Marche aléatoire

I - Le programme

Domaines numériques	**Capacités exigibles**
1. Tableaux	
Tableaux à une ou deux dimensions.	Choisir une structure de données appropriée à la modélisation d'un problème physique. Réaliser des opérations algébriques simples sur des tableaux. Utiliser les fonctions de base de la bibliothèque numpy (leurs spécifications étant fournies) pour manipuler des tableaux.
Contenu thématique	**Capacité numérique**
Approche microscopique du phénomène de diffusion.	À l'aide d'un langage de programmation, simuler la marche au hasard d'un grand nombre de particules à partir d'un centre et caractériser l'étalement spatial de cet ensemble de particules au cours du temps.

II - Les outils python

1 - Les bibliothèques à importer

- numpy

  Pour créer des tableaux et pour le tirage aléatoire
- matplotlib.pyplot

  pour les représentations graphiques

2 - Les commandes utiles

Création de tableaux :
- À partir d'une liste. **np.array([liste])** : crée un tableau à partir d'une liste.
- À partir d'un intervalle et d'un nombre d'éléments. **np.linspace(start, end, nbre)** : génère n valeurs entre start et end.
- À partir d'un intervalle et d'un pas. **np.arange(start, end, pas)** : génère des valeurs entre start et end (exclus) espacées du pas.
- Tableau de zéros : **np.zeros(dim)** crée un tableau de 0 de dimension dim.

Boucle **for** avec la fonction **range()** :
La fonction range() permet de généré une liste de parcours.
range(n) renvoi un itérateur de 0 à n-1.

range(a,b) renvoie un itérateur avec a comme premier entier et b-1 comme dernier entier.
range(a,b,pas) renvoie un l'entier a puis des les entiers avec un intervalle pas jusqu'à b exclus.

Pour tracer des courbes :
plt.plot(x,y) pour créer le graphique
plt.show() pour afficher le graphique

Il existe aussi la commande **mean** de la bibliothèque NumPy.

III - Modèle de la marche au hasard unidimensionnelle

On dispose N_0 particules dans un tuyau d'axe (Ox), de section S en $x = 0$ à la date $t = 0$. Ces particules diffusent selon une direction (Ox) avec un coefficient de diffusion D.

1 - Aspect théorique

1-a. Approche macroscopique

L'équation de diffusion des particules est de la forme :

$$\frac{\partial n}{\partial t} = D\frac{\partial^2 n}{\partial x^2}$$

avec $n(x,t)$ la densité particulaire.

Avec la condition initiale $n(x,0) = 0$ pour $x \neq 0$ et les conditions aux limites $\lim\limits_{x\to\pm\infty} n(x,t) = 0$, la densité particulaire a pour expression :

$$n(x,t) = \frac{N_0}{S\sqrt{4\pi Dt}}\, e^{-\frac{x^2}{4Dt}}$$

On caractérise l'étalement des particules (noté r) par la distance quadratique moyenne des particules à $x = 0$.
On a

$$r = \sqrt{\langle x^2\rangle} = \sqrt{\frac{1}{N_0}\int\limits_{-\infty}^{+\infty} x^2 n(x,t) S \mathrm{d}x} = \sqrt{2Dt}$$

1-b Approche microscopique

Pour étudier cette diffusion à l'échelle microscopique, on exploite le modèle de la marche aléatoire : les particules se déplacent toutes d'une même distance ℓ, soit vers la droite, soit vers la gauche, avec la même vitesse v (vitesse quadratique moyenne), chaque déplacement durant $\tau = \ell/v$.

Conséquence : les positions possibles pour les particules sont discrètes $x_i = i\ell$ avec $i \in \mathbb{Z}$ et les particules se déplacent aux instants $t_k = k\tau$ avec $k \in \mathbb{N}$.

En notant $p(x_i, t_k)$ la probabilité de trouver une particule à la position x_i et à l'instant t_k on a :

$$p(x_i, t_k) = \frac{1}{2}\left(p(x_{i-1}, t_{k-1}) + p(x_{i+1}, t_{k-1})\right)$$

1-c. Lien entre les deux approches

ℓ est très petit devant les dimensions macroscopiques. Un passage au continu nous a permis d'établir que :

$$\frac{\partial p}{\partial t} = \frac{\ell^*}{2\tau}\frac{\partial^2 p}{\partial x^2},$$

avec $p(x,t)$ la probabilité de trouver une particule en x à l'instant t.

Le rapprochement des deux points de vue nous permet d'écrire : $D = \dfrac{\ell^2}{2\tau} = \ell v$. On a alors

$$r^2 = \langle x^2 \rangle = \frac{\ell^2}{\tau} t$$

2 - Mouvement d'une particule

On considère dans un premier temps une particule qui part de $x = 0$ à $t = 0$. On travaille avec des grandeurs adimensionnées : $x^* = \dfrac{x}{\ell}$ et $t^* = \dfrac{t}{\tau}$.

On note k le numéro du saut de la particule étudiée ($k = t^*_k$). Le déplacement de la particule à chaque saut vaut donc ± 1 et le pas de temps vaut 1.

Pour alléger les écritures, x_k est la position de la particule après le saut k.

✍ Quelles sont les valeurs minimale et maximale de x_k ?

✘ Écrire une fonction **un_saut** qui modifie la position de particule pendant 1 saut et qui renvoie la nouvelle position.
Vous utiliserez la bibliothèque random pour tirer aléatoirement ± 1 : `np.random.choice([-1,1])`.

✘ On s'intéresse à la séquence de déplacements de la particule lors de $K = 50$ sauts successifs. Créer deux tableaux : un tableau pos_part qui contient les positions successives de la particule, $x_0, x_1, \dots, x_K$ et un tableau nbr_saut qui contient les numéros des sauts $0, 1, \dots, K$.

✘ Tracer l'évolution des positions en fonction du temps (ou du numéro du saut).

✘ Tracer plusieurs fois ce graphe, que constatez-vous ?

✘ Calculer la position moyenne de la particule des positions successives

$$\langle x \rangle = \frac{1}{K}\sum_{j=1}^{j=K} x_j,$$

ainsi que la moyenne des carrés des abscisses

$$r^2 = \langle x^2 \rangle = \frac{1}{K}\sum_{j=1}^{j=K} x_j^2.$$

À faire pour $k = 10, 20, 50, 100$. Commentaires ?

3 - Étude de l'étalement pour N particules

On s'intéresse maintenant plus particulièrement à l'étalement des particules au cours du temps. À $t = 0$, on place N particules en $x = 0$. Chaque particule suit une marche aléatoire.

✘ Créer un tableau qui va stocker les abscisses des N particules. Ce tableau est actualisé à chaque pas de temps. En parallèle, créer un tableau x2moyen qui va stocker les moyennes des carrés des abscisses à chaque pas de temps.

✘ Tracer les valeurs de x2moyen en fonction du temps et superposer la courbe $\sqrt{\langle x^2 \rangle} = t$. Exécuter votre programme sur 100 pas de temps avec 100, puis 1000 puis 10000 particules.

IV - Marche au hasard dans un plan

On dispose N_0 particules à l'origine d'un plan à $t = 0$. Ils diffusent avec un coefficient de diffusion D.

On veut étudier l'évolution avec le temps de la distance quadratique moyenne parcourue par les atomes : r2moyen $= \langle r^2 \rangle = \langle x^2 + y^2 \rangle$.

1 - Modèle

Chaque atome fait une succession de pas de longueur ℓ accompli en une durée τ.
À chaque pas la direction du vecteur vitesse varie aléatoirement dans l'intervalle $0, 2\pi$.

2 - Implémentation en Python

Les tableaux numpy x et y stockent les coordonnées des particules à chaque instant.

La commande **np.random.uniform(0,2*np.pi)** permet de tirer aléatoirement une valeur entre 0 et 2π.

r2moyen stocke la distance quadratique moyenne en fonction du temps.

✘ Vérifier que la loi $r^2 = \dfrac{\ell^2}{\tau} t$ est bien en accord avec votre simulation.