

# Enriching continuous Lagrange finite element approximation spaces using neural networks

Michel Duprez<sup>1</sup>, Emmanuel Franck<sup>2</sup>, Frédérique Lecourtier<sup>1</sup> and Vanessa Lleras<sup>3</sup>

<sup>1</sup>Project-Team MIMESIS, Inria, Strasbourg, France

<sup>2</sup>Project-Team MACARON, Inria, Strasbourg, France

<sup>3</sup>IMAG, University of Montpellier, Montpellier, France

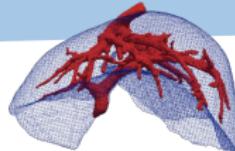
## Joint work with:

H. Barucq, F. Faucher, N. Victorion and V. Michel-Dansac.

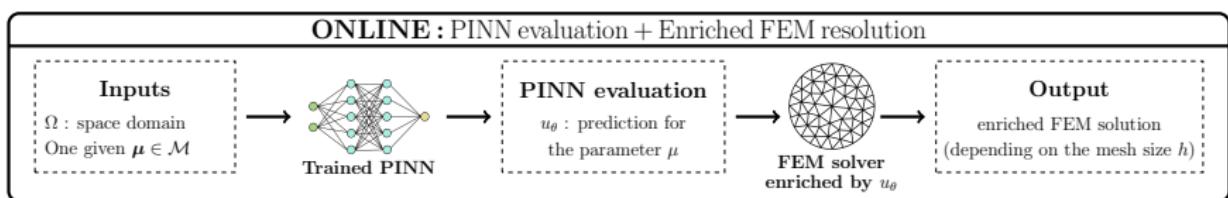
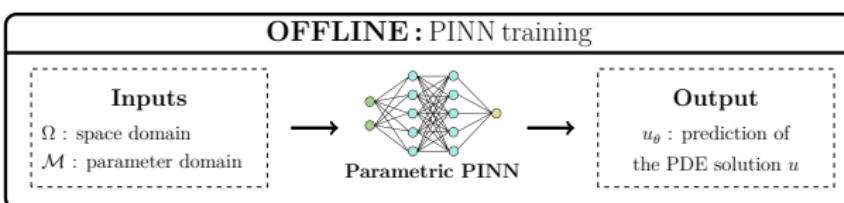


ICOSAHOM  
MONTRÉAL 2025

## Scientific context



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



**Complete ONLINE process : quick + accurate**

# Heated cavity test case

**Stationary incompressible Navier-Stokes equations (with buoyancy and gravity)<sup>1</sup> :**

We consider  $\Omega = [-1, 1]^2$  a squared domain and  $\mathbf{e}_y = (0, 1)$ .

Find the velocity  $\mathbf{u} = (u_1, u_2)$ , the pressure  $p$  and the temperature  $T$  such that

$$\begin{cases} (\mathbf{u} \cdot \nabla) \mathbf{u} + \nabla p - \nu \Delta \mathbf{u} - g(\beta T + 1) \mathbf{e}_y = 0 & \text{in } \Omega \\ \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega \\ \mathbf{u} \cdot \nabla T - k_f \Delta T = 0 & \text{in } \Omega \\ + \text{suitable BC} \end{cases} \quad (\mathcal{P})$$

with  $g = 9.81$  the gravity,  $\beta = 0.1$  the expansion coefficient,  $\nu$  the viscosity and  $k_f$  the thermal conductivity. [Coulaud et al., 2024]

---

<sup>1</sup>The approach will be shown on this example, but can be extended to other test cases.

# Heated cavity test case

**Objective:** Simulation on a range of parameters  $\mu = (\nu, k_f) \in \mathcal{M} = [0.01, 0.1]^2$ .

**Stationary incompressible Navier-Stokes equations (with buoyancy and gravity) :**

We consider  $\mathbf{x} = (x, y) \in \Omega$  and  $\mathbf{e}_y = (0, 1)$ .

Find  $\mathbf{U} = (\mathbf{u}, p, T) = (u_1, u_2, p, T)$  such that

$$\begin{cases} R_{mom}(U; \mathbf{x}, \mu) = 0 & \text{in } \Omega \\ R_{inc}(U; \mathbf{x}, \mu) = 0 & \text{in } \Omega \\ R_{ener}(U; \mathbf{x}, \mu) = 0 & \text{in } \Omega \\ + \text{suitable BC} \end{cases} \quad \begin{array}{l} \text{(momentum)} \\ \text{(incompressibility)} \\ \text{(energy)} \end{array} \quad (\mathcal{P})$$

with  $g = 9.81$  the gravity,  $\beta = 0.1$  the expansion coefficient,  $\nu$  the viscosity and  $k_f$  the thermal conductivity. [Coulaud et al., 2024]

# Heated cavity test case

**Objective:** Simulation on a range of parameters  $\mu = (\nu, k_f) \in \mathcal{M} = [0.01, 0.1]^2$ .

**Stationary incompressible Navier-Stokes equations (with buoyancy and gravity):**

We consider  $\mathbf{x} = (x, y) \in \Omega$  and  $\mathbf{e}_y = (0, 1)$ .

Find  $\mathbf{U} = (\mathbf{u}, p, T) = (u_1, u_2, p, T)$  such that

$$\begin{cases} R_{mom}(U; \mathbf{x}, \mu) = 0 & \text{in } \Omega & \text{(momentum)} \\ R_{inc}(U; \mathbf{x}, \mu) = 0 & \text{in } \Omega & \text{(incompressibility)} \\ R_{ener}(U; \mathbf{x}, \mu) = 0 & \text{in } \Omega & \text{(energy)} \end{cases} \quad (\mathcal{P})$$

with  $g = 9.81$  the gravity,  $\beta = 0.1$  the expansion coefficient,  $\nu$  the viscosity and  $k_f$  the thermal conductivity. [Coulaud et al., 2024]

## Boundary Conditions:

**No-slip BC:**  $\mathbf{u} = 0$  on  $\partial\Omega$

**Isothermal BC:**  $T = 1$  on the left wall ( $x = -1$ )  
 $T = -1$  on the right wall ( $x = 1$ )

**Adiabatic BC:**  $\frac{\partial T}{\partial n} = 0$  on the top and bottom walls ( $y = \pm 1$ , denoted by  $\Gamma_{ad}$ )

# Evaluate quality of solutions

In the following, we are interested in three parameters (rising in complexity) :

$$\boldsymbol{\mu}^{(1)} = (0.1, 0.1), \boldsymbol{\mu}^{(2)} = (0.05, 0.05) \text{ and } \boldsymbol{\mu}^{(3)} = (0.01, 0.01)$$

We evaluate the quality of solutions by comparing them to a reference solution.<sup>1</sup>

---

<sup>1</sup>Computed on an over-refined mesh ( $h = 7.10^{-3}$ ) on a  $\mathbb{P}_3^2 \times \mathbb{P}_2 \times \mathbb{P}_3$  continuous Lagrange FE space.

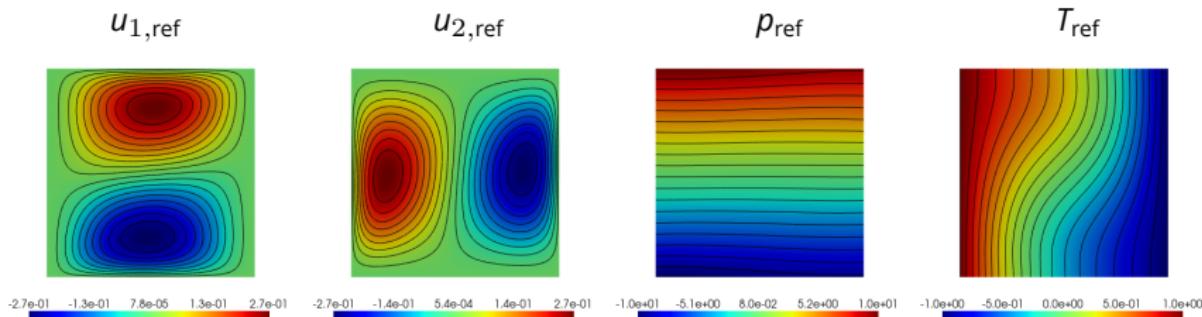
# Evaluate quality of solutions

In the following, we are interested in three parameters (rising in complexity) :

$$\boldsymbol{\mu}^{(1)} = (0.1, 0.1), \boldsymbol{\mu}^{(2)} = (0.05, 0.05) \text{ and } \boldsymbol{\mu}^{(3)} = (0.01, 0.01)$$

We evaluate the quality of solutions by comparing them to a reference solution.<sup>1</sup>

**Reference solution** - Rayleigh number :  $Ra = 1\,569.6$



<sup>1</sup>Computed on an over-refined mesh ( $h = 7.10^{-3}$ ) on a  $\mathbb{P}_3^2 \times \mathbb{P}_2 \times \mathbb{P}_3$  continuous Lagrange FE space.

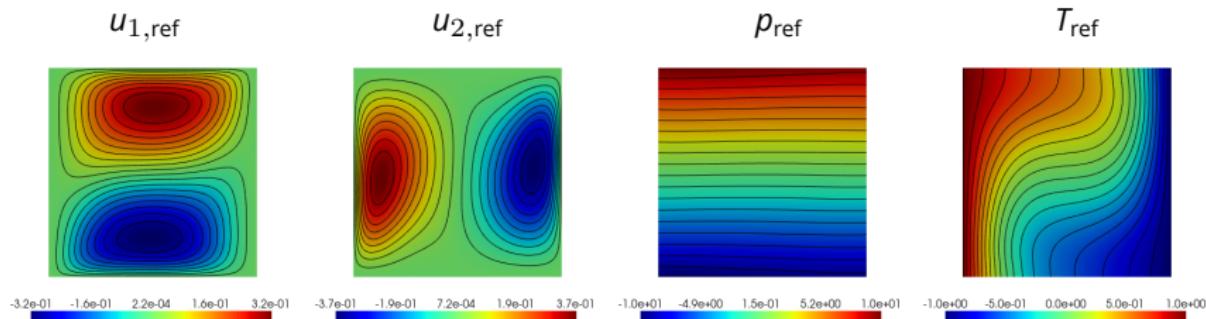
# Evaluate quality of solutions

In the following, we are interested in three parameters (rising in complexity) :

$$\boldsymbol{\mu}^{(1)} = (0.1, 0.1), \boldsymbol{\mu}^{(2)} = (0.05, 0.05) \text{ and } \boldsymbol{\mu}^{(3)} = (0.01, 0.01)$$

We evaluate the quality of solutions by comparing them to a reference solution.<sup>1</sup>

**Reference solution** - Rayleigh number :  $Ra = 6\,278.4$



<sup>1</sup>Computed on an over-refined mesh ( $h = 7.10^{-3}$ ) on a  $\mathbb{P}_3^2 \times \mathbb{P}_2 \times \mathbb{P}_3$  continuous Lagrange FE space.

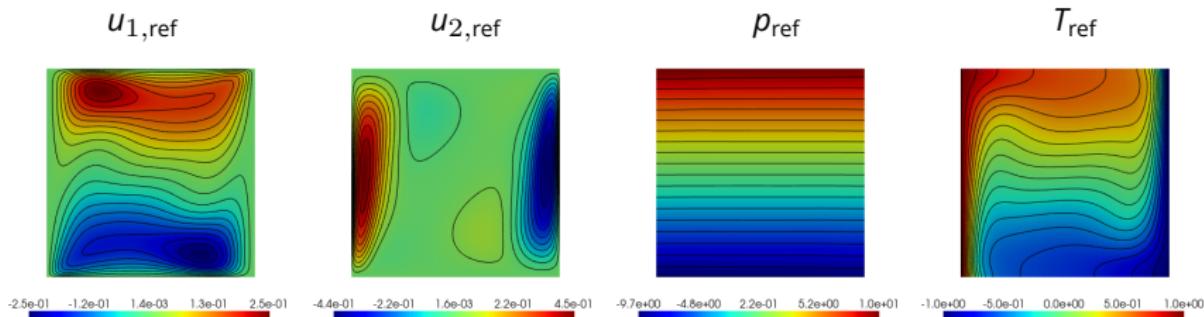
# Evaluate quality of solutions

In the following, we are interested in three parameters (rising in complexity) :

$$\boldsymbol{\mu}^{(1)} = (0.1, 0.1), \boldsymbol{\mu}^{(2)} = (0.05, 0.05) \text{ and } \boldsymbol{\mu}^{(3)} = (0.01, 0.01)$$

We evaluate the quality of solutions by comparing them to a reference solution.<sup>1</sup>

**Reference solution** - Rayleigh number :  $Ra = 156\,960$



<sup>1</sup>Computed on an over-refined mesh ( $h = 7.10^{-3}$ ) on a  $\mathbb{P}_3^2 \times \mathbb{P}_2 \times \mathbb{P}_3$  continuous Lagrange FE space.

# Parametric PINN

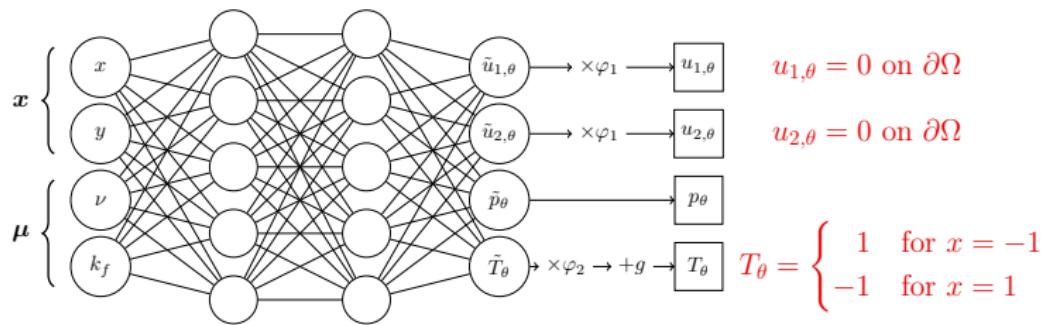
The PINN is parametrized by the  $\mu$  parameter.

# Neural Network considered

We consider a parametric NN with 4 inputs and 4 outputs, defined by

$$U_\theta(\mathbf{x}, \boldsymbol{\mu}) = (u_{1,\theta}, u_{2,\theta}, p_\theta, T_\theta)(\mathbf{x}, \boldsymbol{\mu}).$$

The Dirichlet boundary conditions are imposed on the outputs of the MLP by a **post-processing** step. [Sukumar and Srivastava, 2022]



We consider two levelsets functions  $\varphi_1$  and  $\varphi_2$ , and the linear function  $g$  defined by

$$\varphi_1(x, y) = (x - 1)(x + 1)(y - 1)(y + 1),$$

$$\varphi_2(x, y) = (x - 1)(x + 1) \quad \text{and} \quad g(x, y) = 1 - (x + 1).$$

# PINN training

**Approximate the solution of  $(\mathcal{P})$  by a PINN :** Find the optimal weights  $\theta^*$ , such that

$$\theta^* = \operatorname{argmin}_{\theta} (J_{inc}(\theta) + J_{mom}(\theta) + J_{ener}(\theta) + J_{ad}(\theta)), \quad (\mathcal{P}_\theta)$$

where the different cost functions<sup>1</sup> are defined by

adiabatic condition

$$J_{ad}(\theta) = \int_{\mathcal{M}} \int_{\Gamma_{ad}} \left| \frac{\partial T_\theta(\mathbf{x}, \mu)}{\partial n} \right|^2 d\mathbf{x} d\mu,$$

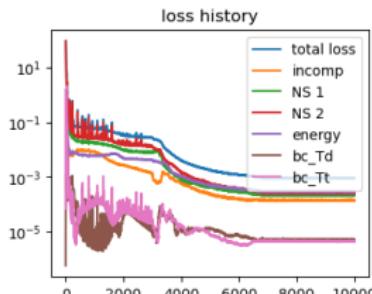
3 residual losses

$$J_{\bullet}(\theta) = \int_{\mathcal{M}} \int_{\Omega} \left| R_{\bullet}(U_\theta(\mathbf{x}, \mu); \mathbf{x}, \mu) \right|^2 d\mathbf{x} d\mu,$$

with  $U_\theta$  the parametric NN and  $\bullet$  the PDE considered (i.e. *inc*, *mom* or *ener*).

Network - MLP	
layers	40, 60, 60, 60, 40
$\sigma$	sine

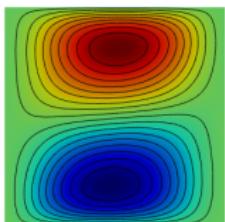
Training (ADAM / LBFGs)			
$lr$	7e-3	$N_{col}$	40000
$n_{epochs}$	10000	$N_{bc}$	30000



<sup>1</sup>Discretized by a random process using Monte-Carlo method.

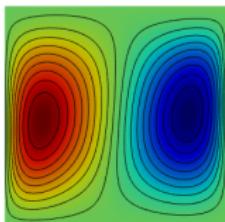
# Prediction on $\mu^{(1)} = (0.1, 0.1)$

$u_{1,\theta}$



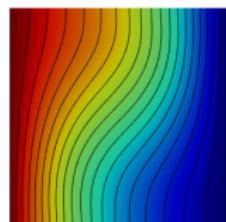
-2.7e-01 -1.4e-01 -3.0e-04 1.3e-01 2.7e-01

$u_{2,\theta}$



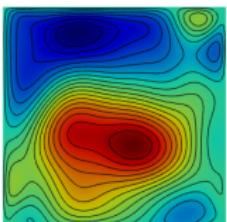
-2.8e-01 -1.4e-01 -4.2e-03 1.3e-01 2.7e-01

$T_\theta$



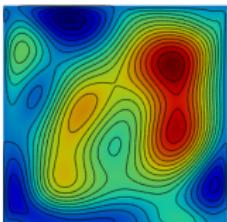
-1.0e+00 -5.0e-01 0.0e+00 5.0e-01 1.0e+00

$u_{1,\text{ref}} - u_{1,\theta}$



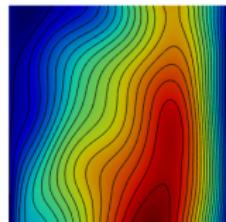
-6.1e-03 -2.3e-03 1.6e-03 5.4e-03 9.2e-03

$u_{2,\text{ref}} - u_{2,\theta}$



-3.6e-03 -2.0e-04 3.2e-03 6.6e-03 1.0e-02

$T_{\text{ref}} - T_\theta$



-1.8e-03 8.4e-03 1.9e-02 2.9e-02 3.9e-02

**Prediction :**

**Error map :**

**$L^2$  error :  
(relative)**

$2.98 \times 10^{-2}$

$3.17 \times 10^{-2}$

$3.90 \times 10^{-2}$

# Finite element method (FEM)

The  $\mu$  parameter is fixed in the FE resolution.

# Discrete weak formulation

We consider a mixed finite element space  $M_h = [V_h^0]^2 \times Q_h \times W_h$  and

$$\left. \begin{array}{lcl} \mathbf{u}_h & \in & [V_h^0]^2 \subset [H_0^1(\Omega)]^2 : \mathbb{P}_2 \\ p_h & \in & Q_h \subset L_0^2(\Omega) : \mathbb{P}_1 \\ T_h & \in & W_h \subset W : \mathbb{P}_2 \end{array} \right\} \text{(Taylor-Hood spaces)}$$

with  $W = \{w \in H^1(\Omega), w|_{x=-1} = 1, w|_{x=1} = -1\}$ .

**Weak problem :** Find  $U_h = (\mathbf{u}_h, p_h, T_h) \in M_h$  s.t.,  $\forall (\mathbf{v}_h, q_h, w_h) \in M_h^0$ ,

$$\int_{\Omega} (\mathbf{u}_h \cdot \nabla) \mathbf{u}_h \cdot \mathbf{v}_h \, d\mathbf{x} + \mu \int_{\Omega} \nabla \mathbf{u}_h : \nabla \mathbf{v}_h \, d\mathbf{x} - \int_{\Omega} p_h \nabla \cdot \mathbf{v}_h \, d\mathbf{x} - g \int_{\Omega} (1 + \beta T_h) \mathbf{e}_y \cdot \mathbf{v}_h \, d\mathbf{x} = 0, \quad (\text{momentum})$$

$$\int_{\Omega} q_h \nabla \cdot \mathbf{u}_h \, d\mathbf{x} + 10^{-4} \int_{\Omega} q_h p_h \, d\mathbf{x} = 0, \quad (\text{incompressibility + pressure penalization})$$

$$\int_{\Omega} (\mathbf{u}_h \cdot \nabla T_h) w_h \, d\mathbf{x} + \int_{\Omega} k_f \nabla T_h \cdot \nabla w_h \, d\mathbf{x} = 0, \quad (\text{energy})$$
 $(\mathcal{P}_h)$

where  $M_h^0 = [V_h^0]^2 \times Q_h \times W_h^0$  with  $W_h^0 \subset \{w \in H^1[\Omega], w|_{x=\pm 1} = 0\}$ .

# Newton method

We consider the following three parameters:

$$\boldsymbol{\mu}^{(1)} = (0.1, 0.1), \quad \boldsymbol{\mu}^{(2)} = (0.05, 0.05) \text{ and } \boldsymbol{\mu}^{(3)} = (0.01, 0.01).$$

Denoting  $N_h$  the dimension of  $M_h$ , we want to solve the non linear system:

$$F(\vec{U}_k) = 0$$

with  $F : \mathbb{R}^{N_h} \rightarrow \mathbb{R}^{N_h}$  a non linear operator and  $\vec{U}_k \in \mathbb{R}^{N_h}$  the unknown vector associated to the  $k$ -th parameter  $\boldsymbol{\mu}^{(k)}$  ( $k = 1, 2, 3$ ). Appendix 1

---

## Algorithm 1: Newton algorithm

---

**Initialization step:** set  $\vec{U}_k^{(0)} = \vec{U}_{k,0}$ ;

**for**  $n \geq 0$  **do**

Solve the linear system  $F(\vec{U}_k^{(n)}) + F'(\vec{U}_k^{(n)})\delta_k^{(n+1)} = 0$  for  $\delta_k^{(n+1)}$ ;

Update  $\vec{U}_k^{(n+1)} = \vec{U}_k^{(n)} + \delta_k^{(n+1)}$ ;

**end**

---

# Newton method

We consider the following three parameters:

$$\boldsymbol{\mu}^{(1)} = (0.1, 0.1), \quad \boldsymbol{\mu}^{(2)} = (0.05, 0.05) \text{ and } \boldsymbol{\mu}^{(3)} = (0.01, 0.01).$$

Denoting  $N_h$  the dimension of  $M_h$ , we want to solve the non linear system:

$$F(\vec{U}_k) = 0$$

with  $F : \mathbb{R}^{N_h} \rightarrow \mathbb{R}^{N_h}$  a non linear operator and  $\vec{U}_k \in \mathbb{R}^{N_h}$  the unknown vector associated to the  $k$ -th parameter  $\boldsymbol{\mu}^{(k)}$  ( $k = 1, 2, 3$ ). Appendix 1

---

## Algorithm 1: Newton algorithm

---

**Initialization step:** set  $\vec{U}_k^{(0)} = \vec{U}_{k,0}$ ;

**for**  $n \geq 0$  **do**

Solve the linear system  $F(\vec{U}_k^{(n)}) + F'(\vec{U}_k^{(n)})\delta_k^{(n+1)} = 0$  for  $\delta_k^{(n+1)}$ ;

Update  $\vec{U}_k^{(n+1)} = \vec{U}_k^{(n)} + \delta_k^{(n+1)}$ ;

**end**

---

How to initialize the Newton solver?

# 3 types of initialization

- **Natural :**
- **PINN :**
- **Continuation method :**

# 3 types of initialization

- **Natural** : Using constant or linear function.

Considering a fixed parameter with  $k \in \{1, 2, 3\}$ , we can use the following initialization:

$$\vec{U}_{k,0} = (\vec{0}, \vec{0}, \vec{0}, \vec{T}_0)$$

where for  $i = 1, \dots, \dim(W_h)$ ,

$$(\vec{T}_0)_i = g(\mathbf{x}^{(i)}) = 1 - (x^{(i)} + 1)$$

with  $\mathbf{x}^{(i)} = (x^{(i)}, y^{(i)})$  the  $i$ -th dofs coordinates of  $W_h$ .

- **PINN** :
- **Continuation method** :

# 3 types of initialization

- **Natural** : Using constant or linear function.

- **PINN** : Using PINN prediction.

(UNet : [Odote et al., 2021] ; FNO : [Aghili et al., 2025])

Considering a fixed parameter with  $k \in \{1, 2, 3\}$ , we can use the following initialization for  $i = 1, \dots, N_h$ ,

$$(\vec{U}_{k,0})_i = U_\theta(\mathbf{x}^{(i)}, \boldsymbol{\mu}^{(k)})$$

with  $\mathbf{x}^{(i)} = (x^{(i)}, y^{(i)})$  the  $i$ -th dofs coordinates of  $M_h$  and  $U_\theta$  the PINN.

- **Continuation method :**

# 3 types of initialization

- **Natural** : Using constant or linear function.
- **PINN** : Using PINN prediction.  
(UNet : [Odote et al., 2021] ; FNO : [Aghili et al., 2025])
- **Continuation method** : Using a coarse FE solution of a simpler parameter.
  - We consider a fixed parameter with  $k \in \{2, 3\}$ .
  - We consider a coarse grid ( $16 \times 16$  grid) and compute the FE solution of  $(P_h)$  for the parameter  $\mu^{(k-1)}$ .
  - We interpolate the coarse solution to the current mesh.
  - We use it as an initialization for the Newton method, i.e.

$$\vec{U}_{k,0} = (\vec{u}_{k-1}, \vec{v}_{k-1}, \vec{p}_{k-1}, \vec{T}_{k-1})$$

where  $\vec{u}_{k-1}$ ,  $\vec{v}_{k-1}$ ,  $\vec{p}_{k-1}$  and  $\vec{T}_{k-1}$  are the FE solutions for the parameter  $\mu^{(k-1)}$ .

# Enriched finite element method using PINN

Very simple linear test case

The heated cavity test case considered

# Enriched finite element method using PINN

Very simple linear test case

The heated cavity test case considered

# What is the purpose of enrichment?

**Poisson problem** (with Dirichlet BC): Find  $u : \Omega \rightarrow \mathbb{R}$  such that

$$\begin{cases} -\Delta u = f, & \text{in } \Omega, \\ u = 0, & \text{on } \partial\Omega. \end{cases}$$

**Variational Problem**: We consider  $V_h^0$  a  $\mathbb{P}_k$  continuous Lagrange FE space ( $k \geq 1$ ).

$$\text{Find } u_h \in V_h^0 \text{ such that, } \forall v_h \in V_h^0, a(u_h, v_h) = l(v_h), \quad (\mathcal{P}_h)$$

with  $h$  the characteristic mesh size,  $a$  and  $l$  the associated bilinear and linear forms.

# What is the purpose of enrichment?

**Poisson problem** (with Dirichlet BC): Find  $u : \Omega \rightarrow \mathbb{R}$  such that

$$\begin{cases} -\Delta u = f, & \text{in } \Omega, \\ u = 0, & \text{on } \partial\Omega. \end{cases}$$

**Variational Problem**: We consider  $V_h^0$  a  $\mathbb{P}_k$  continuous Lagrange FE space ( $k \geq 1$ ).

$$\text{Find } u_h \in V_h^0 \text{ such that, } \forall v_h \in V_h^0, a(u_h, v_h) = l(v_h), \quad (\mathcal{P}_h)$$

with  $h$  the characteristic mesh size,  $a$  and  $l$  the associated bilinear and linear forms.

**Modified variational Problem**: Let  $u_\theta$  be a PINN prediction.

$$\text{Find } C_{h,u}^+ \in V_h^0 \text{ such that, } \forall v_h \in V_h^0, a(C_{h,u}^+, v_h) = l(v_h) - a(u_\theta, v_h), \quad (\mathcal{P}_h^+)$$

with the **enriched trial space**  $V_h^+$  defined by

$$V_h^+ = \left\{ u_h^+ = u_\theta + C_{h,u}^+, \quad C_{h,u}^+ \in V_h^0 \right\}.$$

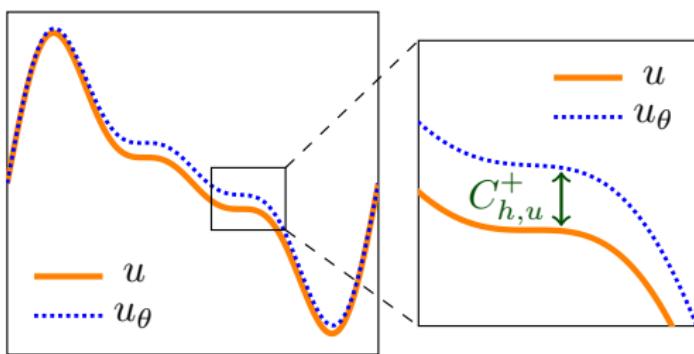
# What is the purpose of enrichment?

**Modified variational Problem :** Let  $u_\theta$  be a PINN prediction.

$$\text{Find } C_{h,u}^+ \in V_h^0 \text{ such that, } \forall v_h \in V_h^0, a(C_{h,u}^+, v_h) = I(v_h) - a(u_\theta, v_h), \quad (\mathcal{P}_h^+)$$

with the enriched trial space  $V_h^+$  defined by

$$V_h^+ = \left\{ u_h^+ = u_\theta + C_{h,u}^+, \quad C_{h,u}^+ \in V_h^0 \right\}.$$



We hope that the modified problem will give the same results as the standard one on coarser meshes.

# Convergence analysis

Theorem 1: Convergence analysis of the standard FEM [Ern and Guermond, 2004]

We denote  $u_h \in V_h$  the solution of  $(\mathcal{P}_h)$  with  $V_h$  the standard trial space. Then,

$$|u - u_h|_{H^1} \leq C_{H^1} h^k |u|_{H^{k+1}},$$

$$\|u - u_h\|_{L^2} \leq C_{L^2} h^{k+1} |u|_{H^{k+1}}.$$

Theorem 2: Convergence analysis of the enriched FEM [F. Lecourtier et al., 2025]

We denote  $u_h^+ \in V_h^+$  the solution of  $(\mathcal{P}_h^+)$  with  $V_h^+$  the enriched trial space. Then,

$$|u - u_h^+|_{H^1} \leq \boxed{\frac{|u - u_\theta|_{H^{k+1}}}{|u|_{H^{k+1}}}} (C_{H^1} h^k |u|_{H^{k+1}}),$$

$$\|u - u_h^+\|_{L^2} \leq \boxed{\frac{|u - u_\theta|_{H^{k+1}}}{|u|_{H^{k+1}}}} (C_{L^2} h^{k+1} |u|_{H^{k+1}}).$$

Gains of the additive approach.

LECOURTIER Frédérique

# Enriched finite element method using PINN

Very simple linear test case

The heated cavity test case considered

# Enriched space using PINN

Considering the PINN prior  $U_\theta = (\mathbf{u}_\theta, p_\theta, T_\theta)$ , we define the **mixed finite element space additively enriched** by the PINN as follows:

$$M_h^+ = \{ U_h^+ = U_\theta + C_h^+, \quad C_h^+ \in M_h^0 \}$$

with  $M_h^0 = [V_h^0]^2 \times Q_h \times W_h^0$ ,  $U_h^+ = (\mathbf{u}_h^+, p_h^+, T_h^+) \in M_h^+$  and  $C_h^+ = (C_{h,\mathbf{u}}^+, C_{h,p}^+, C_{h,T}^+)$ .

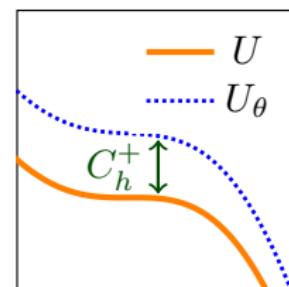
We can then define the three finite element subspaces of  $M_h^+$  as follows:

$$V_h^+ = \{ \mathbf{u}_h^+ = \mathbf{u}_\theta + C_{h,\mathbf{u}}^+, \quad C_{h,\mathbf{u}}^+ \in [V_h^0]^2 \},$$

$$Q_h^+ = \{ p_h^+ = p_\theta + C_{h,p}^+, \quad C_{h,p}^+ \in Q_h \},$$

$$W_h^+ = \{ T_h^+ = T_\theta + C_{h,T}^+, \quad C_{h,T}^+ \in W_h^0 \},$$

where  $C_{h,\mathbf{u}}^+$ ,  $C_{h,p}^+$  and  $C_{h,T}^+$  becomes the unknowns.



# Weak formulation - Additive approach

**Weak problem :** Find  $C_h^+ = (C_{h,u}^+, C_{h,n}^+, C_{h,T}^+) \in M_h^0$  s.t.,  $\forall (\mathbf{v}_h, q_h, w_h) \in M_h^0$ ,

$$\begin{aligned}
& \int_{\Omega} [(\mathbf{u}_\theta \cdot \nabla) \mathbf{u}_\theta + (\mathbf{u}_\theta \cdot \nabla) \mathbf{c}_{h,u}^+ + (\mathbf{c}_{h,u}^+ \cdot \nabla) \mathbf{u}_\theta + (\mathbf{c}_{h,u}^+ \cdot \nabla) \mathbf{c}_{h,u}^+] \cdot \mathbf{v}_h \, d\mathbf{x} \\
& + \mu \left( \int_{\Omega} \nabla \mathbf{u}_\theta : \nabla \mathbf{v}_h \, d\mathbf{x} + \int_{\Omega} \nabla \mathbf{c}_{h,u}^+ : \nabla \mathbf{v}_h \, d\mathbf{x} \right) + \left( \int_{\Omega} \nabla p_\theta \cdot \mathbf{v}_h \, d\mathbf{x} - \int_{\Omega} \mathbf{c}_{h,p}^+ \nabla \cdot \mathbf{v}_h \, d\mathbf{x} \right) \\
& - g \int_{\Omega} (1 + \beta(T_\theta + \mathbf{c}_{h,T}^+)) \mathbf{e}_y \cdot \mathbf{v}_h \, d\mathbf{x} = 0, \text{ (momentum)} \\
& \int_{\Omega} q_h [\nabla \cdot \mathbf{u}_\theta + \nabla \cdot \mathbf{c}_{h,u}^+] \, d\mathbf{x} + 10^{-4} \int_{\Omega} q_h (p_\theta + \mathbf{c}_{h,p}^+) \, d\mathbf{x} = 0, \text{ (incompressibility + penal)} \tag{\mathcal{P}_h^+} \\
& \int_{\Omega} [\mathbf{u}_\theta \cdot \nabla T_\theta + \mathbf{u}_\theta \cdot \nabla \mathbf{c}_{h,T}^+ + \mathbf{c}_{h,u}^+ \cdot \nabla T_\theta + \mathbf{c}_{h,u}^+ \cdot \nabla \mathbf{c}_{h,T}^+] w_h \, d\mathbf{x} \\
& + k_f \left( \int_{\Omega} \nabla T_\theta \cdot \nabla w_h \, d\mathbf{x} + \int_{\Omega} \nabla \mathbf{c}_{h,T}^+ \cdot \nabla w_h \, d\mathbf{x} w_h \, ds \right) = 0, \text{ (energy)}
\end{aligned}$$

with  $\mathcal{U}_\theta = (\mathbf{u}_\theta, p_\theta, T_\theta)$  the PINN prior and some modified boundary conditions.

# Newton method - Additive approach

We want to solve the non linear system:

$$F_\theta(\vec{C}) = 0$$

with  $F_\theta : \mathbb{R}^{N_h} \rightarrow \mathbb{R}^{N_h}$  the non linear operator associated to the weak problem  $(\mathcal{P}_h^+)$  and  $\vec{C} \in \mathbb{R}^{N_h}$  the **correction vector (unknown)**.

---

**Algorithm 2:** Newton algorithm [Aghili et al., 2025]

---

**Initialization step:** set  $\vec{C}^{(0)} = 0$ ;

**for**  $n \geq 0$  **do**

Solve the linear system  $F_\theta(\vec{C}^{(n)}) + F'_\theta(\vec{C}^{(n)})\delta^{(n+1)} = 0$  for  $\delta^{(n+1)}$ ;

Update  $\vec{C}^{(n+1)} = \vec{C}^{(n)} + \delta^{(n+1)}$ ;

**end**

---

**Advantage compared to PINN initialization<sup>1</sup>:**

$u_\theta$  is not required to live in the same discrete space as  $C_h^+$ .

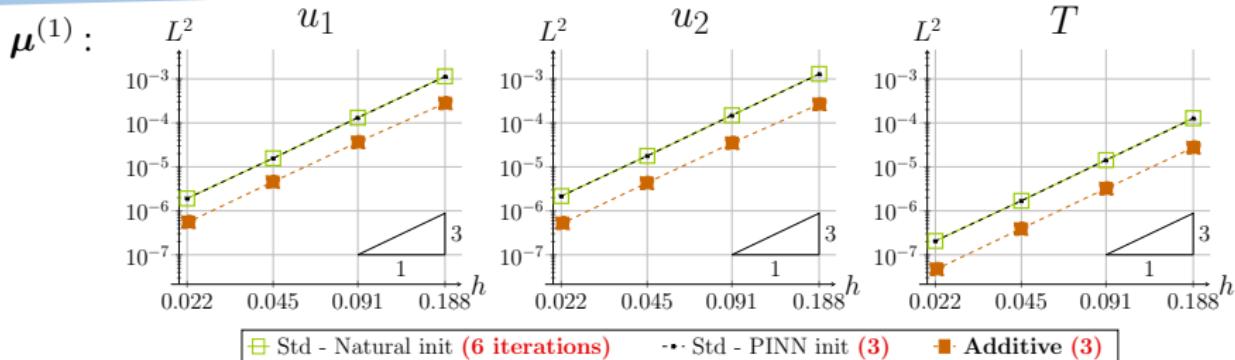
---

<sup>1</sup>Taking  $U_\theta$  and  $C_h^+$  in the same space, additive approach is exactly the same as the PINN initialization.

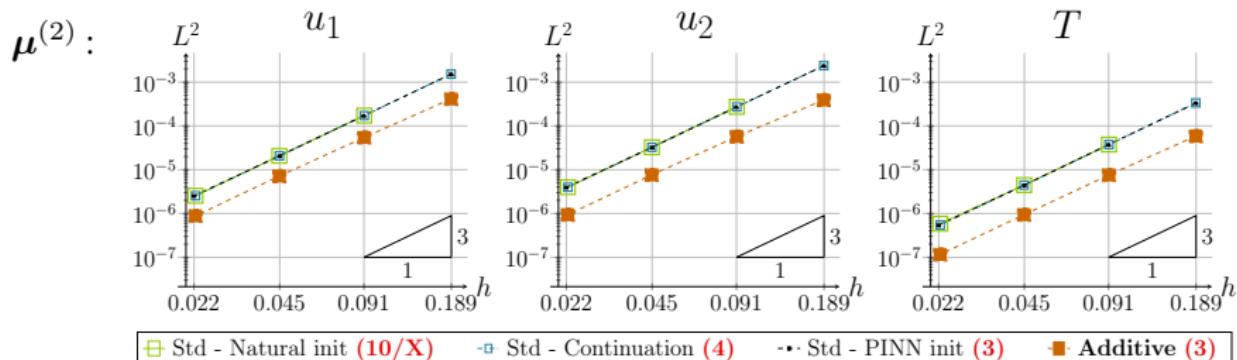
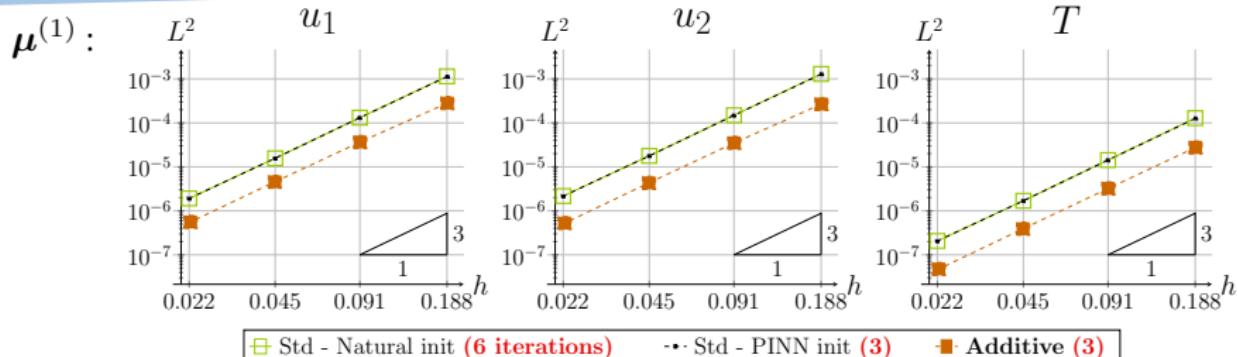
# Numerical results

- Results obtained with a laptop GPU.
- The newton solver is the same for all methods ( $\text{rtol} = 10^{-10}$ ,  $\text{atol} = 10^{-10}$ ,  $\text{max\_it} = 30$ ).
- Additive approach : we consider  $u_\theta$  in a  $\mathbb{P}_3^2 \times \mathbb{P}_2 \times \mathbb{P}_3$  continuous Lagrange FE space (defined on the current mesh).

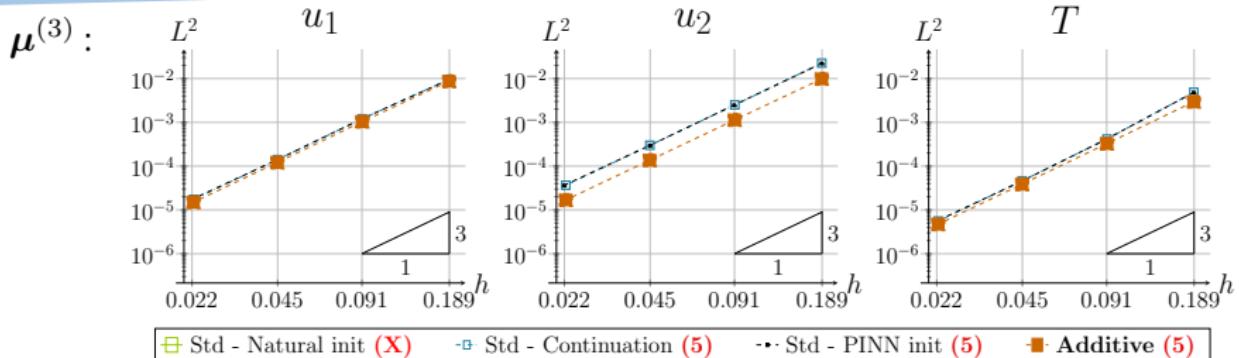
# Error estimates I



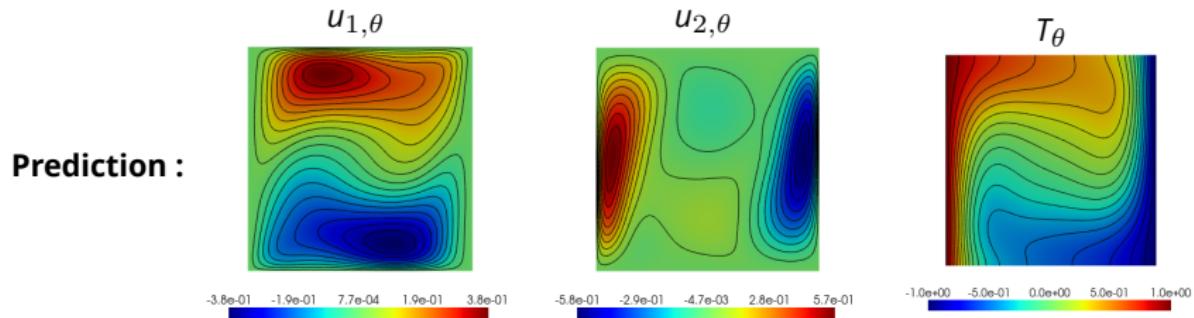
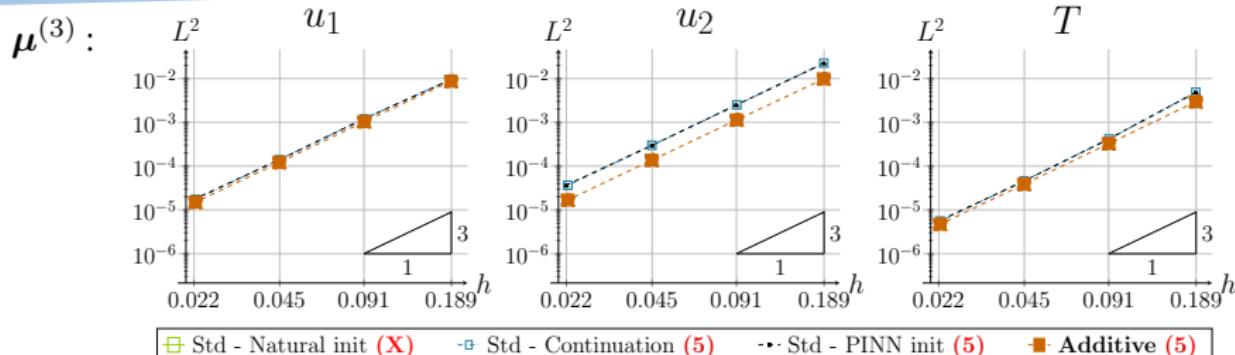
# Error estimates I



# Error estimates II



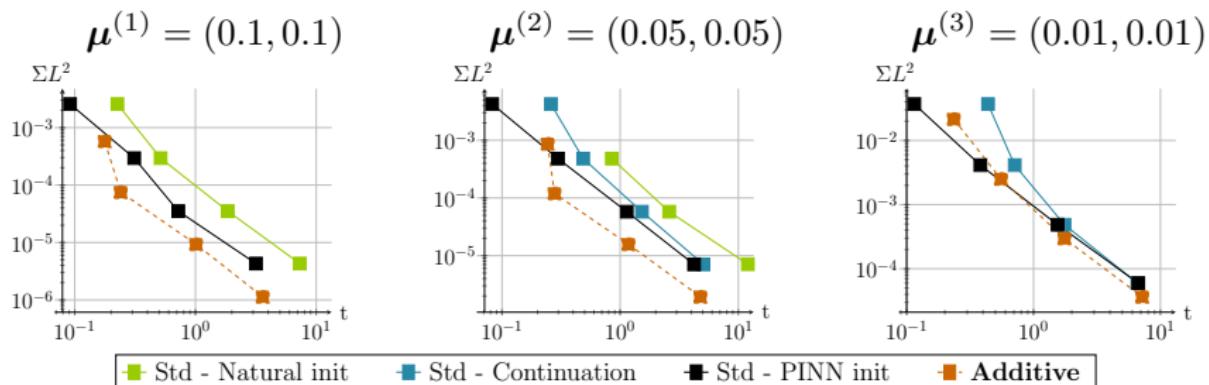
# Error estimates II



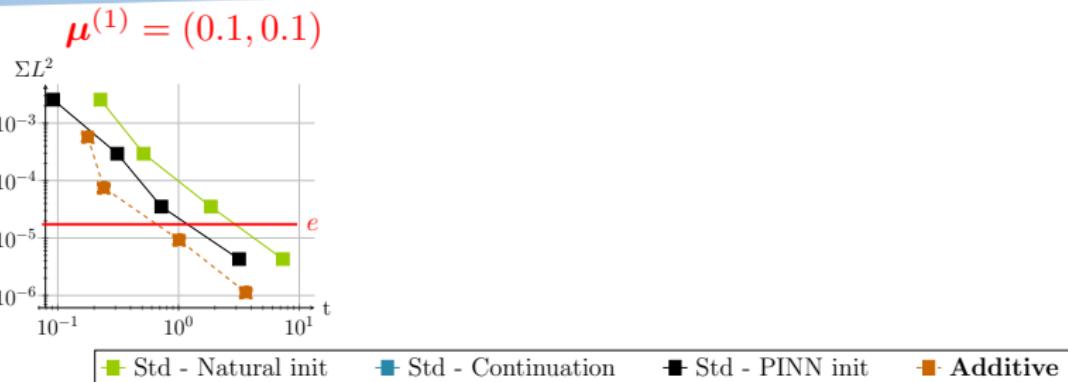
$L^2$  error :  $5.75 \times 10^{-1}$        $4.89 \times 10^{-1}$        $2.57 \times 10^{-1}$   
 (relative)

# Numerical costs

Defining the global error as the sum of the  $L^2$  relatives errors on  $\mathbf{u}$  and  $T$ .



# Numerical costs

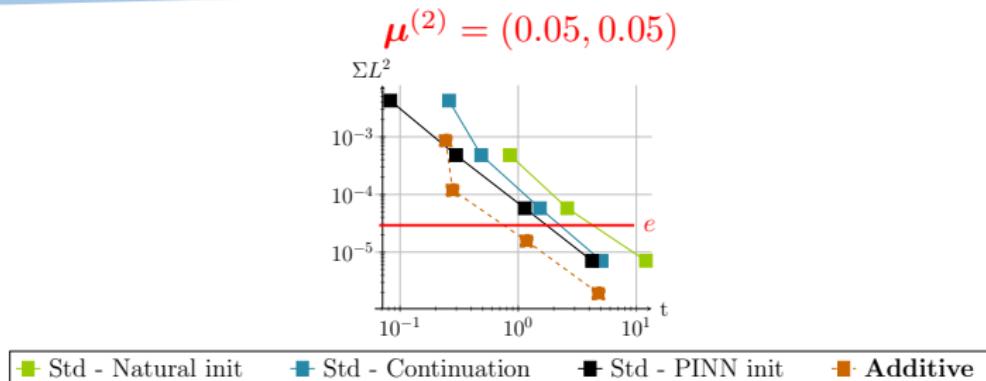


$N_{\text{dofs}}$  and execution time required to reach the same global  $L^2$  relative error  $e$ :

Std vs Add	Number of DoFs		Execution times		
	Std	Add	(nat)	(PINN)	Add
$1 \cdot 10^{-3}$	6,031	<b>2,044</b>	0.32	0.16	<b>0.16</b>
$1 \cdot 10^{-4}$	26,959	<b>10,588</b>	0.99	0.48	<b>0.23</b>
$1 \cdot 10^{-5}$	121,156	<b>49,231</b>	4.21	1.75	<b>0.96</b>

→ ÷ 2.5
→ ÷ 2
→ ÷ 4

# Numerical costs



$N_{\text{dofs}}$  and execution time required to reach the same global  $L^2$  relative error  $e$ :

<b>Std vs Add</b>	<b>Number of DoFs</b>		<b>Execution times</b>			
	<b>Std</b>	<b>Add</b>	<b>(nat)</b>	<b>(cont)</b>	<b>(PINN)</b>	<b>Add</b>
$1 \cdot 10^{-3}$	7,828	2,748	0.58	0.39	0.19	0.24
$1 \cdot 10^{-4}$	35,884	14,623	1.95	1.14	0.8	0.32
$1 \cdot 10^{-5}$	167,583	70,303	9.39	4.16	3.4	1.59

$\div 2.4$

$\div 2$

$\div 6$

$\div 2.5$

# Numerical costs

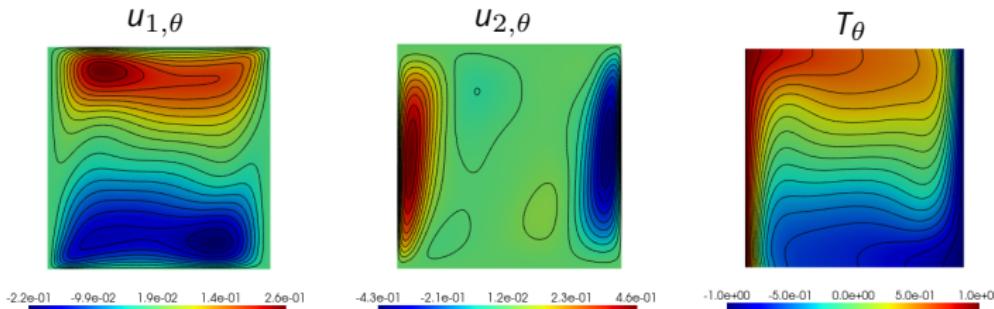


$N_{\text{dofs}}$  and execution time required to reach the same global  $L^2$  relative error  $e$ :

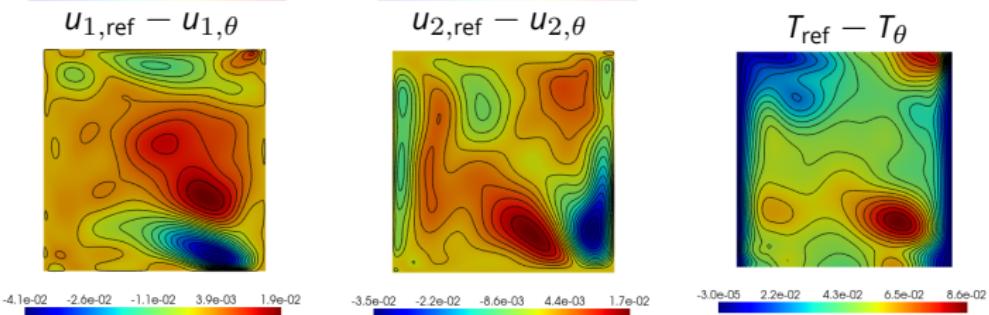
Std vs Add	Number of DoFs		Execution times			
	Std	Add	(nat)	(cont)	(PINN)	Add
$1 \cdot 10^{-3}$	33,204	23,524	X	1.29	0.96	0.91
$1 \cdot 10^{-4}$	150,339	108,931	X	4.76	4.67	3.65
$1 \cdot 10^{-5}$	690,924	502,156	X	20.34	23.3	17.23

# Non parametric PINN<sup>1</sup> for $\mu^{(3)}$

**Prediction :**



**Error map :**



**$L^2$  error :**

**(relative)**

$$7.60 \times 10^{-2}$$

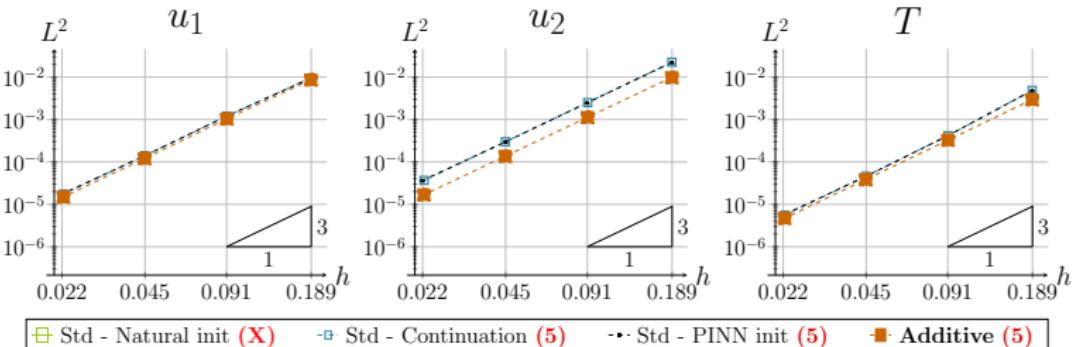
$$5.38 \times 10^{-2}$$

$$9.63 \times 10^{-2}$$

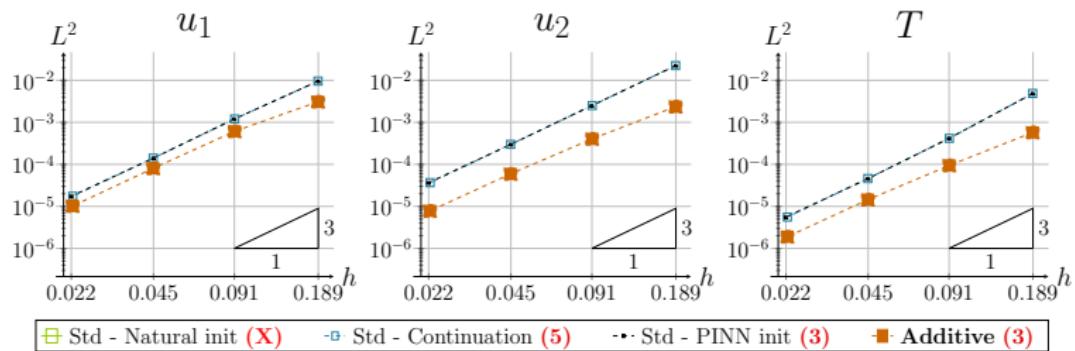
<sup>1</sup>We consider exactly the same architecture, but this time we train the PINN non-parametrically.

# Error estimates on $\mu^{(3)}$

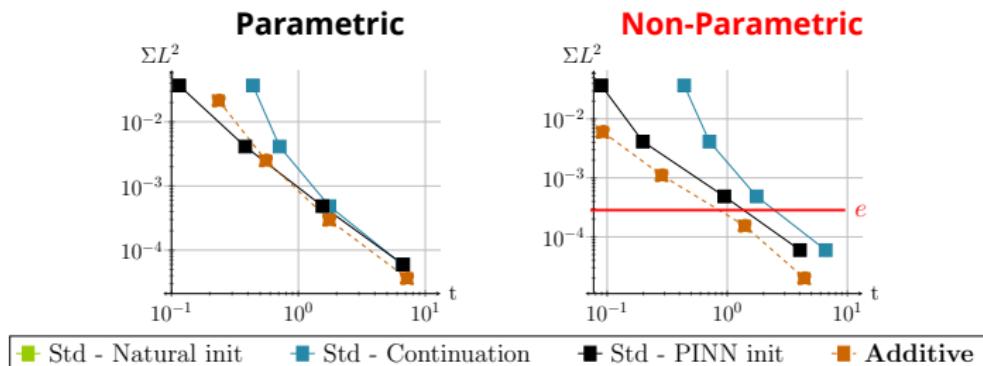
**Parametric**



**Non-Parametric**



# Numerical costs on $\mu^{(3)}$



$N_{\text{dofs}}$  and execution time required to reach the same global  $L^2$  relative error  $e$ :

$e$	Number of DoFs			Execution times			
	(PINN)	Add	Add+	(PINN)	(PINN)+	Add	Add+
$1 \cdot 10^{-3}$	33,204	23,524	13,764	0.96	0.56	0.91	0.31
$1 \cdot 10^{-4}$	150,339	108,931	70,303	4.67	2.82	3.65	1.78
$1 \cdot 10^{-5}$	690,924	502,156	339,231	23.3	13.84	17.23	6.42

÷ 2

÷ 3

# Conclusion

- The enriched approach provides the same results as the standard FEM method, but with **coarser meshes**.  
⇒ Reduction of the computational cost : DoFs, iterations, execution times.
- Theory on linear problems shows that it's the **derivatives** of the prior that are the most crucial. [Appendix 4](#)  
⇒ PINNs are good candidates for the enriched approach.
- The gains obtained on linear problems were much higher. [Appendix 3](#)  
⇒ **Improved training** of parametric PINN (or Neural Operators).

Preprint (linear)



# References

- J. Aghili, E. Franck, R. Hild, V. Michel-Dansac, and V. Vigon. Accelerating the convergence of newton's method for nonlinear elliptic pdes using fourier neural operators. 2025.
- Guillaume Coulaud, Maxime Le, and Régis Duvigneau. Investigations on Physics-Informed Neural Networks for Aerodynamics, 2024.
- A. Ern and J.-L. Guermond. *Theory and Practice of Finite Elements*. 2004.
- A. Odot, R. Haferssas, and S. Cotin. Deepphysics: a physics aware deep learning framework for real-time simulation, 2021.
- N. Sukumar and A. Srivastava. Exact imposition of boundary conditions with distance functions in physics-informed deep neural networks. 2022.
- F. Lecourtier**, H. Barucq, M. Duprez, F. Faucher, E. Franck, V. Lleras, V. Michel-Dansac, and N. Victorion. Enriching continuous lagrange finite element approximation spaces using neural networks, 2025.

Preprint (linear)



# Appendix 1 : FEM

# A1 – Construction of the unknown vector

Considering  $(\phi_i)_{i=1}^{N_u}$ ,  $(\psi_j)_{j=1}^{N_p}$  and  $(\eta_k)_{k=1}^{N_T}$  the basis functions of the finite element spaces  $V_h^0$ ,  $Q_h$  and  $W_h$  respectively, we can write the discrete solutions as:

$$\mathbf{u}_h(\mathbf{x}) = \sum_{i=1}^{N_u} \begin{pmatrix} u_i \\ v_i \end{pmatrix} \phi_i(\mathbf{x}), \quad p_h(\mathbf{x}) = \sum_{j=1}^{N_p} p_j \psi_j(\mathbf{x}) \quad \text{and} \quad T_h(\mathbf{x}) = \sum_{k=1}^{N_T} T_k \eta_k(\mathbf{x}),$$

with the unknown vectors for velocity, pressure and temperature defined by

$$\vec{u} = (u_i)_{i=1}^{N_u} \in \mathbb{R}^{N_u}, \quad \vec{v} = (v_i)_{i=1}^{N_u} \in \mathbb{R}^{N_u},$$

$$\vec{p} = (p_j)_{j=1}^{N_p} \in \mathbb{R}^{N_p} \quad \text{and} \quad \vec{T} = (T_k)_{k=1}^{N_T} \in \mathbb{R}^{N_T}.$$

Considering  $N_h = 2N_u + N_p + N_T$ , we can define the global vector of unknowns as:

$$\vec{U} = (\vec{u}, \vec{v}, \vec{p}, \vec{T}) \in \mathbb{R}^{N_h}.$$

and  $F : \mathbb{R}^{N_h} \rightarrow \mathbb{R}^{N_h}$  the nonlinear operator associated to the weak formulation  $(\mathcal{P}_h)$ .

# Appendix 2 : PINN Initialization / Additive approach

## A2 – Comparison of the 2 approaches

Taking  $U_\theta$  and  $C_h^+$  in the same space, we have :

$$F_\theta(\vec{C}) = F(\vec{U}_\theta + \vec{C}),$$

with  $\vec{C}$  the correction vector and  $\vec{U}_\theta$  the PINN vector (PINN evaluation at the dofs), both of size  $N_h$ .

The first iteration of the additive approach :

$$F_\theta(\vec{C}^{(0)}) + F'_\theta(\vec{C}^{(0)})\delta^{(1)} = 0$$

becomes (as  $C^{(0)} = 0$ ) :

$$F(\vec{U}_\theta) + F'(\vec{U}_\theta)\delta^{(1)} = 0,$$

which is equivalent as the standard method with the PINN initialization.

# Appendix 3 : Results - Linear problem

# A3 – Problem considered

**Problem statement:** Consider the Poisson problem with Dirichlet BC:

$$\begin{cases} -\Delta u = f, & \text{in } \Omega \times \mathcal{M}, \\ u = 0, & \text{on } \partial\Omega \times \mathcal{M}, \end{cases}$$

with  $\Omega = [-0.5\pi, 0.5\pi]^2$  and  $\mathcal{M} = [-0.5, 0.5]^2$  ( $p = 2$  parameters).

**Analytical solution :**

$$u(\mathbf{x}, \boldsymbol{\mu}) = \exp\left(-\frac{(x - \mu_1)^2 + (y - \mu_2)^2}{2}\right) \sin(2x) \sin(2y).$$

**PINN training:** MLP of 5 layers; LBFGs optimizer (5000 epochs).

Imposing the Dirichlet BC exactly in the PINN with the levelset  $\varphi$  defined by

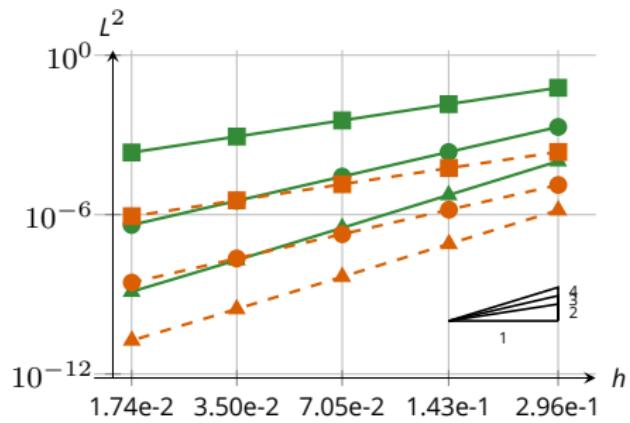
$$\varphi(\mathbf{x}) = (x + 0.5\pi)(x - 0.5\pi)(y + 0.5\pi)(y - 0.5\pi).$$

Training time : less than 1 hour on a laptop GPU.

# A3 – Numerical results

**Error estimates :** 1 set of parameters.

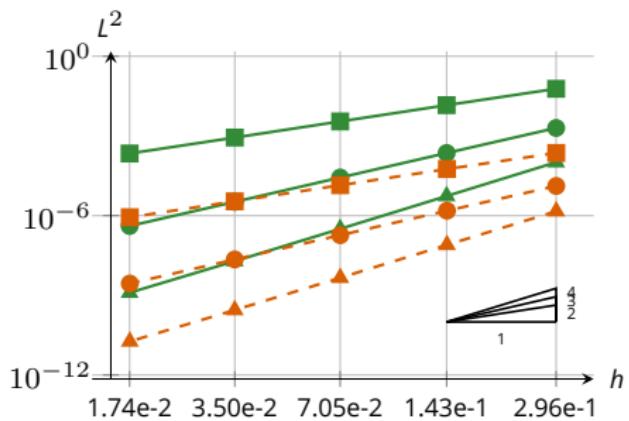
$$\mu^{(1)} = (0.05, 0.22)$$



# A3 – Numerical results

**Error estimates :** 1 set of parameters.

$$\mu^{(1)} = (0.05, 0.22)$$



**Gains achieved :**  $n_p = 50$  sets of parameters.

$$\mathcal{S} = \left\{ \mu^{(1)}, \dots, \mu^{(n_p)} \right\}$$

---

**Gains in  $L^2$  rel error  
of our method w.r.t. FEM**

---

k	min	max	mean
1	134.32	377.36	269.39
2	67.02	164.65	134.85
3	39.52	72.65	61.55

---

$N = 20$

$$\text{Gain} : \|u - u_h\|_{L^2} / \|u - u_h^+\|_{L^2}$$

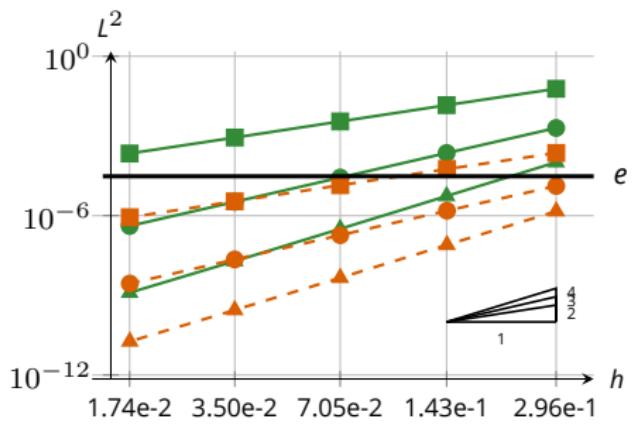
Cartesian mesh :  $N^2$  nodes.

# A3 – Numerical results

**Error estimates :** 1 set of parameters.

**$N_{\text{dofs}}$  required to reach the same error  $e$ :**

$$\mu^{(1)} = (0.05, 0.22)$$



<b>k</b>	<b>e</b>	<b><math>N_{\text{dofs}}</math></b>	
		<b>FEM</b>	<b>Add</b>
1	$1 \cdot 10^{-3}$	14,161	64
	$1 \cdot 10^{-4}$	143,641	576
2	$1 \cdot 10^{-4}$	6,889	225
	$1 \cdot 10^{-5}$	31,329	1,089
3	$1 \cdot 10^{-5}$	6,724	784
	$1 \cdot 10^{-6}$	20,164	2,704

# Appendix 4 : Data-driven vs Physics-Informed training

# A4 – Problem considered

**Problem statement:** Consider the Poisson problem in 1D with Dirichlet BC:

$$\begin{cases} -\partial_{xx}u = f, & \text{in } \Omega \times \mathcal{M}, \\ u = 0, & \text{on } \partial\Omega \times \mathcal{M}, \end{cases}$$

with  $\Omega = [0, 1]^2$  and  $\mathcal{M} = [0, 1]^3$  ( $p = 3$  parameters).

**Analytical solution :**  $u(x; \mu) = \mu_1 \sin(2\pi x) + \mu_2 \sin(4\pi x) + \mu_3 \sin(6\pi x)$ .

**Construction of two priors:** MLP of 6 layers; Adam optimizer (10000 epochs).

Imposing the Dirichlet BC exactly in the PINN with  $\varphi(x) = x(x - 1)$ .

- **Physics-informed training:**  $N_{\text{col}} = 5000$  collocation points.

$$J_r(\theta) \simeq \frac{1}{N_{\text{col}}} \sum_{i=1}^{N_{\text{col}}} \left| \partial_{xx}u_\theta(\mathbf{x}_{\text{col}}^{(i)}; \boldsymbol{\mu}_{\text{col}}^{(i)}) + f(\mathbf{x}_{\text{col}}^{(i)}; \boldsymbol{\mu}_{\text{col}}^{(i)}) \right|^2.$$

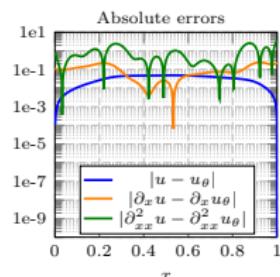
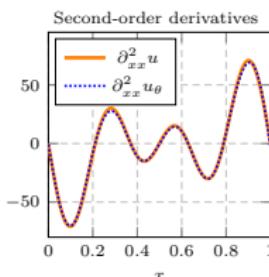
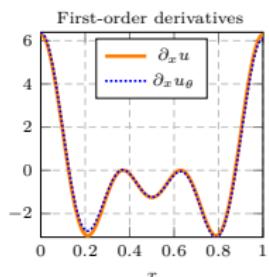
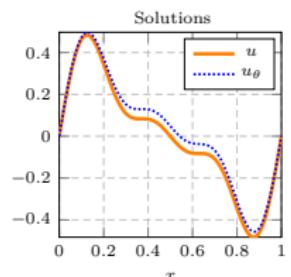
- **Data-driven training:**  $N_{\text{data}} = 5000$  data.

$$J_{\text{data}}(\theta) = \frac{1}{N_{\text{data}}} \sum_{i=1}^{N_{\text{data}}} \left| u_\theta^{\text{data}}(\mathbf{x}_{\text{data}}^{(i)}; \boldsymbol{\mu}_{\text{data}}^{(i)}) - u(\mathbf{x}_{\text{data}}^{(i)}; \boldsymbol{\mu}_{\text{data}}^{(i)}) \right|^2.$$

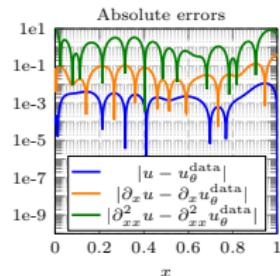
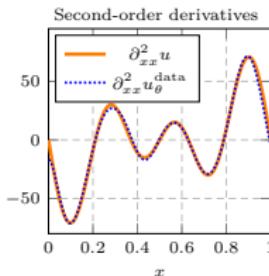
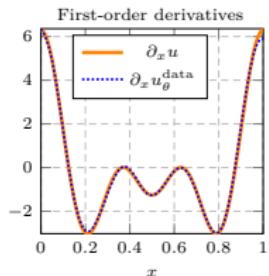
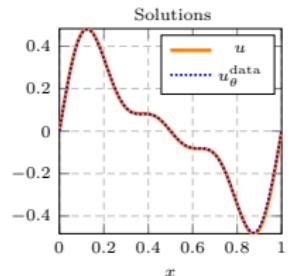
# A4 – Priors derivatives

$$\mu^{(1)} = (0.3, 0.2, 0.1)$$

PINN



Data



# A4 – Additive approach in $\mathbb{P}_1$

**1 set of parameters:**  $\mu^{(1)} = (0.3, 0.2, 0.1)$

FEM		PINN prior $u_\theta$			Data prior $u_\theta^{\text{data}}$	
N	error	N	error	gain	error	gain
16	$5.18 \cdot 10^{-2}$	16	$1.29 \cdot 10^{-3}$	40.34	$3.51 \cdot 10^{-3}$	14.78
32	$1.24 \cdot 10^{-2}$	32	$3.49 \cdot 10^{-4}$	35.41	$8.8 \cdot 10^{-4}$	14.06

**50 set of parameters:**

Gains in $L^2$ rel error of our method w.r.t. FEM						
	PINN prior $u_\theta$			Data prior $u_\theta^{\text{data}}$		
N	min	max	mean	min	max	mean
20	26.49	271.92	140.74	6.91	60.85	26.12
40	23.4	258.37	134.11	7.13	39.34	20.55

$N^2$  nodes.