

Équation de Schrödinger à deux dimensions

1. Introduction

Ce document expose une méthode numérique pour la résolution de l'équation de Schrödinger à deux dimensions, dans le cadre d'un problème de diffusion d'un paquet d'onde par un potentiel, ou de diffraction par un obstacle (diffraction par des fentes).

Pour le même problème à une dimension, voir [Équation de Schrödinger à une dimension](#).

2. Équation de Schrödinger

On considère l'équation de Schrödinger dépendante du temps pour une fonction d'onde $\psi(x, y, t)$:

$$i\hbar \frac{\partial \psi}{\partial t} = -\frac{\hbar^2}{2m} \left(\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} \right) + V(x, y)\psi \quad (1)$$

On s'intéresse au mouvement d'une particule dans un potentiel, limité par le domaine rectangulaire $[0, L_x] \times [0, L_y]$. Sur les bords de ce domaine, la fonction d'onde est nulle. La condition de normalisation s'écrit :

$$\int_0^{L_x} \int_0^{L_y} \psi \psi^* dx dy = 1 \quad (2)$$

La fonction d'onde initiale $\psi(x, y, 0)$ est connue, et prendra la forme d'un paquet d'onde localisé dans le domaine.

L'objectif est de résoudre numériquement l'équation par la méthode des différences finies, en suivant une méthode similaire à celle exposée dans [Équation de diffusion à deux dimensions](#). L'équation de Schrödinger est en effet analogue à l'équation de diffusion en raison de la dérivée première par rapport au temps. Elle a des solutions sous forme d'onde progressives en raison du coefficient imaginaire $i\hbar$.

Soit V_0 l'échelle du potentiel. La fonction $V(x, y)$ utilisée sera sans dimensions. On introduit l'échelle de temps :

$$\tau = \frac{\hbar}{V_0} \quad (3)$$

et l'échelle de longueur :

$$\lambda = \frac{\sqrt{2mV_0}}{\hbar} \quad (4)$$

En utilisant un temps et des coordonnées (x, y) réduits par ces deux échelles, on obtient l'équation :

$$i \frac{\partial \psi}{\partial t} = -\frac{\partial^2 \psi}{\partial x^2} - \frac{\partial^2 \psi}{\partial y^2} + V(x, y)\psi \quad (5)$$

On utilise l'opérateur hamiltonien :

$$H = -\frac{\partial^2}{\partial x^2} - \frac{\partial^2}{\partial y^2} + V(x, y) \quad (6)$$

3. Méthode numérique

3.a. Discrétisation

Les nœuds du maillage sont définis par :

$$x_j = j\Delta x \quad (7)$$

$$y_m = m\Delta y \quad (8)$$

L'indice j varie de 0 à $N_x - 1$, l'indice m de 0 à $N_y - 1$.

Les valeurs sont recherchées aux instants définis par :

$$t_n = n\Delta t \quad (9)$$

où n est un entier positif ou nul. On pose :

$$\psi_{j,m}^n = \psi(j\Delta x, m\Delta y, n\Delta t) \quad (10)$$

$$V_{j,m} = V(j\Delta x, m\Delta y) \quad (11)$$

Dans cette notation, l'exposant sert donc à représenter le temps.

La discrétisation des dérivées secondes par la méthode des différences finies nous amène à définir un opérateur hamiltonien :

$$H\psi_{j,m}^n = -\frac{\psi_{j+1,m}^n - 2\psi_{j,m}^n + \psi_{j-1,m}^n}{(\Delta x)^2} - \frac{\psi_{j,m+1}^n - 2\psi_{j,m}^n + \psi_{j,m-1}^n}{(\Delta y)^2} + V_{j,m}\psi_{j,m}^n \quad (12)$$

Comme il est expliqué dans [Équation de Schrödinger à une dimension](#), un schéma numérique qui conserve la condition de normalisation s'écrit :

$$(1 + i\frac{1}{2}\Delta t H)\psi_{j,m}^{n+1} = (1 - i\frac{1}{2}\Delta t H)\psi_{j,m}^n \quad (13)$$

Le schéma numérique obtenu est implicite.

3.b. Schéma numérique à directions alternées

Il s'agit de modifier le schéma implicite (13) pour obtenir un schéma implicite dans la direction x , alterné avec un schéma implicite dans la direction y , comme exposé dans [Équation de diffusion à deux dimensions](#).

Le schéma implicite à directions alternées consiste à décomposer le pas de temps $(n, n+1)$ en deux sous-pas. Pour le premier, on écrit un schéma implicite pour la dérivée par rapport à x mais explicite pour la dérivée par rapport à y . Pour le second, on écrit un schéma implicite pour la dérivée par rapport à y mais explicite pour la dérivée par rapport à x . On choisit de prendre en compte le potentiel seulement dans le second demi-pas.

Pour alléger les notations, on définit les opérateurs de dérivation suivants :

$$D_x \psi_{j,m}^n = \frac{\psi_{j+1,m}^n - 2\psi_{j,m}^n + \psi_{j-1,m}^n}{(\Delta x)^2} \quad (14)$$

$$D_y \psi_{j,m}^n = \frac{\psi_{j,m+1}^n - 2\psi_{j,m}^n + \psi_{j,m-1}^n}{(\Delta y)^2} \quad (15)$$

On a bien sûr :

$$H = -D_x - D_y + V_{j,m} \quad (16)$$

Le premier demi-pas s'écrit :

$$\left(1 - i \frac{\Delta t}{2} D_x + i \frac{\Delta t}{2} \frac{1}{2} V_{j,m}\right) \psi_{j,m}^{n+\frac{1}{2}} = \left(1 + i \frac{\Delta t}{2} D_y - i \frac{\Delta t}{2} \frac{1}{2} V_{j,m}\right) \psi_{j,m}^n \quad (17)$$

Le second demi-pas s'écrit :

$$\left(1 - i \frac{\Delta t}{2} D_y + i \frac{\Delta t}{2} \frac{1}{2} V_{j,m}\right) \psi_{j,m}^{n+1} = \left(1 + i \frac{\Delta t}{2} D_x - i \frac{\Delta t}{2} \frac{1}{2} V_{j,m}\right) \psi_{j,m}^{n+\frac{1}{2}} \quad (18)$$

Pour simplifier, on pose $\Delta x = \Delta y$ et on définit :

$$\alpha = \frac{2(\Delta x)^2}{\Delta t} \quad (19)$$

Le premier demi-pas s'écrit :

$$\psi_{j,m}^{n+\frac{1}{2}} - \frac{i}{\alpha} \left(\psi_{j+1,m}^{n+\frac{1}{2}} - 2\psi_{j,m}^{n+\frac{1}{2}} + \psi_{j-1,m}^{n+\frac{1}{2}} \right) + \frac{i}{\alpha} (\Delta x)^2 \frac{1}{2} V_{j,m} \psi_{j,m}^{n+\frac{1}{2}} = \psi_{j,m}^n + \frac{i}{\alpha} \left(\psi_{j,m+1}^n - 2\psi_{j,m}^n + \psi_{j,m-1}^n \right) - \frac{i}{\alpha} (\Delta x)^2 \frac{1}{2} V_{j,m} \psi_{j,m}^n \quad (20)$$

Le second demi-pas s'écrit :

$$\psi_{j,m}^{n+1} - \frac{i}{\alpha} \left(\psi_{j,m+1}^{n+1} - 2\psi_{j,m}^{n+1} + \psi_{j,m-1}^{n+1} \right) + \frac{i}{\alpha} (\Delta x)^2 \frac{1}{2} V_{j,m} \psi_{j,m}^{n+1} = \psi_{j,m}^{n+\frac{1}{2}} + \frac{i}{\alpha} \left(\psi_{j+1,m}^{n+\frac{1}{2}} - 2\psi_{j,m}^{n+\frac{1}{2}} + \psi_{j-1,m}^{n+\frac{1}{2}} \right) - \frac{i}{\alpha} (\Delta x)^2 \frac{1}{2} V_{j,m} \psi_{j,m}^{n+\frac{1}{2}} \quad (21)$$

Chacune de ces deux étapes consiste à résoudre un système tridiagonal, ce qui se fait par élimination directe en utilisant l'algorithme décrit dans [équation de diffusion à une dimension](#).

On écrira les deux systèmes à résoudre sous la forme générale suivante :

$$A_{j,m} \psi_{j-1,m}^{n+1/2} + B_{j,m} \psi_{j,m}^{n+1/2} + C_{j,m} \psi_{j+1,m}^{n+1/2} = D_{j,m} \psi_{j,m-1}^n + E_{j,m} \psi_{j,m}^n + F_{j,m} \psi_{j,m+1}^n + G_{j,m} \quad (22)$$

$$A'_{j,m} \psi_{j,m-1}^{n+1} + B'_{j,m} \psi_{j,m}^{n+1} + C'_{j,m} \psi_{j,m+1}^{n+1} = D'_{j,m} \psi_{j-1,m}^{n+1/2} + E'_{j,m} \psi_{j,m}^{n+1/2} + F'_{j,m} \psi_{j+1,m}^{n+1/2} + G'_{j,m} \quad (23)$$

Pour un point non situé sur un bord, les coefficients sont :

$$A_{j,m} = A'_{j,m} = 1 \quad (24)$$

$$B_{j,m} = B'_{j,m} = i\alpha - 2 - (\Delta x)^2 \frac{1}{2} V_{j,m} \quad (25)$$

$$C_{j,m} = C'_{j,m} = 1 \quad (26)$$

$$D_{j,m} = D'_{j,m} = -1 \quad (27)$$

$$E_{j,m} = E'_{j,m} = i\alpha + 2 + (\Delta x)^2 \frac{1}{2} V_{j,m} \quad (28)$$

$$F_{j,m} = F'_{j,m} = -1 \quad (29)$$

$$G_{j,m} = G'_{j,m} = 0 \quad (30)$$

Sur les bords du domaine, ou sur tout point où une valeur nulle est attribuée à la fonction d'onde, on a :

$$B_{j,m} = B'_{j,m} = 1 \quad (31)$$

et tous les autres coefficients sont nuls.

3.c. Algorithme

La maillage est constitué de matrices à N_x colonnes et N_y lignes. La définition des pas de temps et d'espace, des conditions limites et des sources permet de construire les matrices $A, B, C, D, E, F, G, A', B', C', D', E', F', G'$.

Il faut aussi construire les matrices β, γ pour la résolution des systèmes sur les lignes, et les matrices β', γ' pour la résolution des systèmes sur les colonnes. Pour chaque ligne j , la récurrence à appliquer pour i variant de 1 à N_x :

$$\beta_{1,m} = B_{1,m} \quad (32)$$

$$\gamma_{1,m} = \frac{C_{1,m}}{\beta_{1,m}} \quad (33)$$

$$\beta_{j,m} = B_{j,m} - A_{j,m} \gamma_{i-1,m} \quad (34)$$

$$\gamma_{j,m} = \frac{C_{j,m}}{\beta_{j,m}} \quad (35)$$

Pour chaque colonne i , la récurrence à appliquer pour j variant de 1 à N_y :

$$\beta'_{j,1} = B'_{j,1} \quad (36)$$

$$\gamma'_{j,1} = \frac{C'_{j,1}}{\beta'_{j,1}} \quad (37)$$

$$\beta'_{j,m} = B'_{j,m} - A'_{j,m} \gamma'_{j,m-1} \quad (38)$$

$$\gamma'_{j,m} = \frac{C'_{j,m}}{\beta'_{j,m}} \quad (39)$$

Pour chaque pas de temps n , on doit effectuer une résolution tridiagonale pour chaque ligne, puis pour chaque colonne.

Pour chaque ligne j on commence par calculer la suite suivante pour i variant de 1 à N_x :

$$x_1 = \frac{1}{\beta_{1,m}} (D_{1,m}\psi_{1,m-1}^n + E_{1,m}\psi_{1,m}^n + F_{1,m}\psi_{1,m+1}^n + G_{1,m}) \quad (40)$$

$$x_i = \frac{1}{\beta_{j,m}} (D_{j,m}\psi_{j,m-1}^n + E_{j,m}\psi_{j,m}^n + F_{j,m}\psi_{j,m+1}^n + G_{j,m} - A_{j,m}x_{i-1}) \quad (41)$$

Pour la même ligne j , on calcule les nouvelles valeurs à l'instant $n + 1/2$ par récurrence, en faisant varier de i de N_x à 1 :

$$\psi_{N_x,m}^{n+\frac{1}{2}} = x_{N_x} \quad (42)$$

$$\psi_{j,m}^{n+\frac{1}{2}} = x_i - \gamma_{j,m}\psi_{i+1,m}^{n+\frac{1}{2}} \quad (43)$$

Les valeurs à l'instant $n + 1/2$ sont stockées dans la matrice ψ . On procède alors au parcours des colonnes. Pour chaque colonne i , on calcule la suite suivante pour j variant de 1 à N_y :

$$x_1 = \frac{1}{\beta'_{j,1}} (D'_{j,1}\psi_{i-1,1}^{n+1/2} + E'_{j,1}\psi_{j,1}^{n+1/2} + F'_{j,1}\psi_{i+1,1}^{n+1/2} + G'_{j,1}) \quad (44)$$

$$x_i = \frac{1}{\beta'_{j,m}} (D'_{j,m}\psi_{i-1,m}^{n+1/2} + E'_{j,m}\psi_{j,m}^{n+1/2} + F'_{j,m}\psi_{i+1,m}^{n+1/2} + G'_{j,m} - A'_{j,m}x_{i-1}) \quad (45)$$

puis pour j variant de N_y à 1 :

$$\psi_{j,N_y}^n = x_{N_y} \quad (46)$$

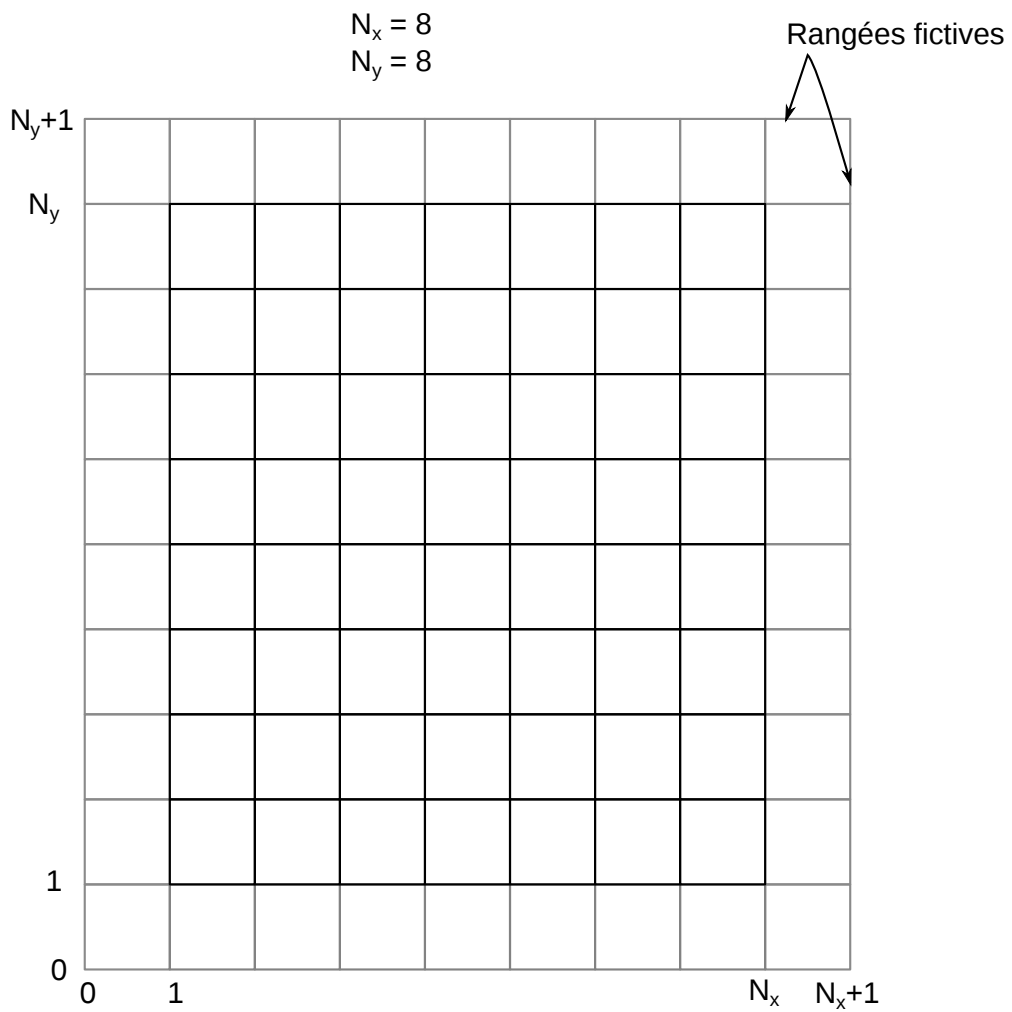
$$\psi_{j,m}^n = x_i - \gamma'_{j,m}\psi_{j,m+1}^n \quad (47)$$

Lorsqu'on traite les lignes, il faut écrire dans une seconde matrice ψ , pour ne pas écraser les valeurs de l'instant n . Il en est de même lorsqu'on traite les colonnes.

Le nombre de matrices utilisées est 20. Pour un maillage $1024x1024$ en flottants 32 bits, la mémoire utilisée est de 80 méga-octets.

Les lignes peuvent être parcourues en parallèle (idem pour les colonnes). On peut donc avec profit utiliser un processeur graphique pour faire ces calculs (via l'API CUDA ou OpenCL). Pour le traitement des lignes, chaque unité de calcul traite une ligne. Les opérations effectuées sur les différentes lignes sont rigoureusement identiques, ce qui est le cas idéal pour une implémentation sur GPU. Bien entendu, toutes les matrices doivent être transférées sur la mémoire de la carte graphique avant le calcul.

On remarque que le traitement des bords (qui doit être identique au traitement des autres points) nécessite un accès à des éléments de ψ hors domaine. Pour résoudre ce problème, on utilise des matrices avec des rangées fictives sur les bords. Le nombre d'éléments dans chaque direction est une puissance de 2 (plus efficace pour l'implémentation GPU). La figure suivante montre un exemple pour $N_x = 2^3$ et $N_y = 2^3$ (les points sont les nœuds de la grille).



Sur cet exemple, les matrices sont 10×10 mais le maillage effectif est 8×8 .