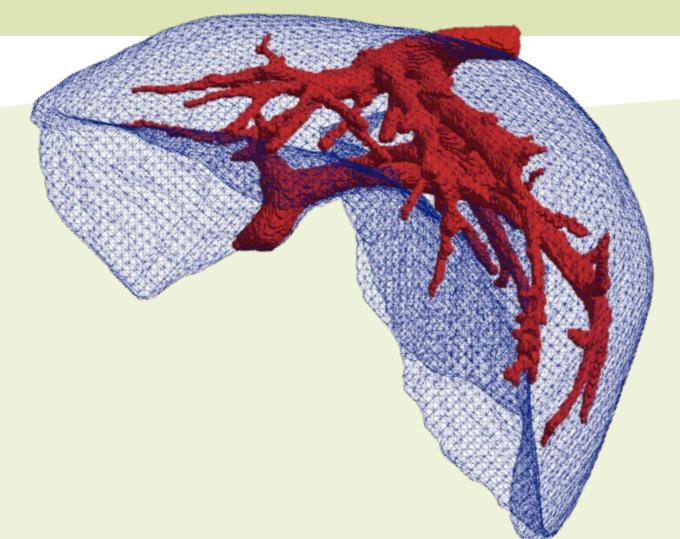


COMBINING FINITE ELEMENT METHODS AND NEURAL NETWORKS TO SOLVE ELLIPTIC PROBLEM ON COMPLEX 2D GEOMETRIES

Hélène BARUCQ³, Michel DUPREZ¹, Florian FAUCHER³, Emmanuel FRANCK², Frédérique LECOURTIER¹, Vanessa LLERAS⁴, Victor MICHEL-DANSAC², and Nicolas VICTORION³

¹ Mimesis team, INRIA Nancy grand Est, Icube ² Macaron team, INRIA Nancy grand Est, IRMA ³ Makutu team, INRIA Bordeaux, TotalEnergies ⁴ Montpellier University

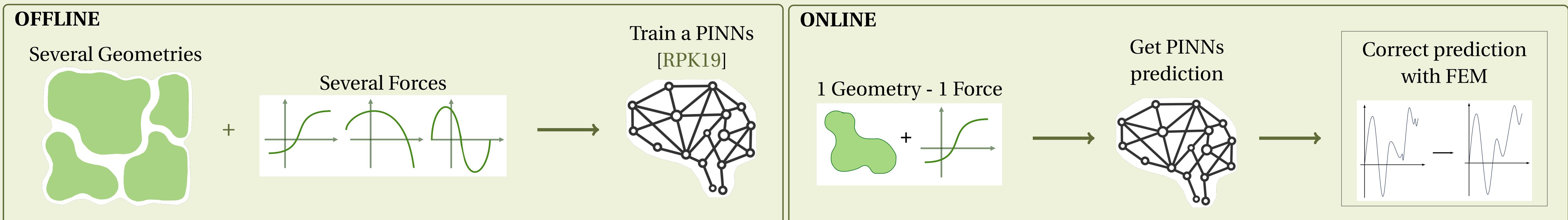
MIMESIS



Scientific context

Create real-time digital twins of an organ (e.g. liver)

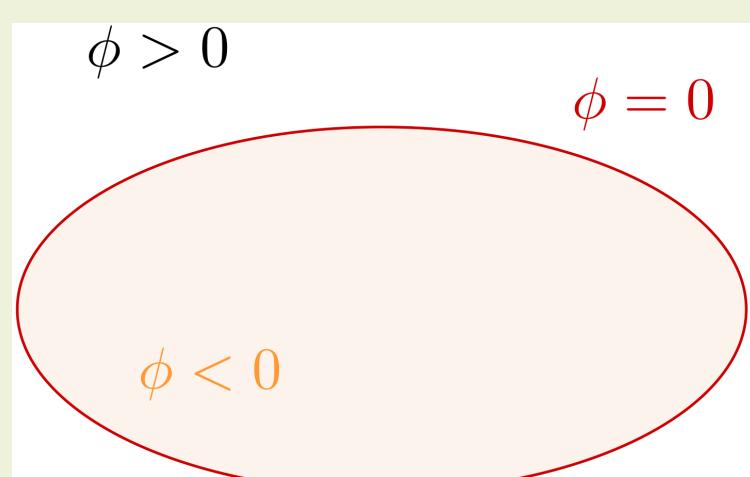
Current Objective : Develop hybrid **finite element / neural network** methods.
accurate quick + parameterized



Problem considered : Poisson problem with Dirichlet boundary conditions.

How to deal with complex geometry in PINNs ?

Approach by levelset. [SS22]



Advantages :

- Sample is easy in this case.
- Allow to impose hard BC (no BC loss) :

$$u_\theta(X) = \phi(X) w_\theta(X) + g(X)$$

with ϕ a levelset function and w_θ a NN.

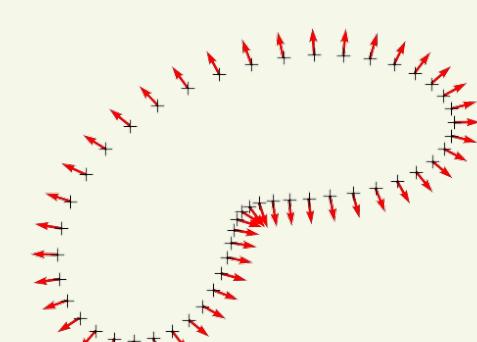
No mesh, so easy to go on complex geometry !

Levelset considered. A regularized Signed Distance Function (SDF).

Theorem 1: Eikonal equation.

If we have a boundary domain Γ , the SDF is solution to:

$$\begin{cases} \|\nabla\phi(X)\| = 1, X \in \mathcal{O} \\ \phi(X) = 0, X \in \Gamma \\ \nabla\phi(X) = n, X \in \Gamma \end{cases}$$



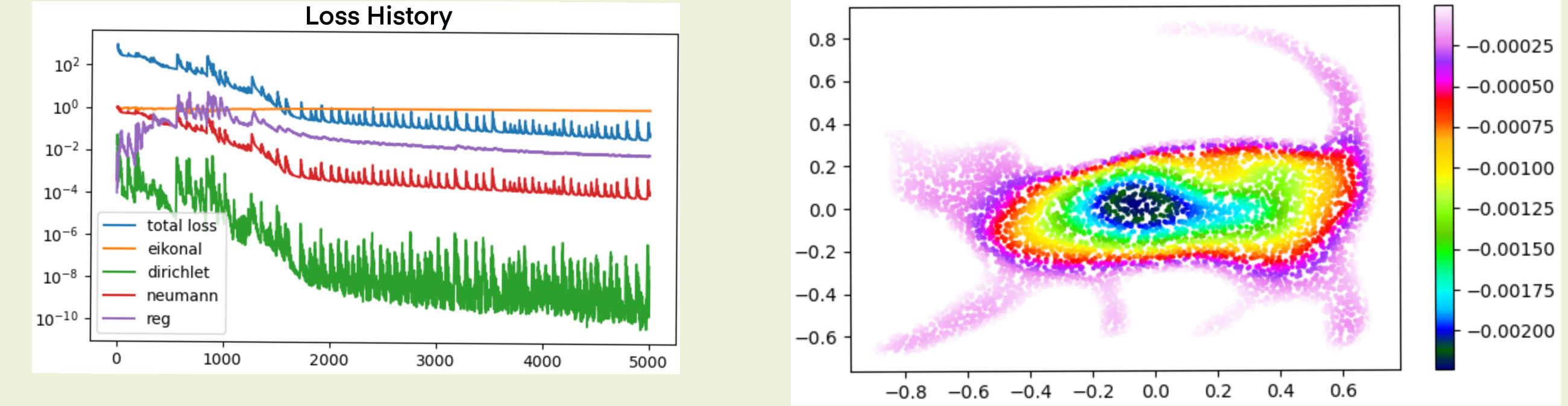
with \mathcal{O} a box which contains Ω completely and n the exterior normal to Γ .

How to do that ? with a PINNs [CD23], by adding the following regularization term

$$J_{\text{reg}} = \int_{\mathcal{O}} |\Delta\phi|^2.$$

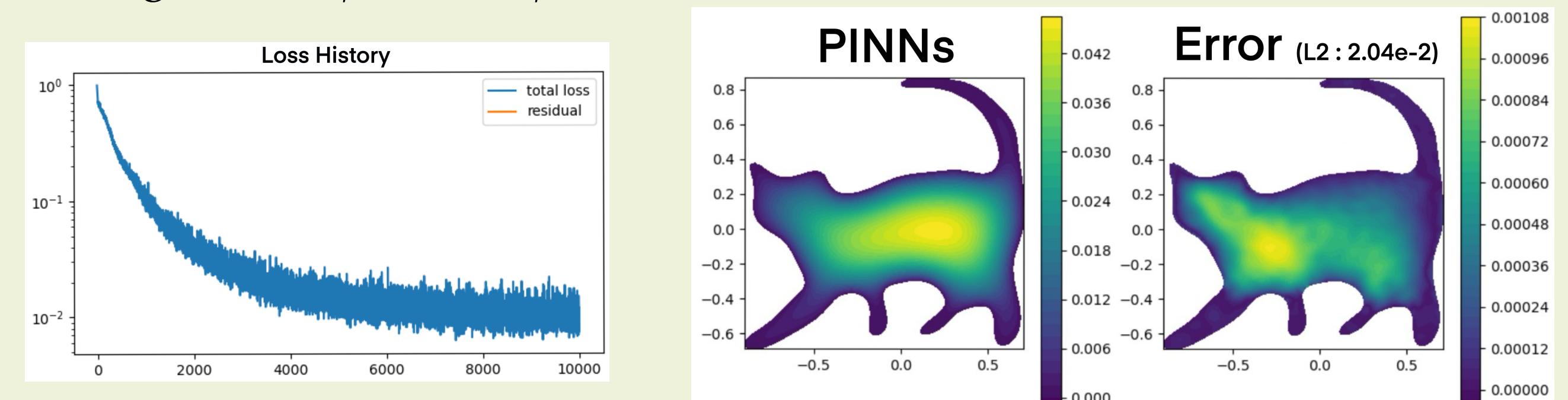
Results - Complex geometry

Levelset learning.



Poisson problem on Cat.

- Taking $f = 1$ (**non parametric**) and homogeneous Dirichlet BC ($g = 0$).
- Looking for $u_\theta = \phi w_\theta$ with ϕ the levelset learned.



How can we improve PINNs prediction ?

Using FEM-type methods

Additive approach. Considering u_θ as the prediction of our PINNs for the Poisson problem, the correction problem consists in writing the solution as

$$\tilde{u} = u_\theta + \tilde{C}$$

and searching $\tilde{C} : \Omega \rightarrow \mathbb{R}^d$ such that

$$\begin{cases} -\Delta\tilde{C} = \tilde{f}, \text{ in } \Omega, \\ \tilde{C} = 0, \text{ on } \Gamma, \end{cases} \quad (\mathcal{P}^+)$$

with $\tilde{f} = f + \Delta u_\theta$.

Poisson problem on Square.

- Considering homogeneous Dirichlet BC ($g = 0$) and $\Omega = [-0.5\pi, 0.5\pi]^2$.
- Analytical levelset function : $\phi(x, y) = (x - 0.5\pi)(x + 0.5\pi)(y - 0.5\pi)(y + 0.5\pi)$
- Analytical solution :

$$u_{ex}(x, y) = \exp\left(-\frac{(x - \mu_1)^2 + (y - \mu_2)^2}{2}\right) \sin(2x) \sin(2y)$$

with $\mu_1, \mu_2 \in [-0.5, 0.5]$ (**parametric**).

Theoretical results. Considering u_θ as the prediction of our PINNs.

Theorem 2: [Lec+ss]

We denote u the solution of the Poisson problem and u_h the discrete solution of the correction problem (\mathcal{P}^+) with V_h a \mathbb{P}_k Lagrange space. Thus

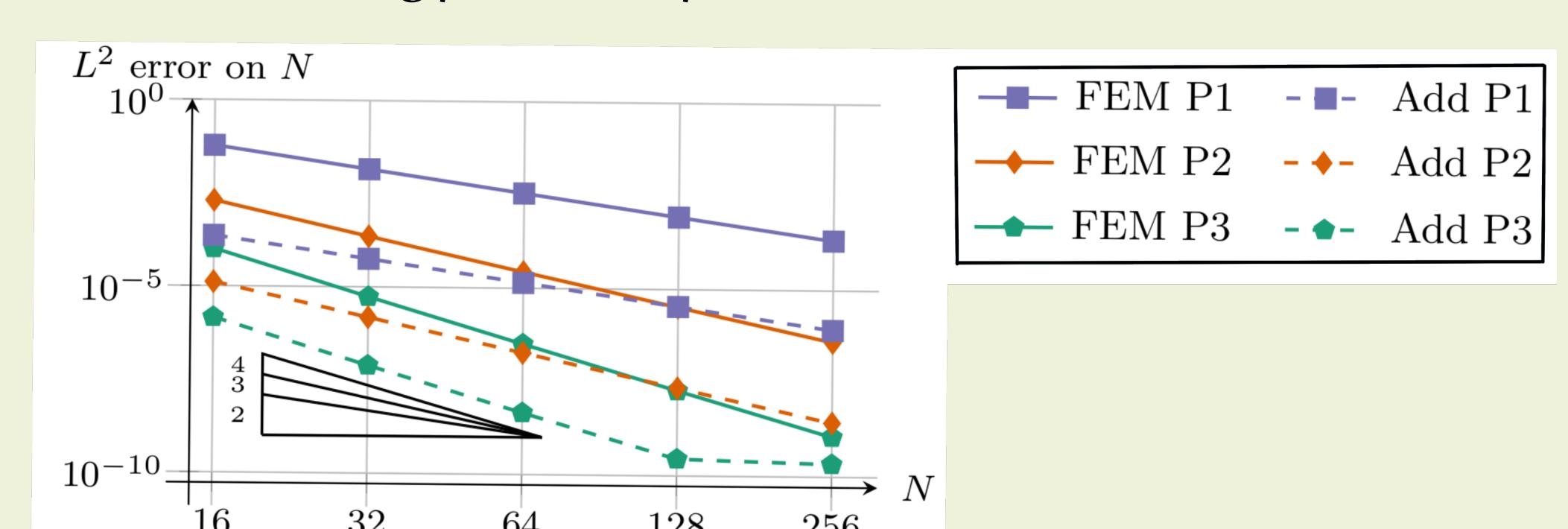
$$\|u - u_h\|_0 \lesssim \frac{\|u - u_\theta\|_{H^{k+1}}}{\|u\|_{H^{k+1}}} h^{k+1} \|u\|_{H^{k+1}}$$

C_{gain}

Remark : The constant C_{gain} shows that the closer the prior is to the solution, the lower the error constant associated with the method.

Results - Improve prediction

Theoretical results. Taking $\mu_1 = 0.05, \mu_2 = 0.22$.



Remark : We note N the number of nodes in each direction of the square.

Gains on error using additive approach.

Considering a set of $n_p = 50$ parameters : $\{(\mu_1^{(1)}, \mu_2^{(1)}), \dots, (\mu_1^{(n_p)}, \mu_2^{(n_p)})\}$.

Solution \mathbb{P}^1	Gains on PINNs				Gains on FEM					
	N	min	max	mean	std	N	min	max	mean	std
20	15.7	48.35	33.64	5.57		134.31	377.36	269.4	43.67	
40	61.47	195.75	135.41	23.21		131.18	362.09	262.12	41.67	

Time/error ratio. Training time for PINNs : $t_{\text{PINNs}} \approx 240s$.

→ At a given precision, how long does each method take to solve 1 problem ?

Precision	N		time (s)	
	FEM	Add	FEM	Add
$1e - 3$	120	8	43	0.24
$1e - 4$	373	25	423.89	1.93

t_{FEM} t_{Add}

→ How many parameters n_p to make our method faster than FEM ?

Total time of Additive approach : Total time of FEM :

$$Tot_{\text{Add}} = t_{\text{PINNs}} + n_p t_{\text{Add}}$$

$$Tot_{\text{FEM}} = n_p t_{\text{FEM}}$$

Let's suppose we want to achieve an **error of $1e - 3$** .

$$Tot_{\text{Add}} < Tot_{\text{FEM}} \Rightarrow n_p > \frac{t_{\text{PINNs}}}{t_{\text{FEM}} - t_{\text{Add}}} \approx 5.61 \Rightarrow n_p = 6$$

[CD23] Clément and Digne. "Neural skeleton: Implicit neural representation away from the surface". In: *Computers and Graphics* (2023).

[Lec+ss] Lecourtier et al. "Enhanced finite element methods using neural networks". In: (in progress).

[RPK19] Raissi, Perdikaris, and Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations". In: *Journal of Computational Physics* (2019).

[SS22] Sukumar and Srivastava. "Exact imposition of boundary conditions with distance functions in physics-informed deep neural networks". In: *Computer Methods in Applied Mechanics and Engineering* (2022).