TP 2 : Marche aléatoire

Cours de modélisation numérique

3 mars 2023

Introduction

Dans cet exercice, vous allez devoir implémenter une méthode de marche aléatoire, pour simuler un cas simple d'un homme ivre cherchant à se déplacer. Dans un second temps, vous implémenterez un modèle similaire pour un système de N particules, puis comparerez le comportement observé avec l'équation de diffusion résolue la semaine précédente.

Marche de l'homme saoûl

Afin de prendre en main le problème, intéressons-nous tout d'abord à l'étude du cas simple de la marche d'un homme ivre. On considère une grille rectangulaire de hauteur H et longueur W, que l'on subdivise en N tranches de longueur $\Delta x := W/N$. Au temps $t = 0$, on place un homme au centre du domaine. L'état au temps suivant est déterminé en le faisant avancer d'une distance Δx dans une des quatre directions possibles, tirée aléatoirement. Si on arrive à un des murs (bords) du domaine, la direction de la vitesse est inversée, afin de rester à l'intérieur du domaine.

Travail demandé

Ecrivez vous-mêmes un code Python simulant cette situation. Représentez sur un graphe, la trajectoire de notre homme au cours du temps. En choisissant des temps de simulation assez longs, arriverait-on à explorer chaque case de l'espace ? Qu'en serait-il d'un modèle ou l'espace serait continu (non discrétisé) ?

Modèle discret pour la diffusion

Dans l'exercice de la semaine précédente, nous avons implémenté un schéma permettant de résoudre une équation de diffusion sur un domaine simple. En réalité, le phénomène modélisé par cette équation implique des corpuscules discrets qui se déplacent au hasard, tout comme notre homme ivre : le fameux mouvement *Brownien* des molécules. On aimerait ainsi montrer que la diffusion émerge comme limite statistique d'une assemblée de particules suivant un comportement simple mais probabiliste.

Considérons un domaine comme décrit précédemment. Au temps $t = 0$, on injecte cette fois-ci une quantité de particules P dans les tranches gauche et inférieure du domaine, de sorte à avoir une densité de particules constante (en moyenne) sur ces bords. Ces particules y sont distribuées de manière aléatoire. En outre, on maintient cette quantité de particules constante sur ces bords au cours du temps, en complétant continuellement le nombre de particules. Enfin, les particules qui sortent par les bords droit et supérieur du domaine, sont éliminées.

Travail demandé

Vous trouverez sur Moodle un code à compléter simulant la situation ci-dessus.

Vous devez implémenter vous même la fonction `move_particles(pos)`, qui prend en paramètre le tableau contenant les positions (x, y) de toutes les particules. Cette fonction met à jour les positions des particules, suivant la règle de la marche aléatoire vue au cours. Pour rappel, il faut déplacer chaque particule d'un $\delta x = \delta\vec{v} \cdot \Delta t$, où $\delta\vec{v}$ est un vecteur aléatoire de norme 1, et Δt est le temps d'une itération, qui sera égal à 1 dans notre cas pour simplifier.

Ensuite, faites une figure montrant le nombre de particules dans le domaine, à mi-hauteur de celui-ci (vous pouvez par exemple utiliser un histogramme). Comparez la distribution avec la courbe de la température à mi-hauteur du domaine obtenue en résolvant l'équation de Laplace dans l'exercice précédent.