Innovative non-conformal finite element methods for augmented surgery

Internship presentation

LECOURTIER Frédérique, DUPREZ Michel, FRANCK Emmanuel, LLERAS Vanessa

Strasbourg University

August 24, 2023

MIMESIS

Introduction

Presentation of the teams

MIMESIS

- project-team as sub-team of MLMS ("Machine Leraning, Modélisation et Simulation") of Inria
- **Aim :** to create real-time digital twins of an organ
- **Scientific challenges :**
  - scientific computing
  - data assimilation
  - machine learning
  - control
- **Main application domains :**
  - surgical training
  - surgical guidance during complex interventions

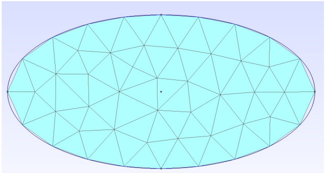
Scientific context

A MODIFIER !!

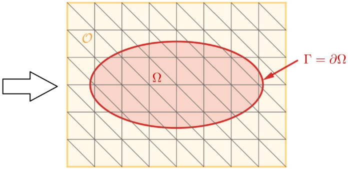
Because of the geometry of organs, mimesis has developed a new method: the ϕ-FEM method.

Abstract : Mimesis propose a new fictitious domain finite element method, well suited for elliptic problems posed in a domain given by a level-set function without requiring a mesh fitting the boundary.

Classic mesh



Fictitious domain



Objectives - Deliverables

Objectives :

- Train a Fourier Neural Operator (FNO), with ϕ-FEM solutions, to predict the solutions of a given PDE.
- Apply a correction on the FNO predictions.

Aim : Neural networks are very fast, but not very accurate
=> finite element methods are used to improve prediction accuracy. **Remark :** Implementation in Python with FEniCS, Pytorch and Tensorflow. **Deliverables :**

- a weekly tracking report (in French)
- a github repository containing all the code allowing to reproduce the results presented in this report
- a report of the internship
- an online report generated with a tool called antora (made by a github CI)

Problem considered

Poisson problem with Dirichlet conditions :
Find $u : \Omega \to \mathbb{R}^d (d = 1, 2, 3)$ such that

$$\begin{cases} -\Delta u = f, & \text{in } \Omega, \\ u = g, & \text{on } \partial\Omega, \end{cases} \qquad (\mathcal{P})$$

with Δ the Laplace operator, Ω a smooth bounded open set and $\partial\Omega$ its boundary.
In this section, we defined by

$$||u_{ex} - u_{method}||_{0,\Omega_h}^{(rel)} = \frac{\int_{\Omega_h} (u_{ex} - u_{method})^2}{\int_{\Omega_h} u_{ex}^2}$$

the relative error between the exact solution u_{ex} and u_{method} a solution obtained by FEM or ϕ-FEM, a correction solver or the prediction of an neural network.

MIMESIS

General methods and tools

Standard FEM method

ϕ-FEM method

Fourier Neural Operator (FNO)

General methods and tools

Standard FEM method

ϕ-FEM method

Fourier Neural Operator (FNO)

Presentation of standard FEM method

Variational Problem :

$$\text{Find } u \in V \text{ such that } a(u,v) = l(v),\ \forall v \in V$$

where V is a Hilbert space, a is a bilinear form and l is a linear form.

Approach Problem :

$$\text{Find } u_h \in V_h \text{ such that } a(u_h,v_h) = l(v_h),\ \forall v_h \in V$$

with u_h an approximate solution in V_h, a finite-dimensional space dependent on h such that $V_h \subset V,\ dimV_h = N_h < \infty,\ \forall h > 0$

As $u_h = \sum_{i=1}^{N_h} u_i \varphi_i$ with $(\varphi_1, \ldots, \varphi_{N_h})$ a basis of V_h, finding an approximation of the PDE solution implies solving the following linear system:

$$AU = b$$

with

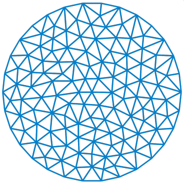
$$A = (a(\varphi_i, \varphi_j))_{1 \leq i,j \leq N_h}, \quad U = (u_i)_{1 \leq i \leq N_h} \quad \text{and} \quad b = (l(\varphi_j))_{1 \leq j \leq N_h}$$

In practice

- Construct a mesh of our Ω geometry with a family of elements (in 2D: triangle, rectangle; in 3D: tetrahedron, parallelepiped, prism) defined by

$$\mathcal{T}_h = \{K_1, \ldots, K_{N_e}\}$$

where N_e is the number of elements.



- Construct a space of piece-wise affine continuous functions, defined by

$$V_h := P^k_{C,h} = \{v_h \in C^0(\bar{\Omega}), \forall K \in \mathcal{T}_h, v_{h|K} \in \mathbb{P}_k\}$$

where $\mathbb{P}_k$ is a vector space of polynomials of total degree less than or equal to k.

- In the case of non-homogeneous condition : use of penalization or elimination methods.

General methods and tools

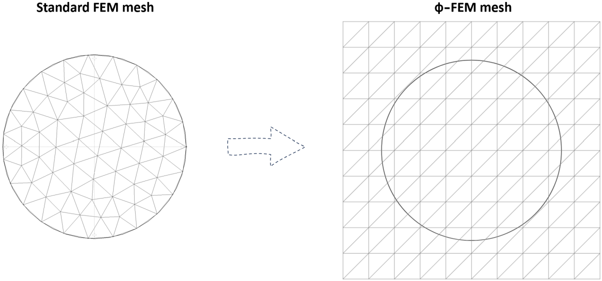
Standard FEM method

ϕ-FEM method

Fourier Neural Operator (FNO)

Context

Idea : ϕ-FEM method = new fictitious domain finite element method that does not require a mesh conforming to the real boundary.



Advantage : boundary represented by a level-set function $\Rightarrow$ only this function will change over time during real-time simulation

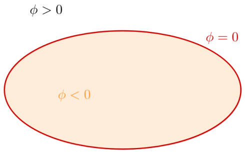
Problem

We pose $u = \phi w$ such that

$$\begin{cases} -\Delta(\phi w) = f, \text{ on } \Omega, \\ u = g, \text{ in } \Gamma, \end{cases}$$

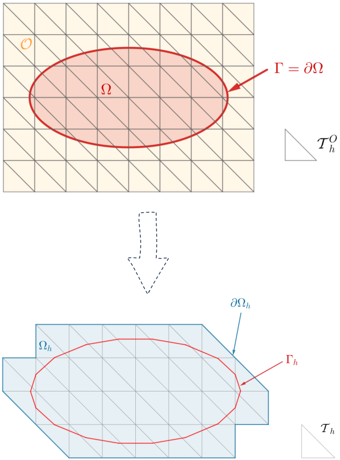
where ϕ is the level-set function and Ω and Γ are given by :

$$\Omega = \{\phi < 0\} \quad \text{and} \quad \Gamma = \{\phi = 0\}.$$



The level-set function ϕ is supposed to be known on $\mathbb{R}^d$ and sufficiently smooth.
For instance, the signed distance to Γ is a good candidate

Fictitious domain



- $\mathcal{O}$: fictitious domain such that $\Omega \subset \mathcal{O}$
- $\mathcal{T}_h^{\mathcal{O}}$: simple quasi-uniform mesh on $\mathcal{O}$
- $\phi_h = I_{h,\mathcal{O}}^{(l)}(\phi) \in V_{h,\mathcal{O}}^{(l)}$: approximation of ϕ with $I_{h,\mathcal{O}}^{(l)}$ the standard Lagrange interpolation operator on

$$V_{h,\mathcal{O}}^{(l)} = \left\{ v_h \in H^1(\mathcal{O}) : v_{h|_T} \in \mathbb{P}_l(T) \ \forall T \in \mathcal{T}_h^{\mathcal{O}} \right\}$$

- $\Gamma_h = \{\phi_h = 0\}$: approximate boundary of Γ
- $\mathcal{T}_h$: sub-mesh of $\mathcal{T}_h^{\mathcal{O}}$ defined by

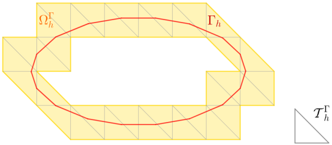
$$\mathcal{T}_h = \left\{ T \in \mathcal{T}_h^{\mathcal{O}} : T \cap \{\phi_h < 0\} \neq \emptyset \right\}$$

- Ω_h : domain covered by the $\mathcal{T}_h$ mesh defined by

$$\Omega_h = \left(\cup_{T \in \mathcal{T}_h} T \right)^O$$

($\partial\Omega_h$ its boundary)

Facets and Cells sets

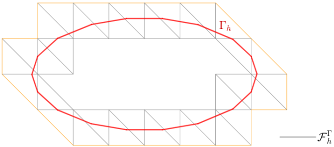


- $\mathcal{T}_h^\Gamma \subset \mathcal{T}_h$: contains the mesh elements cut by Γ_h, i.e.

$$\mathcal{T}_h^\Gamma = \{T \in \mathcal{T}_h : T \cap \Gamma_h \neq \emptyset\},$$

- Ω_h^Γ : domain covered by the $\mathcal{T}_h^\Gamma$ mesh, i.e.

$$\Omega_h^\Gamma = \left(\cup_{T \in \mathcal{T}_h^\Gamma} T\right)^o$$



- $\mathcal{F}_h^\Gamma$: collects the interior facets of $\mathcal{T}_h$ either cut by Γ_h or belonging to a cut mesh element, i.e.

$$\mathcal{F}_h^\Gamma = \{E \text{ (an internal facet of } \mathcal{T}_h) \text{ such that } \exists T \in \mathcal{T}_h : T \cap \Gamma_h \neq \emptyset \text{ and } E \in \partial T\}$$

Application to the Poisson problem

We start by consider the **homogeneous case** ($g = 0$ on Γ).

Approach Problem : Find $w_h \in V_h^{(k)}$ such that

$$a_h(w_h, v_h) = l_h(v_h) \quad \forall v_h \in V_h^{(k)}$$

where

$$a_h(w,v) = \int_{\Omega_h} \nabla(\phi_h w) \cdot \nabla(\phi_h v) - \int_{\partial\Omega_h} \frac{\partial}{\partial n}(\phi_h w)\phi_h v + G_h(w,v),$$

$$l_h(v) = \int_{\Omega_h} f\phi_h v + G_h^{rhs}(v)$$

and

$$V_h^{(k)} = \left\{ v_h \in H^1(\Omega_h) : v_{h|_T} \in \mathbb{P}_k(T),\ \forall T \in \mathcal{T}_h \right\}.$$

MIMESIS

Stabilization terms

Independent parameter of h

Jump on the interface E

$$G_h(w,v) = \boxed{\sigma h \sum_{E\in\mathcal{F}_h^\Gamma} \int_E \left[\frac{\partial}{\partial n}(\phi_h w)\right]\left[\frac{\partial}{\partial n}(\phi_h v)\right]} + \boxed{\sigma h^2 \sum_{T\in\mathcal{T}_h^\Gamma} \int_T \Delta(\phi_h w)\Delta(\phi_h v)}$$

1st order term

$$G_h^{rhs}(v) = \boxed{-\sigma h^2 \sum_{T\in\mathcal{T}_h^\Gamma} \int_T f\Delta(\phi_h v)}$$

-

2nd order term

$$\sigma h^2 \sum_{T\in\mathcal{T}_h^\Gamma} \int_T \left(\Delta(\phi_h w) + f\right)\Delta(\phi_h v)$$

1st term : Ghost penality [4], ensure continuity of the solution by penalizing gradient jumps.
2nd term : require the solution to verify the strong form on Ω_h^Γ.

Purpose :

- reduce the errors created by the "fictitious" boundary
- ensure the correct condition number of the finite element matrix
- permit to restore the coercivity of the bilinear scheme

MIMESIS

Non-homogeneous case - Direct method

Suppose that g is currently given over the entire Ω_h, we have

$$u = \phi w + g, \text{ on } \Omega_h.$$

Problem : Find w_h on Ω_h such that

$$\int_{\Omega_h} \nabla(\phi_h w_h)\nabla(\phi_h v_h) - \int_{\partial\Omega_h} \frac{\partial}{\partial n}(\phi_h w_h)\phi_h v_h + G_h(w_h, v_h) = \int_{\Omega_h} f\phi_h v_h$$
$$- \int_{\Omega_h} \nabla g \nabla(\phi_h v_h) + \int_{\partial\Omega_h} \frac{\partial g}{\partial n}\phi_h v_h + G_h^{rhs}(v_h), \ \forall v_h \in \Omega_h$$

with

$$G_h(w, v) = \sigma h \sum_{E \in \mathcal{F}_h^\Gamma} \int_E \left[\frac{\partial}{\partial n}(\phi_h w)\right]\left[\frac{\partial}{\partial n}(\phi_h v)\right] + \sigma h^2 \sum_{T \in \mathcal{T}_h^\Gamma} \int_T \Delta(\phi_h w)\Delta(\phi_h v)$$

and

$$G_h^{rhs}(v) = -\sigma h^2 \sum_{T \in \mathcal{T}_h^\Gamma} \int_T f\Delta(\phi_h v) - \sigma h \sum_{E \in \mathcal{F}_h^\Gamma} \int_E \left[\frac{\partial g}{\partial n}\right]\left[\frac{\partial}{\partial n}(\phi_h v)\right] - \sigma h^2 \sum_{T \in \mathcal{T}_h^\Gamma} \int_T \Delta g \Delta(\phi_h v)$$

Non-homogeneous case - Dual method

Assuming g defined on Ω_h^Γ, introduce p on Ω_h^Γ in addition to u on Ω_h with

$$u = \phi p + g, \text{ on } \Omega_h^\Gamma.$$

Problem : Find u on Ω_h and p on Ω_h^Γ such that

$$\int_{\Omega_h} \nabla u \nabla v - \int_{\partial\Omega_h} \frac{\partial u}{\partial n} v + \frac{\gamma}{h^2} \sum_{T \in \mathcal{T}_h^\Gamma} \int_T \left(u - \frac{1}{h}\phi p\right)\left(v - \frac{1}{h}\phi q\right) + G_h(u,v) = \int_{\Omega_h} fv$$

$$+ \frac{\gamma}{h^2} \sum_{T \in \mathcal{T}_h^\Gamma} \int_T g\left(v - \frac{1}{h}\phi q\right) + G_h^{rhs}(v), \ \forall v \text{ on } \Omega_h, \ q \text{ on } \Omega_h^\Gamma.$$

with γ an other positive stabilization parameter,

$$G_h(u,v) = \sigma h \sum_{E \in \mathcal{F}_h^\Gamma} \int_E \left[\frac{\partial u}{\partial n}\right]\left[\frac{\partial v}{\partial n}\right] + \sigma h^2 \sum_{T \in \mathcal{T}_h^\Gamma} \int_T \Delta u \Delta v$$

and $$G_h^{rhs}(v) = -\sigma h^2 \sum_{T \in \mathcal{T}_h^\Gamma} \int_T f \Delta v.$$

MIMESIS

General methods and tools

Standard FEM method

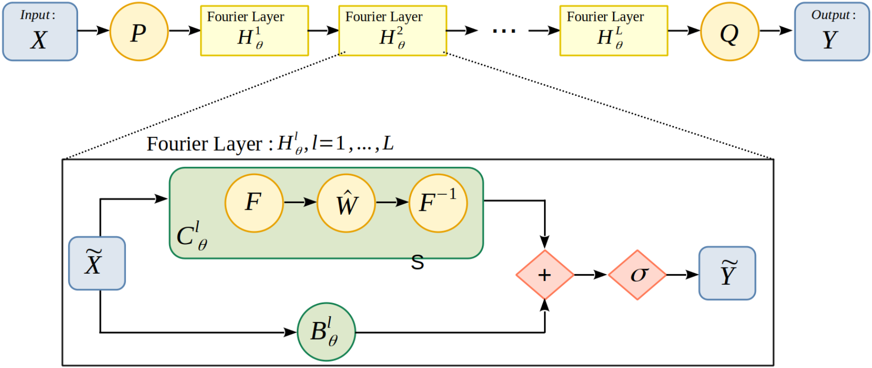
ϕ-FEM method

Fourier Neural Operator (FNO)

Presentation

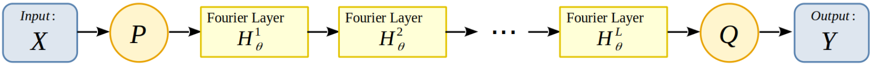
- → widely used in PDE solving and constitute an active field of research
- → FNO are Neural Operator networks : Unlike standard neural networks, which learn using inputs and outputs of fixed dimensions, neural operators **learn operators, which are mappings between spaces of functions**.
- → can be evaluated at any data resolution without the need for retraining

Architecture of the FNO



Input X of shape (bs,ni,nj,nk) **Output** Y of shape (bs,ni,nj,1)
with bs the batch size, ni and nj the grid resolution and nk the number of channels.

Description of the FNO architecture



- perform a P transformation, to move to a space with more channels (to build a sufficiently rich representation of the data)
- apply L Fourier layers defined by

$$\mathcal{H}^l_\theta(\tilde{X}) = \sigma\left(\mathcal{C}^l_\theta(\tilde{X}) + \mathcal{B}^l_\theta(\tilde{X})\right), \; l = 1, \dots, L$$

  with $\tilde{X}$ the input of the current layer and
  - σ an activation function (ReLU or GELU)
  - $\mathcal{C}^l_\theta$: convolution sublayer (convolution performed by Fast Fourier Transform)
  - $\mathcal{B}^l_\theta$: "bias-sublayer"
- return to the target dimension by performing a Q transformation (in our case, the number of output channels is 1)

Fourier Layer Structure

Convolution sublayer : $\mathcal{C}^l_\theta(X) = \mathcal{F}^{-1}(\mathcal{F}(X) \cdot \hat{W})$

C^l_θ: $F \longrightarrow \hat{W} \longrightarrow F^{-1}$

- $\hat{W}$: a trainable kernel
- $\mathcal{F}$: 2D Discrete Fourier Transform (DFT) defined by

$$\mathcal{F}(X)_{ijk} = \frac{1}{ni}\frac{1}{nj}\sum_{i'=0}^{ni-1}\sum_{j'=0}^{nj-1} X_{i'j'k} e^{-2\sqrt{-1}\pi\left(\frac{ii'}{ni}+\frac{jj'}{nj}\right)}$$

$\mathcal{F}^{-1}$: its inverse.

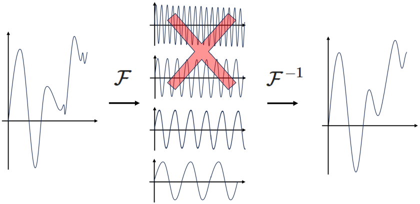
- $(Y \cdot \hat{W})_{ijk} = \sum_{k'} Y_{ijk'} \hat{W}_{ijk'}$ $\quad\Rightarrow\quad$ applied channel by channel

Bias-sublayer : $\mathcal{B}^l_\theta(X)_{ijk} = \sum_{k'} X_{ijk} W_{k'k} + B_k$ $\quad B^l_\theta$

- 2D convolution with a kernel of size 1
- allowing channels to be mixed via a kernel without allowing interaction between pixels.

Some details on the FNO

- **Mesh resolution independent :**
  - P and Q = fully-connected multi-layer perceptron $\Rightarrow$ perform local transformations at each point
  - Fourier layers also independent of mesh resolution : learn in Fourier space so the value of the Fourier modes does not depend on the mesh resolution
- **Low pass filter :** truncate high Fourier modes to ignore high frequencies $\Rightarrow$ enable a kind of regularization that helps the generalization



Limitation : Use of regular grid $\Rightarrow$ only $\mathbb{P}_1$ or $\mathbb{P}_2$ solution

Application

FNO Training :

Input : Train data .

$$X_{train} = \{ f_i, g_i, \phi_i \}$$
$$Y_{train} = \{ w_i \}_{i=1,\dots,n_{data}}$$

→

FNO training .

$$loss_\theta = \| \phi w_\theta - \phi w_{true} \|_{H^2(\Omega_h)}$$

→

Output .

$$Y_{pred} = \{ \phi_i w_{\theta,i} \}_{i=1,\dots,n_{data}}$$

Correction on the FNO predictions :

Input : Test data .

$$X_{test} = \{ f_i^{test}, g_i^{test}, \phi_i^{test} \}_{i=1,\dots,n_{test}}$$

→

FNO

$$Y_{test} = \{ \phi_i^{test} w_{\theta,i}^{test} \}$$

→

Correction :
- → *by adding*
- → *by multiplying*
- → *by multiplying on an elevated problem*

Correction

Problems considered

Methods considered

Numerical results

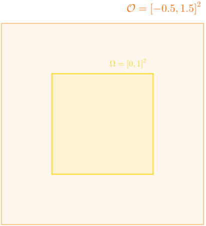
Correction

Problems considered

Methods considered

Numerical results

Trigonometric solution on a Square



→ Level-set function (for formulation) :

$$\phi(x,y) = x(1-x)y(1-y)$$

→ Level-set function (for construction) :

$$\phi_C(X) = ||X - 0.5||_\infty - 0.5$$

→ Analytical solution : (Homogeneous if $\varphi = 0$)

$$u_{ex}(x,y) = S \times sin\left(2\pi f x + \varphi\right) \times sin\left(2\pi f y + \varphi\right)$$

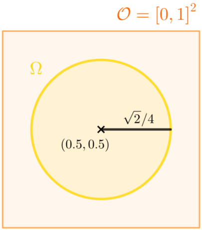
→ Source term :

$$f(x,y) = 8\pi^2 S f^2 sin\left(2\pi f x + \varphi\right) sin\left(2\pi f y + \varphi\right)$$

- $S \in [0,1]$: amplitude of the signal
- $f \in \mathbb{N}$: "frequency" of the signal
- $\varphi \in [0,1]$: phase at the origin

		f = 1	f = 2	f = 3	f = 4
Homogeneous	**fem**	2.61e-03	1.04e-02	2.34e-02	4.12e-02
	phifem	1.84e-03	1.44e-02	3.69e-02	6.84e-02
Non-homogeneous	**fem**	2.30e-03	9.53e-03	2.19e-02	3.88e-02
	phifem	1.20e-04	5.74e-04	1.34e-03	2.39e-03

Unknown solution on a Circle



→ Level-set function :

$$\phi(x,y) = -1/8 + (x - 1/2)^2 + (y - 1/2)^2$$

→ Source term :

$$f(x,y) = exp\left(-\frac{(x-\mu_0)^2 + (y-\mu_1)^2}{2\sigma^2}\right)$$

- $\sigma \sim \mathcal{U}([0.1, 0.6])$
- $\mu_0, \mu_1 \sim \mathcal{U}([0.5 - \sqrt{2}/4, 0.5 + \sqrt{2}/4])$

with the condition $\phi(\mu_0, \mu_1) < -0.05$

→ a reference solution u_{ref} : over-refined $\mathbb{P}^1$ solution (with FEM)

Correction

Problems considered

Methods considered

Numerical results

Correction by adding

We are given $\tilde{\phi}$ an "initial" solution to the problem under consideration.
We will consider

$$\tilde{u} = \tilde{\phi} + \tilde{C}$$

We want to find $\tilde{C} : \Omega \to \mathbb{R}^d$ solution to the problem

$$\begin{cases} -\Delta \tilde{u} = f, & \text{on } \Omega, \\ \tilde{u} = g, & \text{in } \Gamma. \end{cases}$$

with $\tilde{C}\phi C$ for the ϕ-FEM method Rewriting the problem, we seek to find $\tilde{C} : \Omega \to \mathbb{R}^d$ solution to the problem

$$\begin{cases} -\Delta \tilde{C} = \tilde{f}, & \text{on } \Omega, \\ \tilde{C} = 0, & \text{in } \Gamma. \end{cases} \qquad (\mathcal{C}_+)$$

with $\tilde{f} = f + \Delta\tilde{\phi}$.
In practice, it may be useful to integrate by parts the term containing $\Delta\tilde{\phi}$.

MIMESIS

Correction by multiplying

We will considering

$$\tilde{u} = \tilde{\phi} C$$

We want to find $C : \Omega \to \mathbb{R}^d$ solution to the problem

$$\begin{cases} -\Delta(\tilde{\phi} C) = f, & \text{on } \Omega, \\ C = 1, & \text{on } \Gamma. \end{cases} \tag{$\mathcal{C}_\times$}$$

In the non-homogeneous case, it is important to impose the boundary conditions either by the direct method or by the dual method.

MIMESIS

Correction by multiplying (elevated problem)

We introduced an initial modified problem : Find $\hat{u} : \Omega \to \mathbb{R}^d$ such that

$$\begin{cases} -\Delta \hat{u} = f, & \text{in } \Omega, \\ \hat{u} = g + m, & \text{on } \Gamma, \end{cases} \qquad (\mathcal{P}^{\mathcal{M}})$$

with $\hat{u} = u + m$ and m a constant.

We then apply the multiplication correction on the elevated problem by considering

$$\hat{\phi} = \tilde{\phi} + m$$

and so we look for $C : \Omega \to \mathbb{R}^d$ solution to the problem

$$\begin{cases} -\Delta(\hat{\phi} C) = f, & \text{in } \Omega, \\ C = 1, & \text{on } \Gamma. \end{cases} \qquad (\mathcal{C}_{\times}^{\mathcal{M}})$$

In the case of this correction, it is important to impose the boundary conditions either by the direct method, or by the dual method.

Theoretical results

Here, we're interested by $(\mathcal{C}_{\times}^{\mathcal{M}})$ with standard FEM.

➞ We can prove the following property:

$$||\hat{u_{ex}} - \hat{u_h}||_0 \leq ch^{k+1}||\hat{\phi}||_\infty \, |C|_{k+1}$$

with $\hat{u_{ex}} = u_{ex} + m$ the exact solution of $(\mathcal{P}^{\mathcal{M}})$, $\hat{u_h}$ the solution obtained of $(\mathcal{C}_{\times}^{\mathcal{M}})$ such that $\hat{u_h} = \hat{\phi} C_h$ with $\hat{\phi} = \tilde{\phi} + m = u_{ex} + \epsilon P + m$.

➞ With the previous property, we can show that, with m sufficiently large, the error no longer depends on the solution but only on the perturbation P :

$$\left|\left|\frac{\hat{u_{ex}}}{\hat{\phi}} - C_h\right|\right|_{0,\Omega} \leq ch^{k+1}\epsilon \, ||P''||_{0,\Omega}$$

➞ We can also prove that when m tends to infinity, the solution obtained with multiplication correction on an elevated problem $(\mathcal{C}_{\times}^{\mathcal{M}})$ converges to the solution obtained with correction by adding $(\mathcal{C}_{+})$.

MIMESIS

Correction

Problems considered

Methods considered

Numerical results

MIMESIS

Correction on exact solution

We consider the trigonometric solution on the square ($n_{vert} = 100$) and

$$\tilde{\phi} = u_{ex} \in \mathbb{P}_{10}$$

Results with FEM :

Homogeneous case :

	FEM	Corr_add	Corr_add_IPP	Corr_mult
f = 1	2.61e-03	5.16e-11	1.28e-13	2.48e-13
f = 2	1.04e-02	1.43e-11	1.27e-13	2.49e-13
f = 3	2.34e-02	8.07e-12	1.26e-13	2.49e-13
f = 4	4.12e-02	1.28e-11	1.24e-13	2.49e-13

Non homogeneous case :

	FEM	Corr_add	Corr_add_IPP	Corr_mult
f = 1	2.30e-03	4.74e-11	1.10e-13	1.72e-13
f = 2	9.53e-03	1.79e-11	1.15e-13	2.16e-13
f = 3	2.19e-02	1.19e-11	1.17e-13	2.29e-13
f = 4	3.88e-02	8.32e-12	1.18e-13	2.35e-13

Results with ϕ-FEM :

Homogeneous case :

	PhiFEM	Corr_add	Corr_add_IPP	Corr_mult
f = 1	1.84e-03	2.21e-11	2.21e-13	3.08e-13
f = 2	1.44e-02	8.80e-12	4.77e-12	4.80e-12
f = 3	3.69e-02	1.30e-10	1.29e-10	1.73e-10
f = 4	6.84e-02	1.27e-09	1.27e-09	2.02e-09

Non homogeneous case :

	PhiFEM	Corr_add	Corr_add_IPP	Corr_mult
f = 1	1.20e-04	1.42e-11	2.96e-13	2.02e-11
f = 2	5.74e-04	9.83e-12	4.45e-12	2.90e-11
f = 3	1.34e-03	1.16e-10	1.15e-10	1.78e-10
f = 4	2.39e-03	1.13e-09	1.14e-09	1.46e-09

Correction on disturbed solution

We consider the trigonometric solution on the square ($n_{vert} = 100$) and

$$\tilde{\phi} = u_{ex} + \epsilon P \in \mathbb{P}_{10}$$

where u_{ex} the exact solution to the problem, P the perturbation and ϵ a real number (amplitude of P).

We will choose to consider P as being of the same form as our exact solution defined by the parameters (S_p, f_p, φ_p) with $\varphi_p = 0$.

TO DO !

Correction on disturbed solution - Elevated Problem

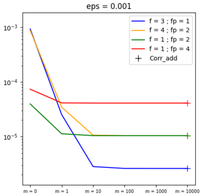
Try the correction by multiplying on the elevated problem with FEM and ϕ-FEM :

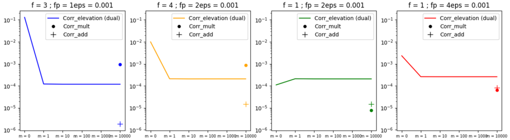
Standard FEM method :

		m = 0	m = 1	m = 10	m = 100	m = 1000	m = 10000	Corr_add
f > fp	f = 3 ; fp = 1	9.40e-04	2.54e-05	2.82e-06	2.61e-06	2.61e-06	2.61e-06	2.61e-06
	f = 4 ; fp = 2	8.82e-04	3.47e-05	1.06e-05	1.04e-05	1.04e-05	1.04e-05	1.04e-05
f < fp	f = 1 ; fp = 2	3.98e-05	1.13e-05	1.04e-05	1.04e-05	1.04e-05	1.04e-05	1.04e-05
	f = 1 ; fp = 4	7.39e-05	4.15e-05	4.12e-05	4.12e-05	4.12e-05	4.12e-05	4.12e-05

Φ−FEM method :

		Corr_mult	m = 0	m = 1	m = 10	m = 100	m = 1000	m = 10000	Corr_add
f > fp	f = 3 ; fp = 1	9.41e-04	1.30e-01	1.21e-04	1.19e-04	1.19e-04	1.19e-04	1.19e-04	1.86e-06
	f = 4 ; fp = 2	8.86e-04	9.99e-03	2.09e-04	2.07e-04	2.07e-04	2.07e-04	2.07e-04	1.50e-05
f < fp	f = 1 ; fp = 2	7.67e-06	1.10e-04	2.10e-04	2.07e-04	2.07e-04	2.07e-04	2.07e-04	1.51e-05
	f = 1 ; fp = 4	6.38e-05	2.33e-03	2.61e-04	2.60e-04	2.60e-04	2.60e-04	2.60e-04	8.11e-05





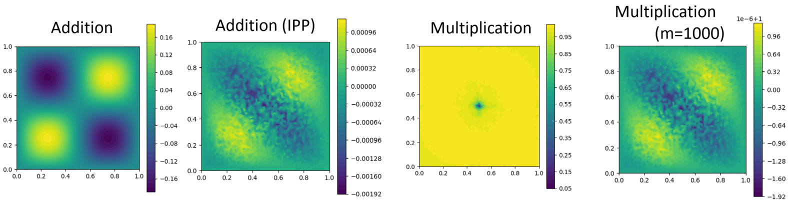
MIMESIS

Correction on a ϕ-FEM solution

We consider the trigonometric solution on the square and

$$\tilde{\phi} = u_{\phi-FEM} \in \mathbb{P}_2$$

where $u_{\phi-FEM}$ is the solution obtained with ϕ-FEM ($n_{vert} = 32$).
We will correct this solution with the different correction methods by using FEM.
We obtain the following correction terms ($\tilde{C}$ for addition and C for multiplication) :

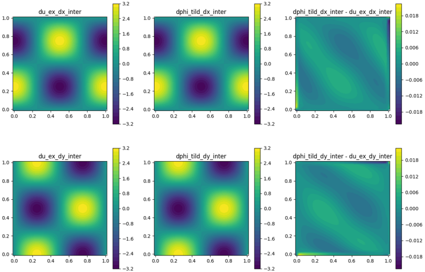


	Add	Add (IPP)	Mult	Mult (m=1000)
L2_rel	3.64e-01	1.03e-03	1.91e-03	1.03e-03

MIMESIS

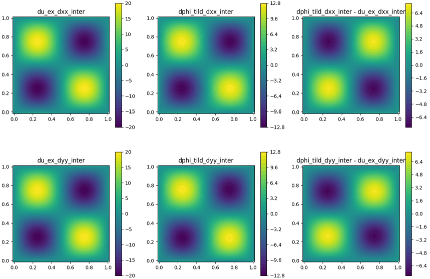
Correction on a ϕ-FEM solution - Derivatives

We compute the first and second derivatives of $\tilde{\phi}$ according to x and y.



1st derivatives

2nd derivatives



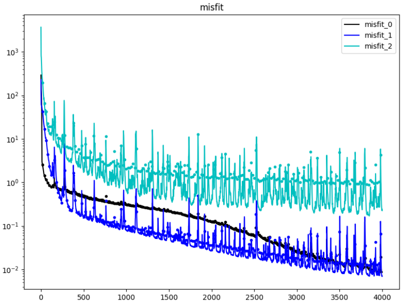
We can see that the second derivatives are quite far from the true derivatives.

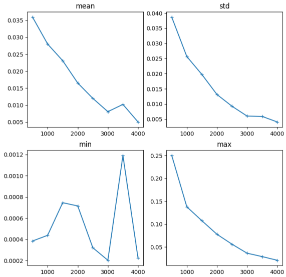
Correction on a FNO prediction I

We consider the unknown solution on the circle (with f Gaussian) and $\tilde{\phi} = u_{FNO} \in \mathbb{P}_2$ where u_{FNO} is the FNO prediction ($n_{vert} = 32$).

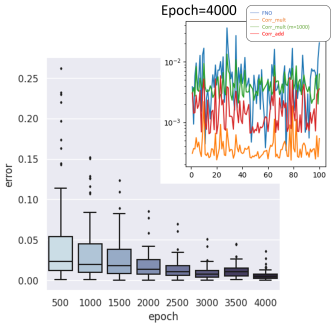
We will apply the different correction methods on the FNO prediction (of a test sample of size $n_{test} = 100$).

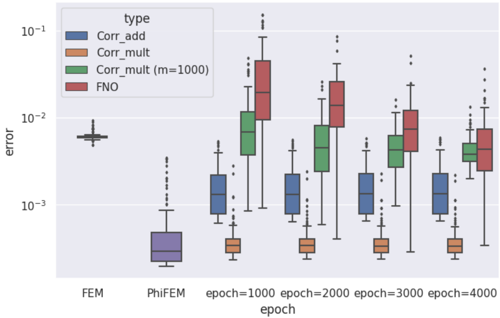
Training on 4000 epochs (with bs=64 and lr=0.01) :





Correction on a FNO prediction II





increase the degree of the $\mathbb{P}_k$ space ?

interpolation of $\tilde{\phi}$: decomposition into a series of Legendre polynomials $\Rightarrow$ analytical expression of the solution valid at any point of $\Omega \Rightarrow$ application of the correction in $\mathbb{P}^1 0$

MIMESIS

Correction with other networks I

Idea : using a neural network to predict a single solution at any point in the domain

input data = a collection of n_{pts} 2D points : $\{(x_i, y_i)\}_{i=1,\dots,n_{pts}}$

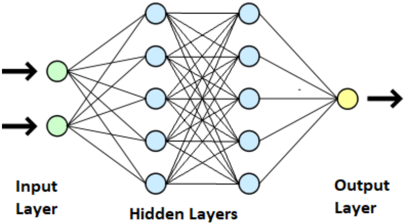
output = solution at each of these points : $\{u_i\}_{i=1,\dots,n_{dots}}$

with $u_i = \phi(x_i, y_i) w_i$ and $w_i = w_\theta(x_i, y_i)$.

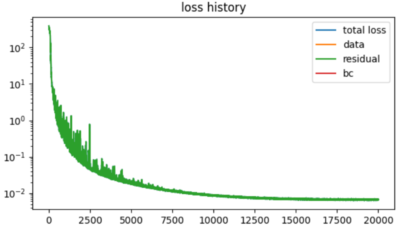
→ start with Fully-connected Multi-Layer Perceptron (MLP) but bad derivatives

→ try with a Physics-Informed Neural Networks (PINNs)

model = MLP with 6-layer network ($\{10, 20, 60, 60, 20, 10\}$)



Training over 20000 epochs (first half with lr=0.01, second half with lr=0.001) :



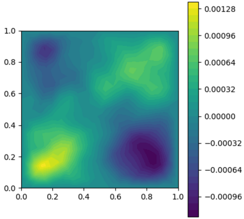
Correction with other networks II

We consider the trigonometric solution on the square and $\tilde{\phi} = u_{PINNs} \in \mathbb{P}_{10}$ where u_{PINNs} is the PINNs prediction ($n_{vert} = 32$) and we have

$$||u_{ex} - \tilde{\phi}||_{0,\Omega}^{(rel)} = 1.93e-3.$$

We will correct this solution with the correction by adding (with FEM and ϕ-FEM).

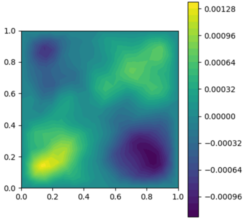
Standard FEM method :



$$||u_{ex} - \tilde{\phi}C||_{0,\Omega}^{(rel)} = 1.13e-4$$

divided by 246.60 (FEM : 2.80e-2)

ϕ-FEM method :



$$||u_{ex} - \tilde{\phi}C||_{0,\Omega}^{(rel)} = 1.27e-4$$

divided by 151.34 (ϕ-FEM : 1.92e-2)

Conclusion

Conclusion

- obtain numerical results on analytical solutions $\Rightarrow$ correction methods considered functional and theoretical results confirmed
- correction methods on the FNO predictions not satisfactory
- try to increase the degree of the solution :
  - Legendre and MLP not satisfactory
  - PINNs : reduction of the error made by conventional methods (FEM and ϕ-FEM) by a factor of around 100 (correction by addition).

Perspectives :

- try adding PINNs to the output of the FNO (add the PINNs as a layer output that would replace the decomposition into a series of polynomials) $\Rightarrow$ solution at any point in the domain
- carry out some documentation work to find more suitable models than the FNO
- consider more complex and time-varying geometries (such as 3D organ geometries)

Bibliography

[1] Allal Bedlaoui. *Les éléments finis : de la théorie à la pratique*. URL: https://www.academia.edu/36497995/Les_%C3%A9l%C3%A9ments_finis_de_la_th%C3%A9orie_%C3%A0_la_pratique (visited on 08/16/2023).

[2] Allal Bedlaoui. *Les éléments finis : de la théorie à la pratique*. URL: https://www.academia.edu/36497995/Les_%C3%A9l%C3%A9ments_finis_de_la_th%C3%A9orie_%C3%A0_la_pratique (visited on 08/16/2023).

[3] Haim Brezis. *Functional Analysis, Sobolev Spaces and Partial Differential Equations*. New York, NY: Springer New York, 2011. ISBN: 978-0-387-70913-0 978-0-387-70914-7. DOI: 10.1007/978-0-387-70914-7. URL: https://link.springer.com/10.1007/978-0-387-70914-7 (visited on 08/17/2023).

[4] Erik Burman. "Ghost penalty". In: *Comptes Rendus. Mathématique* 348.21 (2010). Number: 21-22, pp. 1217–1220. ISSN: 1778-3569. DOI: 10.1016/j.crma.2010.10.006. URL: http://www.numdam.org/articles/10.1016/j.crma.2010.10.006/ (visited on 07/26/2023).

[5] Erik Burman et al. "CutFEM: Discretizing geometry and partial differential equations". In: *International Journal for Numerical Methods in Engineering* 104.7 (2015). Number: 7 _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.4823, pp. 472–501. ISSN: 1097-0207. DOI: 10.1002/nme.4823. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.4823 (visited on 07/26/2023).

[6] Stéphane Cotin et al. *ϕ-FEM: an efficient simulation tool using simple meshes for problems in structure mechanics and heat transfer*.

[7] Michel Duprez, Vanessa Lleras, and Alexei Lozinski. “ϕ-FEM: an optimally convergent and easily implementable immersed boundary method for particulate flows and Stokes equations”. In: *ESAIM: Mathematical Modelling and Numerical Analysis* 57.3 (May 2023). Number: 3, pp. 1111–1142. ISSN: 2822-7840, 2804-7214. DOI: 10.1051/m2an/2023010. URL: https://www.esaim-m2an.org/10.1051/m2an/2023010 (visited on 07/26/2023).

[8] Michel Duprez, Vanessa Lleras, and Alexei Lozinski. “A new ϕ-FEM approach for problems with natural boundary conditions”. In: *Numerical Methods for Partial Differential Equations* 39.1 (2023). Number: 1 _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/num.22878, pp. 281–303. ISSN: 1098-2426. DOI: 10.1002/num.22878. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/num.22878 (visited on 07/26/2023).

[9] Michel Duprez and Alexei Lozinski. “ϕ-FEM: A Finite Element Method on Domains Defined by Level-Sets”. In: *SIAM Journal on Numerical Analysis* 58.2 (Jan. 2020). Number: 2 Publisher: Society for Industrial and Applied Mathematics, pp. 1008–1028. ISSN: 0036-1429. DOI: 10.1137/19M1248947. URL: https://epubs.siam.org/doi/10.1137/19M1248947 (visited on 07/26/2023).

[10] Zongyi Li et al. *Neural Operator: Graph Kernel Network for Partial Differential Equations*. Mar. 6, 2020. DOI: 10.48550/arXiv.2003.03485. arXiv: 2003.03485[cs,math,stat]. URL: http://arxiv.org/abs/2003.03485 (visited on 08/17/2023).

[11] Zongyi Li et al. *Fourier Neural Operator for Parametric Partial Differential Equations*. Issue: arXiv:2010.08895. May 16, 2021. DOI: 10.48550/arXiv.2010.08895. arXiv: 2010.08895[cs,math]. URL: http://arxiv.org/abs/2010.08895 (visited on 07/26/2023).

[12] Zongyi Li et al. *Fourier Neural Operator with Learned Deformations for PDEs on General Geometries*. July 11, 2022. DOI: 10.48550/arXiv.2207.05209. arXiv: 2207.05209[cs,math]. URL: http://arxiv.org/abs/2207.05209 (visited on 08/17/2023).

[13] Zongyi Li et al. *Physics-Informed Neural Operator for Learning Partial Differential Equations*. July 29, 2023. DOI: 10.48550/arXiv.2111.03794. arXiv: 2111.03794[cs,math]. URL: http://arxiv.org/abs/2111.03794 (visited on 08/17/2023).

[14] *Méthodes numériques pour les EDP*. URL: https://feelpp.github.io/csmi-edp/#/ (visited on 08/16/2023).

[15] Nicolas Moës and Ted Belytschko. "X-FEM, de nouvelles frontières pour les éléments finis". In: *Revue Européenne des Éléments Finis* 11.2 (Jan. 2002). Number: 2-4, pp. 305–318. ISSN: 1250-6559. DOI: 10.3166/reef.11.305-318. URL: https://www.tandfonline.com/doi/full/10.3166/reef.11.305-318 (visited on 07/26/2023).

[16] Alfio Quarteroni et al. *Méthodes numériques: algorithmes, analyse et applications*. Ed. revue et augmentée. Milan: Springer, 2007. ISBN: 978-88-470-0495-5.