

```

# -*- coding: utf-8 -*-

import numpy as np
import matplotlib.pyplot as plt

#####
#1er exemple T(0,t)=0, T(L,t)=0, T(x,0)=sin(pi.x/L)
#####
D = 0.5      #diffusivité
L = 1.0      #taille du domaine
tmax = 0.5   #date de fin d'intégration
N = 101      #nombre de points pour la discrétisation spatiale
M = 10000    #nombre de dates (0 excepté) pour la discrétisation temporelle
delta = L/(N-1) #pas spatial
epsilon = tmax/M #pas temporel
alpha=epsilon*D/delta**2
print('Pas spatial: ',delta)
print('Pas temporel: ',epsilon)
print('alpha: ',alpha)
print('Echelle temporelle caractéristique: ',L**2/D)

if alpha>0.5:
    raise Exception('alpha>0.5 !')
# Initialisation
#tableau des abscisses
x = np.linspace(0.0,L,N)
#ajout est le tableau des quantités à ajouter au T de la date précédente pour obtenir
le nouveau
ajout = np.zeros((N))
#conditions initiales
T = np.sin(np.pi*x/L)
#conditions aux limites
def T0_ex1(t):
    return(0)
def TL_ex1(t):
    return(0)

plt.figure(1)
#tracé de T(x) à la date t=0
plt.plot(x,T,label='t=0')

# boucle temporelle
for n in range(1,M+1):
    #schéma numérique des points intérieurs
    ajout[1:-1] = alpha*(T[:-2]-2*T[1:-1]+T[2:])
    T=T+ajout
    date=n*epsilon
    #points extrêmes, Dirichlet
    T[0]=T0_ex1(date)
    T[N-1]=TL_ex1(date)

#tracé toutes les M/10 dates
if (n%(M/10) == 0):
    plt.plot(x,T, label='t(s)= '+ str(n * epsilon))
    # et la solution exacte
    Texact=np.exp(-np.pi**2*D*date/L**2)*np.sin(np.pi*x/L)
    plt.plot(x,Texact,'x', markersize=3)

plt.xlabel('x (m)')
plt.ylabel('T (°C)',rotation=0)

```

```

plt.title('Equation de diffusion thermique, exemple de test (trait plein: solution
numérique; x: solution exacte)')
plt.legend()
plt.grid()
plt.show()

#####
#2nd exemple  $T(0,t)=0$ ,  $T(L,t)=1$ ,  $T(x,0)=0$ 
#####
D = 0.5      #diffusivité
L = 1.0      #taille du domaine
tmax = 1     #date de fin d'intégration
N = 101      #nombre de points pour la discrétisation spatiale
M = 20000    #nombre de dates (0 excepté) pour la discrétisation temporelle
delta = L/(N-1) #pas spatial
epsilon = tmax/M #pas temporel
alpha=epsilon*D/delta**2
print('Pas spatial: ',delta)
print('Pas temporel: ',epsilon)
print('alpha: ',alpha)
print('Echelle temporelle caractéristique: ',L**2/D)

if alpha>0.5:
    raise Exception('alpha>0.5 !')
# Initialisation
#tableau des abscisses
x = np.linspace(0.0,L,N)
#ajout est le tableau des quantités à ajouter au T de la date précédente pour obtenir
le nouveau
ajout = np.zeros((N))
#conditions initiales
T = np.zeros((N))
#conditions aux limites
def T0_ex2(t):
    return(0)
def TL_ex2(t):
    return(1)

plt.figure(2)
#tracé de T(x) à la date t=0
plt.plot(x,T,label='t=0')

# boucle temporelle
for n in range(1,M+1):
    #schéma numérique des points intérieurs
    ajout[1:-1] = alpha*(T[:-2]-2*T[1:-1]+T[2:])
    T=T+ajout
    date=n*epsilon
    #points extrêmes, Dirichlet
    T[0]=T0_ex2(date)
    T[N-1]=TL_ex2(date)

#tracé toutes les M/10 dates
    if (n%(M//10) == 0):
        plt.plot(x,T, label='t(s)= '+ str(n * epsilon))
#tracé solutionstationnaire exacte
plt.plot(x,x/L,'x',label='Sol.statio. exacte')

plt.xlabel('x (m)')
plt.ylabel('T (°C)',rotation=0)
plt.title('Equation de diffusion thermique, exemple  $T(0,t)=0$ ,  $T(L,t)=1$ ,  $T(x,0)=0$ ')
plt.legend()
plt.grid()
plt.show()

```

```
#####
#3eme exemple jQ(0,t)=1, T(L,t)=1, T(x,0)=0
#####
D = 0.5      #diffusivité
L = 1        #taille du domaine
tmax = 3     #date de fin d'intégration
N = 101      #nombre de points pour la discrétisation spatiale
M = 30000    #nombre de dates (0 excepté) pour la discrétisation temporelle
delta = L/(N-1) #pas spatial
epsilon = tmax/M #pas temporel
alpha=epsilon*D/delta**2
Psurf=1
TL=1
conductivite=1 #conductivite thermique
print('Pas spatial: ',delta)
print('Pas temporel: ',epsilon)
print('alpha: ',alpha)
print('Echelle temporelle caractéristique: ',L**2/D)

if alpha>0.5:
    raise Exception('alpha>0.5 !')
# Initialisation
#tableau des abscisses
x = np.linspace(0.0,L,N)
#ajout est le tableau des quantités à ajouter au T de la date précédente pour obtenir
le nouveau
ajout = np.zeros((N))
#conditions initiales
T = np.zeros((N))
#conditions aux limites
def TL_ex3(t):
    return(TL)
def jQ0(t):
    return(Psurf)

plt.figure(3)
#tracé de T(x) à la date t=0
plt.plot(x,T,label='t=0')

# boucle temporelle
for n in range(1,M+1):
    date=n*epsilon
    #schéma numérique des points intérieurs
    ajout[1:-1] = alpha*(T[:-2]-2*T[1:-1]+T[2:])
    #ajout pour point auche, de type Neumann
    ajout[0]=2*alpha*(T[1]-T[0]-delta*(-jQ0(date)/conductivite))
    T=T+ajout
    #points extrêmes, Dirichlet
    T[N-1]=TL_ex3(date)

#tracé toutes les M/10 dates
    if (n%(M//10) == 0):
        plt.plot(x,T, label='t(s)= '+ str(n * epsilon))
#tracé solutionstationnaire exacte
plt.plot(x,-Psurf*x/conductivite+TL+Psurf*L/conductivite,'x',label='Sol.statio.
exacte')

plt.xlabel('x (m)')
plt.ylabel('T (°C)',rotation=0)
plt.title('Equation de diffusion thermique, exemple jQ(0,t)=1, T(L,t)=1, T(x,0)=0')
```

```

plt.legend()
plt.grid()
plt.show()

#####
#4eme exemple barre Al: résistance chauffante à gauche, Newton à droite
#####
a=0.02
b=0.01
S=a*b          #section de la barre
P=11           #puissance thermique de la résistance chauffante
conductivite=237 #conductivite thermique aluminium
rho=2699        #Emasse volumique
c=897          #Wcapacité thermique massique
D = conductivite/rho/c    #diffusivité
L = 0.12       #longueur barre
tmax = 2500    #date de fin d'intégration
h=1600        #coefficient d'échange barre-air
Tair=20       #temperature de l'air
N = 101       #nombre de points pour la discrétisation spatiale
M = 400000    #nombre de dates (0 excepté) pour la discrétisation temporelle
delta = L/(N-1) #pas spatial
epsilon = tmax/M    #pas temporel
alpha=epsilon*D/delta**2
print('Pas spatial: ',delta)
print('Pas temporel: ',epsilon)
print('alpha: ',alpha)
print('Echelle temporelle caractéristique: ',L**2/D)

if alpha>0.5:
    raise Exception('alpha>0.5 !')
# Initialisation
#tableau des abscisses
x = np.linspace(0.0,L,N)
#ajout est le tableau des quantités à ajouter au T de la date précédente pour obtenir
le nouveau
ajout = np.zeros((N))
#conditions initiales
T =Tair*np.ones((N))    #barre uniformément à la température de l'air
#conditions aux limites
def jQ0_ex4(t):
    return(P/S)

plt.figure(4)
#tracé de T(x) à la date t=0
plt.plot(x,T,label='t=0')

# boucle temporelle
for n in range(1,M+1):
    date=n*epsilon
    #schéma numérique des points intérieurs
    ajout[1:-1] = alpha*(T[:-2]-2*T[1:-1]+T[2:])
    #ajout pour point anche, de type Neumann
    ajout[0]=2*alpha*(T[1]-T[0]-delta*(-jQ0_ex4(date)/conductivite))
    #ajout pour point droit, type Newton
    ajout[N-1]=2*alpha*(T[N-2]-T[N-1]-delta*h/conductivite*(T[N-1]-Tair))
    T=T+ajout

#tracé toutes les M/10 dates
    if (n%(M//10) == 0):
        plt.plot(x,T, label='t(s)= '+ str(n * epsilon))
#tracé solutionstationnaire exacte
plt.plot(x,-P/conductivite/S*x+Tair+P/S*(1/h+L/conductivite),'x',label='Sol.statio.

```

```
exacte')
```

```
plt.xlabel('x (m)')  
plt.ylabel('T (°C)',rotation=0)  
plt.title('Equation de diffusion thermique, résistance chauffante à gauche,  
ventilateur à droite')  
plt.legend()  
plt.grid()  
plt.show()
```