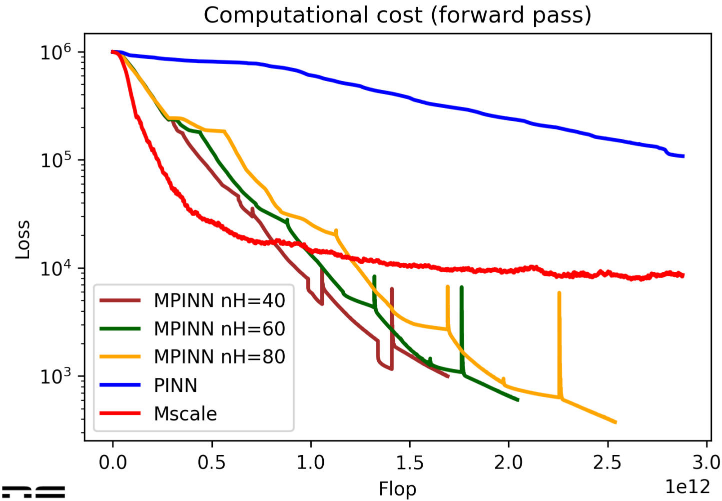


Convergence of MSE (extrapolation)



Computational cost for two levels...

...as a function of coarse grid size (nH)



Conclusions

- We have presented a new multigrid-inspired training framework using recent advances in NN to efficiently solve PINN-type problems.
- We have proposed an algorithm which works without prior knowledge of frequency content and which is promising.
- We have demonstrated that exploiting spectral complementarity using our framework may bring significant computational benefits (faster convergence).

Perspectives

- Perform further extensive testing, including more complex problems.
- Pursue the sensitivity analysis for
  - the relative sizes of the grids,
  - the relative sizes of the networks.
- Investigate theoretical aspects:
  - convergence of the iterates from an optimization point of view,
  - convergence to the solution in functional space.

UNIVERSITÉ DE LYON ENS DE LYON

Thank you for your attention!

A few references

- S. Gratton, A. Sartenaer, Ph. L. Toint. Recursive trust-region methods for multiscale nonlinear optimization, SIAM J. Opt., 19:414–444, 2008
- W. Briggs, V. Henson, S. McCormick. A Multigrid Tutorial, SIAM, 2000
- H. Calandra, S. Gratton, E. Riccietti, X. Vasseur. On high-order multilevel optimization strategies. SIAM J. Opt., 31.1: 307-330, 2021.
- S. Wang, H. Wang, P. Perdikaris. On the eigenvector bias of Fourier feature networks: From regression to solving multi-scale PDEs with physics-informed neural networks. CMAME, 384, 2021
- Z. Liu, W. Cai, Z. Xu. Multi-scale deep neural network (MscaleDNN) for solving Poisson-Boltzmann equation in complex domains. arXiv:2007.11207, 2020
- E. Riccietti, V. Mercier, S. Gratton, P. Boudier, Multilevel physics informed neural networks (MPINNs), technical report, 2021
- V. Mercier, S. Gratton, P. Boudier. A coarse space acceleration of deep-DDM, arxiv:2112.03732, 2022
- H. Calandra, S. Gratton, E. Riccietti, X. Vasseur. On a multilevel Levenberg-Marquardt method for the training of artificial neural networks and its application to the solution of partial differential equations, OMS, 2020