

Jeremy Uzan

Implémentation Python par la méthode de différence finie

(Partie programmation du devoir maison)

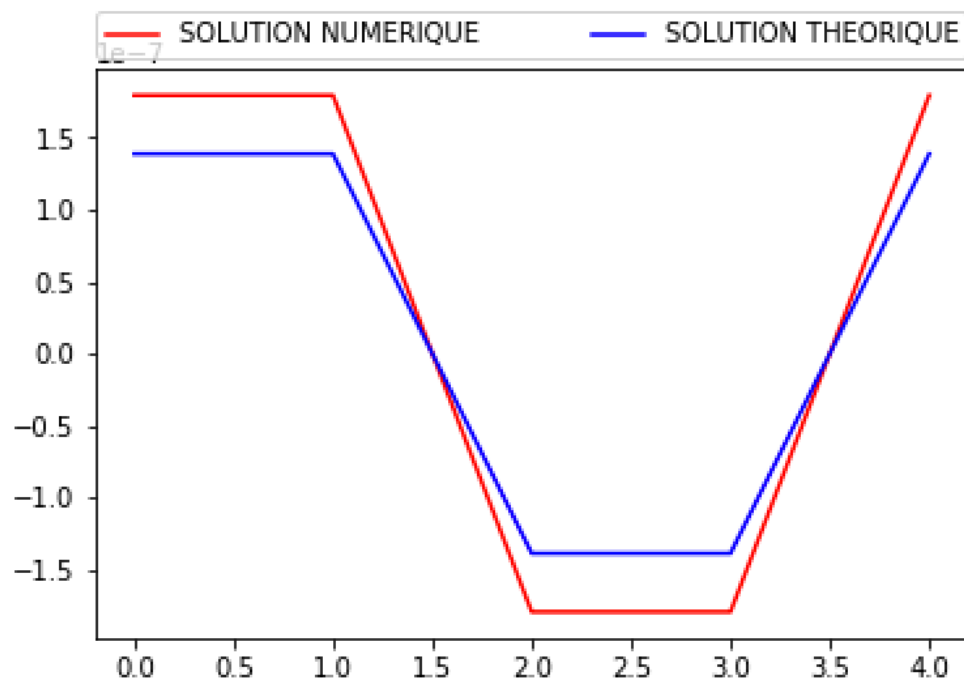
1.IMPLEMENTATION

- Depuis le terminal, aller vers le dossier contenant le fichier « DM_UZAN.py » et effectuez la commande
« python DM_UZAN.py » permettant de lancer python depuis le terminal
- POUR FAIRE AFFICHER LES GRAPHES POUR LE NOMBRE DE PAS DE GRILLE SUIVANT, IL SUFFIT DE FERMER LA PAGE DU GRAPHE QUI VIENT DE S’AFFICHER. LE SUIVANT S’AFFICHERA ALORS DIRECTEMENT.

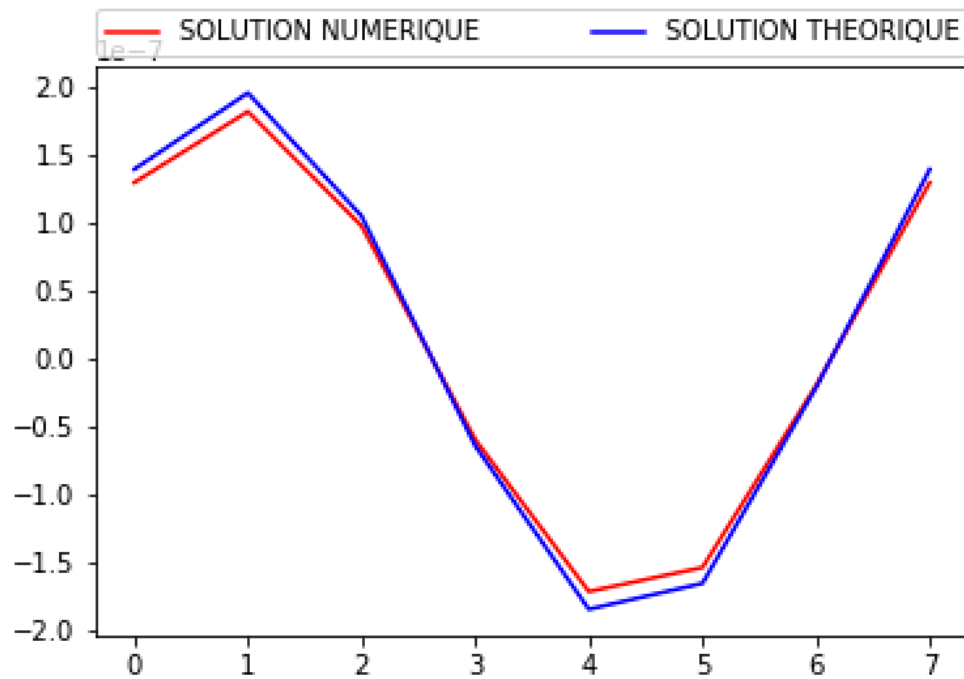
2.INTERPRETATION DES RESULTATS

Réponse et interprétation des résultats numériques obtenus
Question F) G) de la partie 2 du devoir maison

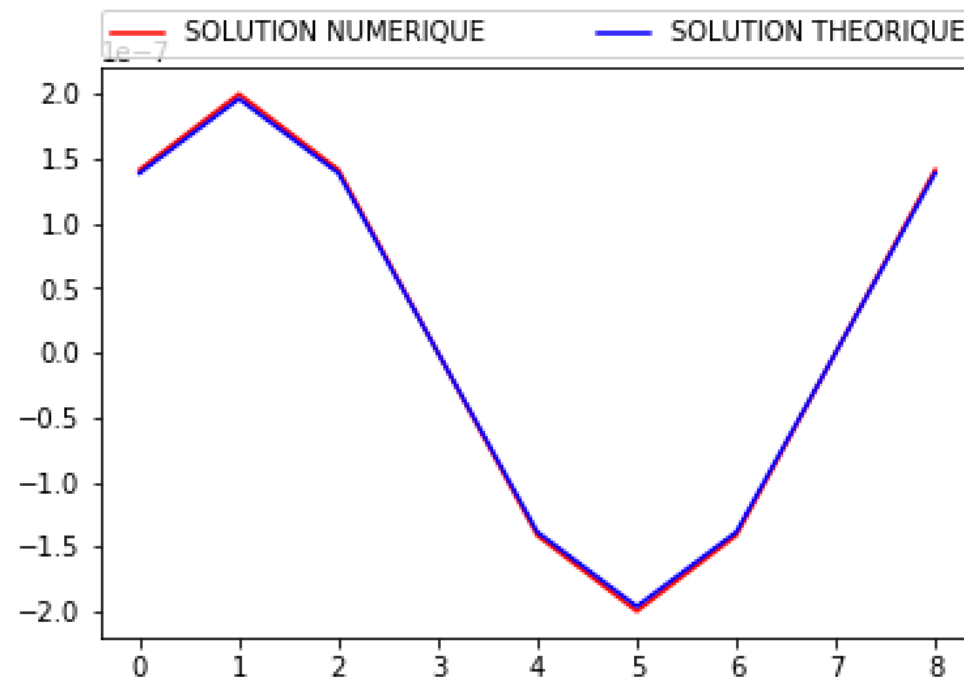
POUR 4 POINTS DE GRILLE, L'ERREUR DE NORME L^2 EST 0.2



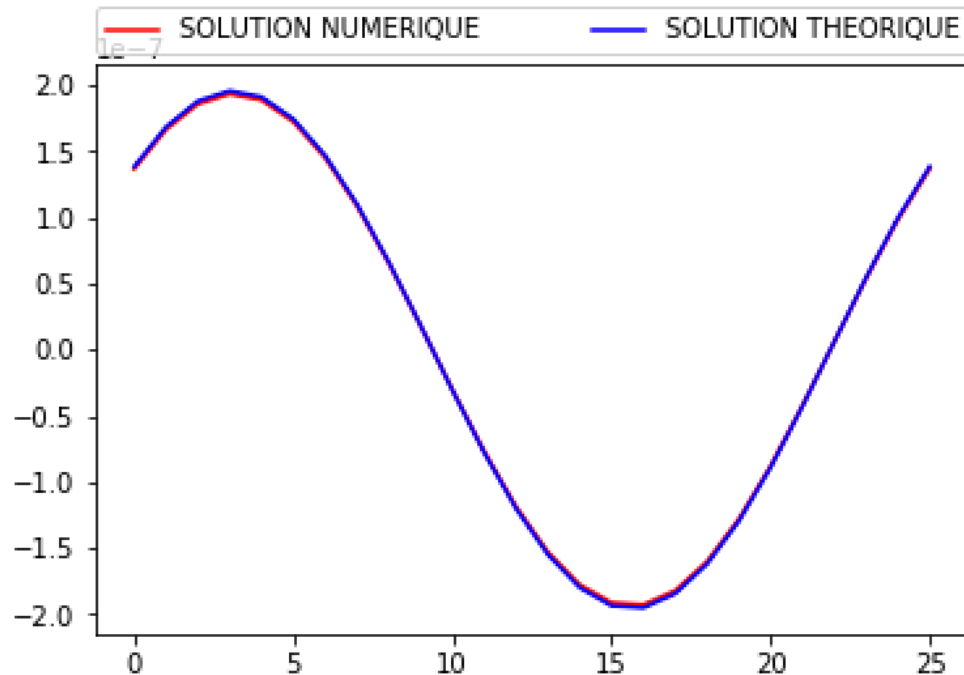
POUR 7 POINTS DE GRILLE, L'ERREUR DE NORME L^2 EST 0.0704



POUR 8 POINTS DE GRILLE, L'ERREUR DE NORME L^2 EST 0



POUR 25 POINTS DE GRILLE, L'ERREUR DE NORME L^2 EST 0.01;



- On compare sur un graphe la solution exacte et la solution discrète.
On observe assez naturellement que plus le nombre de points de grille est élevé, plus on se rapproche la solution discrète se r
approche de la solution exacte.

On veut s'assurer que le schéma discret va converger vers la solution exacte. Un moyen de l'intuiter numériquement est d'observer la norme L^2 de la différence entre la solution exacte et la solution exacte pour un nombre de points de grille de 4, de 5, 7, 8, 10 puis 25 et d'espérer observer que plus le nombre de points de grille est grand, plus la norme de cette différence est petite.

- On trace donc un graphe qui représente l'évolution de la norme de la différence entre la solution exacte et la solution approchée
EN FONCTION du nombre de point de grille.

DEPUIS Spyder, ce dernier graphe s'affiche normalement, comme on peut le voir ci dessous :

Mais il ne semble pas se « lancer » depuis le terminal. En tout cas, les résultats permettent bien de conclure que plus le nombre de points de grille est grand, plus l'erreur en norme L^2 diminue. Donc on peut conjecturer que le schéma converge en norme L^2 .

```
plt.show()
```

```
#Calcul de l'erreur numérique en fonction du n
print("Erreur en norme L^2: ", inversef, Err_list)
inversef = [1/x for x in List_NX]
#plt.figure(0)

plt.plot(List_NX, inversef, "black", label = "f(x)=1/x")
plt.plot(List_NX, Err_list, "yellow", label = "Err L^2")
plt.legend(bbox_to_anchor=(0., 1.02, 1., .10),
           ncol=2, mode="expand", borderaxes=False)
```

```
0.012089903117643601]
```

```
Out[6]: <matplotlib.legend.Legend at 0x181da79
```

