

Modélisation d'un gaz parfait

Groupe 19 - 24 Novembre 2023

Membres du groupe

LUCIE BROUTIN

GUILLAUME DUBOIS

VIANNEY PARENT

SOLÈNE DUQUENNE

HADRIEN HENRIOT

SALOMÉ FONVIELLE

Sommaire

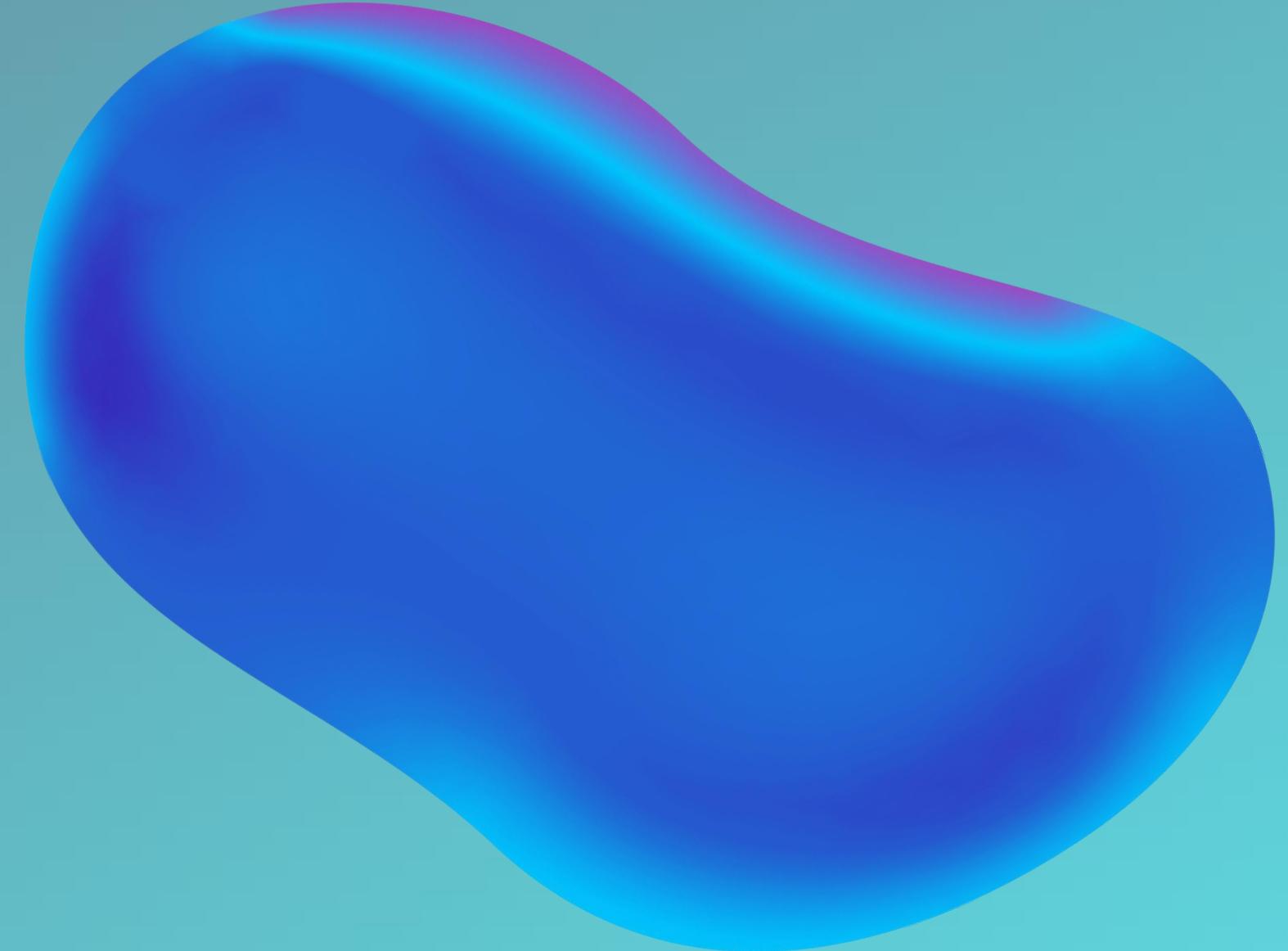
Introduction

I - Modèles physiques

II - Objectifs et structure du code

III - Démonstration du code

Conclusion



Introduction

UNE INTERFACE DYNAMIQUE

Mise en mouvement de N particules dans une enceinte avec prise en compte des chocs.



A DESTINATION DES ENSEIGNANTS ET DES ELEVES

Les présentations sont des outils de communication pouvant être utilisés pour des démonstrations ou des conférences.

POUR OBSERVER LE COMPORTEMENT D'UN GAZ PARFAIT

L'utilisateur peut régler la vitesse initiale, le nombre de particules et leur dimension.



Introduction

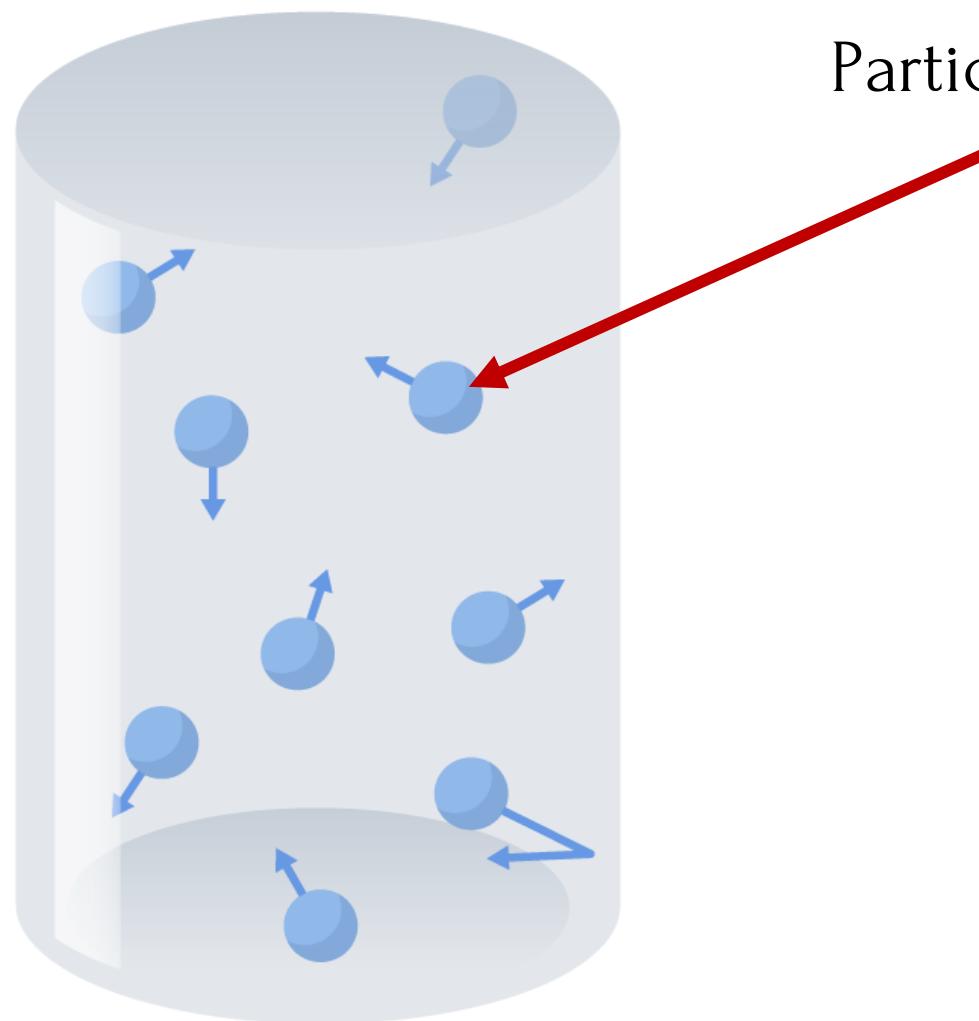
MVP :

Visualisation du mouvement d'une particule dans un gaz.

I – Modèles physiques

Modélisation du gaz

Modèle des sphères dures



Particule de rayon fixé, soumise uniquement à de possibles chocs

Modélisation des chocs entre les particules

Deux particules: (m_1, v_1) et (m_2, v_2)

Modèle des chocs élastiques:

$$\vec{u}_1 = \vec{v}_1 - \frac{2m_2(\vec{v}_1 - \vec{v}_2) \cdot (\vec{r}_1 - \vec{r}_2)}{M|\vec{r}_1 - \vec{r}_2|^2} (\vec{r}_1 - \vec{r}_2)$$

$$\vec{u}_2 = \vec{v}_2 - \frac{2m_1(\vec{r}_2 - \vec{r}_1) \cdot (\vec{v}_2 - \vec{v}_1)}{M|\vec{r}_1 - \vec{r}_2|^2} (\vec{r}_2 - \vec{r}_1)$$

Définition des vitesses initiales

$$P = \frac{n^* m u^2}{3}$$

$$\frac{1}{2} m u^2 = \frac{3}{2} k_b T$$

$$u = \sqrt{\frac{3}{m} k_b T}$$

P la pression

n^* densité particulaire

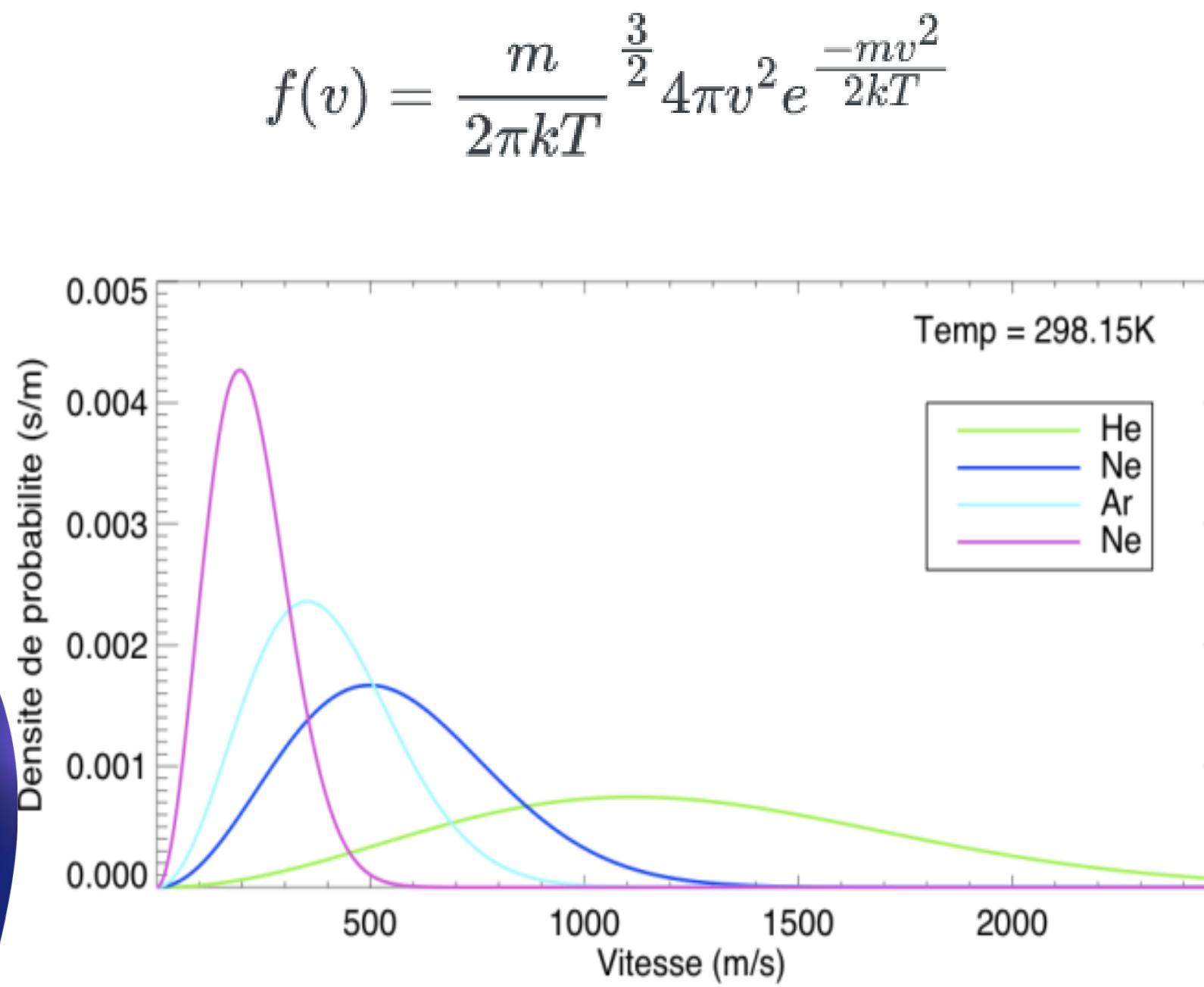
m la masse d'une particule

u la vitesse moyenne d'une particule

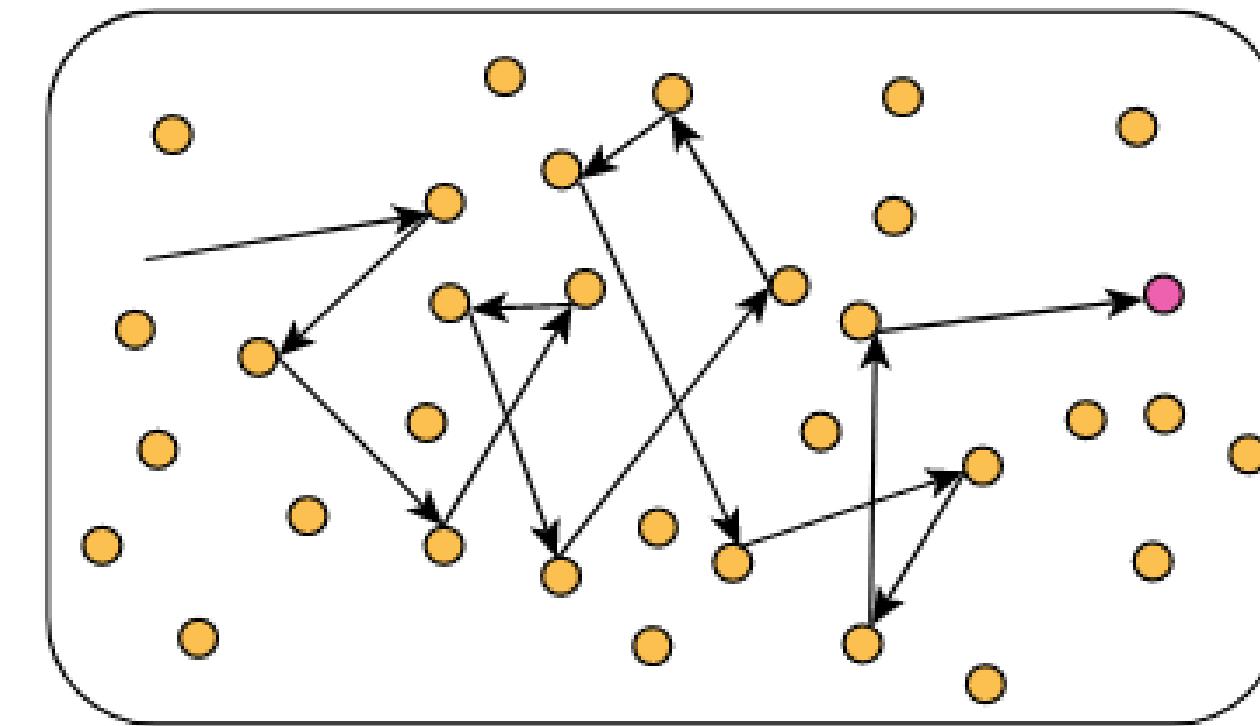
k_b constante de Boltzmann

Phénomènes observés dans la simulation

Répartition des vitesses suivant la loi de Boltzmann



Mouvement Brownien et diffusion

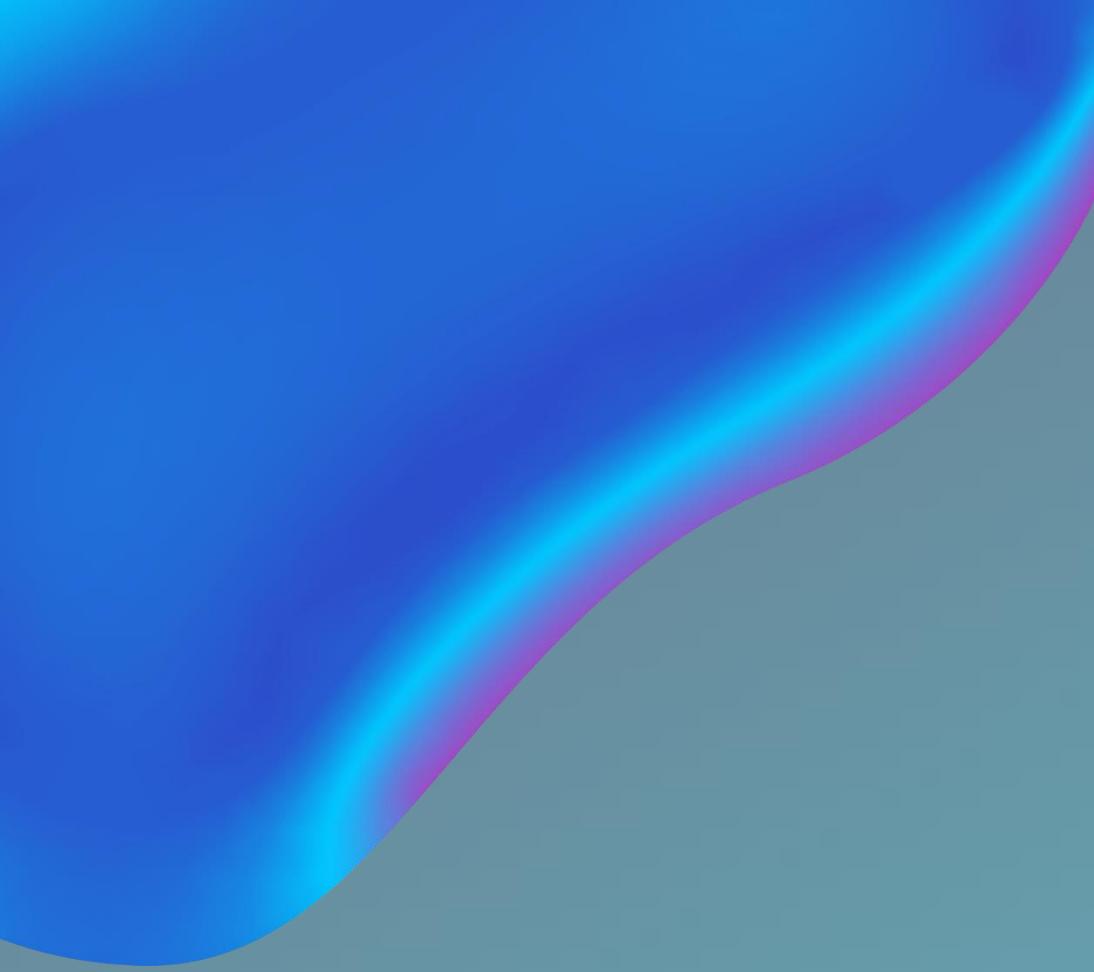


$$D_\phi \Delta \phi = \frac{\partial \phi}{\partial t}$$

Cohérence physique de la simulation

Gaz de référence : Diazote (N₂)

Réalité	Simulation
Longueur : L	Longueur : $L' = L/1000$
Vitesse : V	Vitesse : $V' = V/1000$
Masse d'une particule : m	Masse d'une particule : $m' = m \times 4,6 \cdot 10^{-26}$ kg
1 particule	10^{22} particules



II - Objectifs et structure du code

Objectifs

01 Marche aléatoire

Jalon 1 : Déplacement aléatoire d'une particule dans 4 directions

Jalon 2 : Déplacement aléatoire d'une particule dans toutes les directions

02 Création d'un nuage de points

Jalon 1 : Représentation fixe d'un nuage

Jalon 2 : Mise en mouvement du nuage

Jalon 3 : Rebonds sur les bords du cadre

03 Collisions entre les particules et suivi d'une particule

Jalon 0: représentation sous pymunk/pygame de la situation

Jalon 1 : Mise en place des collisions

Jalon 2 : Mise en évidence d'une particule

Jalon 3 : Suivi de la trajectoire de cette particule

Objectifs

04 Crédit de l'interface graphique

Jalon 1 : Crédit d'une maquette

Jalon 2 : Implémentation simple de l'interface graphique

Jalon 3 : Permettre le paramétrage de l'interface

05 Amélioration, autre interface graphique, vérification des lois physiques (Bonus)

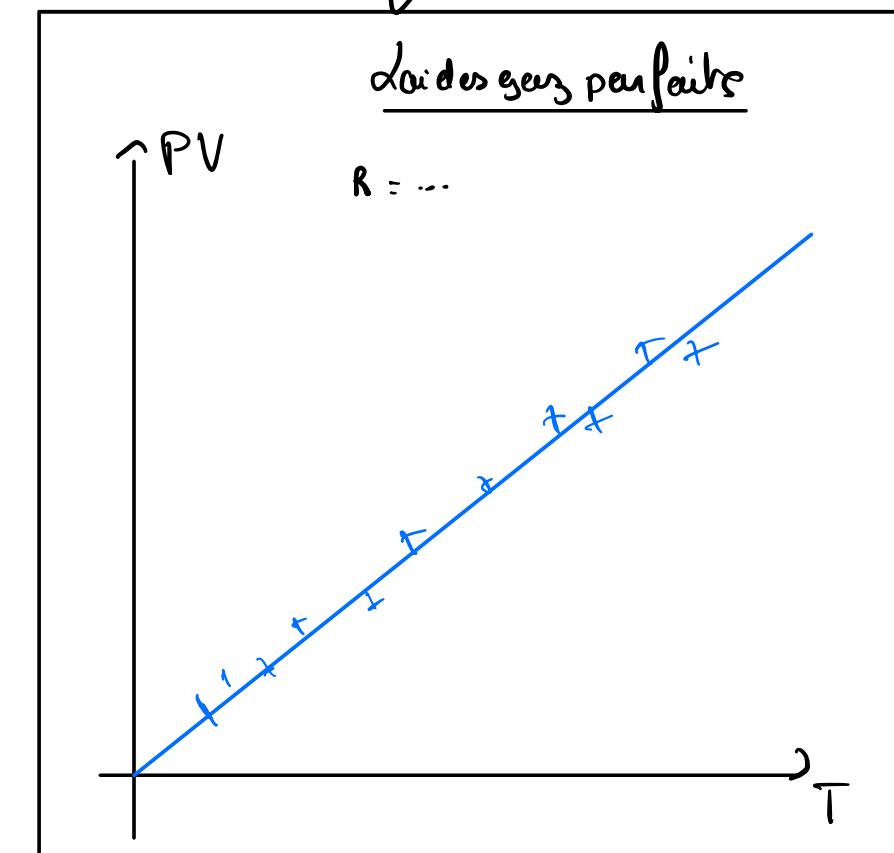
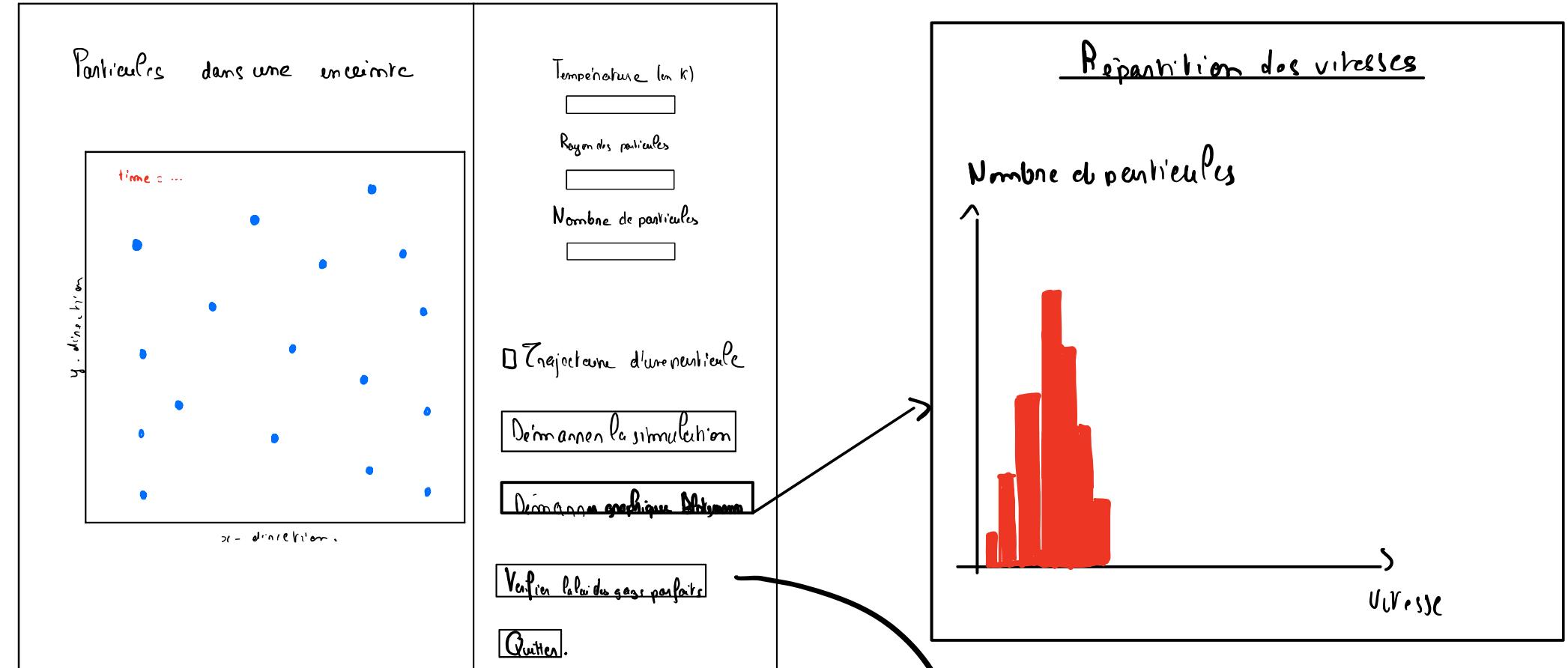
Jalon 1 : Vérification des lois physiques par le modèle et mise en cohérence des grandeurs physiques

Jalon 2 : Qualité de l'interface graphique et options supplémentaires

Jalon 3 : Tentative 3 D

Objectifs

Maquette de
l'interface graphique



Structure du code

Code principal :

- Partie 1: Définition de la classe particule : masse, rayon, vitesse et position
- Partie 2 : Implémentation de fonctions auxiliaires :
 - Détection des chocs
 - Déplacement dans le cas d'une collision entre deux particules
 - Chocs avec les bords
 - Calcul de la pression dans l'enceinte
- Partie 3 : Création de l'animation
 - Positionnement initial des particules de manière aléatoire
 - Fonction "animation" donnant la position d'une particule à partir de sa position précédente avec prise en compte des collisions.
- Partie 4 : Amélioration de l'interface graphique avec Tkinter

Structure du code

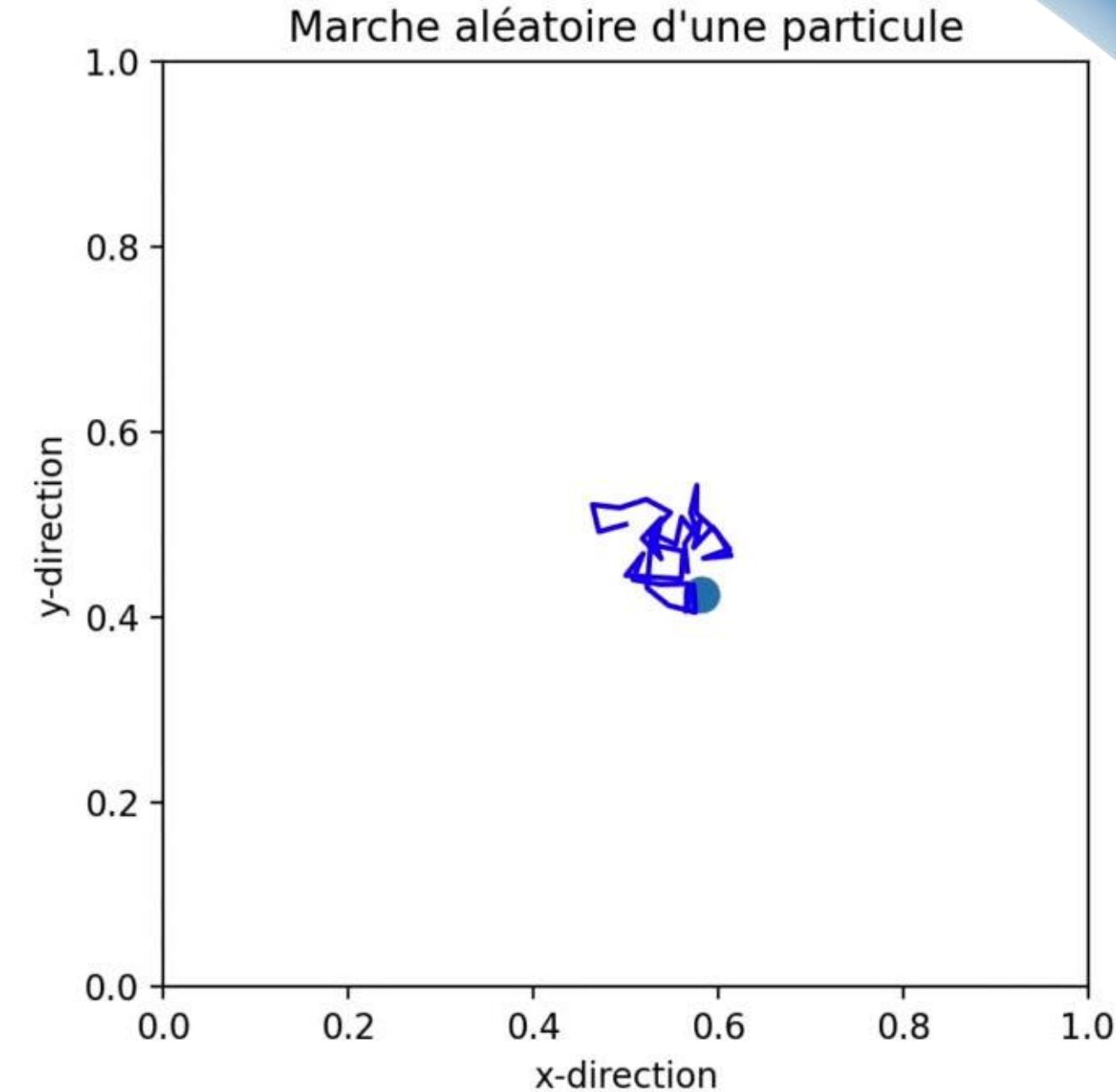
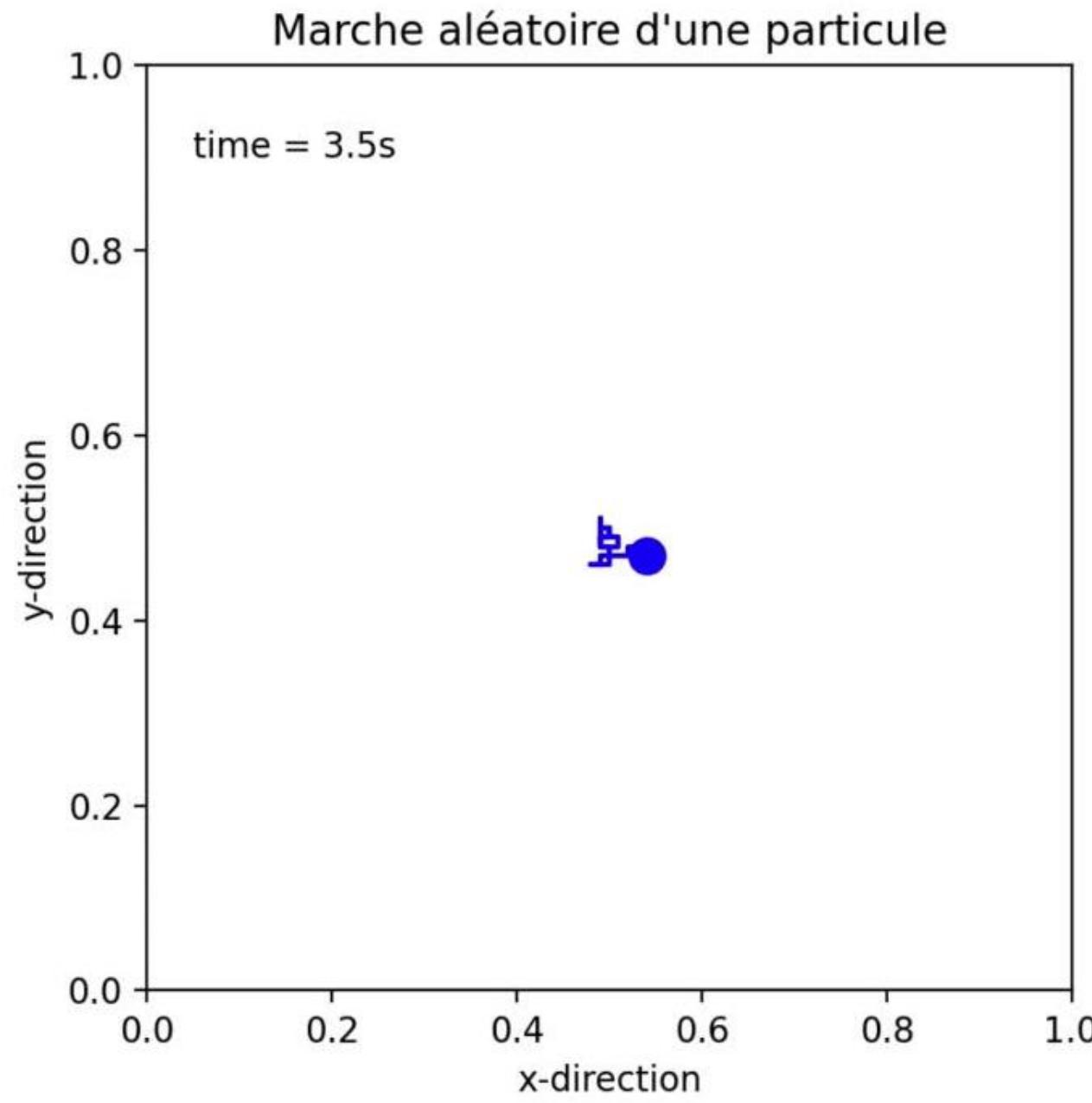
Codes secondaires :

- Marche aléatoire :
 - Déplacement dans 4 directions équiprobables et tracé de la trajectoire
 - Déplacement dans toutes les directions
- Mouvement brownien sur Pymunk :
 - Création des particules et des bords avec le module "body"
 - Choix de l'élasticité entre les corps et Pymunk gère lui-même les collisions

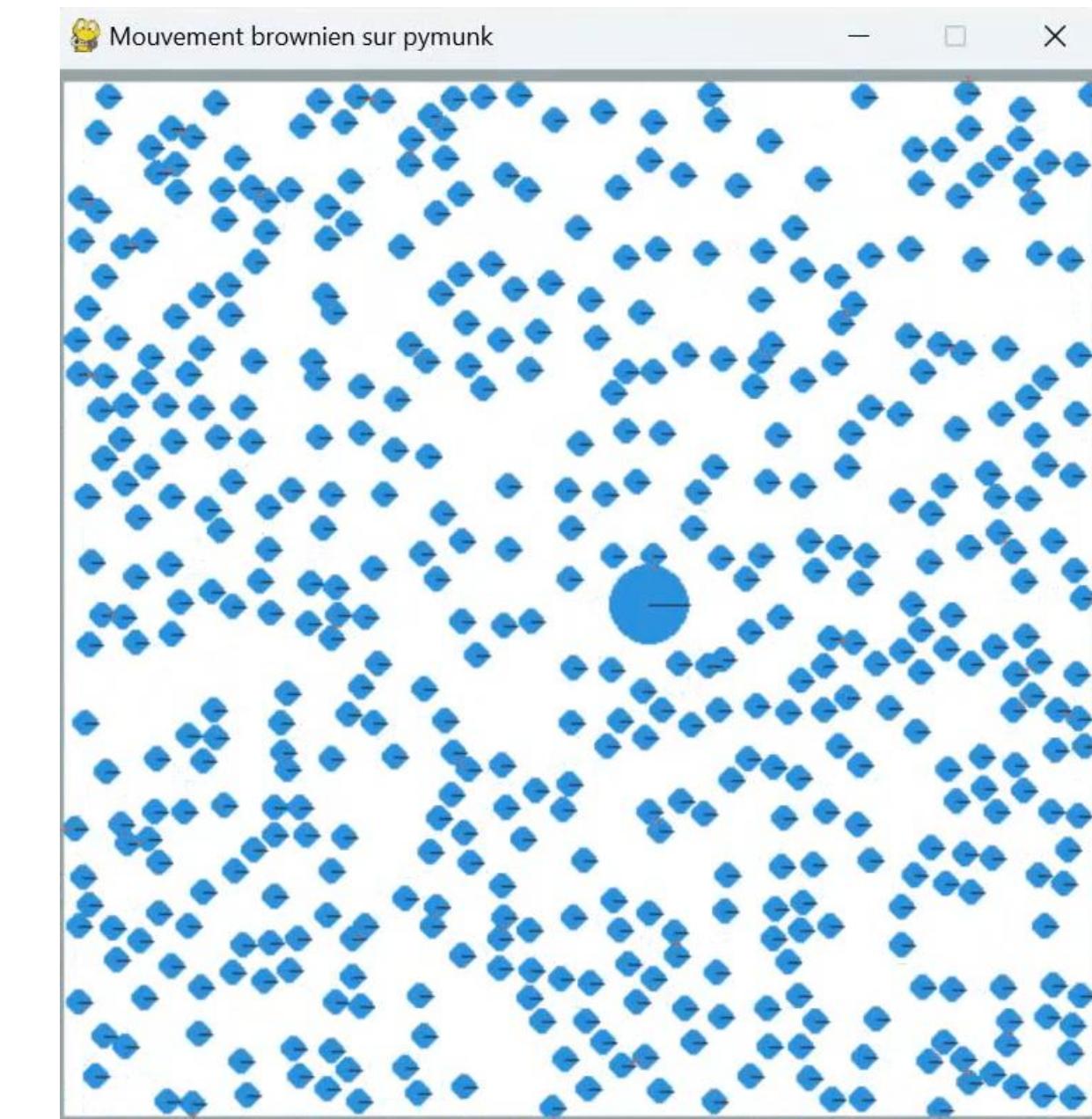
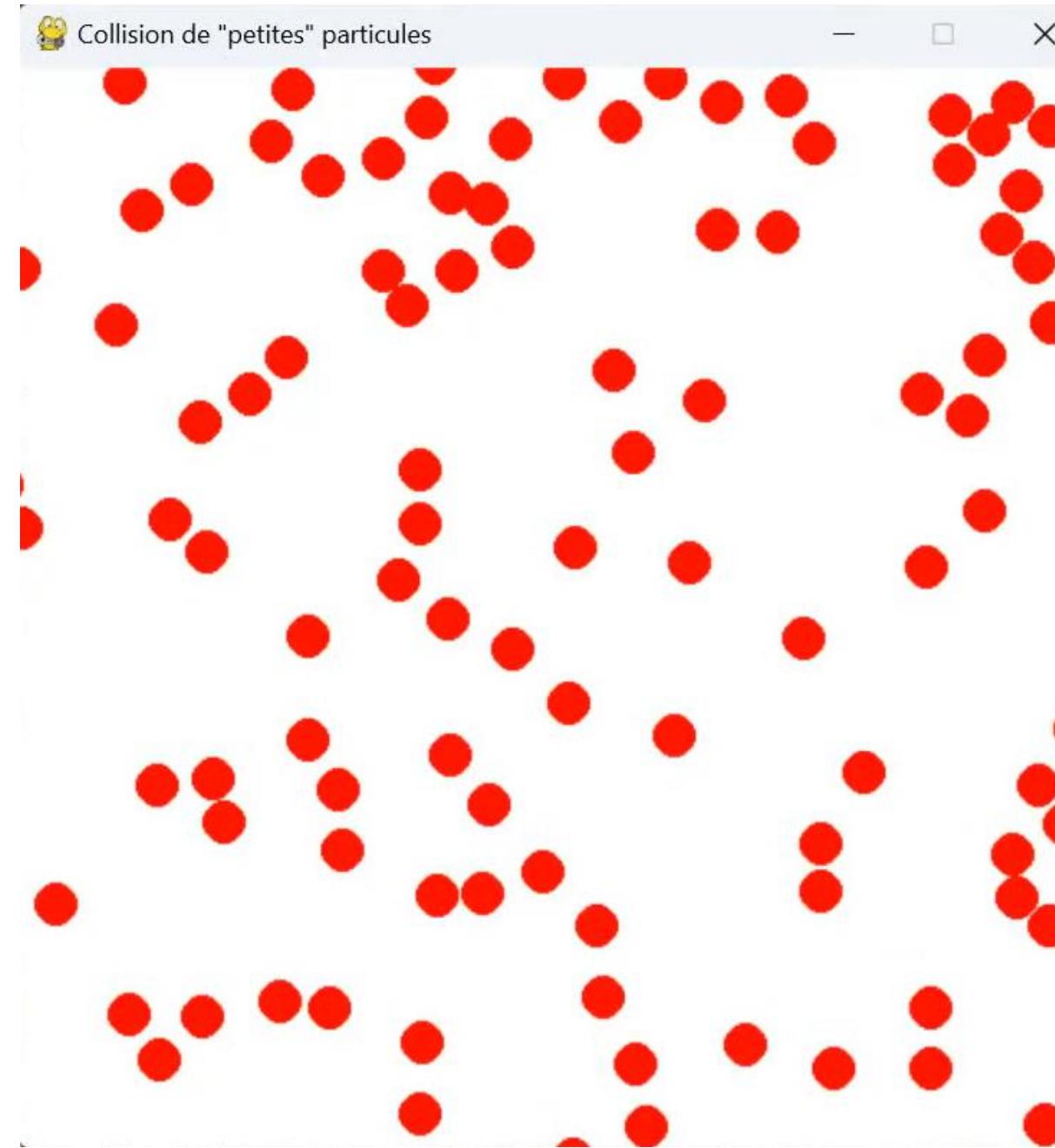


III – Démonstration du code

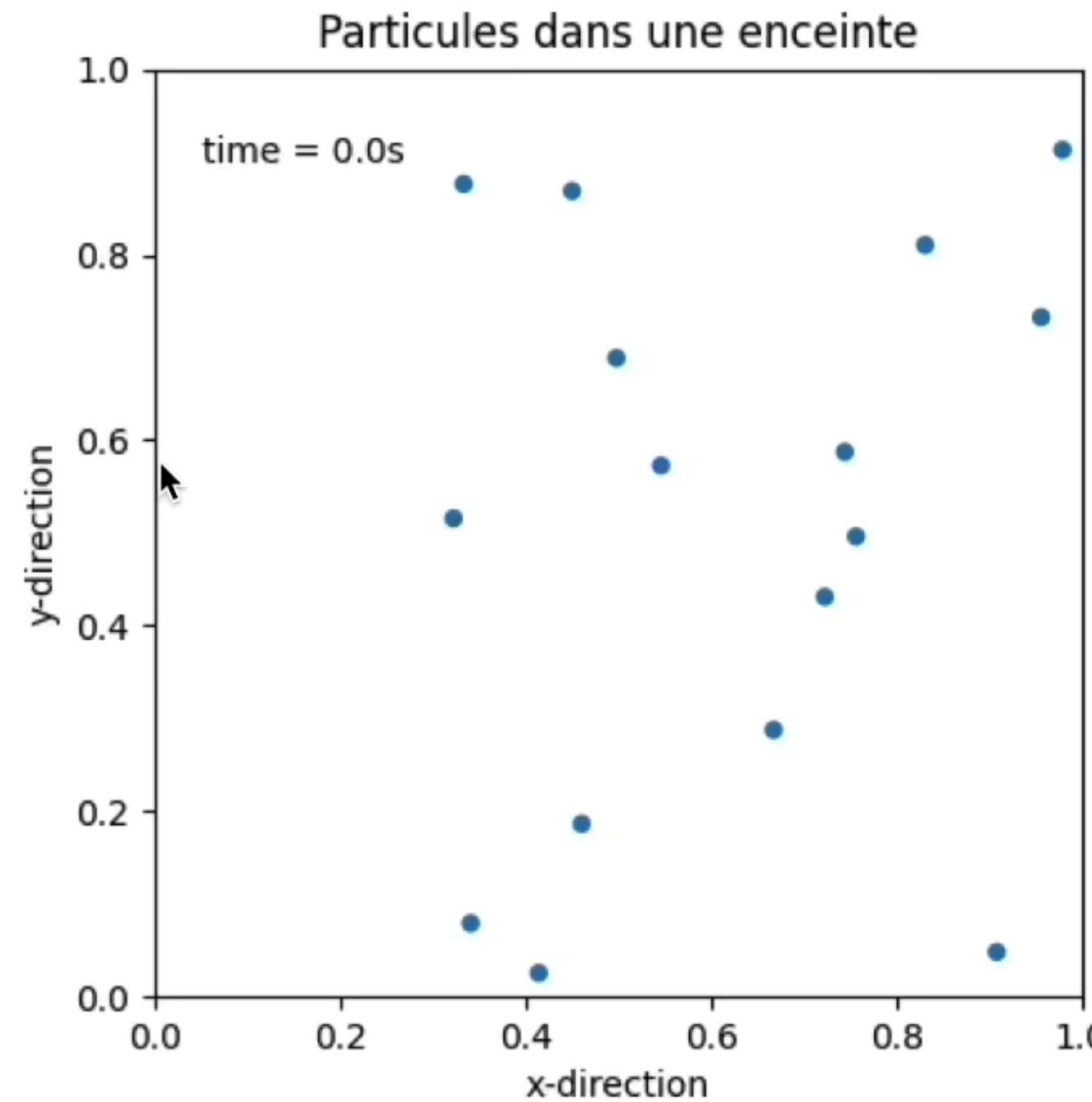
1. Marche aléatoire (Objectif 1)



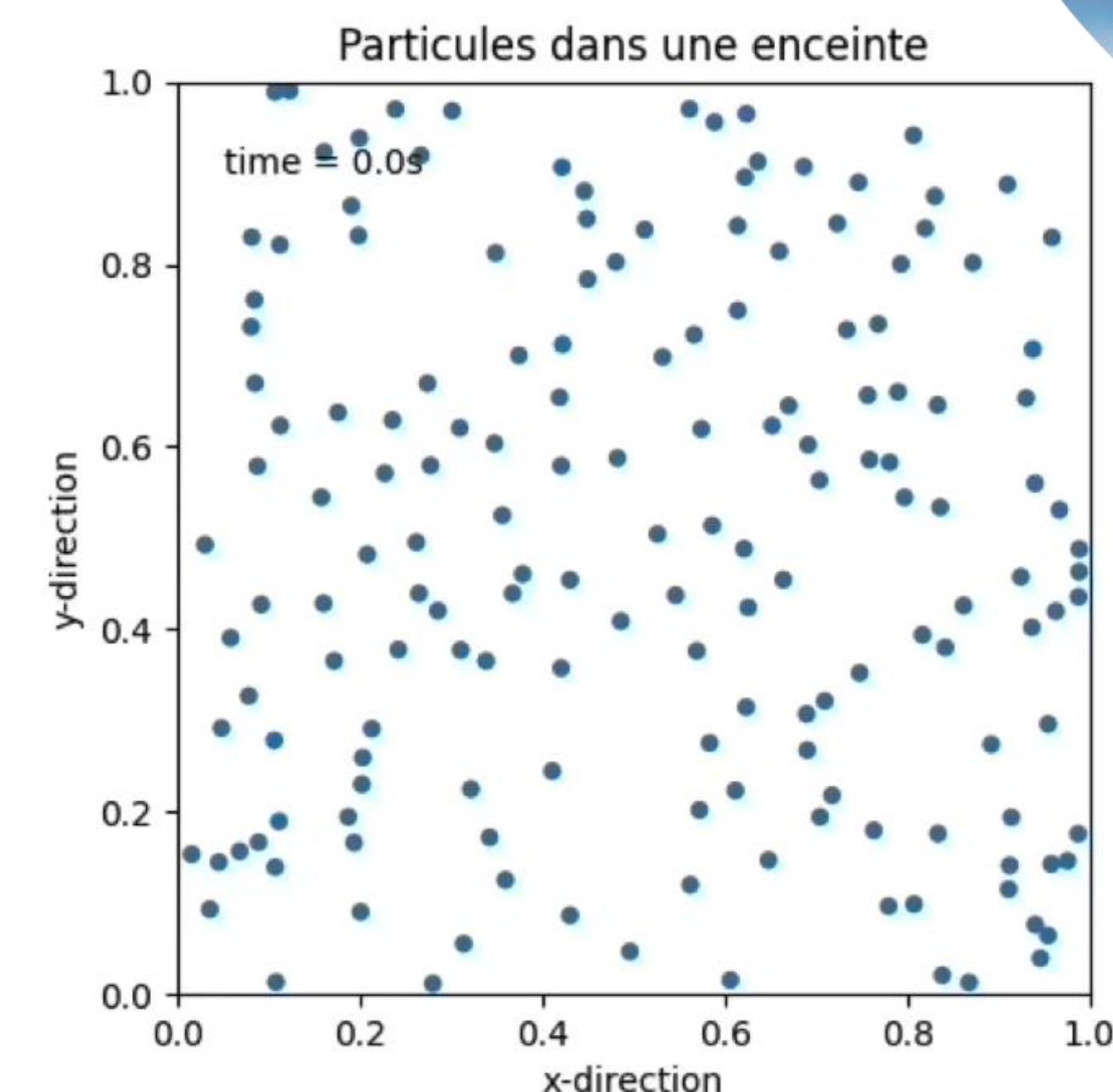
2. Représentation Pymunk (Objectif 2)



3. Représentation Matplotlib et Tkinter (Objectifs 3, 4)

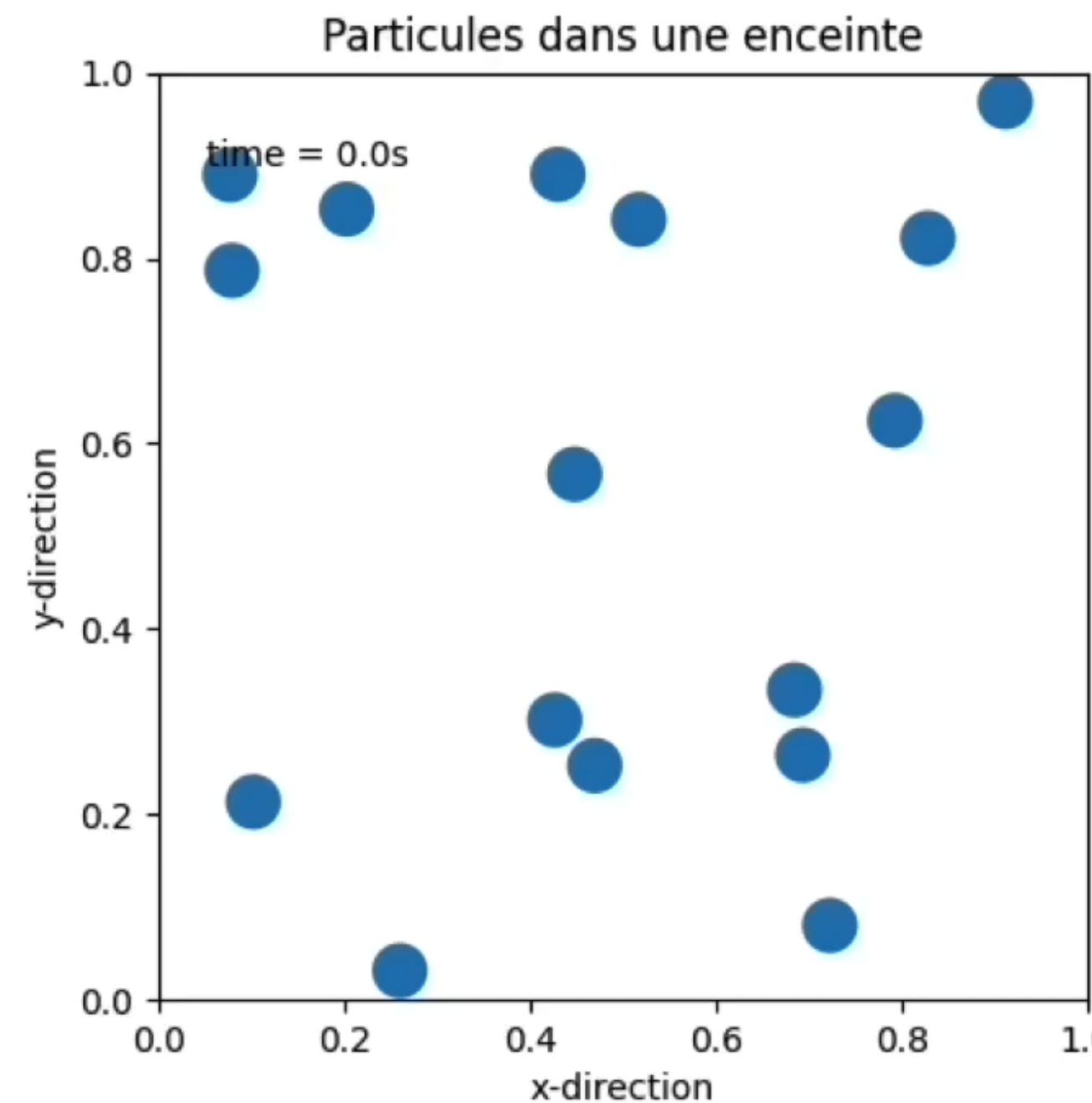


T = 300 K
r = 1.0
N = 16

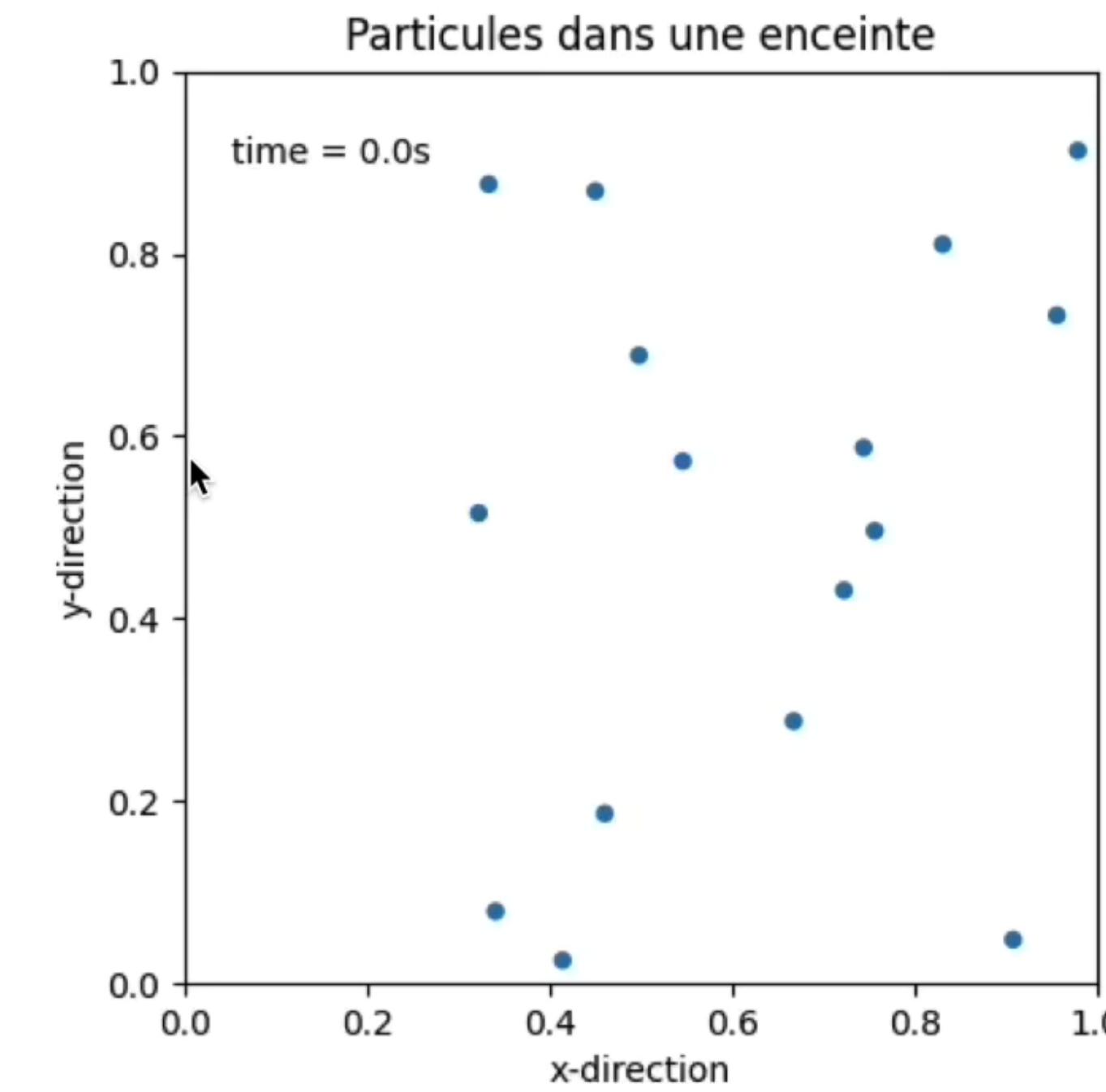


T = 300 K
r = 1.0
N = 163

3. Représentation Matplotlib et Tkinter (Objectifs 3, 4)



T = 300 K
r = 3.0
N = 16



T = 300 K
r = 1.0
N = 16

3. Représentation Matplotlib et Tkinter (Objectifs 3, 4)

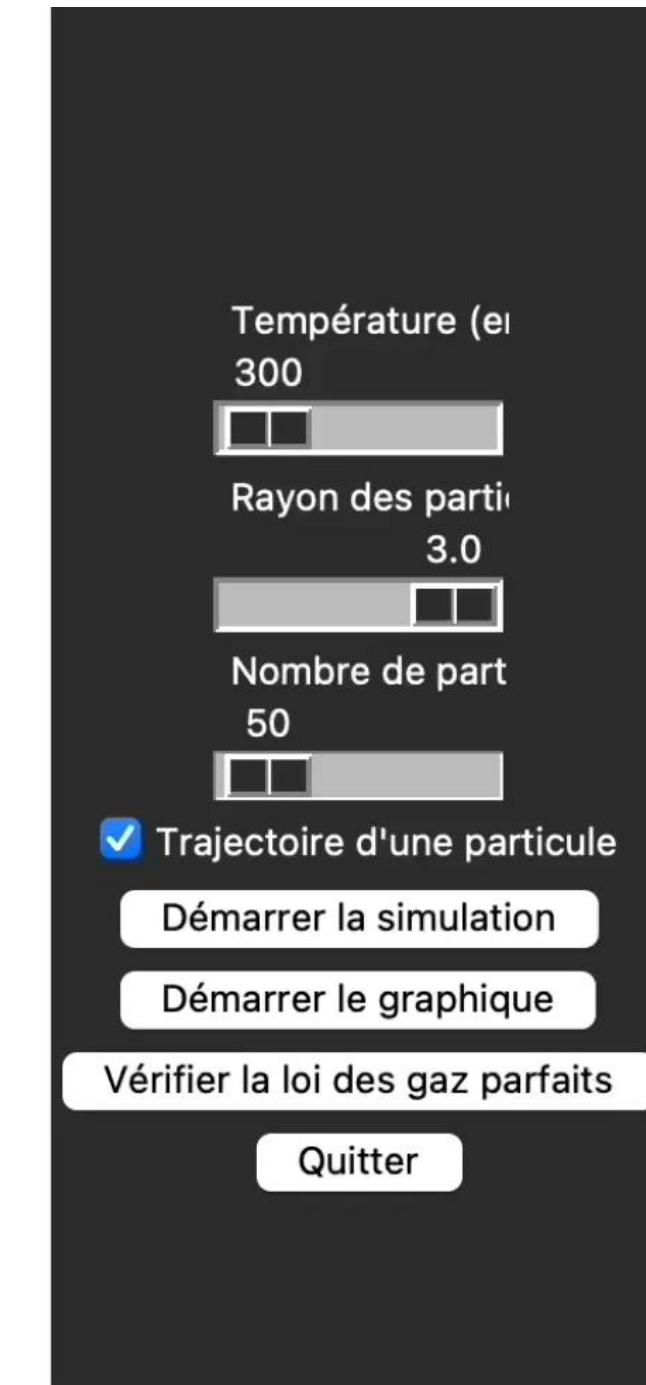
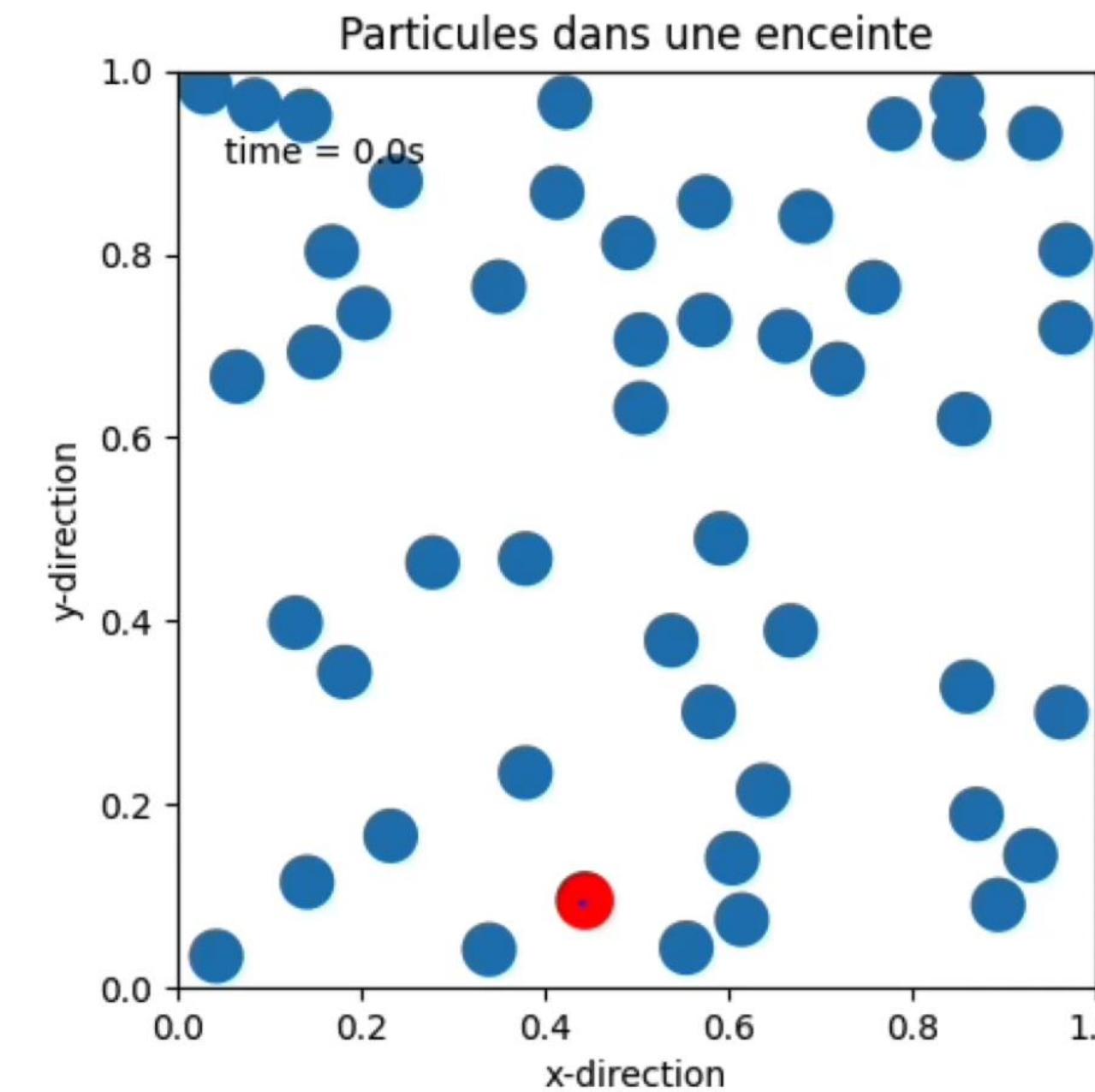
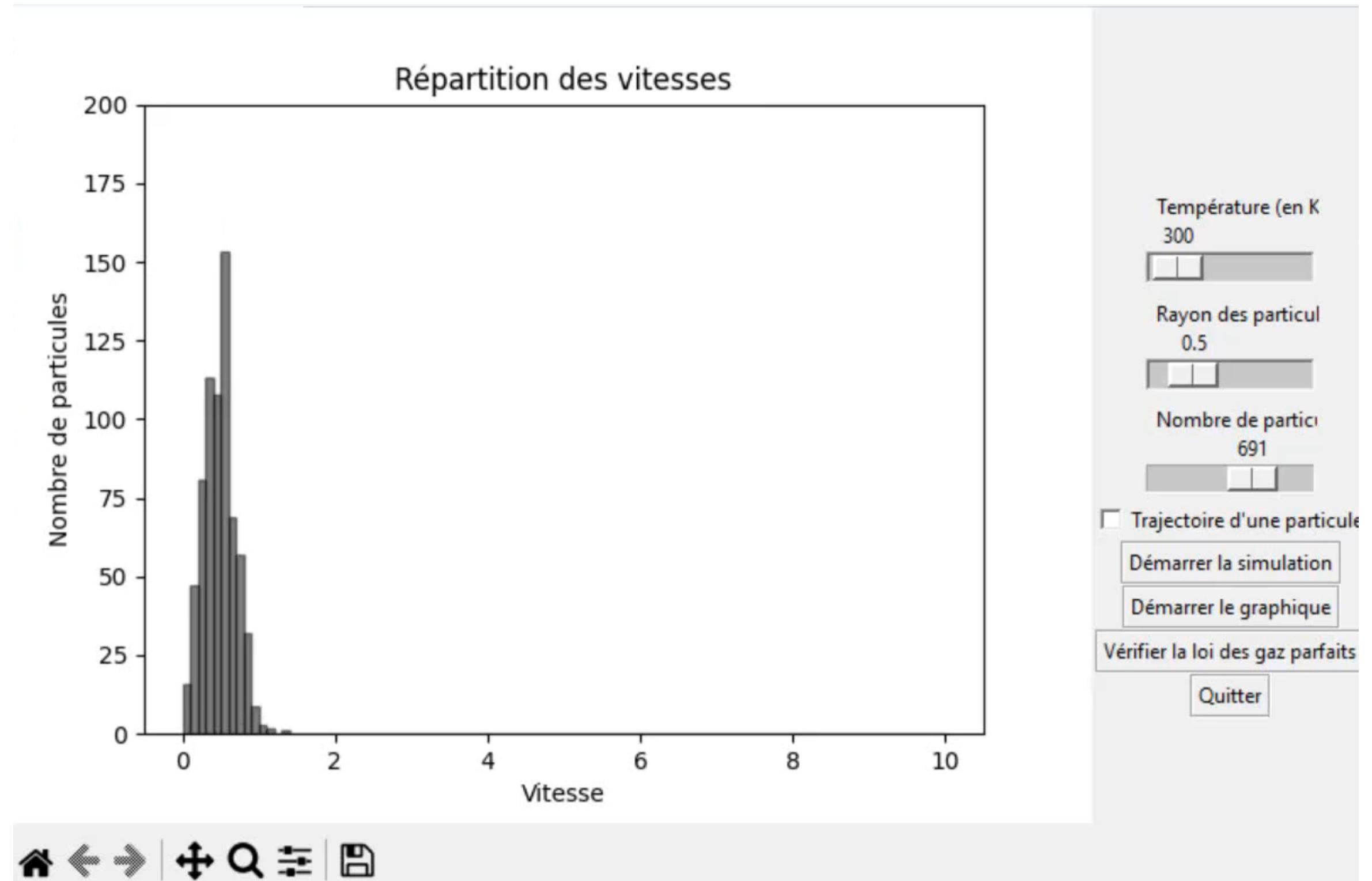
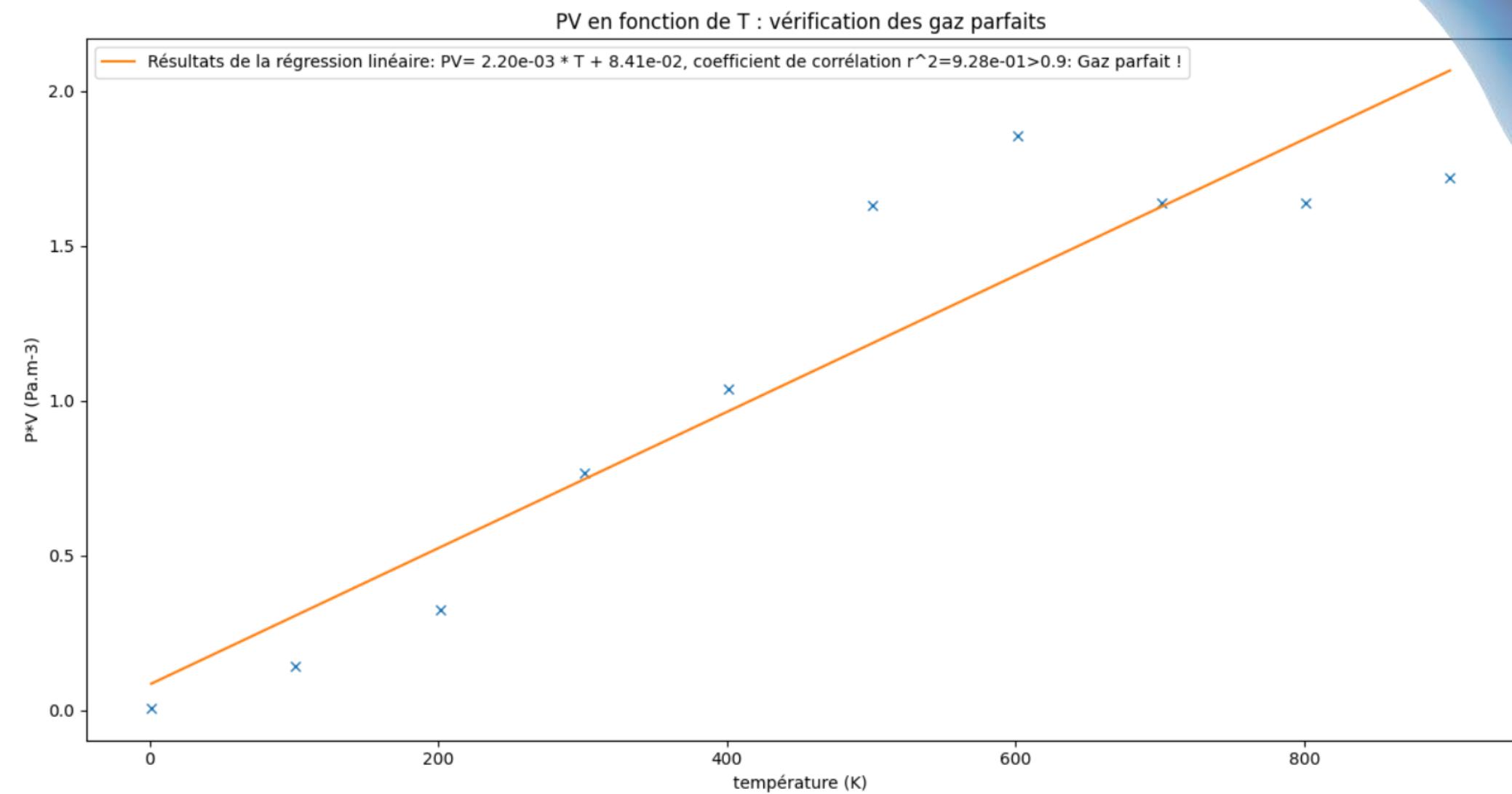
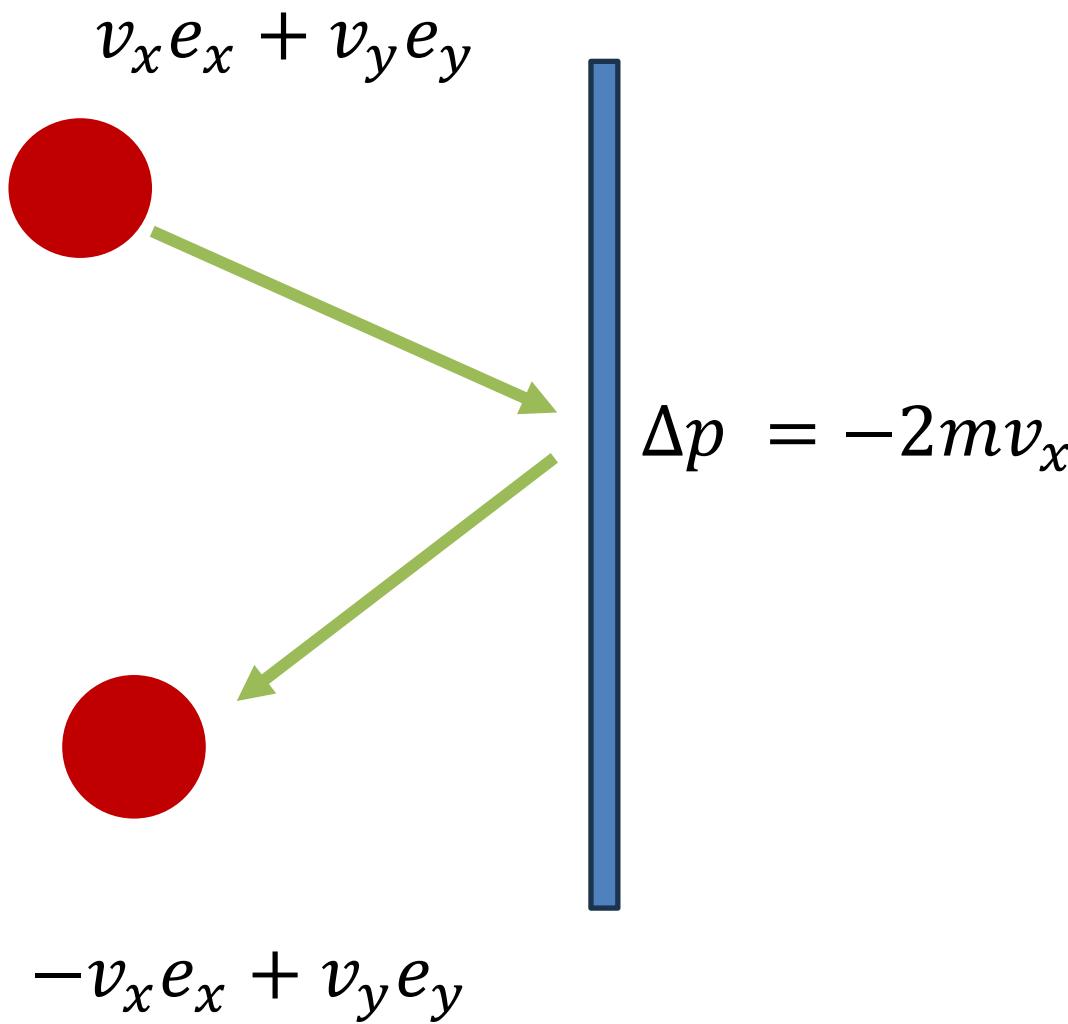


Diagramme de répartition des vitesses



4. Capteur de Pression et vérification du comportement de gaz parfait

Principe de la mesure :



5. Couverture du code par les tests

Coverage report: 81%

coverage.py v7.3.2, created at 2023-11-22 16:11 +0100

<i>Module</i>	<i>statements</i>	<i>missing</i>	<i>excluded</i>	<i>coverage</i>
<code>interface_graphique__init__.py</code>	0	0	0	100%
<code>interface_graphique\definitions_objets.py</code>	61	0	0	100%
<code>interface_graphique\fonctions_auxiliaires.py</code>	51	32	0	37%
<code>test_fonction_avance.py</code>	9	0	0	100%
<code>test_fonction_choc_rebord.py</code>	16	0	0	100%
<code>test_fonction_collision.py</code>	9	0	0	100%
<code>test_fonction_place_particules.py</code>	7	0	0	100%
<code>test_fonction_rebord.py</code>	13	0	0	100%
<code>test_particule_alea.py</code>	6	0	0	100%
Total	172	32	0	81%

coverage.py v7.3.2, created at 2023-11-22 16:11 +0100

IV - Conclusion

MVP atteint :

- Visualisation d'un gaz
- Influence des différents paramètres (température, densité, rayon)
- Interface facilement utilisable

Améliorations envisageables :

- Représentation 3D du gaz
- Étude du comportement du gaz si l'enceinte comporte une ouverture
- Optimisation du code (temps de calcul et fluidité)