TD 6

Tom Wozniak

13 octobre 2025

1 Optimisation

On travaille sur un nouveau code, qui est une optimisation de l'équation ADRS. Il repose sur le calcul de la minimisation d'une fonctionnelle J permettant de mesurer la différence entre une solution numérique et une solution donnée, via une méthode des moindres carrés. On commence par visualiser la fonction J en en faisant varier que deux paramètres et en gardant les autres fixés. On obtient alors une surface.

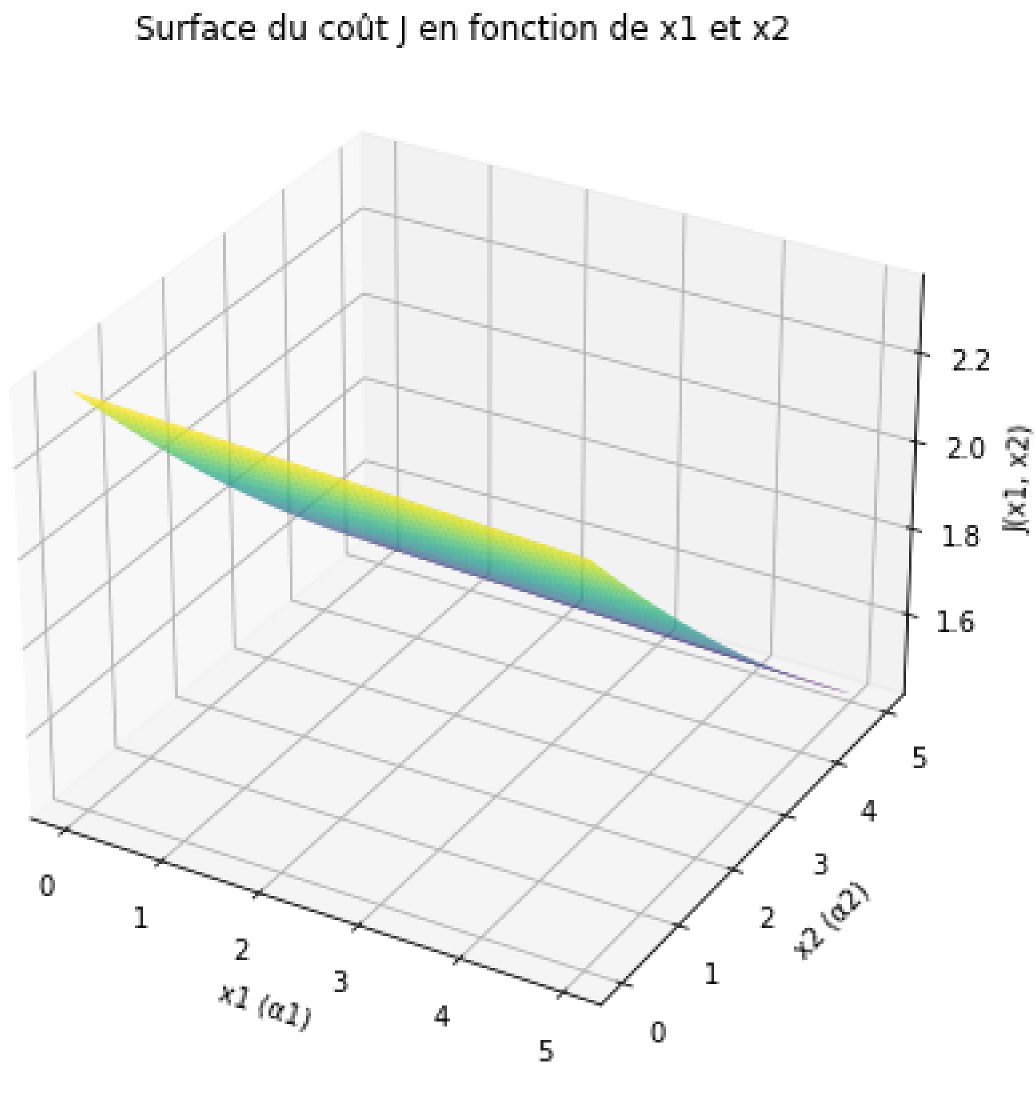

Figure 1 – La surface J

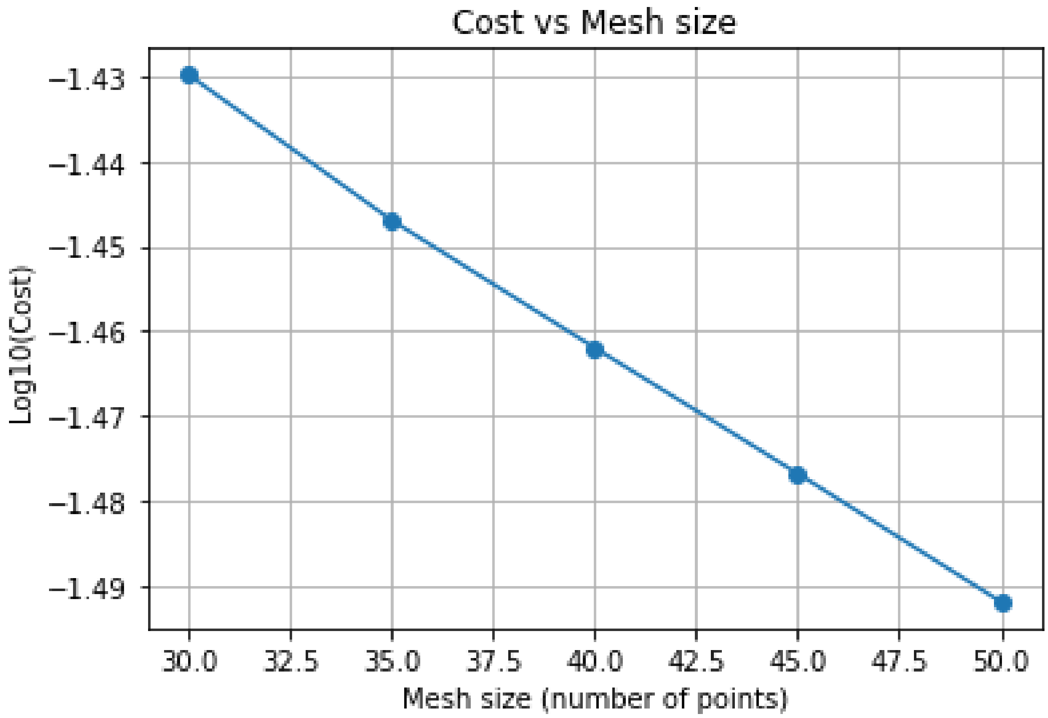

Figure 2 – Graphe montrant l'évolution de l'erreur par rapport à la taille du maillage

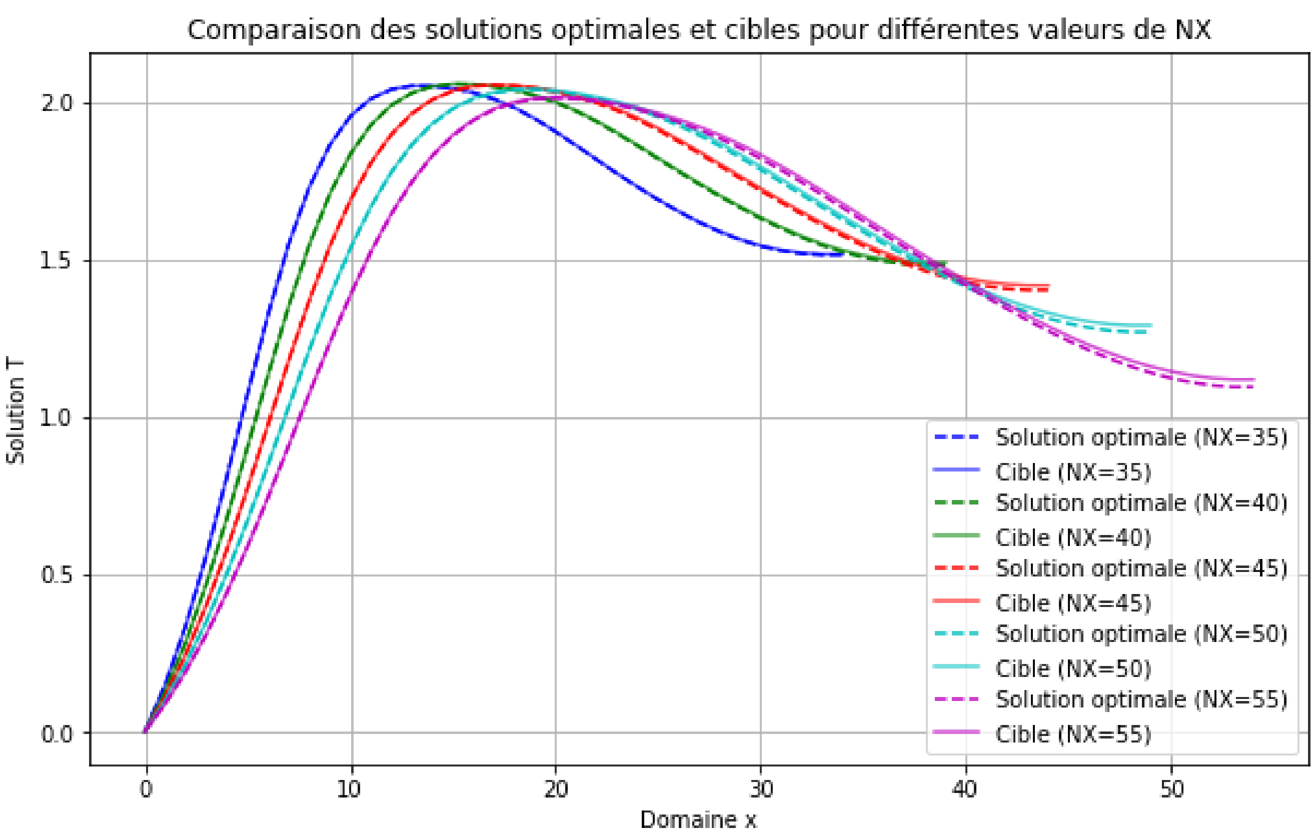

Figure 3 – Graphe comparant les solutions

On observe que les solutions élémentaires $u_{i,h}, u_{j,h}$ et $u_{0,h}$ sont calculées sur des maillages différents. Ainsi, les intégrales $A_{i,j}$ et B_i ne peuvent pas être calculées comme des produits scalaires sur un vecteur commun, il est donc nécessaire de faire une interpolation sur un maillage commun. De plus, la précision de l'intégration numérique de $A_{i,j}$ et B_i ne doit pas être supérieure à l'erreur d'interpolation. Nous avons modifié le code pour mettre ceci en place. Ensuite, nous comparons le contrôle optimal obtenu avec la solution sur maillage fixe $NX = 500$.

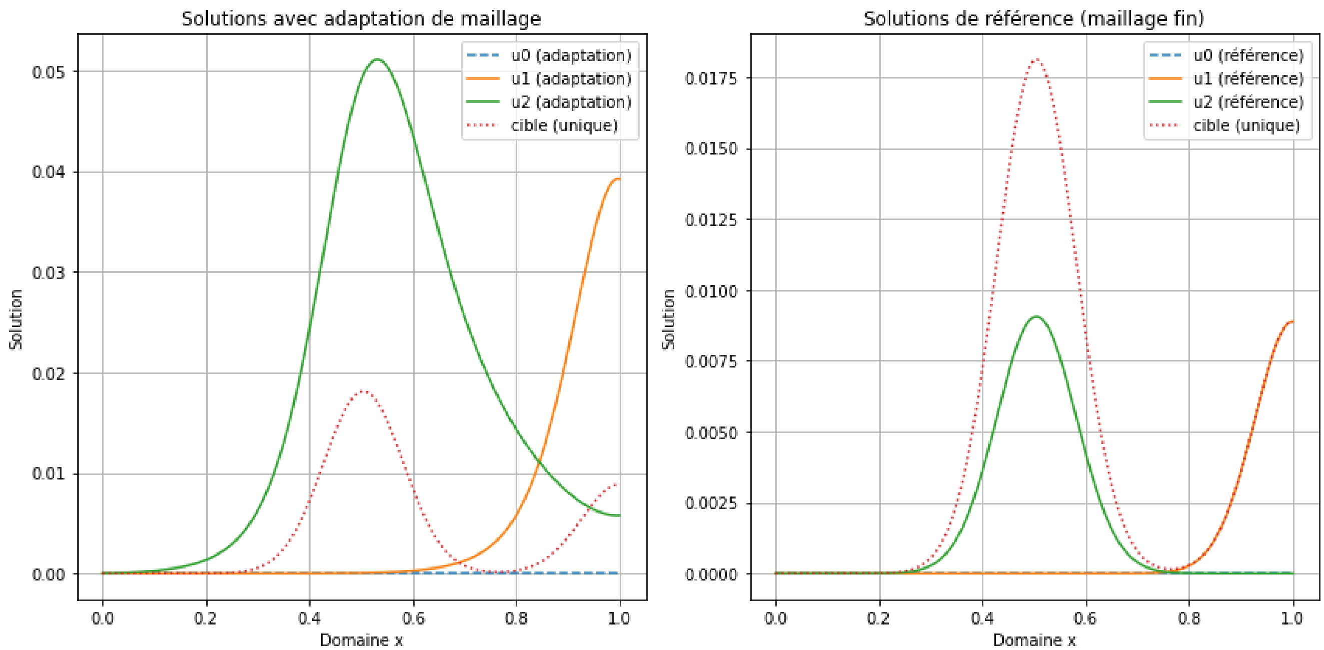

FIGURE 4 –

On observe des différences importantes entre les deux graphes, liées très probablement aux erreurs d'interpolation et à la différence du maillage.

2 L'algorithme

Nous avons ensuite implémenter via Python l'algorithme présenté dans le support du cours. On a ensuite introduit une boucle de raffinement pour observer comment la solution $X^*(h)$ évolue avec les itérations.

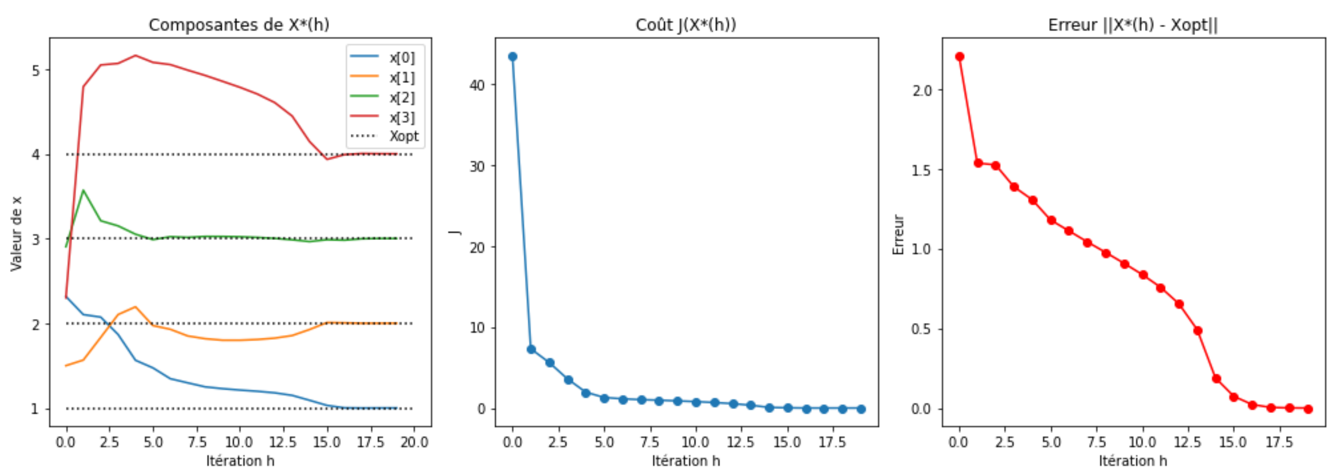

FIGURE 5 –