2nd CSI

Development of hybrid finite element/neural network methods to help create digital surgical twins

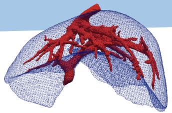
Michel Duprez[1], Emmanuel Franck[2], **Frédérique Lecourtier**[1] and Vanessa Lleras[3]

[1]Project-Team MIMESIS, Inria, Strasbourg, France
[2]Project-Team MACARON, Inria, Strasbourg, France
[3]IMAG, University of Montpellier, Montpellier, France

June 12, 2025

☆ - Update 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

☆ **Parametric elliptic convection/diffusion PDE :** For one or several $\boldsymbol{\mu} \in \mathcal{M}$, find $u : \Omega \to \mathbb{R}$ such that

$$\mathcal{L}\big(u\,;\boldsymbol{x},\boldsymbol{\mu}\big) = f(\boldsymbol{x},\boldsymbol{\mu}), \qquad (\mathcal{P})$$

where $\mathcal{L}$ is the parametric differential operator defined by

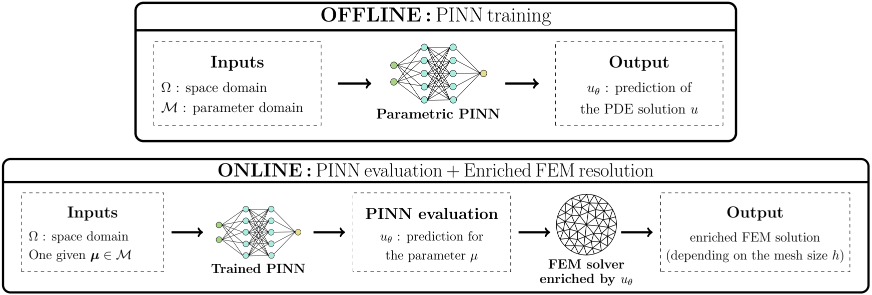
$$\mathcal{L}(\cdot;\boldsymbol{x},\boldsymbol{\mu}) : u \mapsto R(\boldsymbol{x},\boldsymbol{\mu})u + C(\boldsymbol{\mu}) \cdot \nabla u - \frac{1}{\mathrm{Pe}}\nabla \cdot (D(\boldsymbol{x},\boldsymbol{\mu})\nabla u),$$

and some Dirichlet, Neumann or Robin BC (which can also depend on $\boldsymbol{\mu}$).

Pipeline of the Enriched FEM

Enriched FEM = Combination of 2 standard methods

- **PINNs** : Physics Informed Neural Networks Appendix 1.1
- **FEMs** : Finite Element Methods Appendix 1.2



Remark : The PINN prediction enriched Finite element approximation spaces.

Enriched finite element method

Additive approach
Numerical results ☆

This section is based on [Lecourtier et al., 2025].

Enriched finite element method

Additive approach
Numerical results ☆

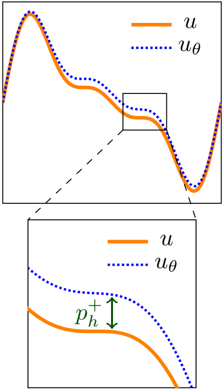
Additive approach

Variational Problem : Let $u_\theta \in H^{k+1}(\Omega) \cap H_0^1(\Omega)$.

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

with the enriched trial space V_h^+ defined by

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



General Dirichlet BC : If $u = g$ on $\partial\Omega$, then

$$p_h^+ = g - u_\theta \quad \text{on } \partial\Omega,$$

with u_θ the PINN prior.

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,

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

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

Theorem 2: Convergence analysis of the enriched FEM [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,

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

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

Gains of the additive approach.

Enriched finite element method

Additive approach

Numerical results ☆

1st problem considered

Problem statement: Considering an Anisotropic Elliptic problem with Dirichlet BC:

$$\begin{cases} -\mathrm{div}(D\nabla u) = f, & \text{in } \Omega, \\ u = 0, & \text{on } \partial\Omega, \end{cases}$$

with $\Omega = [0,1]^2$ and $\mathcal{M} = [0.4, 0.6] \times [0.4, 0.6] \times [0.01, 1] \times [0.1, 0.8]$ ($p = 4$).

Right-hand side :

$$f(\boldsymbol{x}, \boldsymbol{\mu}) = \exp\left(-\frac{(x-\mu_1)^2 + (y-\mu_2)^2}{0.025\sigma^2} \right).$$

Diffusion matrix : (symmetric and positive definite)

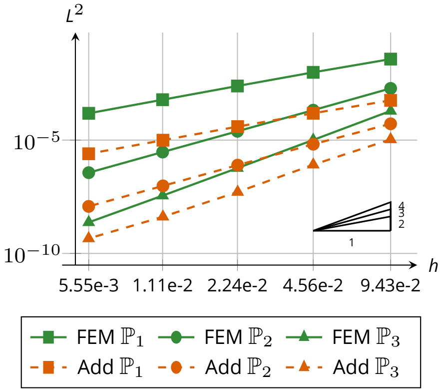
$$D(\boldsymbol{x}, \boldsymbol{\mu}) = \begin{pmatrix} \epsilon x^2 + y^2 & (\epsilon - 1)xy \\ (\epsilon - 1)xy & x^2 + \epsilon y^2 \end{pmatrix}.$$

PINN training: Imposing BC exactly with a level-set function.

Numerical results

Error estimates : 1 set of parameters.

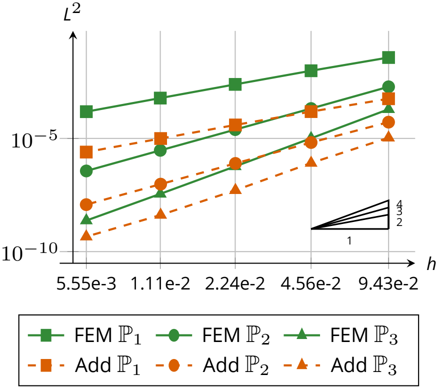
$$\boldsymbol{\mu}^{(1)} = (0.51, 0.54, 0.52, 0.55)$$



Numerical results

Error estimates : 1 set of parameters.

$$\boldsymbol{\mu}^{(1)} = (0.51, 0.54, 0.52, 0.55)$$



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

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

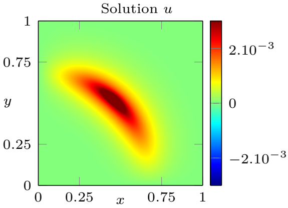
	Gains in L^2 rel error of our method w.r.t. FEM		
k	**min**	**max**	**mean**
1	7.12	82.57	35.67
2	3.54	35.88	18.32
3	1.33	26.51	8.32

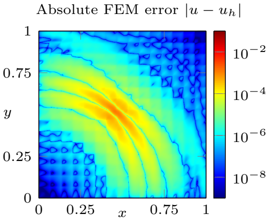
$N = 20$

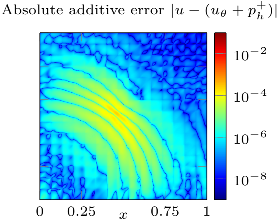
Gain : $\|u - u_h\|_{L^2} / \|u - u_h^+\|_{L^2}$

Cartesian mesh : N^2 nodes.

Numerical solutions







$$\boldsymbol{\mu}^{(2)} = (0.46, 0.52, 0.05, 0.12)$$

2nd problem considered

Problem statement: Considering the Poisson problem with mixed BC:

$$\begin{cases} -\Delta u = f, & \text{in } \Omega \times \mathcal{M}, \\ u = g, & \text{on } \Gamma_E \times \mathcal{M}, \\ \dfrac{\partial u}{\partial n} + u = g_R, & \text{on } \Gamma_I \times \mathcal{M}, \end{cases}$$

with $\Omega = \{(x,y) \in \mathbb{R}^2,\ 0.25 \leq x^2 + y^2 \leq 1\}$ and $\mathcal{M} = [2.4, 2.6]$ ($p = 1$).

Analytical solution :

$$u(\boldsymbol{x}; \boldsymbol{\mu}) = 1 - \frac{\ln\left(\mu_1 \sqrt{x^2 + y^2}\right)}{\ln(4)},$$

Boundary conditions :

$$g(\boldsymbol{x}; \boldsymbol{\mu}) = 1 - \frac{\ln(\mu_1)}{\ln(4)} \quad \text{and} \quad g_R(\boldsymbol{x}; \boldsymbol{\mu}) = 2 + \frac{4 - \ln(\mu_1)}{\ln(4)}.$$

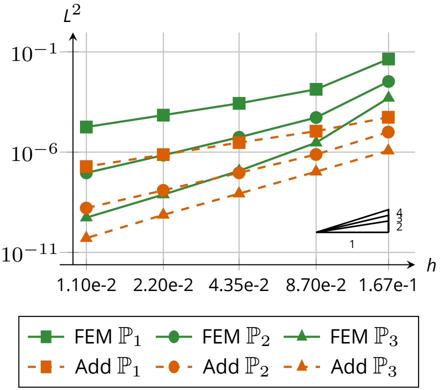
PINN training: Imposing mixed BC exactly in the PINN[1].

[1] [Sukumar and Srivastava, 2022]

Numerical results

Error estimates : 1 set of parameters.

$$\boldsymbol{\mu}^{(1)} = 2.51$$



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

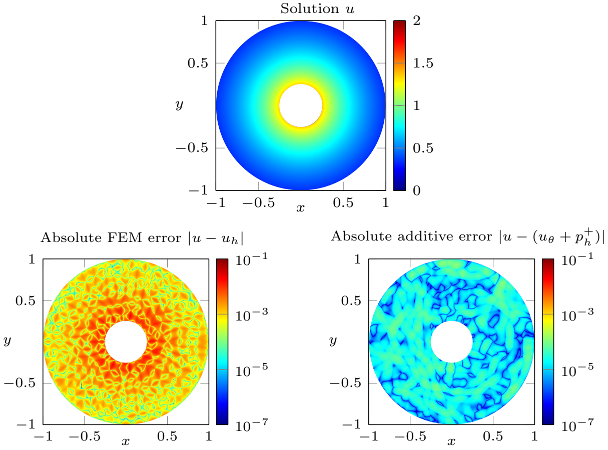
$$\mathcal{S} = \left\{ \boldsymbol{\mu}^{(1)}, \dots, \boldsymbol{\mu}^{(n_p)} \right\}$$

	Gains in L^2 rel error of our method w.r.t. FEM		
k	**min**	**max**	**mean**
1	15.12	137.72	55.5
2	31	77.46	58.41
3	18.72	21.49	20.6

$$h = 1.33 \cdot 10^{-1}$$

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

Numerical solutions



$$\mu^{(1)} = 2.51$$

Supplementary work

Supplementary work I

Teaching

- 2024/2025 :
  - 64h of Computer Science Practical Work - L1S2 and L2S3 (Python) / L3S6 (C++)
  - 3 days supervising a group of high school girls in RJMI ("Rendez-vous des Jeunes Mathématiciennes et Informaticiennes")
- 2023/2024 : 50h of Computer Science Practical Work - L2S3 (Python) / L3S6 (C++)

Training courses (Total : 176h35)

- A dozen seminars organized by IRMA ($\approx 10h$)
- 1 Deep Learning introductory course - FIDLE ($\approx 40h$)
- 2 workshops on Scientific Machine Learning ($\approx 2 \times 21h$)
- 1 summer school on "New Trend in computing" ($\approx 27h$)
- several cross-disciplinary courses - Methodology, scientific English, etc. ($\approx 58h$)

Supplementary work II

Talks

- **ICOSAHOM 2025, Montréal** - July 2025 *(Coming soon...)*
  ”Enriching continuous Lagrange finite element approximation spaces using neural networks”
- **DTE & AICOMAS 2025, Paris** - February 20, 2025
  ”Combining Finite Element Methods and Neural Networks to Solve Elliptic Problems on 2D Geometries”
- **Exama project, WP2 reunion** - March 26, 2024
  ”How to work with complex geometries in PINNs ?”
- **Retreat (Macaron/Tonus)** - February 6, 2024
  ”Mesh-based methods and physically informed learning”
- **Team meeting (Mimesis)** - December 12, 2023
  ”Development of hybrid finite element/neural network methods to help create digital surgical twins”

Supplementary work III

Posters

- **EMS-TAG-SciML 2025, Milan** - March 24, 2025 - ”Enriching continuous Lagrange finite element approximation spaces using neural networks”
- **CJC-MA 2024, Lyon** - October 29, 2024 - ”Combining Finite Element Methods and Neural Networks to Solve Elliptic Problems on 2D Geometries”
- **MSII poster day, Strasbourg** - October 24, 2024
- **SciML 2024, Strasbourg** - July 08, 2024

Publications

- Enriching continuous lagrange finite element approximation spaces using neural networks. *(submitted in February 2025, M2AN journal)*
  H. Barucq, M. Duprez, F. Faucher, E. Franck, **F. Lecourtier**, V. Lleras, V. Michel-Dansac, and N. Victorion.

Conclusion

TODO

References

A. Ern and J.-L. Guermond. *Theory and Practice of Finite Elements*. Springer New York, 2004.

E. Franck, V. Michel-Dansac, and L. Navoret. Approximately well-balanced Discontinuous Galerkin methods using bases enriched with Physics-Informed Neural Networks. *J. Comput. Phys.*, 512:113144, 2024. ISSN 0021-9991.

I. E. Lagaris, A. Likas, and D. I. Fotiadis. Artificial neural networks for solving ordinary and partial differential equations. *IEEE Trans. Neural Netw.*, 9(5):987–1000, 1998. ISSN 1045-9227.

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.

M. Raissi, P. Perdikaris, and G. E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. Phys.*, 378:686–707, 2019.

N. Sukumar and A. Srivastava. Exact imposition of boundary conditions with distance functions in physics-informed deep neural networks. *Comput. Method. Appl. M.*, 389:114333, 2022. ISSN 0045-7825.

M. Tancik, P. Srinivasan, and al. Fourier Features Let Networks Learn High Frequency Functions in Low Dimensional Domains. In *Advances in Neural Information Processing Systems*, volume 33, pages 7537–7547. Curran Associates, Inc., 2020.

Appendix 1 : Standard methods

A1.1 – Physics-Informed Neural Networks

Standard PINNs[1] **(Weak BC) :** Find the optimal weights $\theta^\star$, such that

$$\theta^\star = \underset{\theta}{\operatorname{argmin}} \left(\omega_r\, J_r(\theta) + \omega_b\, J_b(\theta)\right), \qquad (\mathcal{P}_\theta)$$

with

residual loss $$\boxed{J_r(\theta) = \int_{\mathcal{M}} \int_{\Omega} \left|\mathcal{L}\left(u_\theta(\boldsymbol{x}, \boldsymbol{\mu}); \boldsymbol{x}, \boldsymbol{\mu}\right) - f(\boldsymbol{x}, \boldsymbol{\mu})\right|^2 d\boldsymbol{x} d\boldsymbol{\mu},}$$

boundary loss $$\boxed{J_b(\theta) = \int_{\mathcal{M}} \int_{\partial\Omega} \left|u_\theta(\boldsymbol{x}, \boldsymbol{\mu}) - g(\boldsymbol{x}, \boldsymbol{\mu})\right|^2 d\boldsymbol{x} d\boldsymbol{\mu},}$$

where u_θ is a neural network, $g = 0$ is the Dirichlet BC.

In $(\mathcal{P}_\theta)$, ω_r and ω_b are some weights.

Monte-Carlo method : Discretize the cost functions by random process.

[1] [Raissi et al., 2019]

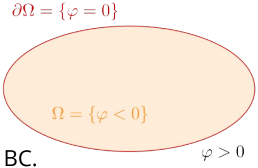
A1.1 – Physics-Informed Neural Networks

Improved PINNs[1] **(Strong BC) :** Find the optimal weights $\theta^\star$ such that

$$\theta^\star = \underset{\theta}{\operatorname{argmin}} \left(\omega_r J_r(\theta) + \cancel{\omega_b J_b(\theta)}\right),$$

with $\omega_r = 1$ and

residual loss $\boxed{J_r(\theta) = \int_{\mathcal{M}} \int_{\Omega} \left| \mathcal{L}\big(u_\theta(\boldsymbol{x}, \boldsymbol{\mu}); \boldsymbol{x}, \boldsymbol{\mu}\big) - f(\boldsymbol{x}, \boldsymbol{\mu}) \right|^2 d\boldsymbol{x} d\boldsymbol{\mu},}$



where u_θ is a neural network defined by

$$u_\theta(\boldsymbol{x}, \boldsymbol{\mu}) = \varphi(\boldsymbol{x}) w_\theta(\boldsymbol{x}, \boldsymbol{\mu}) + g(\boldsymbol{x}, \boldsymbol{\mu}),$$

with φ a level-set function, w_θ a NN and $g = 0$ the Dirichlet BC.

Thus, the Dirichlet BC is imposed exactly in the PINN : $u_\theta = g$ on $\partial\Omega$.

[1] [Lagaris et al., 1998; Franck et al., 2024]

A1.2 – Finite Element Methods[1]

Variational Problem :

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

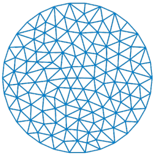
with h the characteristic mesh size, a and l the bilinear and linear forms given by

$$a(u_h, v_h) = \frac{1}{\mathrm{Pe}} \int_\Omega D \nabla u_h \cdot \nabla v_h + \int_\Omega R\, u_h\, v_h + \int_\Omega v_h\, C \cdot \nabla u_h, \quad l(v_h) = \int_\Omega f\, v_h,$$

and V_h^0 the finite element space defined by

$$V_h^0 = \left\{ v_h \in C^0(\Omega),\ \forall K \in \mathcal{T}_h,\ v_h|_K \in \mathbb{P}_k, v_h|_{\partial\Omega} = 0 \right\},$$

where $\mathbb{P}_k$ is the space of polynomials of degree at most k.



$\mathcal{T}_h = \{K_1, \ldots, K_{N_e}\}$
(N_e : number of elements)

Linear system : Let $(\phi_1, \ldots, \phi_{N_h})$ a basis of V_h^0.

Find $U \in \mathbb{R}^{N_h}$ such that $\qquad AU = b$

with

$$A = \left(a(\phi_i, \phi_j)\right)_{1 \le i,j \le N_h} \quad \text{and} \quad b = \left(l(\phi_j)\right)_{1 \le j \le N_h}.$$

[1] [Ern and Guermond, 2004]