

Enriching continuous Lagrange finite element approximation spaces using neural networks

Michel Duprez¹, Emmanuel Franck², **Frédérique Lecourtier¹** and Vanessa Lleras³

¹Project-Team MIMESIS, Inria, Strasbourg, France

²Project-Team MACARON, Inria, Strasbourg, France

³IMAG, University of Montpellier, Montpellier, France

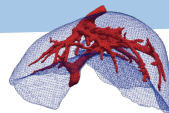
Joint work with:

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



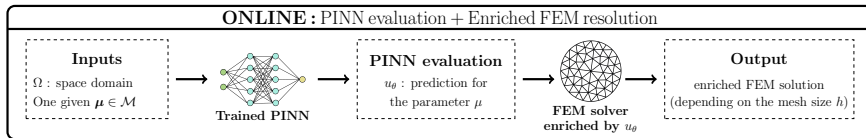
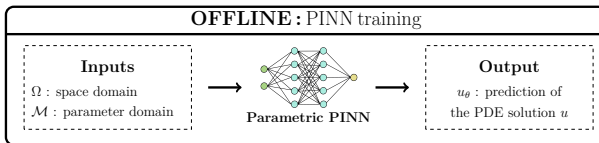
ICOSAHOM
MONTREAL 2025

Scientific context



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

Objective : Develop an hybrid finite element / neural network method.
accurate quick + parameterized



Problem considered

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

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

$$\left\{ \begin{array}{ll} (\mathbf{u} \cdot \nabla) \mathbf{u} + \nabla p - \mu \Delta \mathbf{u} - g(\beta T + 1) \mathbf{e}_y = 0 & \text{in } \Omega \quad (\text{momentum}) \\ \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega \quad (\text{incompressibility}) \\ \mathbf{u} \cdot \nabla T - k_f \Delta T = 0 & \text{in } \Omega \quad (\text{energy}) \\ + \text{suitable BC} & \end{array} \right. \quad (\mathcal{P})$$

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

Problem considered

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

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

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

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

Problem considered

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

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

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

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

Boundary Conditions:

- $\mathbf{u} = 0$ on $\partial\Omega$
- $T = 1$ on the left wall ($x = -1$) and $T = -1$ on the right wall ($x = 1$)
- $\frac{\partial T}{\partial n} = 0$ on the top and bottom walls ($y = \pm 1$, denoted by Γ_{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.¹

¹Computed on a 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.

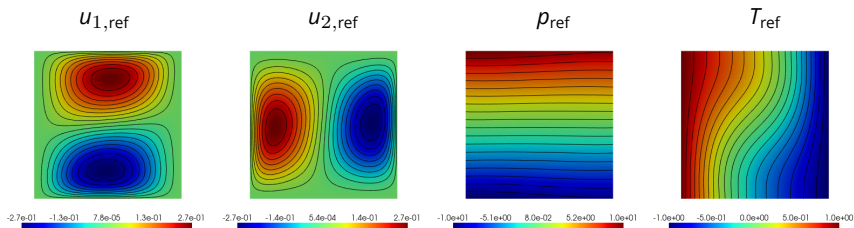
Evaluate quality of solutions

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.¹

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



¹Computed on a 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.

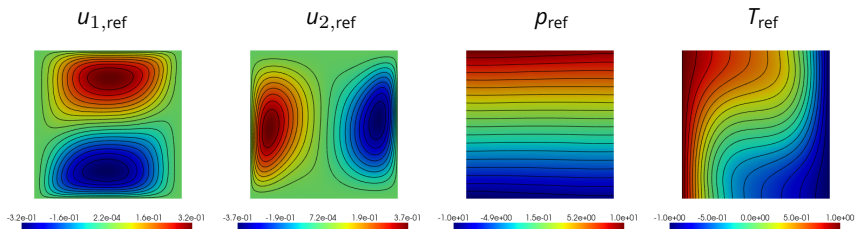
Evaluate quality of solutions

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

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

We evaluate the quality of solutions by comparing them to a reference solution.¹

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



¹Computed on a 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.

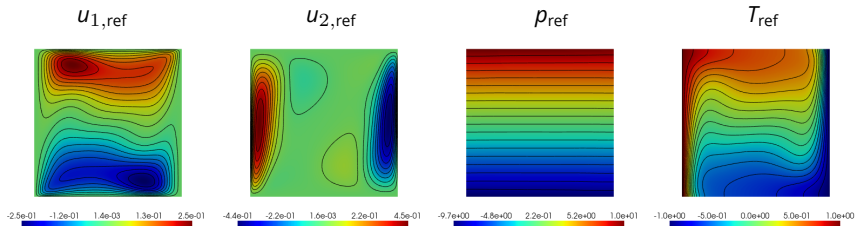
Evaluate quality of solutions

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

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

We evaluate the quality of solutions by comparing them to a reference solution.¹

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



¹Computed on a 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.

Physics-informed neural network (PINN)

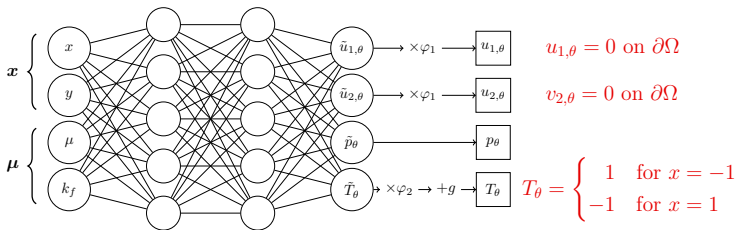
The PINN is parametrized by the μ parameter.

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 φ_1 and φ_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 θ^* , such that

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

where the different cost functions¹ are defined by

adiabatic condition

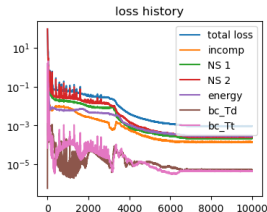
$$J_{ad}(\theta) = \int_{\mathcal{M}} \int_{\Gamma_{ad}} \left| \frac{\partial T_\theta(\mathbf{x}, \mu)}{\partial n} \right|^2 d\mathbf{x} d\mu,$$

3 residual losses

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

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

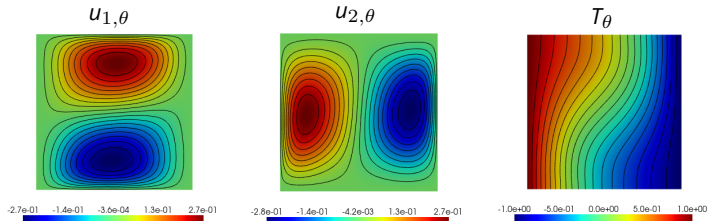
| Network - MLP | | Training (ADAM / LBFGS) | | | |
|---------------|--------------------|-------------------------|-------|-----------|-------|
| <i>layers</i> | 40, 60, 60, 60, 40 | <i>lr</i> | 7e-3 | N_{col} | 40000 |
| σ | sine | <i>n_epochs</i> | 10000 | N_{bc} | 30000 |



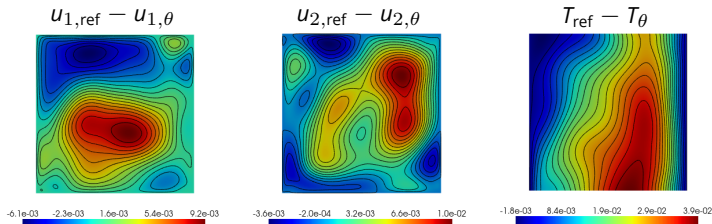
¹Discretized by a random process using Monte-Carlo method.

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

Prediction :



Error map :



L^2 error :

2.98×10^{-2}

3.17×10^{-2}

3.90×10^{-2}

Finite element method (FEM)

The μ 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

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

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

Discrete weak formulation

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

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

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., $\forall (\mathbf{v}_h, q_h, w_h) \in M_h^0$,

$$\begin{aligned} & \int_{\Omega} (\mathbf{u}_h \cdot \nabla) \mathbf{u}_h \cdot \mathbf{v}_h \, d\mathbf{x} + \mu \int_{\Omega} \nabla \mathbf{u}_h : \nabla \mathbf{v}_h \, d\mathbf{x} \\ & - \int_{\Omega} p_h \nabla \cdot \mathbf{v}_h \, d\mathbf{x} - g \int_{\Omega} (1 + \beta T_h) \mathbf{e}_y \cdot \mathbf{v}_h \, d\mathbf{x} = 0, \quad \text{(momentum)} \\ & \int_{\Omega} q_h \nabla \cdot \mathbf{u}_h \, d\mathbf{x} + 10^{-4} \int_{\Omega} q_h p_h \, d\mathbf{x} = 0, \quad \text{(incompressibility + pressure penalization)} \\ & \int_{\Omega} (\mathbf{u}_h \cdot \nabla T_h) w_h \, d\mathbf{x} + \int_{\Omega} k_f \nabla T_h \cdot \nabla w_h \, d\mathbf{x} = 0, \quad \text{(energy)} \end{aligned} \quad (\mathcal{P}_h)$$

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:

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

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 [Aghili et al., 2025]

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:

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

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 [Aghili et al., 2025]

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 initialization :**
- **DeepPhysics initialization :**
- **Incremental initialization.**

3 types of initialization

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

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

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

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

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

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

- **DeepPhysics initialization** :
- **Incremental initialization**.

3 types of initialization

- **Natural initialization** : Using constant or linear function.
- **DeepPhysics initialization** : Using PINN prediction [Odote et al., 2021].
 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_θ the PINN.

- **Incremental initialization.**

3 types of initialization

- **Natural initialization** : Using constant or linear function.
- **DeepPhysics initialization** : Using PINN prediction [[Odot et al., 2021](#)].
- **Incremental initialization.** Using a coarse FE solution of a simpler parameter.
 - We consider a fixed parameter with $k \in \{2, 3\}$.
 - We consider a coarse grid (16×16 grid) and compute the FE solution of (\mathcal{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

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 $C_h^+ = (\mathbf{c}_{h,u}^+, c_{h,p}^+, c_{h,T}^+)$.

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

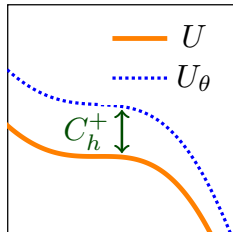
$$V_h^+ = \{\mathbf{u}_h^+ = \mathbf{u}_\theta + \mathbf{c}_{h,u}^+, \quad \mathbf{c}_{h,u}^+ \in [V_h^0]^2\},$$

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

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

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

à ajouter : dans quoi vit U_θ ?



Weak formulation - Additive approach

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

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

with $\mathbf{U}_{\theta} = (\mathbf{u}_{\theta}, p_{\theta}, T_{\theta})$ 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 $\vec{C}^{(0)} = \mathbf{0}$;

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

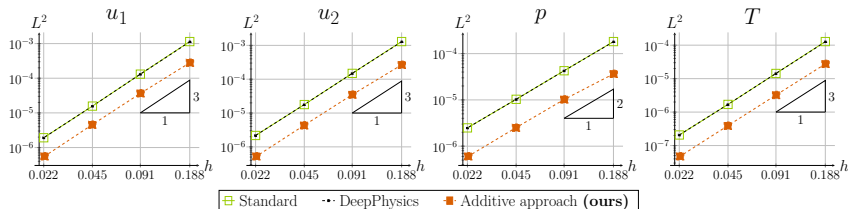
Advantage compared to DeepPhysics¹: Appendix 2

u_{θ} is not required to live in the same discrete space as C_h^+ .

¹The additive approach is exactly the same as DeepPhysics if we take U_{θ} in the same space as C_h^+ .

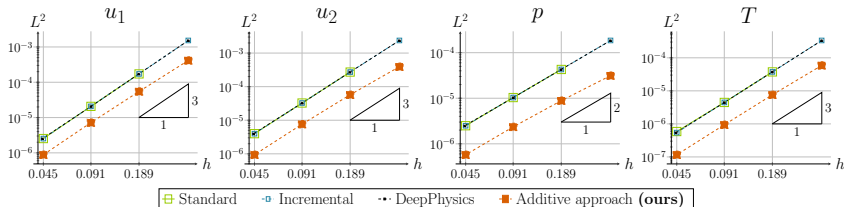
Numerical results

Error estimates on $\mu^{(1)}$



| Number of iterations | |
|----------------------|---|
| Standard | 6 |
| DeepPhysics | 3 |
| Additive | 3 |

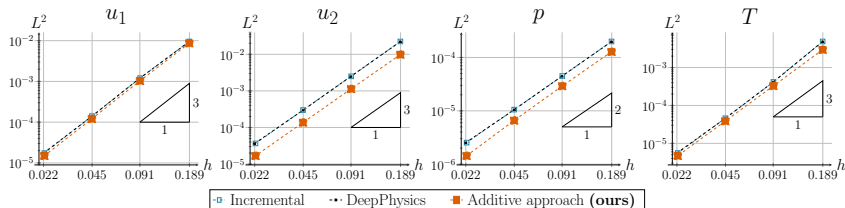
Error estimates on $\mu^{(2)}$



Number of iterations

| | |
|-----------------|-----------|
| Standard | 10 / None |
| Incremental | 4 |
| DeepPhysics | 3 |
| Additive | 3 |

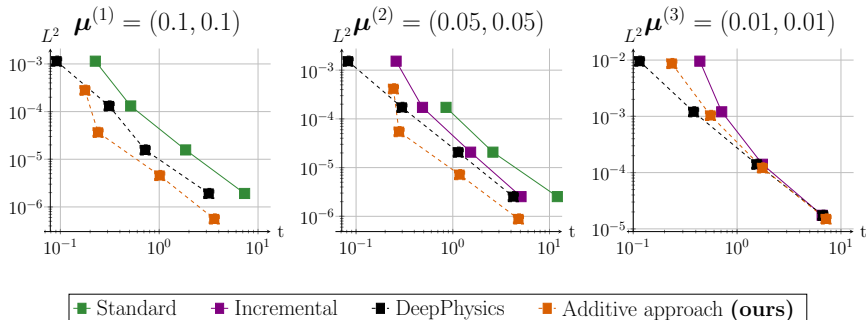
Error estimates on $\mu^{(3)}$



Number of iterations

| Standard | None |
|-----------------|----------|
| Incremental | 5 |
| DeepPhysics | 5 |
| Additive | 5 |

Execution times



Conclusion

TODO

Parler du papier en linéaire et dire que dans ce cadre on a des résultats théoriques de convergence.

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. 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.

Appendix 1 : Finite element method (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 : DeepPhysics / Additive approach

A2 – ??