

Résultats

Table des matières

1 Étape 1 : Apprentissage d'une fonction Levelset ϕ	2
1.1 Génération des Inputs	2
1.2 Trouver une levelset	2
1.3 Résultats	3
2 Étape 2 : Résolution de Poisson	5
3 Étape 3 : Correction par addition	9
3.1 Avec FEM	9
3.2 Avec ϕ -FEM	11

1 Étape 1 : Apprentissage d'une fonction Levelset ϕ

1.1 Génération des Inputs

On considère une courbe paramétrique $c(t)$ définissant un surface \mathcal{S} (en 2D). En pratique, nous allons considérer les 3 formes définies par les courbes paramétriques, pour $t \in [0, 1]$:

$$c_{circle}(t) = \begin{pmatrix} x_0 + r \cos(2\pi t) \\ y_0 + r \sin(2\pi t) \end{pmatrix}, (x_0, y_0) = (0.5, 0.5), r = \sqrt{2}/4$$

$$c_{bean}(t) = \begin{pmatrix} \sin(2\pi t) \times (\sin(2\pi t)^a + \cos(2\pi t)^b) \\ -\cos(2\pi t) \times (\sin(2\pi t)^a + \cos(2\pi t)^b) \end{pmatrix}, a = 3, b = 5$$

$$c_{pumpkin}(t) = \begin{pmatrix} \cos(2\pi t) + 0.3 \cos(6\pi t) + 0.1 \cos(10\pi t) \\ \sin(2\pi t) + 0.3 \sin(6\pi t) + 0.1 \sin(10\pi t) \end{pmatrix}$$

A partir de ces différentes courbes, on va générer un nombre n_{bc} de points sur le surface \mathcal{S} et on peut facilement calculer les normales à la surface en les points considérés. En considérant la dérivée $c'(t)$ de la courbe c et $c'^\perp(t)$ définie comme la rotation de c' d'un angle de $-\pi/2$, c'est-à-dire pour $\theta = \pi/2$, on a :

$$c'^\perp(t) = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix} c'(t)$$

et ainsi

$$n(t) = \frac{c'^\perp(t)}{\|c'^\perp(t)\|}$$

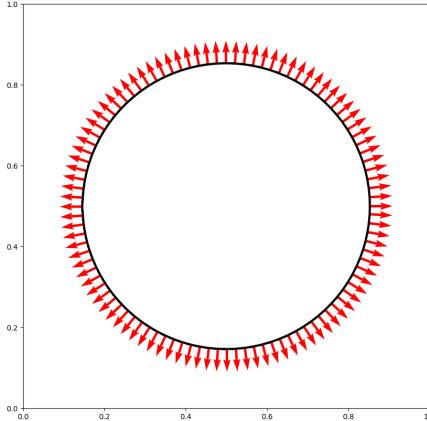


FIGURE 1 – Cercle.

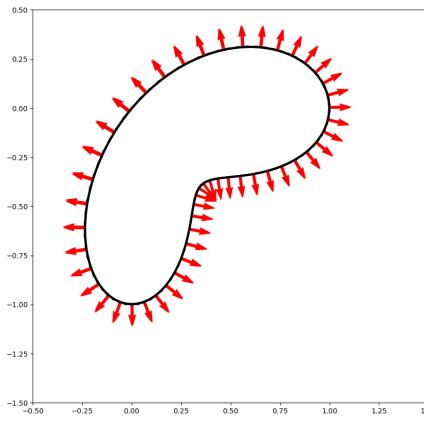


FIGURE 2 – "Haricot".

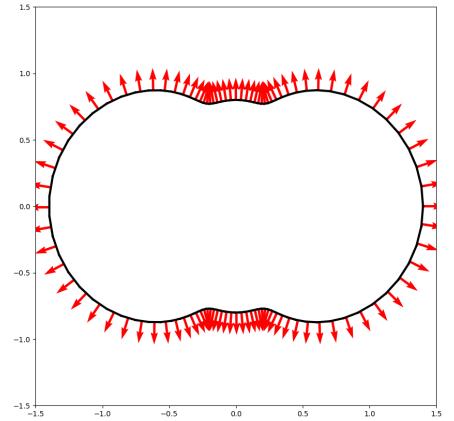


FIGURE 3 – "Citrouille".

Remarque. Ce n'est que pour une question de simplicité qu'on considère une courbe paramétrique. En pratique, on peut partir uniquement d'un ensemble de points.

1.2 Trouver une levelset

Pour déterminer ϕ , on s'est basé sur l'article [1] dont l'objectif est la génération d'une distance signée par la résolution de l'équation Eikonale dans le but de générer le squelette de géométrie complexe. Comme notre objectif n'est pas le même on a du remanier ce qui était présenté dans le bus que la fonction apprise soit utilisable dans l'apprentissage d'une solution à un problème donné. On a alors considéré les loss suivantes présentées dans l'article afin d'obtenir une solution à l'équation Eikonal :

$$\mathcal{L}_{eik} = \int_{\mathbb{R}^2} (1 - \|\nabla \phi(p)\|)^2 dp$$

$$\mathcal{L}_{dir} = \int_{\partial\Omega} \phi^2 dp$$

$$\mathcal{L}_{neu} = \int_{\partial\Omega} 1 - \frac{n(p) \cdot \nabla\phi(p)}{\|n(p)\| \| \nabla\phi(p)\|} dp$$

Dans notre cas, comme l'objectif est différent, on ne cherchera pas à approcher la sdf de la géométrie, on souhaite en fait déterminer une fonction levelset qui a certaines bonnes propriétés qui permettraient de l'utiliser par la suite. En particulier, on cherchera à limiter l'explosion des dérivées secondes et pour cela, on ajoutera la loss suivante :

$$\mathcal{L}_{lap} = \int_{\mathbb{R}^2} \Delta\phi^2 dp$$

Pour la résolution du problème, on considérera un PINNs dont le modèle sera défini par un simple MLP et on cherchera $\phi_\theta(x, y)$ avec la loss définie par

$$\mathcal{L} = w_{eik}\mathcal{L}_{eik} + w_{dir}\mathcal{L}_{dir} + w_{neu}\mathcal{L}_{neu} + w_{lap}\mathcal{L}_{lap}$$

Remarque. Pour l'instant, on entraîne un modèle par géométrie mais dans la suite on cherchera à construire un modèle paramétrique.

1.3 Résultats

Circle :

layers	activation	lr	decay	w_eik	w_dir	w_neu	w_lap	n_epochs	n_collocations	n_bc_collocations
[64, 64, 64, 64, 64, 64]	sine	0,001	0,99	100	1000	100	1	10000	4000	2000

FIGURE 4 – Configuration.

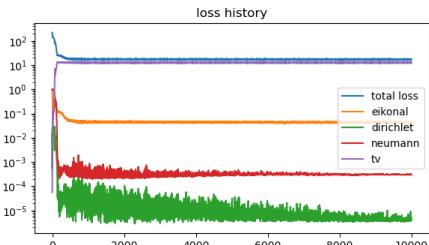


FIGURE 5 – Loss (tv=lap).

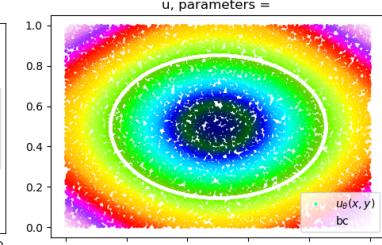


FIGURE 6 – Levelset ϕ_θ .

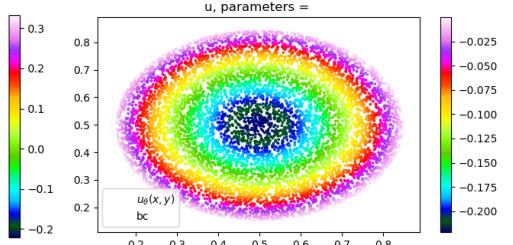


FIGURE 7 – Levelset $\phi_\theta < 0$.

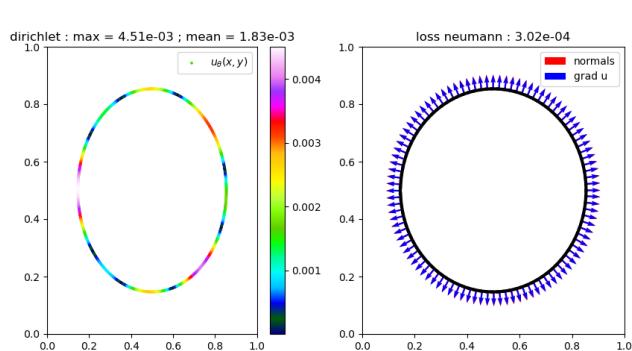


FIGURE 8 – Loss (tv=lap).

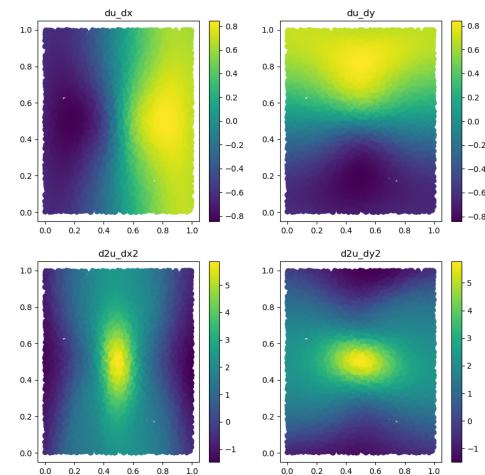


FIGURE 9 – Levelset ϕ_θ .

Bean :

layers	activation	lr	decay	w_eik	w_dir	w_neu	w_lap	n_epochs	n_collocations	n_bc_collocations
[64, 64, 64, 64, 64, 64]	sine	0,007	0,99	100	10000	100	1	10000	4000	2000

FIGURE 10 – Configuration.

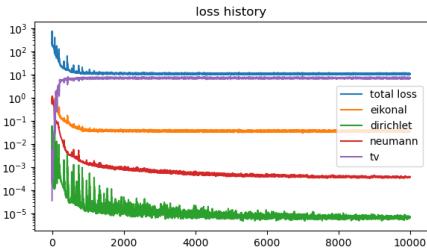


FIGURE 11 – Loss (tv=lap).

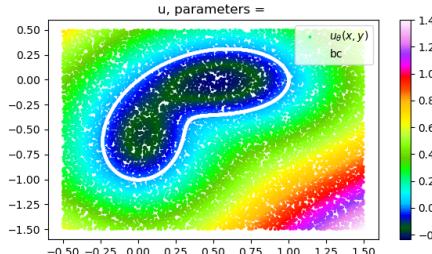


FIGURE 12 – Levelset ϕ_θ .

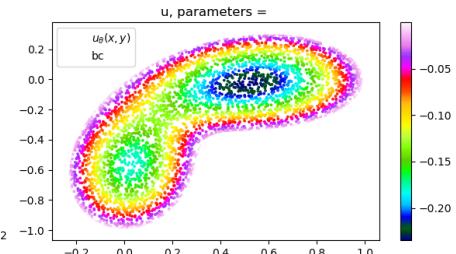


FIGURE 13 – Levelset $\phi_\theta < 0$.

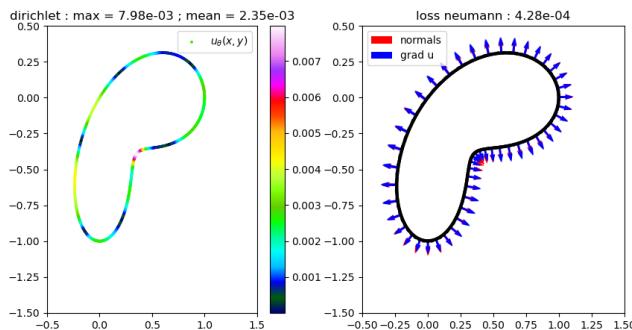


FIGURE 14 – Loss (tv=lap).

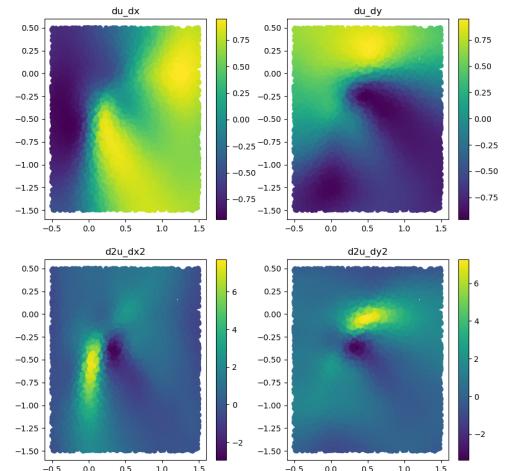


FIGURE 15 – Levelset ϕ_θ .

Pumpkin :

layers	activation	lr	decay	w_eik	w_dir	w_neu	w_lap	n_epochs	n_collocations	n_bc_collocations
[64, 64, 64, 64, 64, 64]	sine	0,007	0,99	100	10000	100	1	10000	4000	2000

FIGURE 16 – Configuration.

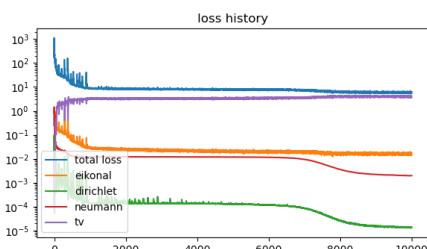


FIGURE 17 – Loss (tv=lap).

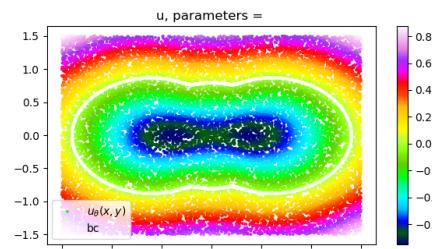


FIGURE 18 – Levelset ϕ_θ .

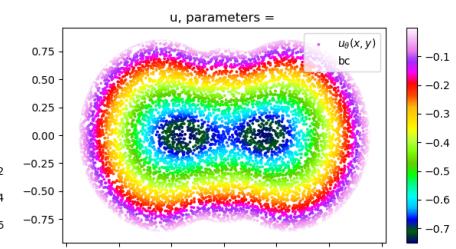


FIGURE 19 – Levelset $\phi_\theta < 0$.

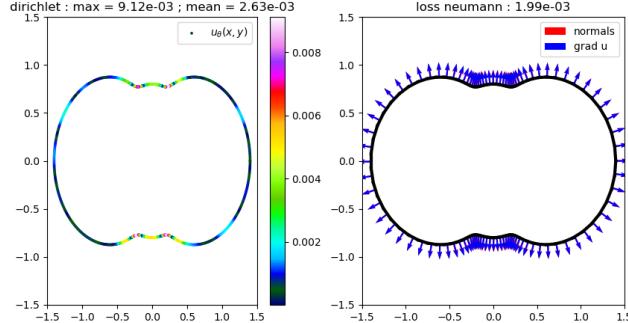


FIGURE 20 – Loss (tv=lap).

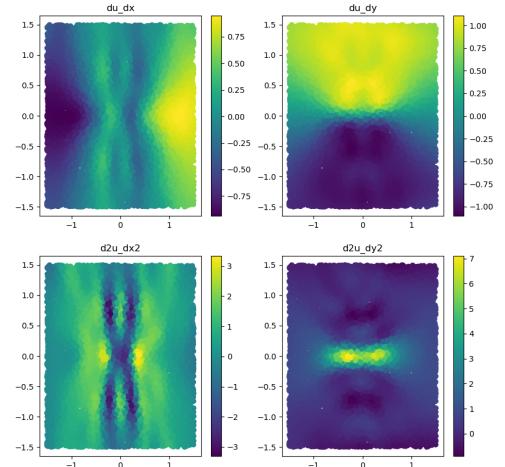


FIGURE 21 – Levelset ϕ_θ .

2 Étape 2 : Résolution de Poisson

On considère le problème suivant

$$\begin{cases} -\Delta u = 1 \text{ dans } \Omega \\ u = 0 \text{ sur } \partial\Omega \end{cases}$$

où Ω pourra représenter le cercle, le haricot ou la citrouille.

On cherchera à entraîner un nouveau PINNs à apprendre w tel que $u = \phi_\theta w$ avec ϕ_θ la levelset apprise précédemment.

Circle :

layers	activation	lr	decay	w_data	w_res	n_epochs	n_collocations	n_data
[64, 64, 64, 64, 64, 64]	sine	0,007	0,99	0	1	10000	2000	0

FIGURE 22 – Configuration.

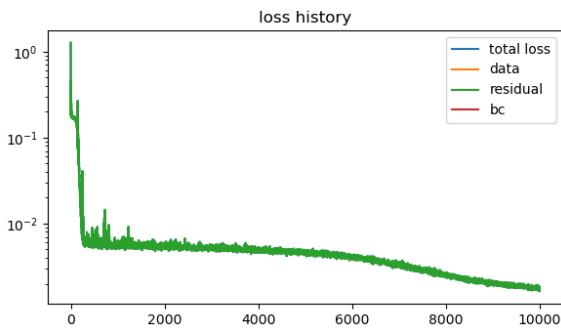


FIGURE 23 – Loss.

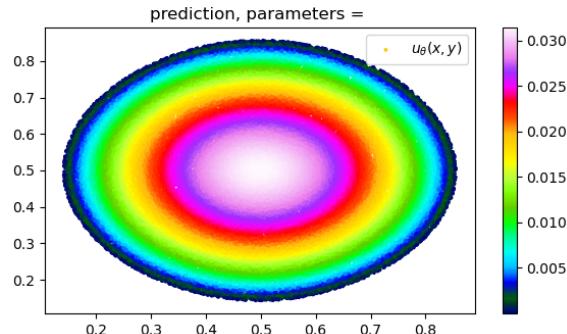


FIGURE 24 – Sol u_θ .

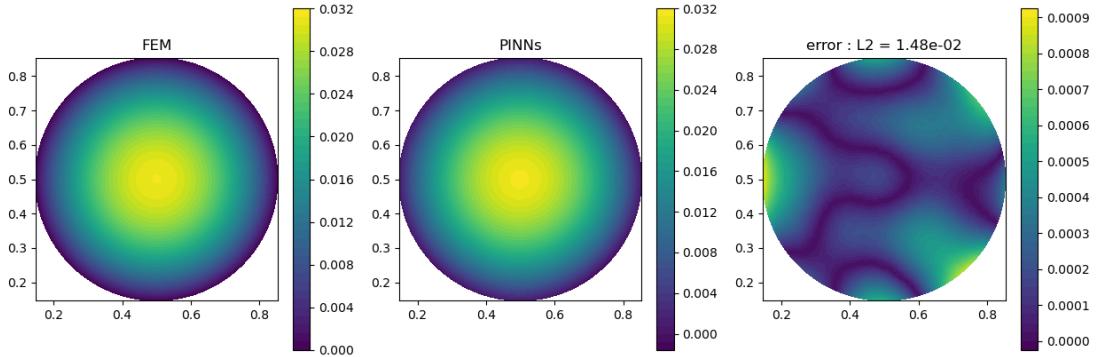


FIGURE 25 – Comparaison avec une solution FEM sur-rafinée.

Bean - 1 :

layers	activation	lr	decay	w_data	w_res	n_epochs	n_collocations	n_data
[20, 20, 20, 20]	sine	0,01	0,99	0	1	10000	2000	0

FIGURE 26 – Configuration.

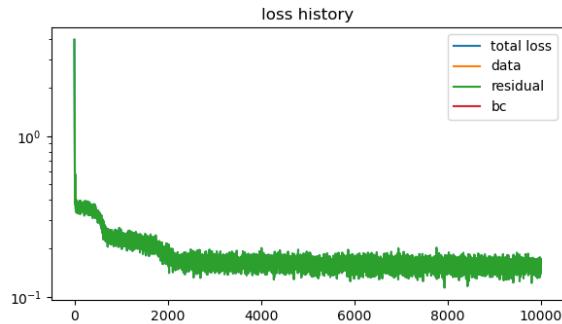


FIGURE 27 – Loss.

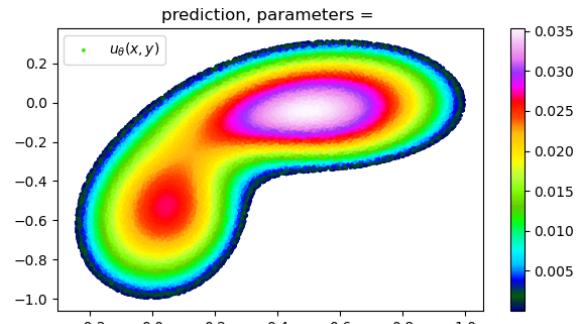


FIGURE 28 – Sol u_θ .

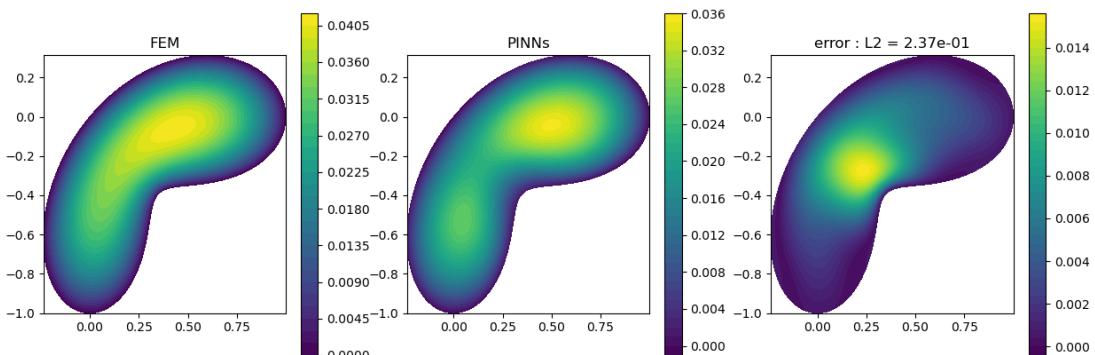


FIGURE 29 – Comparaison avec une solution FEM sur-rafinée.

Bean - 2 :

layers	activation	lr	decay	w_data	w_res	n_epochs	n_collocations	n_data
[64, 64, 64, 64, 64, 64]	sine	0,007	0,99	0	1	10000	2000	0

FIGURE 30 – Configuration.

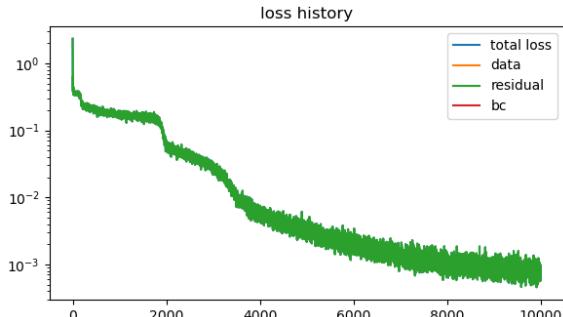


FIGURE 31 – Loss.

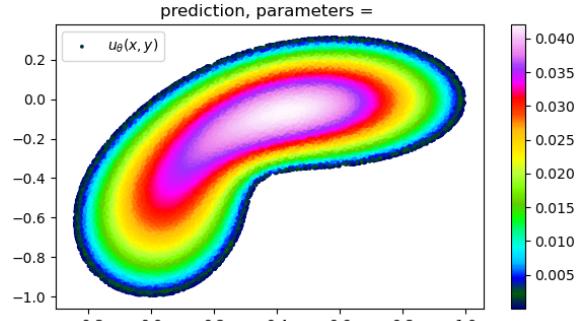


FIGURE 32 – Sol u_θ .

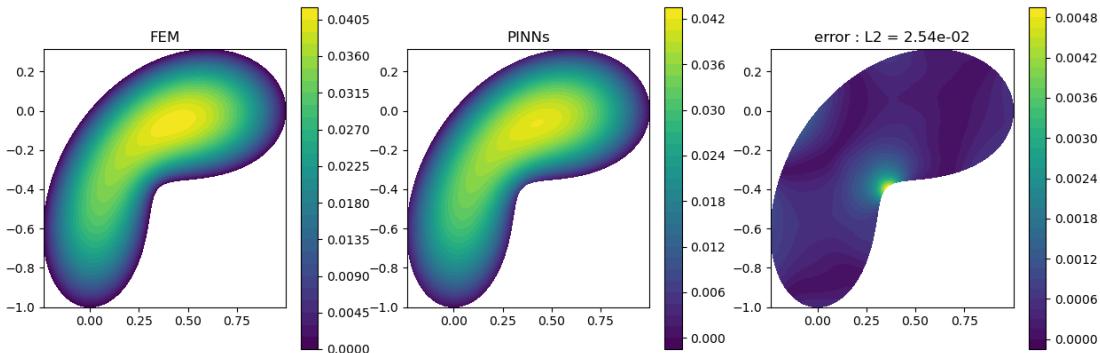


FIGURE 33 – Comparaison avec une solution FEM sur-rafinée.

Pumpkin :

layers	activation	lr	decay	w_data	w_res	n_epochs	n_collocations	n_data
[64, 64, 64, 64, 64, 64]	sine	0,007	0,99	0	1	10000	2000	0

FIGURE 34 – Configuration.

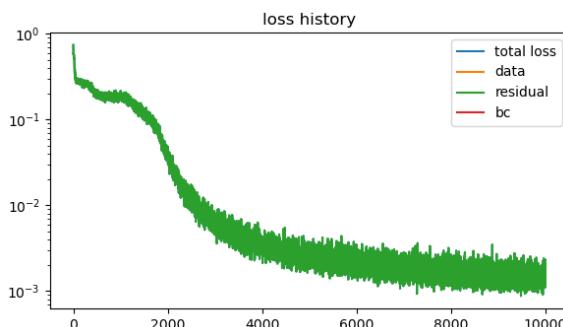


FIGURE 35 – Loss.

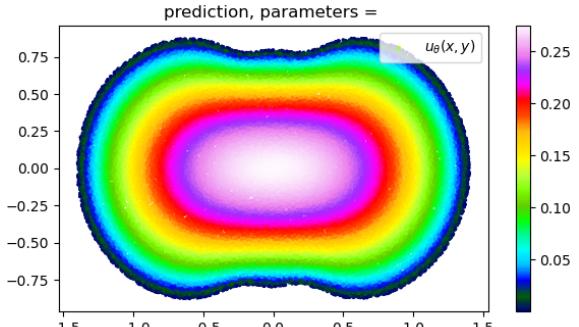


FIGURE 36 – Sol u_θ .

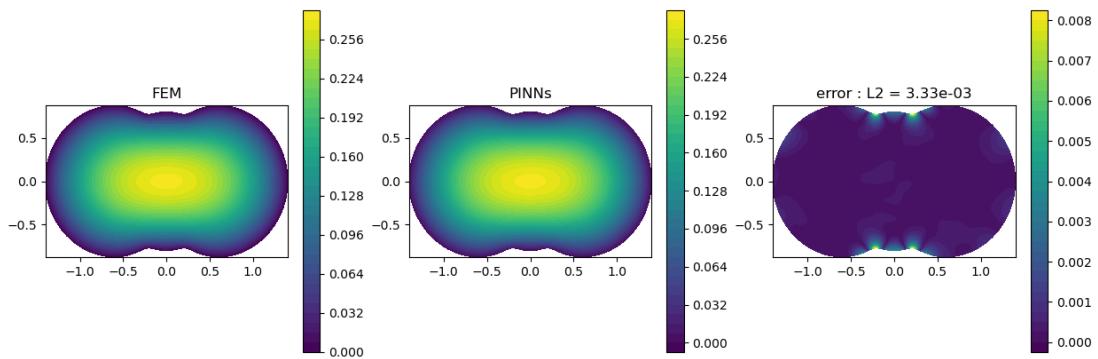


FIGURE 37 – Comparaison avec une solution FEM sur-rafinée.

3 Étape 3 : Correction par addition

3.1 Avec FEM

Circle :

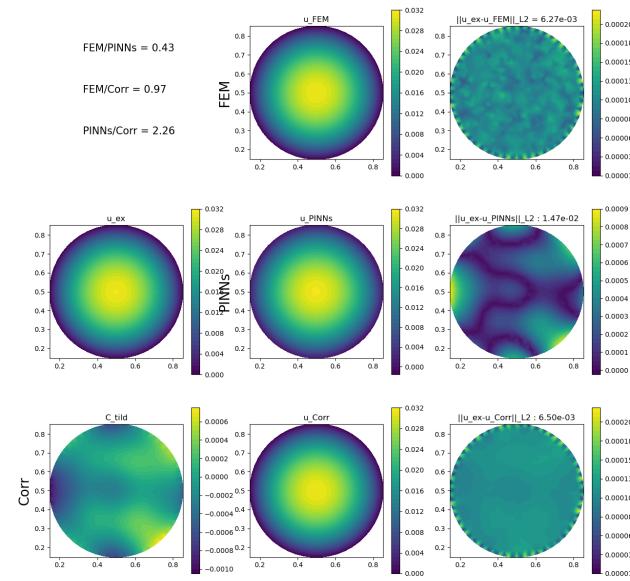


FIGURE 38 – Correction par addition avec FEM.

Bean - 1 :

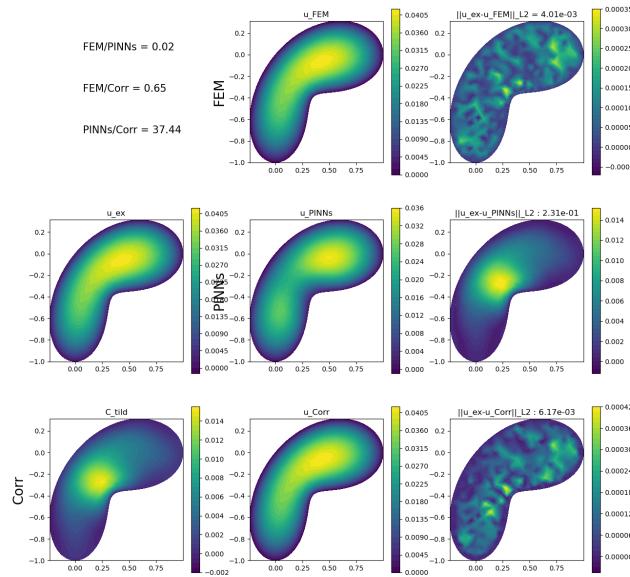


FIGURE 39 – Correction par addition avec FEM.

Bean - 2 :

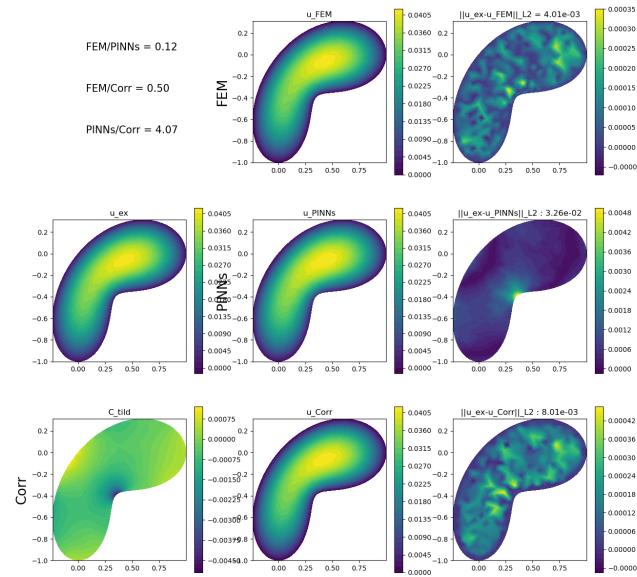


FIGURE 40 – Correction par addition avec FEM.

Pumpkin :

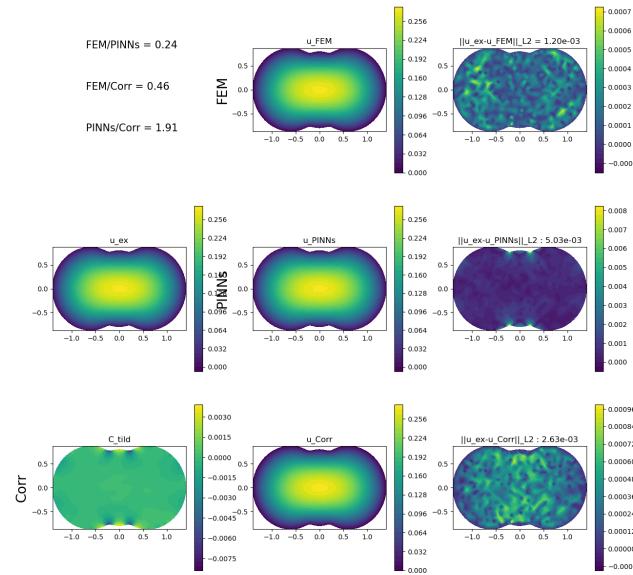


FIGURE 41 – Correction par addition avec FEM.

3.2 Avec ϕ -FEM

Circle :

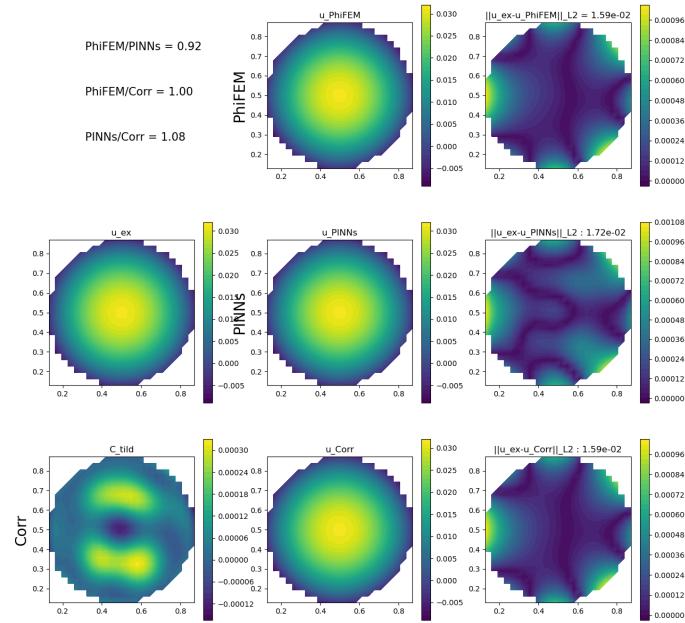


FIGURE 42 – Correction par addition avec ϕ -FEM.

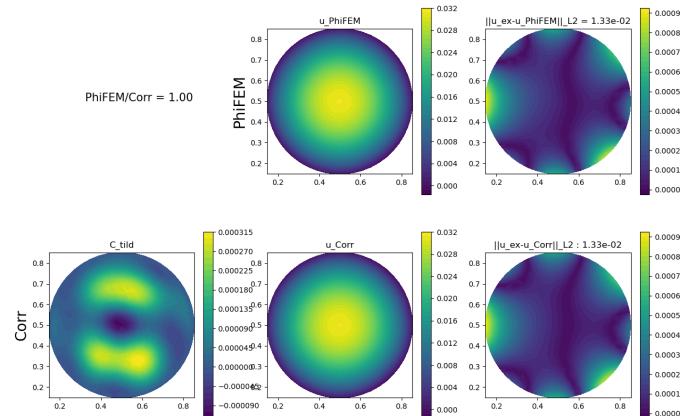


FIGURE 43 – Correction par addition avec ϕ -FEM (projété).

Bean - 1 :

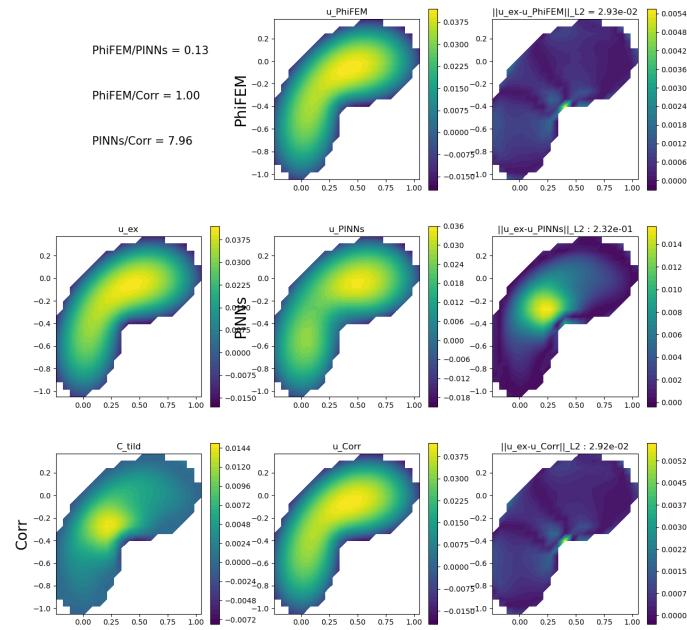


FIGURE 44 – Correction par addition avec ϕ -FEM.

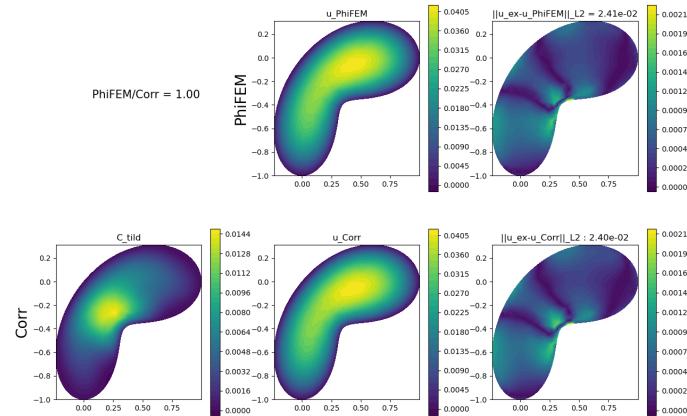


FIGURE 45 – Correction par addition avec ϕ -FEM (projété).

Bean - 2 :

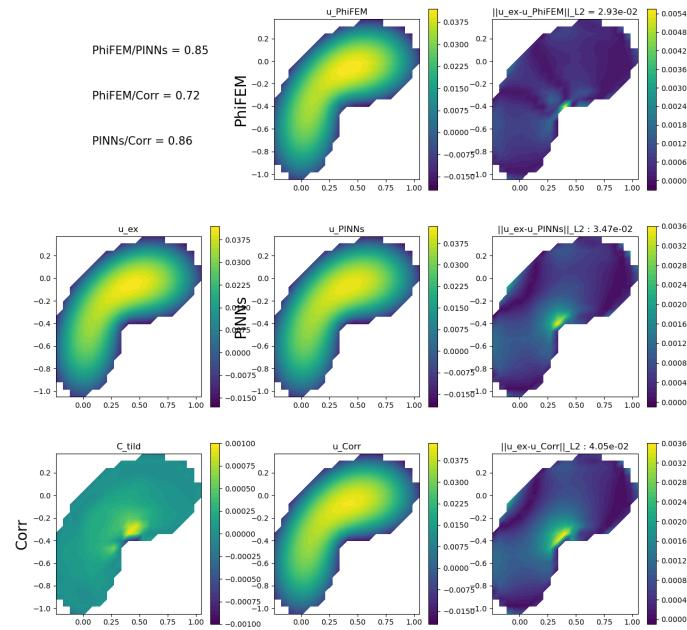


FIGURE 46 – Correction par addition avec ϕ -FEM.

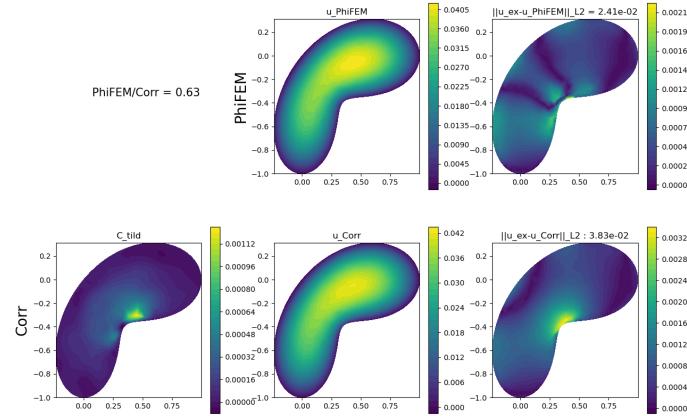


FIGURE 47 – Correction par addition avec ϕ -FEM (projété).

Pumpkin :

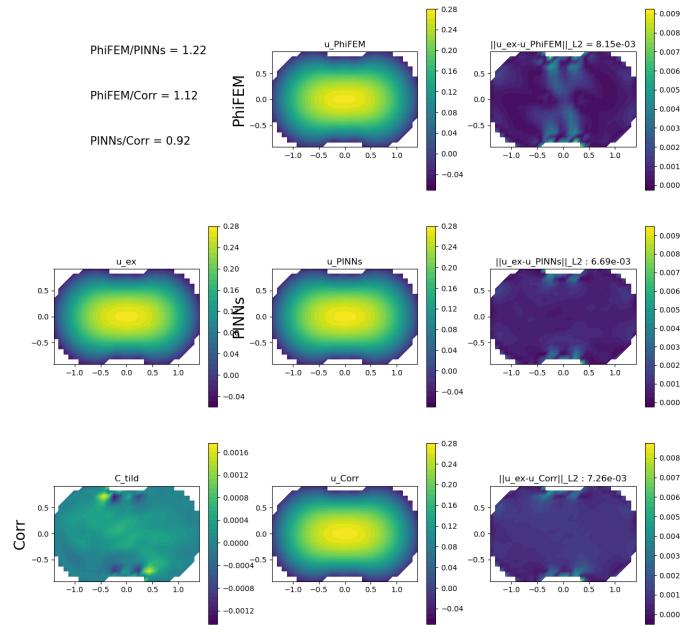


FIGURE 48 – Correction par addition avec ϕ -FEM.

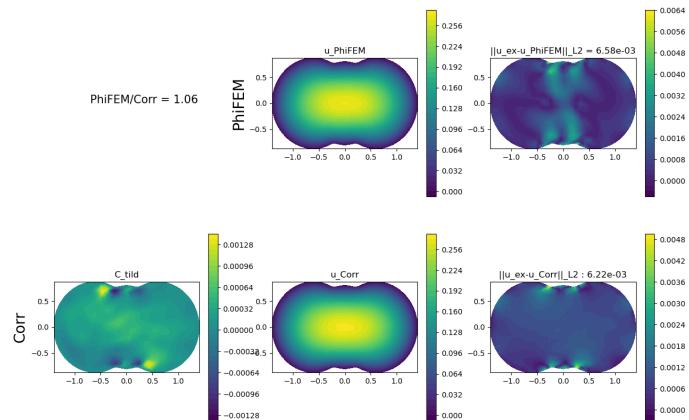


FIGURE 49 – Correction par addition avec ϕ -FEM (projété).

Références

- [1] Mattéo CLÉMOT et Julie DIGNE. “Neural skeleton : Implicit neural representation away from the surface”. In : *Computers and Graphics* 114 (2023), p. 368-378. ISSN : 0097-8493. DOI : <https://doi.org/10.1016/j.cag.2023.06.012>. URL : <https://www.sciencedirect.com/science/article/pii/S0097849323001085>.