

# Conduction thermique à une dimension

## 1. Introduction

On considère quelques exemples de résolution numérique de l'équation de la chaleur à une dimension. La méthode numérique est expliquée dans [Équation de diffusion à une dimension](#). Le module python utilisé est présenté dans [Équation de diffusion à une dimension avec Python](#).

L'équation de conduction thermique (ou équation de la chaleur) est :

$$\frac{\partial T}{\partial t} = D \frac{\partial^2 T}{\partial x^2} + s(x) \quad (1)$$

où  $D$  est le coefficient de diffusion thermique et  $s(x)$  un terme de source thermique, qui représente par exemple une puissance volumique dissipée.

En raison de la linéarité de cette équation, il est possible de raisonner sur une température sans dimension, avec une origine fixée arbitrairement. Par exemple, un problème comportant deux température constantes  $T_1$  et  $T_2 > T_1$  sera traité en attribuant les valeurs 0 et 1 à ces températures.

## 2. Conduction dans une plaque

On considère une plaque (perpendiculaire à l'axe  $x$ ) de conductivité thermique uniforme, soumise en  $x = 0$  à une température constante  $T = 1$  et en  $x = 1$  à une température constante  $T = 0$ . Il n'y a aucune source thermique dans la plaque. Initialement la température est nulle sur l'intervalle  $[0, 1]$ .

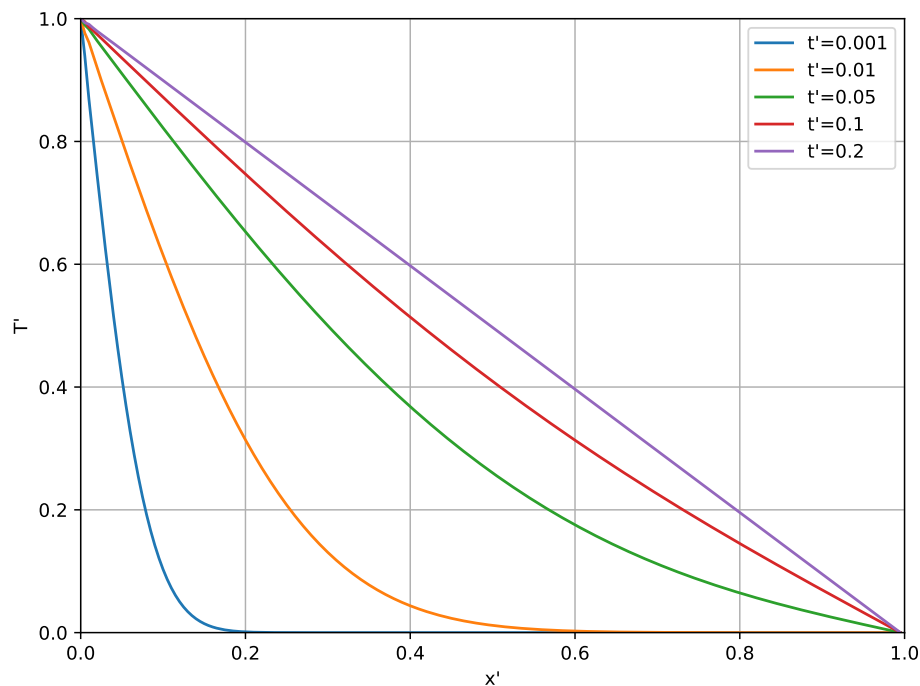
```
import numpy
import sys
sys.path.append('../..../numerique/diffusion/pydiffusion')
from Diffusion1D import Diffusion1D
from matplotlib.pyplot import *

N=200
U=numpy.zeros(N)
S=numpy.zeros(N)
x=numpy.arange(N)*1.0/N
coef = [[1,1]]
t=0
diffusion = Diffusion1D(N)
diffusion.config(0.0001,coef,"dirichlet",1,"dirichlet",0,S)
[U1,t]=diffusion.iterations(U,t,0.001)
diffusion.config(0.001,coef,"dirichlet",1,"dirichlet",0,S)
[U2,t]=diffusion.iterations(U1,t,0.01)
[U3,t]=diffusion.iterations(U2,t,0.05)
diffusion.config(0.01,coef,"dirichlet",1,"dirichlet",0,S)
[U4,t]=diffusion.iterations(U3,t,0.1)
diffusion.config(0.1,coef,"dirichlet",1,"dirichlet",0,S)
[U5,t]=diffusion.iterations(U4,t,0.2)
figure(figsize=(8,6))
plot(x,U1,label="t'=0.001")
plot(x,U2,label="t'=0.01")
```

```

plot(x,U3,label="t'=0.05")
plot(x,U4,label="t'=0.1")
plot(x,U5,label="t'=0.2")
legend(loc="upper right")
xlabel("x' ")
ylabel("T' ")
axis([0,1,0,1])
grid()

```



### 3. Conduction dans deux plaques

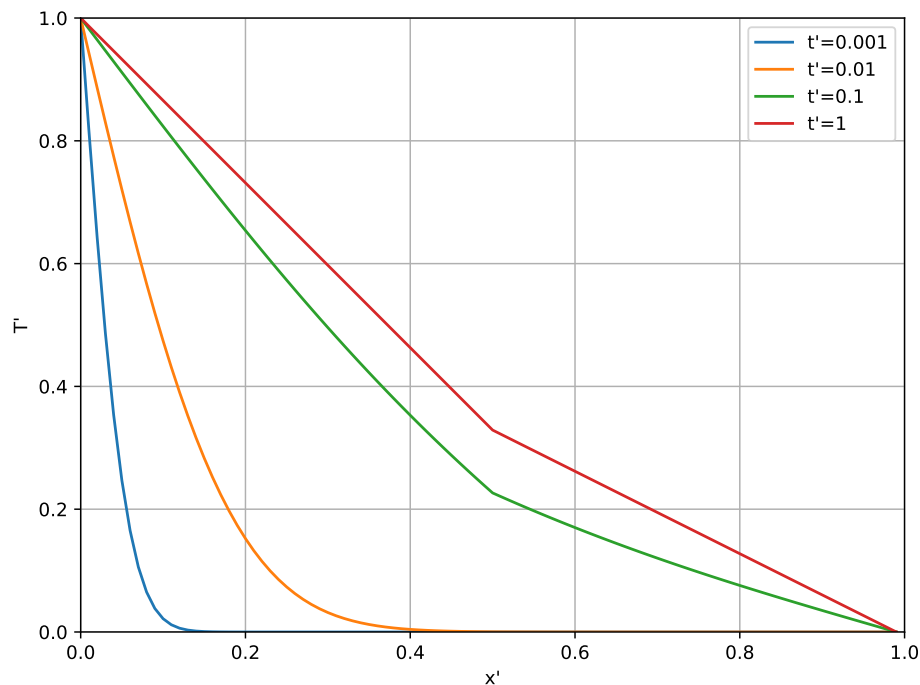
On reprend le problème précédent avec deux plaques de conductivités thermiques différentes.

```

N=100
U=numpy.zeros(N)
S=numpy.zeros(N)
x=numpy.arange(N)*1.0/N
coef = [[0.5,0.5],[1,1]]
t=0
diffusion = Diffusion1D(N)
diffusion.config(0.0001,coef,"dirichlet",1,"dirichlet",0,S)
[U1,t]=diffusion.iterations(U,t,0.001)
diffusion.config(0.001,coef,"dirichlet",1,"dirichlet",0,S)
[U2,t]=diffusion.iterations(U1,t,0.01)
diffusion.config(0.01,coef,"dirichlet",1,"dirichlet",0,S)
[U3,t]=diffusion.iterations(U2,t,0.1)
diffusion.config(0.1,coef,"dirichlet",1,"dirichlet",0,S)

```

```
[U4,t]=diffusion.iterations(U3,t,1)
figure(figsize=(8,6))
plot(x,U1,label="t'=0.001")
plot(x,U2,label="t'=0.01")
plot(x,U3,label="t'=0.1")
plot(x,U4,label="t'=1")
legend(loc="upper right")
xlabel("x' ")
ylabel("T' ")
axis([0,1,0,1])
grid()
```



#### 4. Échange thermique entre deux solides identiques

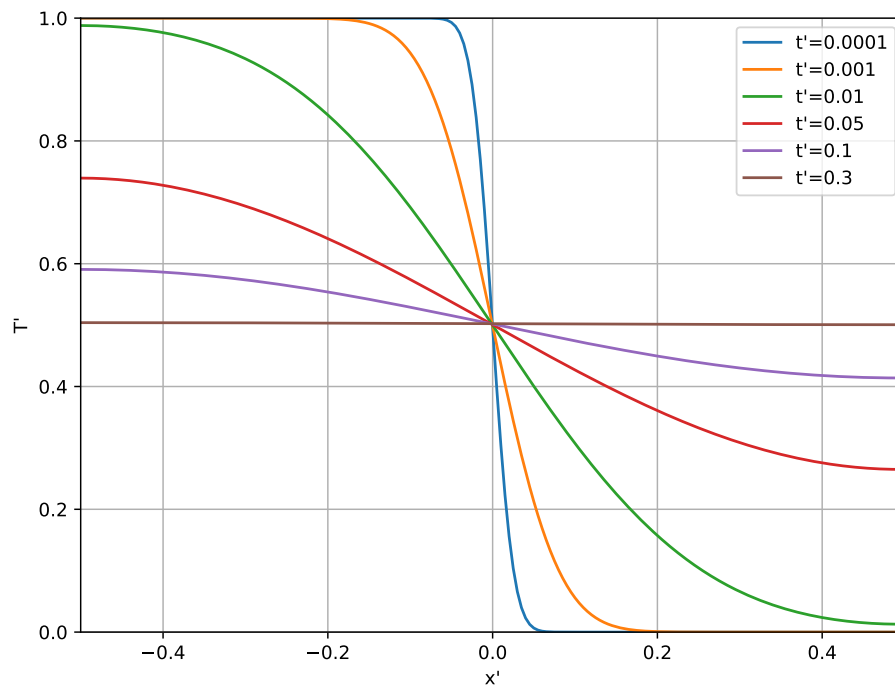
Deux solides identiques ayant initialement deux températures différentes sont mis en contact à l'instant  $t = 0$ . On néglige la résistance de contact et l'ensemble des deux solides est supposé isolé thermiquement.

```
N=200
U=numpy.zeros(N)
S=numpy.zeros(N)
x=numpy.arange(N)*1.0/N
coef = [[0.5,1],[1,1]]
for i in range(int(N/2)):
    U[i] = 1
t=0
diffusion = Diffusion1D(N)
diffusion.config(0.000001,coef,"neumann",0,"neumann",0,S)
```

```

[U0,t]=diffusion.iterations(U,t,0.0001)
diffusion.config(0.00001,coef,"neumann",0,"neumann",0,S)
[U1,t]=diffusion.iterations(U0,t,0.001)
diffusion.config(0.0001,coef,"neumann",0,"neumann",0,S)
[U2,t]=diffusion.iterations(U1,t,0.01)
[U22,t]=diffusion.iterations(U2,t,0.05)
diffusion.config(0.001,coef,"neumann",0,"neumann",0,S)
[U3,t]=diffusion.iterations(U22,t,0.1)
diffusion.config(0.01,coef,"neumann",0,"neumann",0,S)
[U4,t]=diffusion.iterations(U3,t,0.3)
figure(figsize=(8,6))
x=x-0.5
plot(x,U0,label="t'=0.0001")
plot(x,U1,label="t'=0.001")
plot(x,U2,label="t'=0.01")
plot(x,U22,label="t'=0.05")
plot(x,U3,label="t'=0.1")
plot(x,U4,label="t'=0.3")
legend(loc="upper right")
xlabel("x' ")
ylabel("T' ")
axis([-0.5,0.5,0,1])
grid()

```



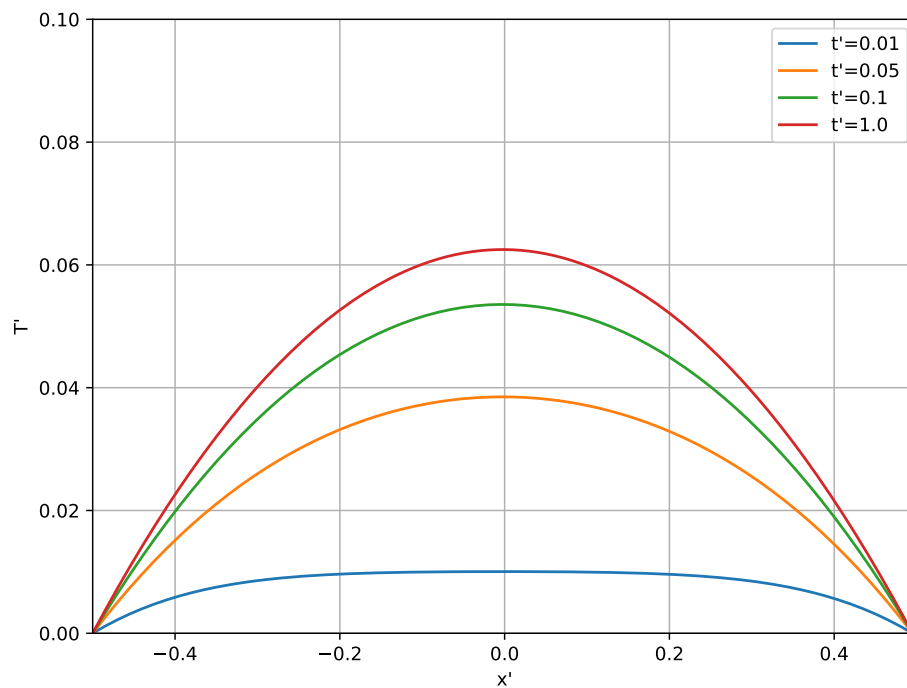
## 5. Dissipation uniforme

Une plaque est soumise à une dissipation uniforme (par exemple une dissipation électrique). Le terme de source est uniforme dans la plaque. La température extérieure est constante.

```

N=200
U=numpy.zeros(N)
S=numpy.ones(N)
x=numpy.arange(N)*1.0/N
coef = [[1,1]]
t=0
diffusion = Diffusion1D(N)
diffusion.config(0.0001,coef,"dirichlet",0,"dirichlet",0,S)
[U0,t]=diffusion.iterations(U,t,0.01)
diffusion.config(0.001,coef,"dirichlet",0,"dirichlet",0,S)
[U1,t]=diffusion.iterations(U0,t,0.05)
[U2,t]=diffusion.iterations(U1,t,0.1)
[U3,t]=diffusion.iterations(U2,t,1.0)
figure(figsize=(8,6))
x=x-0.5
plot(x,U0,label="t'=0.01")
plot(x,U1,label="t'=0.05")
plot(x,U2,label="t'=0.1")
plot(x,U3,label="t'=1.0")
legend(loc="upper right")
xlabel("x' ")
ylabel("T' ")
axis([-0.5,0.5,0,0.1])
grid()

```



Même chose avec un flux nul sur le bord  $x = 0$  :

```

N=200
U=numpy.zeros(N)
S=numpy.ones(N)

```

```
x=numpy.arange(N)*1.0/N
coef = [[1,1]]
t=0
diffusion = Diffusion1D(N)
diffusion.config(0.0001,coef,"neumann",0,"dirichlet",0,S)
[U0,t]=diffusion.iterations(U,t,0.01)
diffusion.config(0.001,coef,"neumann",0,"dirichlet",0,S)
[U1,t]=diffusion.iterations(U0,t,0.05)
[U2,t]=diffusion.iterations(U1,t,0.1)
[U3,t]=diffusion.iterations(U2,t,1.0)
figure(figsize=(8,6))
plot(x,U0,label="t=0.01")
plot(x,U1,label="t=0.05")
plot(x,U2,label="t=0.1")
plot(x,U3,label="t=1.0")
legend(loc="upper right")
xlabel("x")
ylabel("T")
axis([0,1,0,0.3])
grid()
```

