

# Differences Finies

Sébastien Joannès

September 24, 2015

## Contents

<b>1 Introduction</b>	<b>1</b>
<b>2 L'équation de diffusion comme exemple</b>	<b>1</b>
2.1 La physique de la diffusion unifiée . . . . .	1
2.2 Discrétisation en espace et en temps . . . . .	2
2.3 Approximation des opérateurs par la formule de Taylor . . . . .	2
<b>3 Schémas d'approximation pour l'équation de diffusion</b>	<b>4</b>
3.1 Différence avant en temps et centrée en espace . . . . .	4
3.2 Différence arrière en temps et centrée en espace . . . . .	7
3.3 Schéma de Crank-Nicolson & theta-method . . . . .	8

## 1 Introduction

## 2 L'équation de diffusion comme exemple

### 2.1 La physique de la diffusion unifiée

L'équation de diffusion s'écrit sous une même et unique forme qu'il s'agisse de diffusion moléculaire, de conduction thermique ou électrique. L'équation de Fick est ici choisie comme support à cette présentation sur les différences finies est dans un cas instationnaire ( $t$  indiquant le temps) prend la forme ci-dessous:

$$-\frac{\partial}{\partial t} \int_{\Omega} C \, d\Omega = \int_{\Omega} \underline{\nabla} \cdot \underline{J} \, d\Omega \quad \Rightarrow \quad \frac{\partial C}{\partial t} = -\underline{\nabla} \cdot \underline{J} = \underline{\nabla} \cdot (D \underline{\nabla} C) \quad (1)$$

Où  $C$  désigne la concentration,  $\underline{\nabla}$  l'opérateur divergence et  $\underline{J}$  le flux de diffusion, proportionnel au gradient de la concentration. Le coefficient de proportionnalité  $D$  se nomme **diffusivité** ou **coefficient de diffusion**.

Dans le cas mono-dimensionnel, la variable d'espace étant notée  $x$ , la seconde loi de Fick s'écrit:

$$\frac{\partial C}{\partial t} = \frac{\partial}{\partial x} \left( D \frac{\partial C}{\partial x} \right) \quad (2)$$

$$\frac{\partial C}{\partial t} = D \frac{\partial^2 C}{\partial x^2} \quad \text{si } D \text{ est constant} \quad (3)$$

La résolution de cette équation nécessite d'introduire des **conditions aux limites** pour  $x = 0$  et  $x = L$  (épaisseur ou longueur du système dans lequel la diffusion se propage) ainsi que des **conditions initiales** lorsque  $t = 0$ .

## 2.2 Discrétisation en espace et en temps

La première étape de la méthode des différences finies consiste à bâtir une grille sur laquelle l'équation pourra être résolue. L'approximation de l'équation aux dérivées partielles est obtenue par un jeu de relations algébriques entre les valeurs de  $C$  aux points (noeuds de la grille)  $x_1, x_2, \dots, x_i, \dots$ . La forme continue des dérivées est donc remplacée par une forme discrétisée appelée **différence finie** et basée sur les séries de Taylor.

La méthode des différences finies propose une approximation de  $C(x, t)$  sur un ensemble fini de points  $x$  et  $t$ . Nous considérons que ces points sont distribués de manière uniforme et désignés par  $x_i$  et  $t_m$  avec  $i = 1 \rightarrow N$  et  $m = 1 \rightarrow P$ .  $C_i^m$  désigne donc la solution numérique approchée de l'équation et doit être la plus proche possible de  $C(x_i, t_m)$  qui est la solution continue évaluée aux noeuds de la grille.

## 2.3 Approximation des opérateurs par la formule de Taylor

Les séries de Taylor sont surtout utilisées pour approximer des fonctions lorsque l'on peut identifier un "petit" paramètre. Pour la présente application, le petit paramètre peut être un petit pas de temps ou d'espace. Un développement en série de Taylor est la représentation d'une fonction en une somme infinie de termes calculés à partir des valeurs de la fonction et de ses dérivés en un point particulier.

Soit  $f : x \mapsto f(x)$  une fonction continue et indéfiniment dérivable sur l'intervalle considéré.  $f$  prend les valeurs  $f(x_i)$  au point  $x_i$  et à ce point, la dérivée première de  $f$  est notée  $f^{(1)}(x_i)$ . De même, les dérivées suivantes de  $f$  en  $x_i$  se notent  $f^{(2)}(x_i), f^{(3)}(x_i), \dots, f^{(n)}(x_i)$ :

$$f^{(1)}(x_i) = \left. \frac{df}{dx} \right|_{x_i}, \quad f^{(2)}(x_i) = \left. \frac{d^2 f}{dx^2} \right|_{x_i} \quad \text{et} \quad f^{(n)}(x_i) = \left. \frac{d^n f}{dx^n} \right|_{x_i} \quad (4)$$

Considérons un développement en série de Taylor de  $f$  dans un voisinage du point  $x_i + \Delta x$  (où  $\Delta x$  est une petite variation de  $x$ ) et isolons la valeur de la dérivée première  $f^{(1)}(x_i)$ :

$$f(x_i + \Delta x) = f(x_i) + \Delta x f^{(1)}(x_i) + \frac{(\Delta x)^2}{2!} f^{(2)}(x_i) + \dots \quad (5)$$

$$f^{(1)}(x_i) = \frac{f(x_i + \Delta x) - f(x_i)}{\Delta x} - \frac{\Delta x}{2!} f^{(2)}(x_i) + \dots \quad (6)$$

En utilisant les notations  $f_i \approx f(x_i)$ ,  $f_{i+1} \approx f(x_i + \Delta x)$  et  $\mathcal{O}$ , nous obtenons:

$$f^{(1)}(x_i) = \frac{f_{i+1} - f_i}{\Delta x} + \mathcal{O}(\Delta x) \quad (7)$$

Où  $\mathcal{O}(\Delta x)$  reflète l'**erreur de troncature**. Cela signifie que l'erreur d'approximation décroît comme le pas  $\Delta x$  lorsque  $\Delta x$  devient de plus en plus petit.

Il s'agit d'une **différence avant** du **premier ordre**.

On peut écrire de même, une **différence arrière** du **premier ordre** en développant au voisinage de  $x_i - \Delta x$ :

$$f(x_i - \Delta x) = f(x_i) - \Delta x f^{(1)}(x_i) + \frac{(\Delta x)^2}{2!} f^{(2)}(x_i) + \dots \quad (8)$$

$$f^{(1)}(x_i) = \frac{f_i - f_{i-1}}{\Delta x} + \mathcal{O}(\Delta x) \quad (9)$$

Une **différence centrée** du **premier ordre** s'obtient en soustrayant les deux approximations précédentes:

$$f^{(1)}(x_i) = \frac{f_{i+1} - f_{i-1}}{2\Delta x} + \mathcal{O}(\Delta x^2) \quad (10)$$

Une **différence centrée** du **second ordre** s'obtient en additionnant les deux approximations précédentes:

$$f^{(2)}(x_i) = \frac{f_{i+1} - 2f_i + f_{i-1}}{(\Delta x)^2} + \mathcal{O}(\Delta x^2) \quad (11)$$

Il est de même possible de construire des approximations sur une grille non uniforme en introduisant  $\Delta x_i$  l'espace entre les points  $x_{i+1}$  et  $x_i$ , et  $\Delta x_{i-1}$  l'espace entre les points  $x_i$  et  $x_{i-1}$ .

$$f(x_i + \Delta x_i) = f(x_i) + \Delta x_i f^{(1)}(x_i) + \frac{(\Delta x_i)^2}{2!} f^{(2)}(x_i) + \dots \quad (12)$$

$$f(x_i - \Delta x_{i-1}) = f(x_i) - \Delta x_{i-1} f^{(1)}(x_i) + \frac{(\Delta x_{i-1})^2}{2!} f^{(2)}(x_i) + \dots \quad (13)$$

Ce qui conduit aux approximations du premier ordre suivantes:

$$f^{(1)}(x_i) = \frac{f_{i+1} - f_i}{\Delta x_i} + \mathcal{O}(\Delta x) \quad (14)$$

$$f^{(1)}(x_i) = \frac{f_i - f_{i-1}}{\Delta x_{i-1}} + \mathcal{O}(\Delta x) \quad (15)$$

$$f^{(1)}(x_i) = \frac{f_{i+1} - f_{i-1}}{\Delta x_i + \Delta x_{i-1}} + \mathcal{O}(\Delta x) \quad (16)$$

### 3 Schémas d'approximation pour l'équation de diffusion

#### 3.1 Différence avant en temps et centrée en espace

Différence **avant** en temps:

$$\left. \frac{\partial C}{\partial t} \right|_{t_m, x_i} = \frac{C_i^{m+1} - C_i^m}{\Delta t} + \mathcal{O}(\Delta t) \quad (17)$$

Différence **centrée** en espace:

$$\left. \frac{\partial^2 C}{\partial x^2} \right|_{t_m, x_i} = \frac{C_{i+1}^m - 2C_i^m + C_{i-1}^m}{(\Delta x)^2} + \mathcal{O}(\Delta x^2) \quad (18)$$

L'approximation de la seconde loi de Fick s'écrit:

$$\frac{C_i^{m+1} - C_i^m}{\Delta t} = D \frac{C_{i+1}^m - 2C_i^m + C_{i-1}^m}{(\Delta x)^2} + \mathcal{O}(\Delta t) + \mathcal{O}(\Delta x^2) \quad (19)$$

Ce qui permet de déterminer explicitement  $C_i^{m+1}$  :

$$C_i^{m+1} = C_i^m + \frac{D\Delta t}{(\Delta x)^2} (C_{i+1}^m - 2C_i^m + C_{i-1}^m) \quad (20)$$

En posant  $r = D\Delta t / (\Delta x)^2$ , il vient:

$$C_i^{m+1} = rC_{i+1}^m + (1 - 2r)C_i^m + rC_{i-1}^m \quad (21)$$

La stabilité du schéma n'est obtenue que pour  $r < 1/2$ . Les pas d'espace et de temps sont donc liés.

L'implémentation du schéma est très simple:

```
>>> import numpy as np
... import matplotlib as mpl
... import matplotlib.pyplot as plt
... %matplotlib inline
```

Définition des paramètres:

```
>>> D = .05;
... L = 1.;
... tmax = 1.;
... nx = 6;
... nt = 10;
... dx = L/(nx-1);
... dt = tmax/(nt-1);
... r = D*dt/dx**2; r2 = 1 - 2*r;
```

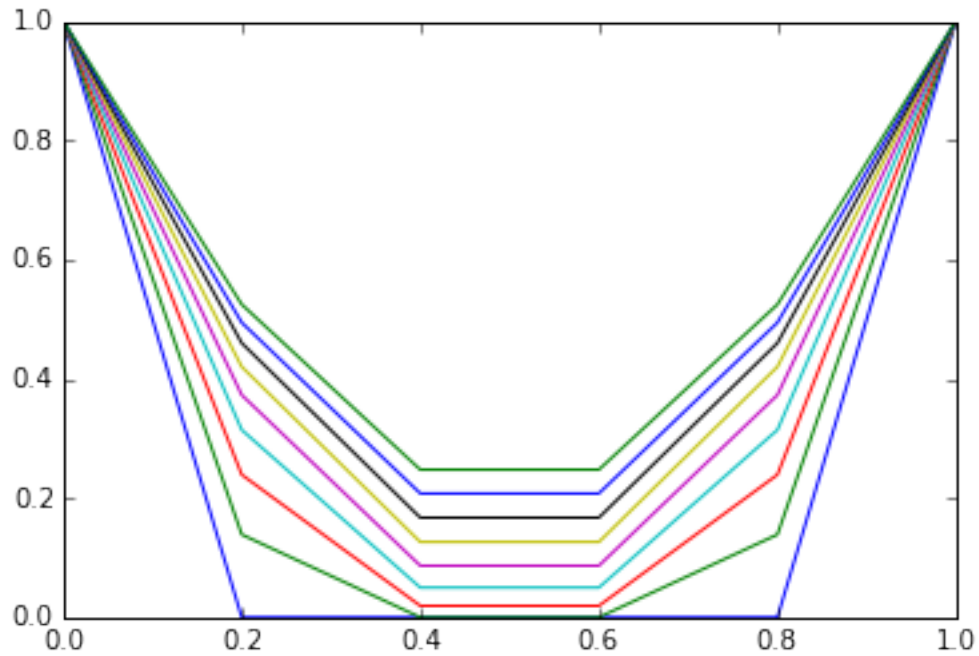
Deux boucles imbriquées pour déterminer la solution:

```
>>> t = 0.;
... C = np.zeros((nt,nx));
... for m in range(nt-1):
...     C[:,0] = 1.;
...     C[:, -1] = 1.;
...     t = t + dt;
...     for i in range(1,nx-1):
...         C[m+1,i] = r*C[m,i-1] + r2*C[m,i] + r*C[m,i+1];
... print C
```

```
[[ 1.         0.         0.         0.         0.         1.        ]
 [ 1.         0.13888889 0.         0.         0.13888889 1.        ]
 [ 1.         0.23919753 0.01929012 0.01929012 0.23919753 1.        ]
 [ 1.         0.31432184 0.04983282 0.04983282 0.31432184 1.        ]
 [ 1.         0.37282034 0.08656741 0.08656741 0.37282034 1.        ]
 [ 1.         0.42017127 0.12632476 0.12632476 0.42017127 1.        ]
 [ 1.         0.45989102 0.16713677 0.16713677 0.45989102 1.        ]
 [ 1.         0.49424585 0.20779709 0.20779709 0.49424585 1.        ]
 [ 1.         0.52470493 0.24758164 0.24758164 0.52470493 1.        ]
 [ 1.         0.55222879 0.28607098 0.28607098 0.55222879 1.        ]]
```

L'évolution de la concentration  $C$  peut être tracée en fonction de l'espace et du temps:

```
>>> x=np.linspace(0.,L,nx)
... for m in range(nt-1):
...     plt.plot(x,C[m,:]);
... plt.show()
```



>>> Augmentation du nombre de points et adaptation du temps de simulation:

```
>>> D = .05;
... L = 1.;
... tmax = 10.;
... nx = 20;
... nt = 400;
... dx = L/(nx-1);
... dt = tmax/(nt-1);
... r = D*dt/dx**2; r2 = 1 - 2*r;
... print r
...
... t = 0.;
... C = np.zeros((nt,nx));
... for m in range(nt-1):
...     C[:,0] = 1.;
...     C[:, -1] = 1.;
...     t = t + dt;
...     for i in range(1,nx-1):
...         C[m+1,i] = r*C[m,i-1] + r2*C[m,i] + r*C[m,i+1];
```



$$\frac{C_i^m - C_i^{m-1}}{\Delta t} = D \frac{C_{i+1}^m - 2C_i^m + C_{i-1}^m}{(\Delta x)^2} + \mathcal{O}(\Delta t) + \mathcal{O}(\Delta x^2) \quad (24)$$

Ce qui conduit à un système d'équations:

$$\frac{1}{\Delta t} C_i^{m-1} = -\frac{D}{(\Delta x)^2} C_{i-1}^m + \left( \frac{1}{\Delta t} + \frac{2D}{(\Delta x)^2} \right) C_i^m - \frac{D}{(\Delta x)^2} C_{i+1}^m \quad (25)$$

Qui peut se mettre sous forme matricielle en posant:

$$d_i = a_i C_{i-1}^m + b_i C_i^m + c_i C_{i+1}^m \quad \text{avec} \quad i = 2 \rightarrow N-1 \quad (26)$$

$$\begin{bmatrix} b_1 = 1 & c_1 = 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ a_2 & b_2 & c_2 & 0 & \cdots & 0 & 0 & 0 \\ 0 & a_3 & b_3 & c_3 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & a_{N-1} & b_{N-1} & c_{N-1} \\ 0 & 0 & 0 & 0 & \cdots & 0 & a_N = 0 & b_N = 1 \end{bmatrix} \begin{bmatrix} C_1^m \\ C_2^m \\ C_3^m \\ \vdots \\ C_{N-1}^m \\ C_N^m \end{bmatrix} = \begin{bmatrix} d_1 = C_0 \\ d_2 \\ d_3 \\ \vdots \\ d_{N-1} \\ d_N = C_L \end{bmatrix} \quad (27)$$

Ce schéma est inconditionnellement stable.

### 3.3 Schéma de Crank-Nicolson & theta-method

$$\frac{\partial}{\partial x} \left( D \frac{\partial C}{\partial x} \right) \Big|_{t_m, x_i} \approx A C_{i-1}^m + B C_i^m + \Gamma C_{i+1}^m \quad (28)$$

$$\begin{aligned} \frac{C_i^m - C_i^{m-1}}{\Delta t_{m-1}} &= \theta [A C_{i-1}^m + B C_i^m + \Gamma C_{i+1}^m] \cdots \\ &+ (1 - \theta) [A C_{i-1}^{m-1} + B C_i^{m-1} + \Gamma C_{i+1}^{m-1}] \end{aligned} \quad (29)$$