

Particules micrométriques proches d'un mur

Stage optionnel de M1 Mathématiques appliquées

Élodie Millan

Juin - juillet 2020

Sous la direction de :
Thomas Salez
Yacine Amarouchene

Université de Bordeaux,
Master MIMSE

Table des matières

Remerciements	2
Notation mathématique	2
Introduction	3
0 Outils mathématiques pour l'analyse de marches aléatoires	4
1 Particule dans un espace infini	5
1.1 Mise en équation du problème	5
1.2 Cas sur-amorti	7
1.3 Cas inertiel	9
2 Particule brownienne près d'un mur dur	12
2.1 Mise en équation du problème	12
2.2 Cas sur-amorti	13
3 Choix d'optimisation numérique et perspectives d'amélioration	17
3.1 Choix numériques et d'optimisation	17
3.2 Perspective de parallélisation	18
Conclusion	20

Remerciements

Je remercie Thomas Salez^I et Yacine Amarouchene^{II} pour leur partage de connaissances et les échanges avec eux, ainsi que de m’avoir acceptée dans leur équipe pour ce projet de stage puis prochainement de thèse et de m’avoir permis d’effectuer ce stage fort enrichissant. Un grand merci à Maxime Lavaud^{III} pour son aide et ses conseils numérique et à mon ami Pierre^{IV} pour ses relectures de ce manuscrit. Ce stage et mon projet de thèse n’auraient pas pu être possibles sans eux.

Enfin, je remercie cette période de confinement de m’avoir permis d’être encore plus autonome que je ne l’étais.

Notations mathématiques

$\langle \cdot \rangle$: Valeur moyenne
$\ \cdot \ $: Norme
$ \cdot $: Valeur absolue
$\vec{\cdot}$: Vecteur

I. Chargé de recherche au LOMA, équipe théorie de la matière condensée, Université de Bordeaux.

II. Chargé de recherche au LOMA, équipe photonique et matériaux, Université de Bordeaux.

III. Doctorant au LOMA, équipe théorie de la matière condensée, Université de Bordeaux.

IV. plus connu sous le nom de Thopi.

Introduction

Sous l'effet de l'agitation thermique des molécules d'un fluide de viscosité η , des particules micrométriques de rayon R immergées présentent un mouvement aléatoire, généralement appelé mouvement Brownien, comme présenté en figure 1. Sa première description mathématique est due à Albert Einstein en 1905 et les physiciens le décrivent généralement avec un coefficient de diffusion $D = \frac{k_B T}{6\pi\eta R}$, avec $k_B T$ l'énergie thermique valant ≈ 4 pN.nm à température ambiante. L'autre principale caractéristique du mouvement Brownien est que le déplacement aléatoire suit une loi de probabilité normale.

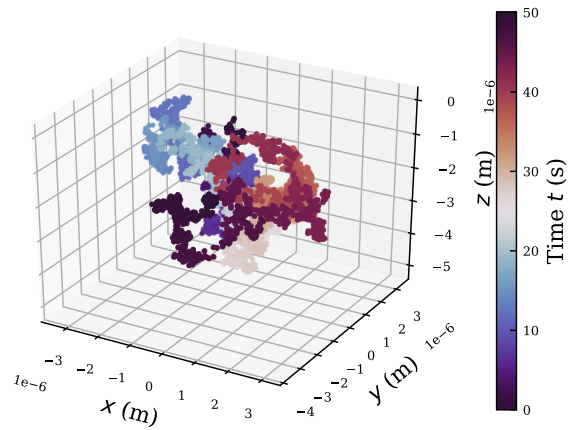


FIGURE 1 – Mouvement brownien d'une particule de rayon $R = 10^{-6}$ m.

Ce type de description est très bien connu à ce jour et n'est valable que dans le cas où le fluide est un milieu "infini" mais ne s'applique qu'à de très rares cas de la vie de tous les jours. En chimie, en médecine ou même en biologie, nous rencontrons essentiellement des systèmes confinés due à la présence de parois molles, d'interface liquide, de membrane, etc. Par exemple, Bocquet et Charlaix [1] ont étudié les canaux d'aquaporine^V qui jouent un rôle de filtre pour l'eau et sont sélectifs pour certaines molécules dans les membranes biologiques. Ce processus biologique a alors donné l'idée d'osmose inverse qui consiste à inverser le déplacement de molécule de sel dans un canal osmotique^{VI}, ceci dans le but de pouvoir contrôler le déplacement de molécules dans des canaux. Beaucoup d'autres applications montrent qu'il est important d'étudier le confinement d'une particule brownienne proche d'une paroi dure, molle ou en interaction avec la particule.

C'est pourquoi, au cours de ce stage, j'ai étudié l'influence d'un mur dur sur une particule micrométrique. Dans une première partie, j'ai modélisé le cas simple d'une particule dans un fluide infini sans inertie puis avec inertie. Dans une seconde partie j'ai ajouté l'influence d'un mur dur. Ce stage étant le préambule de ma thèse, l'idée est de construire les bases qui pourront me servir à l'étude d'un ensemble de particules actives^{VII} proches de parois déformables ou encore l'influence d'une forte concentration de particules proches d'une paroi dure que je vais effectuer au LOMA^{VIII}.

V. Protéines membranaires.

VI. Une paroi osmotique est une paroi poreuse laissant passer certaines molécules, comme l'eau, mais pas d'autre comme le sel NaCl

VII. Une particule active est un objet microscopique auto-propulsé comme une bactérie ou une particule avec des propriétés asymétriques en surface.

VIII. Laboratoire Ondes et Matière d'Aquitaine, Université de Bordeaux, Talence.

0 Outils mathématiques pour l'analyse de marches aléatoires

Du fait de l'aspect aléatoire des trajectoires des particules que j'ai étudiées, il nous faut définir quelques outils mathématiques d'analyse d'une marche aléatoire. On définit \vec{e}_x , \vec{e}_y et \vec{e}_z les trois vecteurs directeurs des dimensions d'espace, x , y et z les variables de positions le long de ces axes et t le temps.

La $\text{MSD}_{1D}^{\text{IX}}$ d'une trajectoire en une dimension spatiale correspondant au moment d'ordre 2 de la position selon un axe \vec{e}_x tel que :

$$\langle |x(t + \Delta t) - x(t)|^2 \rangle = \frac{1}{N} \sum_{i=1}^N |x^{(i)}(t + \Delta t) - x^{(i)}(t)|^2, \quad (1)$$

où $x^{(i)}(t)$ est la position selon une dimension d'espace, ici $i \in \{1, 2, 3\}$ que l'on associe à $x^{(i)} \in \{x, y, z\}$ et Δt la durée entre deux mesures de positions de la particule. Dans le cas à 3 dimensions, la MSD_{3D} s'écrit :

$$\langle \|\vec{OM}(t + \Delta t) - \vec{OM}(t)\|^2 \rangle = \langle |x(t + \Delta t) - x(t)|^2 + |y(t + \Delta t) - y(t)|^2 + |z(t + \Delta t) - z(t)|^2 \rangle, \quad (2)$$

où $\vec{OM} = x\vec{e}_x + y\vec{e}_y + z\vec{e}_z$ représente le vecteur position. Par linéarité de la moyenne sur la somme, on a :

$$\begin{aligned} \langle \|\vec{OM}(t + \Delta t) - \vec{OM}(t)\|^2 \rangle &= \langle |x(t + \Delta t) - x(t)|^2 \rangle + \langle |y(t + \Delta t) - y(t)|^2 \rangle \\ &\quad + \langle |z(t + \Delta t) - z(t)|^2 \rangle, \end{aligned} \quad (3)$$

La PDF^{X} de $\Delta x = x(t + \Delta t) - x(t)$, correspondant à la distribution de Δx s'écrit telle que :

$$p(\Delta x) = \frac{1}{\int_0^{+\infty} e^{-\frac{(\Delta x)^2}{2\sigma^2}} d(\Delta x)} e^{-\frac{(\Delta x)^2}{2\sigma^2}}, \quad (4)$$

où σ est l'écart-type de la distribution.

À l'aide de ces outils, nous pourrons, par la suite, analyser les données de trajectoires de particules générées numériquement pour les comparer à la théories physiques.

IX. mean square displacement, c'est-à-dire déplacement carré moyen en français

X. Probability density function, c'est-à-dire la fonction de densité de probabilité en français

1 Particule dans un espace infini

1.1 Mise en équation du problème

J'ai commencé par étudier une particule seule, de rayon R et de masse m , placée dans un fluide de viscosité η et dans un domaine Ω de volume infini. Dans cette première partie je présente la méthode utilisée pour modéliser un mouvement brownien d'une particule. Je vais simplifier le problème en négligeant toute autre force extérieure exceptée la force de friction. De plus, je vais d'abord m'intéresser au cas sur-amorti en négligeant la masse qui est plus simple à simuler.

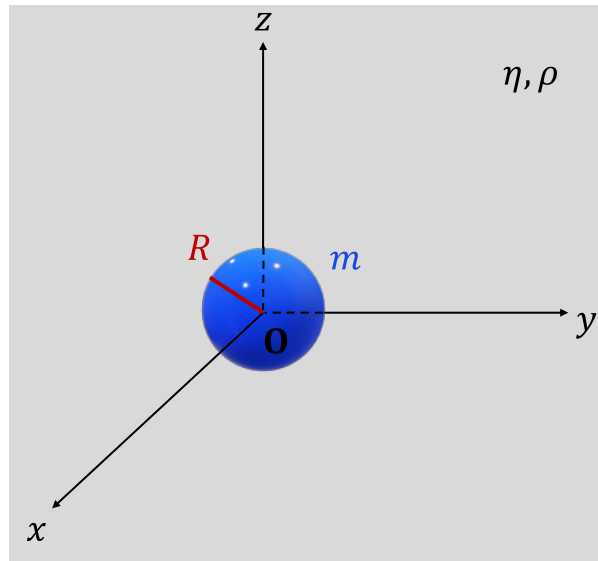


FIGURE 2 – Schéma du problème physique étudié. Une particule de masse m et de rayon $R = 10^{-6}$ m, placée à l'origine du repère, plongée dans un fluide considéré comme un espace infini de viscosité $\eta = 0.001$ Pa.s, de masse volumique $\rho = 1000$ Kg.m $^{-3}$ et à la température $T = 300$ K.

L'équation de Langevin associée à ce problème est :

$$m \frac{d^2x}{dt^2} = -\gamma \frac{dx}{dt} + F(t) , \quad (5)$$

où $-\gamma \frac{dx}{dt} = 6\pi\eta Rv$ est la force de Stokes^{XI}, $\frac{dx}{dt} = v_t$, la vitesse instantanée de la particule, x sa position à l'instant t et $F(t)$ la force aléatoire. Pour caractériser la force aléatoire, on doit faire des hypothèses statistiques [9] : (i) le fluide étant supposé isotrope, il n'y a pas de direction privilégiée dans l'espace et donc en moyenne la force est nulle sur le temps $\langle F(t) \rangle_t = 0$ et (ii) il n'y a pas de corrélation entre deux positions séparées d'une durée supérieure à un temps τ_c tel que $\langle F(t + \tau_c)F(t) \rangle \approx a \delta(t + \tau_c - t)$, où $F(t) = a \frac{dB_t}{dt}$ est la force aléatoire, a l'amplitude de la force aléatoire, $\frac{dB_t}{dt}$ un bruit blanc gaussien et $\delta(t + \tau_c - t) = \delta(\tau_c)$ la fonction de Dirac au temps τ_c .

Il est possible que l'amplitude des fluctuations thermiques $a(x)$ dépende de la position et

XI. Forces de traînée correspondant à la force de frottement entre le fluide et la particule.

elle doit être calculée afin de tout connaître du système étudié et le résoudre. On a, d'après l'équation (5),

$$dv_t = -\frac{\gamma}{m}v_t dt + \frac{a}{m}dB_t. \quad (6)$$

En calculant le carré de cette quantité et en ne gardant que les termes d'ordre 1 en dt , on obtient :

$$(dv_t)^2 = \frac{a^2}{m^2}(dB_t)^2 - 2\frac{a\gamma}{m^2}v_t dt dB_t = \frac{a^2}{m^2}dt - 2\frac{a\gamma}{m^2}v_t dt dB_t. \quad (7)$$

Par suite, on cherche à définir la valeur moyenne de la vitesse à l'équilibre thermodynamique $\langle v_t \rangle$. Pour cela, on calcule le développement de Taylor à l'ordre 2 de $dS = d(v_t^2)$ et à l'aide de (6) et (7) on obtient :

$$\begin{aligned} d(v_t^2) &= \frac{\partial(v_t^2)}{\partial v} dv_t + \frac{1}{2} \frac{\partial^2(v_t^2)}{\partial v^2} (dv_t)^2 \\ &= 2v_t \left[-\frac{\gamma}{m}v_t dt + \frac{a}{m}dB_t \right] + \frac{a^2}{m^2}dt - 2\frac{a\gamma}{m^2}v_t dt dB_t. \end{aligned} \quad (8)$$

Sachant que $\langle dB_t \rangle = 0$, en prenant la valeur moyenne de $d(v_t^2)$ on obtient :

$$\langle d(v_t^2) \rangle = \left\langle -\frac{2\gamma v_t^2}{m} dt + \frac{a^2}{m^2} dt \right\rangle, \quad (9)$$

or les quantités m , $\gamma = 6\pi\eta R$ et a sont constantes, d'où,

$$\frac{d\langle v_t^2 \rangle}{dt} = -\frac{2\gamma\langle v_t^2 \rangle}{m} + \frac{a^2}{m^2}. \quad (10)$$

Par résolution de cette équation différentielle du second degré à l'aide de la méthode de variation de la constante, puis en se plaçant à l'équilibre lorsque t tend vers $+\infty$, on trouve :

$$\langle v_t^2 \rangle_{\text{eq}} = \frac{a^2}{2\gamma m}. \quad (11)$$

En utilisant le théorème d'équipartition de l'énergie en 1D, tel que $\frac{1}{2}m\langle v_t^2 \rangle_{\text{eq}} = \frac{1}{2}k_B T$, où k_B est la constante de Boltzmann et T la température du milieu, on obtient alors,

$$\frac{1}{2} \left(\frac{a^2}{2\gamma} \right) = \frac{1}{2}k_B T, \quad (12)$$

d'où, l'amplitude des fluctuations :

$$a = \sqrt{2k_B T \gamma}. \quad (13)$$

On remarque que la magnitude des fluctuations ne dépend pas de la position de la particule, elle est donc constante. On peut à présent simuler numériquement une trajectoire de particules actives en trois dimensions indépendantes. Dans cette section 1, j'ai choisi de fixer la position initiale de la particule en $\vec{x}_0 = (0, 0, 0)$, son rayon à $R = 10^{-6}$ m, la température à $T = 300$ K et la viscosité du fluide à $\eta = 0.001$ Pa.s.

1.2 Cas sur-amorti

Dans cette sous-partie nous allons négliger l'inertie de la particule, que l'on nommera le cas sur-amorti. Comme les dimensions sont indépendantes, on peut appliquer (5) en 1D à chaque dimension. On obtient alors l'équation suivante :

$$-\gamma \frac{dx}{dt} + a \frac{dB_t}{dt} = 0, \quad (14)$$

réécrite tel que :

$$dx = a_2 dB_t, \quad (15)$$

où $a_2 = \frac{a}{\gamma} = \sqrt{\frac{2k_B T}{\gamma}}$ est la nouvelle amplitude des fluctuations thermiques avec $\gamma = 6\pi\eta R$. Le déplacement Δx étant toujours gaussien, la MSD_{nDim} pour n dimensions spatiale s'écrit [3] [4],

$$MSD_{nDim} = 2 n D \Delta t, \quad (16)$$

où $D = \frac{k_B T}{\gamma}$ est le coefficient de diffusion et Δt la durée du déplacement Δx . La PDF du déplacement Δx [3] [4] s'écrit :

$$p(\Delta x) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(\Delta x)^2}{2\sigma^2}}, \quad (17)$$

où $\sigma = \sqrt{2D\Delta t}$ est l'écart-type.

En normalisant Δx par l'écart-type tel que $\Delta x_{\text{nor}} = \frac{\Delta x}{\sqrt{2 \frac{\Delta t}{D}}}$ afin de s'affranchir de la dépendance en Δt , puis en normalisant $p(\Delta x)$ tel que l'air sous la courbe soit égale à 1, on obtient :

$$p_{\text{nor}}(\Delta x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(\Delta x)^2}{2}}, \quad (18)$$

de déviation standard égale à 1. Par suite, j'ai discrétisé numériquement l'équation (15) tel que :

$$x_i = x_{i-1} + a_2 dB_t[i-1]. \quad (19)$$

où x_i est la position de la particule sur un axe donné à l'instant $t = i dt$, x_{i-1} celle à l'instant $t - dt = (i-1)dt$, dt le pas de temps numérique et $dB_t[i-1]$ la $(i-1)^{\text{ième}}$ valeur du vecteur dB_t . Il est composé de N_t nombres aléatoires de loi Normale de moyenne nulle et d'écart-type \sqrt{dt} , construites à l'aide de la fonction `Numpy.Random.Normal()` en Python pour chaque position x , y et z .

J'ai créé un objet en Python nommé `Langevin3D` afin d'automatiser le code qui est construit tel que :

`Langevin3D class.`

Attributes of Langevin3D :

— `dt` : Pas de temps [s].

- N_t : Nombre de points.
- R : Rayon de la particule [m].
- η : Viscosité du fluide (défaut = 0.001 [Pa/s]).
- T : Température (défaut = 300 [K]).
- x_0 : Position initiale de la particule (défaut = (0,0,0) [m]).

Methodes of Langevin3D :

- `Trajectory()`
- `plotTrajectory()`
- `MSD1D(axis)`
- `MSD3D()`
- `speedDistribution1D(axis, nbTimesIntervalle=1, bins=50)`
- `dXDistribution1D(axis, nbTimesIntervalle=1)`

Pour construire la trajectoire dans `Trajectory()`, j'ai utilisé la fonction `cumsum()`^{XII} de Numpy afin d'éviter d'utiliser une boucle `for` qui peut prendre du temps de calcul sous Python.

J'ai choisit un pas de temps $dt = 0.01$ s et $N_t = 500000$ nombre de points. Une trajectoire possible, obtenue à la figure 3.a), ne peut pas être étudiée en l'état. Il est donc nécessaire de calculer la MSD pour vérifier les propriétés de diffusion.

Les MSD_{1D} selon \vec{e}_x , \vec{e}_y et \vec{e}_z et la MSD_{3D} (cf. figure 3.b)), peuvent être comparées à la théorie (16). À l'aide de la MSD_{3D} on peut vérifier l'indépendance des dimensions. On voit clairement que la modélisation numérique est en accord avec la théorie. On peut cependant observer qu'à partir de 10^2 s, la modélisation s'éloigne très légèrement de la théorie puisque à cette échelle de temps, le nombre de points n'est plus suffisant pour obtenir une statistique raisonnable. Ces résultats montrent que le déplacement quadratique moyen de la particule augmente linéairement avec le temps lorsqu'elle est placée dans un espace infini.

On trace également la densité de probabilité des déplacements Δx (elle est identique selon Δy et Δz) pour différentes durées de déplacement Δt , comme on peut le voir sur les figures 3.c). De cette courbe on ne peut pas vraiment comparer les résultats entre les différentes durées de déplacement puisque la PDF dépend de Δt également. Pour cela, on normalise la PDF comme expliqué dans la partie 1.2.1 et le résultat est donné sur la figure 3.d). De celle-ci on voit bien que le comportement de la particule est identique pour tout temps. La PDF théorique (18) valide bien la simulation numérique du mouvement brownien de la particule. Ce résultat montre que la densité de probabilité de déplacement de la particule est bien de forme gaussienne. C'est

^{XII}. Une somme cumulative. Exemple : si `list = [1, 2, 3, 4]` alors `np.cumsum(list) = [1, (1 + 2), (1 + 2 + 3), (1 + 2 + 3 + 4)] = [1, 3, 6, 10]`.

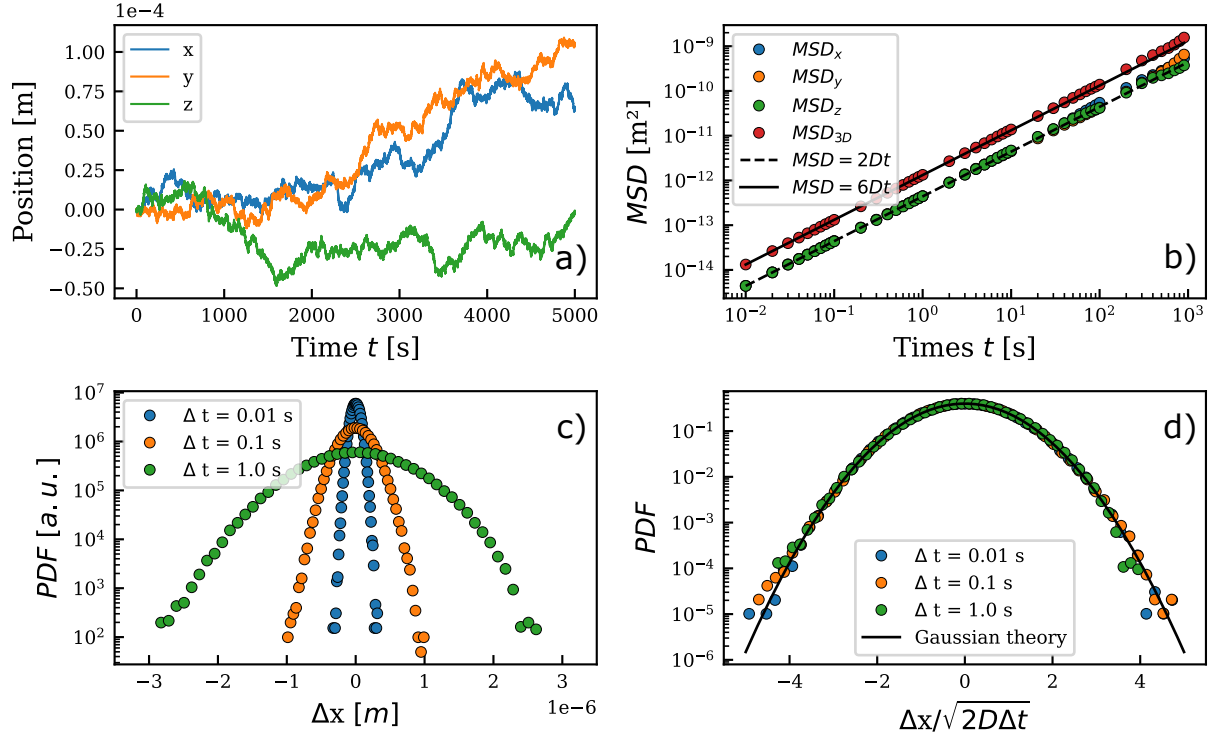


FIGURE 3 – Simulation sur-amortie de la diffusion d’une particule de rayon $R = 10^{-6}$ m dans un espace infini, calculée sur $N_t = 500000$ points et un intervalle de temps $dt = 0.01$ s. **a)** La trajectoire simulée de la particule, **b)** les MSD_{1D} et MSD_{3D} comparées aux courbes théoriques attendues, **c)** la PDF de Δx pour diverses durées et **d)** la PDF normalisée comparée à la théorie.

à dire que les déplacements les plus courts sont les plus probables en moyenne.

J’ai bien construit numériquement une particule se déplaçant avec une marche aléatoire due à l’agitation thermique. Suite à cela, j’ajoute le terme inertiel dans la simulation numérique.

1.3 Cas inertiel

Pour prendre en compte l’inertie, on considère une masse de particule m non nulle. Dans ce cas, il faut que le pas de temps numérique soit très petit devant le temps caractéristique du passage du régime inertiel au régime de diffusion, $\tau = \frac{m}{\gamma}$, puisque nous cherchons à observer les effets inertiels. Notons que pour le cas sur-amorti, il ne fallait pas prendre un pas de temps $dt < \tau$ puisque cette simulation ne prenait pas les effets inertiels en compte. Dans ce cas, on reprend l’équation (5) que l’on peut discrétiser tel que :

$$m \frac{x_i - 2x_{i-1} + x_{i-2}}{dt^2} = -\gamma \frac{x_i - x_{i-1}}{dt} + a \frac{dB_t}{dt}[i-1], \quad (20)$$

donnant la position x_i telle que :

$$x_i = \frac{2 + dt \frac{\gamma}{m}}{1 + dt \frac{\gamma}{m}} x_{i-1} - \frac{1}{1 + dt \frac{\gamma}{m}} x_{i-2} + \frac{dt}{m(1 + dt \frac{\gamma}{m})} dB_t[i-1], \quad (21)$$

avec $a = \sqrt{2k_B T \gamma}$ donné en (13), x_i la position de la particule sur un axe donné à l’instant

$t = idt$, x_{i-1} celle à l'instant $t - dt = (i - 1)dt$, x_{i-2} celle à l'instant $t - 2dt = (i - 2)dt$, dt le pas de temps et $dB_t[i - 1]$ la $(i - 1)^{\text{ième}}$ valeur du vecteur dB_t . Le vecteur dB_t un bruit blanc gaussien d'écart-type \sqrt{dt} .

Afin de limiter la taille du code, j'ai construit une classe nommée `InertialLangevin3D` qui hérite de la classe `Langevin3D` telle que :

`InertialLangevin3D class(Langevin3D).`

Attributes :

- `super().__init__(dt, Nt, R, eta = eta, T = T, x0 = x0)`
- ρ : masse volumique de la particule
- $m = \rho V$: masse de la particule [kg]
- $\tau = m/\gamma$: temps caractéristique de changement de régime.

Méthodes :

- `_PositionXi(xi-1, xi-2, rng)` :
 param : x_{i-1} : position à $t - dt$
 param : x_{i-2} : position à $t - 2dt$
 return : position de la particule à t .
- `Trajectory()`.

Ici, il est inévitable de passer par une boucle `for` selon chaque dimension pour calculer la trajectoire d'une particule du fait de la dérivée seconde en temps. En bouclant pour chaque dimension sur les $N_t - 2$ points et en utilisant `_PositionXi(xi-1, xi-2, rng)` qui calcule la position à t à l'aide des positions à $t - dt$ et $t - 2dt$, on construit une trajectoire de la particule sur une durée de $N_t dt$. Il nous faut donc deux conditions initiales pour lancer la simulation mais par simplicité elles seront toutes les deux égales à x_0 . Ce choix de simplification est justifié car il est peu probable que la particule effectue un grand déplacement entre deux points puisque $dt \ll \tau$ et que $N_t \gg 1$. Cette perturbation numérique est minime.

J'ai choisi $dt = 10^{-9}$ s, $N_t = 10^7$ et une particule de rayon $R = 10^{-6}$ m, de masse volumique $\rho = 1050$ kg.m⁻³ et de masse $m = \rho \frac{4}{3}\pi R^3 \approx 4.4 \cdot 10^{-15}$ Kg, plongée dans un fluide de viscosité $\eta = 10^3$ Pa.s et de masse volumique $\rho_F = 1000$ kg.m⁻³. En simulant le mouvement de la particule, on obtient la trajectoire en figure 4.a). De cette trajectoire on calcule le déplacement quadratique moyen de la particule en 3D en figure 4.b). On remarque que les effets inertiels font apparaître deux régimes. Ici, le temps caractéristique de passage du régime inertiels au régime brownien est $\tau = \frac{m}{\gamma} \approx 2.33 \cdot 10^{-7}$ s. Dans le régime brownien, on retrouve toujours la théorie diffusive (16), par contre, lorsque $\Delta t \ll \tau$, les effets inertiels ne sont plus négligeables. Dans ce cas, la théorie de Maxwell-Boltzmann prédit une augmentation de la

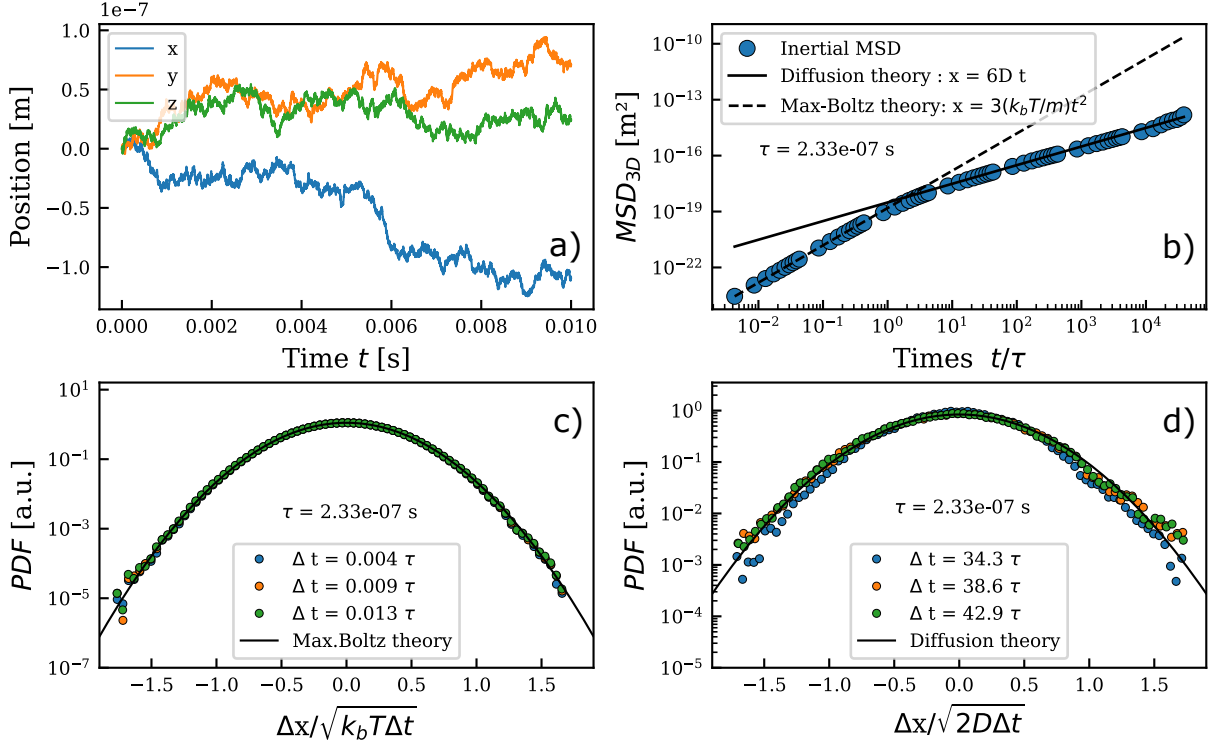


FIGURE 4 – Résultats numériques de la simulation inertielle pour une particule de rayon $R = 10^{-6}$ m dans un espace infini, calculée sur $N_t = 10^7$ points et un intervalle de temps $dt = 10^{-9}$ s. **a)** La trajectoire simulée de la particule, **b)** la MSD_{3D} comparée aux régimes théoriques attendus - régime inertiel et régime brownien -, **c)** la PDF de $\frac{\Delta x}{\sqrt{k_B T \Delta t}}$ pour divers pas de temps $\Delta t \ll \tau$ en régime inertiel et **d)** celle en régime brownien $\Delta t \gg \tau$, $PDF(\frac{\Delta x}{\sqrt{2D \Delta t}})$, comparées aux théories respectives.

MSD pour n dimensions aux temps courts tel que :

$$MSD_{nDim} = n \frac{k_B T}{m} \Delta t^2, \quad (22)$$

ainsi on retrouve un changement de loi de puissance en t de la MSD par rapport au régime diffusif, qui est quadratique aux temps courts ($\Delta t \ll \tau$) et linéaire aux temps longs ($\Delta t \gg \tau$), ce qui crée une sur-diffusion aux temps courts. La densité de probabilité de présence de la particule prédite par Maxwell-Boltzmann aux temps courts est :

$$p(\Delta x) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{\Delta x^2}{2\sigma^2}}, \quad (23)$$

avec $\sigma^2 = \frac{k_B T}{m} \Delta t^2$ l'écart-type. Comme on peut le voir sur la figure 4.c), on retrouve bien ce résultat numériquement. Enfin, la théorie diffusive (17) valide la simulation lorsque l'on est aux temps longs comme on peut le voir sur la figure 4.d). On peut cependant remarquer une sous-diffusion pour $|\frac{\Delta x}{2D \Delta t}| > 1.5$, probablement à cause d'un manque de statistiques aux temps longs.

2 Particule brownienne près d'un mur dur

2.1 Mise en équation du problème

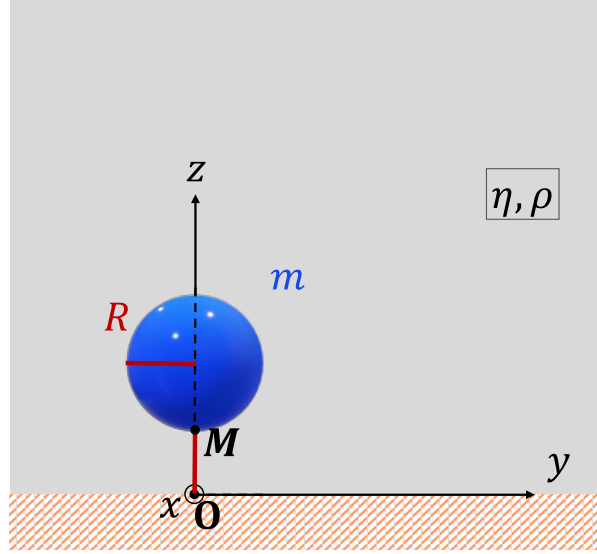


FIGURE 5 – Schéma du problème physique. On étudie une particule de masse $m = \rho_P \frac{4}{3}\pi R^3$ avec $\rho_P = 1050 \text{ kg.m}^{-3}$ sa masse volumique et $R = 1.5 \cdot 10^{-6} \text{ m}$ son rayon. Sa trajectoire est repérée par le vecteur $\vec{OM} = x \vec{e}_x + y \vec{e}_y + z \vec{e}_z$ où les effets rotationnels sont négligés. La particule est placée en $\vec{OM}(t=0) = (0, 0, R)$ dans un fluide infini selon x et y et semi-infini selon z proche d'un mur dur placé sur l'axe $z = 0$. Le fluide est de viscosité $\eta = 0.001 \text{ Pa.s}$, de masse volumique $\rho_F = 1000 \text{ Kg.m}^{-3}$ et à la température $T = 300 \text{ K}$.

Dans cette partie, on souhaite ajouter l'influence d'une paroi rigide dans la direction z de la particule comme on peut le voir sur la figure 5. Pour cela, on considère toujours une particule seule, placée dans un fluide de viscosité η . On place cette particule proche d'un mur rigide qui est à l'origine du repère selon l'axe z . Cette fois-ci, on tient compte du poids et de la poussée d'Archimède selon l'axe z tel que,

$$\vec{P} + \vec{A} = -m\vec{g} + m_F\vec{g} = -\frac{4}{3}\pi R(\rho - \rho_F)g \vec{e}_z, \quad (24)$$

où \vec{P} est le poids, \vec{A} la poussée d'Archimède, m et ρ respectivement la masse et la masse volumique de la particule et m_F et ρ_F celle du fluide déplacé par la particule. Pour tenir compte de la présence d'un mur rigide, supposé infranchissable, on ajoute une force électrostatique d'écrantage obtenue par approximation de Debye-Hückel qui s'écrit [7] :

$$\vec{R}_{\text{elec}} \approx \frac{\alpha k_B T}{l_D} e^{-\frac{z}{l_D}} \vec{e}_z, \quad (25)$$

où $l_D \approx 70 \cdot 10^{-9} \text{ m}$ est la constante de Debye du problème, $\alpha = 4$ le facteur déterminé expérimentalement par l'équipe de recherche.

Par projection sur les axes, les équations selon x et y sont inchangées (5), par contre pour celle selon z il est impératif d'ajouter ces trois forces, donnant ainsi,

$$m \frac{d^2 z}{dt^2} = -\gamma \frac{dz}{dt} + a(z) \frac{dB_t}{dt} + P + A + R_{elec}. \quad (26)$$

Le calcul de l'amplitude des fluctuations thermiques de (6) à (13) est inchangé donc on conserve toujours $a = \sqrt{2k_B T \gamma}$. Cependant, la résolution de l'équation de Stokes proche d'un mur rigide montre que [2] les coefficients de friction γ_i , où $i = \{x, y, z\}$, ne sont plus indépendants de la position de la particule et diffèrent en fonction de la direction i considérée. Dans les directions parallèles au mur, \vec{e}_x et \vec{e}_y , la diffusion est déviée par des effets hydrodynamiques : plus la bille est proche du mur, où la vitesse tend vers zéro, plus la force de traînée augmente. Les coefficients de friction selon \vec{e}_x et \vec{e}_y dépendent alors de la position z et s'expriment à l'aide de l'approximation de Libchaber telle que [6] :

$$\gamma_{x/y} = \left(6\pi R \eta \left(1 - \frac{9R}{16(z+R)} + \left(\frac{R}{8(z+R)} \right)^3 - \left(\frac{45R}{256(z+R)} \right)^4 - \left(\frac{R}{16(z+R)} \right)^5 \right) \right)^{-1}. \quad (27)$$

Dans la direction transverse au mur, la diffusion décroît anisotropiquement avec la distance z et le coefficient de friction s'exprime à l'aide de l'approximation de Padé telle que [8] :

$$\gamma_z = \left(6\pi R \eta \left(\frac{6z^2 + 2Rz}{6z^2 + 9Rz + 2R^2} \right) \right)^{-1}. \quad (28)$$

Puisque $\gamma_{x/y}$ et γ_z dépendent de la position selon z (l'axe perpendiculaire au mur rigide), alors l'amplitude des fluctuations thermiques a dépend aussi de la position z quelque soit l'axe sur lequel on cherche à la calculer. Dans un premier temps, je résous le problème sans inertie pour obtenir le modèle théorique aux temps longs $t > \tau_{eq}$, où τ_{eq} est le temps caractéristique de la mise en équilibre du système. Puis, dans un second temps, j'ajoute l'inertie afin de rendre compte des effets inertiels aux temps courts $t < \langle \tau_i(z) \rangle$, entre le régime inertiel et le régime diffusif, où $\langle \tau_i(z) \rangle = \frac{m}{\langle \gamma_i(z) \rangle}$ et i est la dimension considérée. Il y a alors trois régimes, le régime inertiel, diffusif et l'équilibre thermodynamique. Pour cette étude, j'ai choisi de fixer la position initiale de la particule à $\vec{x}_0 = (0, 0, R)$, son rayon à $R = 10^{-6}$ m, la température du milieu à $T = 300$ K, la viscosité du fluide à $\eta = 0.001$ Pa.s, sa masse volumique $\rho_F = 1000$ kg.m⁻³ (eau), celle de la particule $\rho = 1050$ kg.m⁻³ (bille de polystyrène).

2.2 Cas sur-amorti

Dans le cas sur-amorti, (26) projeté sur l'axe z s'écrit :

$$0 = -\gamma \frac{dz}{dt} + a(z) \frac{dB_t}{dt} + \frac{4}{3} \pi R (\rho_P - \rho_F) + F_{correction}, \quad (29)$$

où $F_{correction}$ est une force fictive selon z due au gradient de D_z [5] [10]. Cette force fictive s'écrit :

$$F_{\text{correction}} = \frac{D'_z}{D_z}(z) = k_B T \frac{42Rz^2 + 24R^2z + 4R^3}{(6z^2 + 9Rz + 2R^2)(6z^2 + 2z)}, \quad (30)$$

où $D_z(z) = \frac{k_B T}{\gamma_z(z)}$ est le coefficient de diffusion de la particule selon l'axe z lorsqu'elle est proche d'un mur dur et $D'_z(z)$ sa dérivée par rapport à z .

Selon les axes x et y on retrouve la même équation sans le poids, la poussée d'Archimède et la force correctrice.

On s'attend à ce qu'en régime diffusif, $t < \tau_{eq}$, avant que l'équilibre thermodynamique ne soit atteint, la MSD_z soit en moyenne la même que celle sans l'ajout du mur dur [3] [4],

$$MSD_i^{\text{short}} = 2\langle D_i(z) \rangle t, \quad (31)$$

où $\langle D_i(z) \rangle = \frac{k_B T}{6\pi\langle\eta_i(z)\rangle R}$ est la valeur moyenne du coefficient de diffusion de la particule selon un axe i donné. Ici, on ne parlera pas de la MSD en 3D puisque les positions x , y et z dépendent de la position z à cause de l'amplitude des fluctuations thermiques $a(z)$. Puis lorsque l'équilibre est atteint, la MSD_z sature vers $\langle \Delta z^2 \rangle_{\text{inf}}$ calculée grâce à la PDF [8] telle que :

$$p(z) = \frac{1}{M} \exp \left(-4 \exp \left(-\frac{z}{l_D} \right) - \frac{z}{l_B} \right), \quad (32)$$

où $l_D = 70.10^{-9} \text{ m}$ pour une particule de polystyrène est la longueur de Debye correspondant à la longueur caractéristique d'écrantage par les contre-ions des interactions coulombiennes, $l_B = \frac{k_B T}{\Delta m g}$ la longueur de Boltzmann correspondant à la longueur qui caractérise la longueur à partir de laquelle l'énergie potentielle de pesanteur l'emporte sur l'agitation thermique et M est une constante de normalisation de la PDF_z .

Ce qui permet de calculer :

$$p(\Delta z) = \int_0^{+\infty} p(z) p(z + \Delta z) dz, \quad (33)$$

donnant,

$$\langle \Delta z^2 \rangle_{\text{inf}} = \int_0^{+\infty} (\Delta z)^2 p(\Delta z) d(\Delta z). \quad (34)$$

Le plateau de la MSD_z se situe donc à $\langle \Delta z^2 \rangle_{\text{inf}} = 7.26.10^{-13} \text{ m}^2$ d'après le calcul théorique.

Puis, pour simuler numériquement la trajectoire de la particule, on détermine le schéma numérique de la solution tel que :

$$z_i = z_{i-1} - \frac{\Delta m g}{\gamma(z_{i-1})} dt + R_{\text{elec}}(z_{i-1}) + a(z_{i-1}) dB_t dt, \quad (35)$$

que l'on calcule selon z pour $N_t = 10^7$ points et un pas de temps $dt = 10^{-2} \text{ s}$. Le choix de dt est ici complètement lié au fait que nous sommes dans le cas sur-amorti. Comme on ne tient pas compte de l'inertie qui voit ses effets aux temps courts, on ne peut pas mettre des pas de temps $dt < \tau$.

Finalement, on obtient une trajectoire qui semble libre selon les axes x et y et confinée selon z comme on peut le voir sur la figure 6.a). Si on calcule la PDF de la position z , comme on peut le voir sur la figure 6.b), on observe que, comme la théorie le décrit, la position de la particule est plus probable à $z \approx 0.3 \cdot 10^{-6}$ m. Puis cette probabilité décroît de manière asymétrique de part et d'autre de ce maximum. Lorsque la particule s'éloigne du mur (à la position $z = 0$), la probabilité décroît exponentiellement et tend vers zéro lorsque $z \rightarrow +\infty$. Lorsqu'elle se rapproche du mur, les positions $z < 0.3 \cdot 10^{-6}$ m sont plus probables que celles éloignées du mur $z > 0.3 \cdot 10^{-6}$ m, puisque la particule sédimente. La simulation est bien en accord avec ce que la théorie prévoit.

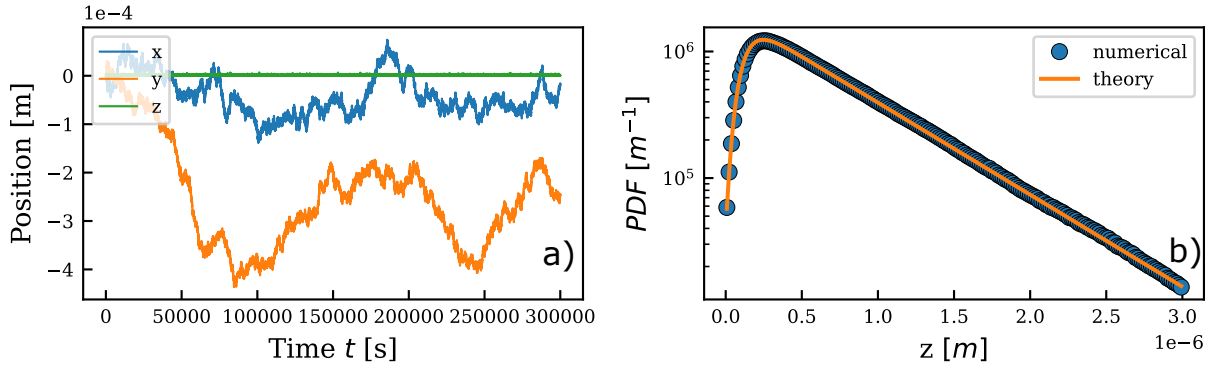


FIGURE 6 – Particule proche d'un mur dur dans le cas sur-amorti, de rayon $R = 1,5 \cdot 10^{-6}$ m, de masse volumique $\rho = 1050 \text{ kg.m}^3$ et de position initiale $\vec{x}_0 = (0, 0, R)$, plongée dans de l'eau de masse volumique $\rho_F = 1000 \text{ kg.m}^3$ et de viscosité $\eta = 0,001 \text{ Pa.s}$ et à la température $T = 300 \text{ K}$. La simulation est calculée avec $N_t = 3 \cdot 10^7$ points et le pas de temps est $dt = 10^{-2} \text{ s}$. **a)** Trajectoire numérique de la particule. **b)** PDF de z numérique et théorique.

Puis on trace les MSD_{1D} selon x , y et z sur la figure 7.a) et on les compare aux différentes théories attendues. D'une part, les MSD_x et MSD_y sont identiques et s'accordent bien sur la moyenne de la $\text{MSD}_{\text{theo}} = 2\langle D_x(z) \rangle t$ que l'on avait obtenue dans le cas sans mur dur. D'autre part, on observe deux régimes de fonctionnement pour la MSD_z . À $t < \tau_{eq}$, cette quantité vaut la moyenne de la valeur théorique dans l'espace infini $\text{MSD}_{z,\text{theo}} = 2\langle D_z(z) \rangle t$. On remarque que la MSD_z est plus faible pour tout t que celles selon x et y malgré le fait que toutes les MSD aient le même coefficient directeur égale à 2. Puis, à $t > \tau_{eq}$, lorsque l'équilibre est atteint, la MSD_z tend vers une valeur constante $\langle \Delta x^2 \rangle$. En effectuant un fit de la forme $a(1 - \exp(bt))$, où a et b sont les paramètres recherchés, on trouve une valeur numérique de $\langle \Delta x^2 \rangle_{\text{num}} = a = 7.27 \cdot 10^{-13} \text{ m}^2$ contre la valeur attendue de $\langle \Delta x^2 \rangle_{\text{theo}} = 7.26 \cdot 10^{-13} \text{ m}^2$ théoriquement. Ainsi, ce résultat confirme une fois de plus la construction numérique de la trajectoire de la particule.

Enfin, pour comprendre les phénomènes physiques en jeu lorsqu'on ajoute une paroi rigide, on trace également les densités de probabilité de déplacement $p(\Delta x)$ sur la figure 7.b) et $p(\Delta z)$ sur les figures 7.c) et 7.d), respectivement pour les cas hors équilibre et à l'équilibre thermodynamique. Sur les figures a) et b), on remarque que la théorie de Maxwell-Boltzmann

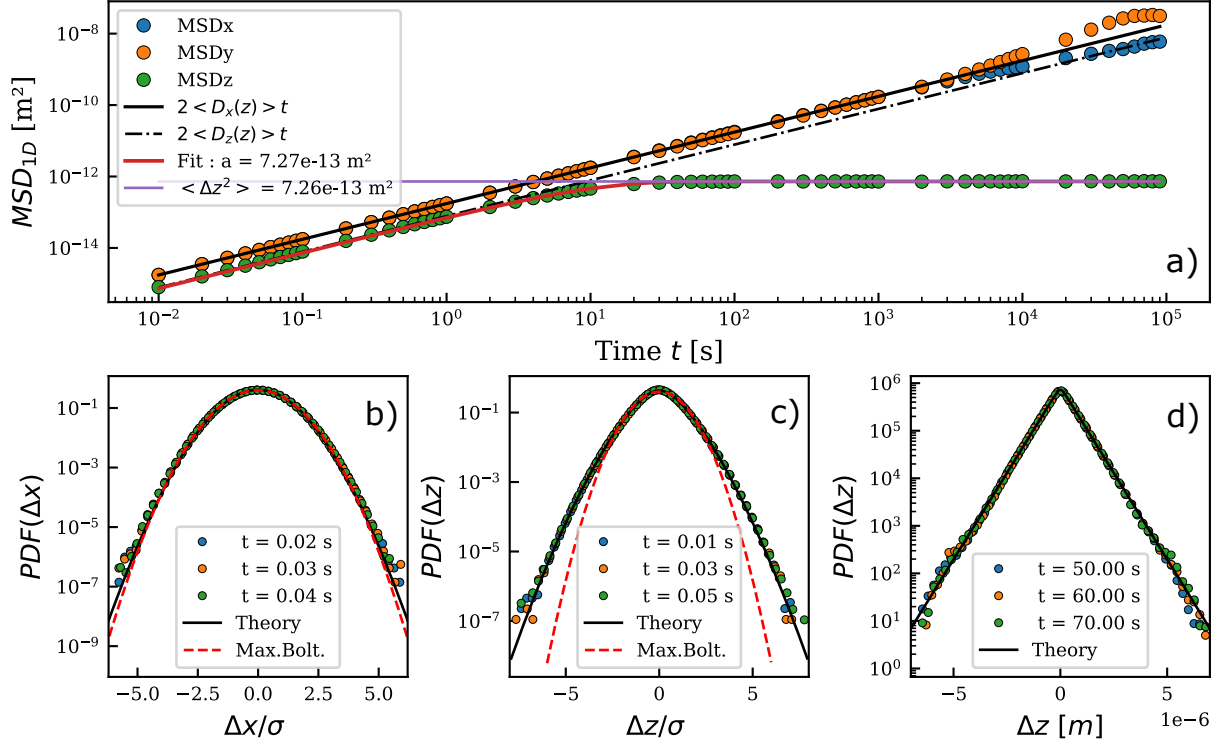


FIGURE 7 – Particule proche d'un mur dur dans le cas sur-amorti, de rayon $R = 1,5 \cdot 10^{-6} \text{ m}$, de masse volumique $\rho = 1050 \text{ kg.m}^3$ et de position initiale $\vec{x}_0 = (0, 0, R)$, plongée dans de l'eau de masse volumique $\rho_F = 1000 \text{ kg.m}^3$ et de viscosité $\eta = 0,001 \text{ Pa.s}$ et à la température $T = 300 \text{ K}$. La simulation est calculée avec $N_t = 3 \cdot 10^7$ points et le pas de temps est de $dt = 10^{-2} \text{ s}$. **a)** MSD en 1 dimension selon x , y et z comparées aux moyennes des MSD théorique en régime de diffusion en noir (obtenu sans mur). En rouge on retrouve l'ajustement des points de la simulation sous la forme $a(1 - \exp(-bx))$ où a représente la constante de saturation et en violet la valeur de saturation théorique attendue. **b)** La densité de probabilité $p(\frac{\Delta x}{\sigma})$ en comparaison avec la théorie dans le cas d'un mur dur (noir) et la théorie de Maxwell-Boltzmann obtenue dans un espace infini (rouge). **c)** Celle, $p(\frac{\Delta z}{\sigma})$, aux temps courts ($t \ll \tau_{eq}$) et **d)** aux temps longs ($t \gg \tau_{eq}$), où τ_{eq} est le temps caractéristique pour atteindre l'équilibre thermodynamique.

décrivant les temps courts ne fonctionne plus. En effet, la présence de la paroi rigide modifie les coefficients de diffusion D_x , D_y et D_z , il faut alors tenir compte du fait qu'il ne sont plus des constantes de la position. Dans ce cas, la $p_{theo}(\Delta x_i, \Delta t)$ [8] s'écrit :

$$p_{theo}(\Delta x_i, \Delta t) = \int p(D_i) \exp\left(-\frac{\Delta x_i^2}{4 D_i \Delta t}\right) d(D_i), \quad (36)$$

où x_i correspond à la direction étudiée, D_i le coefficient de diffusion selon la direction x_i et $p(x_i)$ sa densité de probabilité. Ce calcul théorique est tracé en noir dans chacun des cas et on voit bien que la présence de la paroi rigide influence la probabilité de déplacement qui n'est alors plus gaussienne. Ainsi, pour chaque point de l'espace où la particule a une chance de se trouver, le coefficient de diffusion est différent et influence la probabilité que la particule puisse s'y déplacer.

L'étude de la simulation numérique que j'ai développée est en accord avec la théorie et la trajectoire obtenue est bien celle d'une particule proche d'une paroi rigide.

3 Choix d'optimisation numérique et perspectives d'amélioration

Les travaux effectués au cours de ce stage sont les premières briques d'un projet plus général de thèse. Ainsi, l'objectif numérique du stage est de mettre en place un code robuste et réutilisable dans le futur. Dans cette partie je vais expliquer les choix d'outils numériques que j'ai faits et également les perspectives d'amélioration.

3.1 Choix numériques et d'optimisation

J'ai choisi de coder le problème physique en orienté objet afin de faciliter l'ajout de briques plus complexes dans mon programme `Python`. En effet, comme j'ai déjà pu le décrire dans les sections 1 et 2, j'utilise des classes afin d'utiliser l'héritage de classe afin d'ajouter l'inertie, un mur rigide etc, au problème physique de base d'une trajectoire de particule sur-amortie dans un espace infini. Ceci dans le but de ne pas tout redéfinir à chaque étape de complexification du code puisque tous ces problèmes physiques partent d'une même base. De plus, travailler avec des objets permet de généraliser les constantes sous forme d'attributs de classe qui ne sont alors implémentées qu'une seule fois et de pouvoir faire appel à des méthodes construites dans chaque classe afin de faciliter l'utilisation du programme. Ces attributs et méthodes sont des paramètres et des fonctions associés à l'objet. Par exemple, lorsque j'ai créé une instance de l'objet `Langevin3D`, je peux faire appel aux méthodes `Langevin3D.Trajectory()` pour calculer une trajectoire, `Langevin3D.PlotTrajectory()` pour tracer la trajectoire de la particule, `Langevin3D.MSD1D(axis)` pour calculer la MSD en 1D selon un axe donné, etc. Je peux également accéder aux attributs `Langevin3D.gamma` ou `Langevin3D.kb` par exemple, afin d'utiliser les valeurs de γ et de k_B de l'instance de la classe, etc. Cela permet de gagner en vitesse d'utilisation du code.

Par la suite, je me suis confrontée au problème de vitesse de calcul des trajectoires, notamment dans les cas de particules proches d'une paroi rigide. Dans chaque cas, la simulation numérique demande de calculer entre 1 million et quelques dizaines de millions de points et la boucle `for` de `Python` demande beaucoup de temps contrairement au langage `C`, du fait en partie du manque de typage. J'ai alors choisi d'augmenter la vitesse de calcul avec le langage de programmation `Cython`, qui est un sous-langage de `Python` utilisant la compilation en `C` pour augmenter les performances du code. L'optimisation du code sous `Cython` m'a pris beaucoup de temps de développement car c'est un langage que je ne connaissais pas avant cela. Pour ce faire, j'ai commencé par définir l'ensemble des classes mères et filles sous `Cython` avant de me rendre compte que la programmation en orienté objet sous ce langage était plutôt compliquée. Suite à ça, j'ai simplifié mon approche en créant une fonction `Trajectory_cython()` afin de calculer seulement la trajectoire sous `Cython`. Pour cythoniser la fonction trajectoire, on a besoin d'ajouter le type de chaque variable et d'utiliser des bibliothèques mathématiques de `C`

plutôt que celles de Python.

J'ai alors comparé les performances de temps de calcul de la trajectoire de la particule sous Python et sous Cython que j'ai résumées dans le tableau ci-dessous. J'y indique t_{Py} le temps moyen de calcul sous Python sur 10 runs, t_{Cy} celui sous Cython et $\frac{t_{Py}}{t_{Cy}}$ le rapport entre les deux.

	Cas physiques étudiés :	t_{Py}	t_{Cy}	$\frac{t_{Py}}{t_{Cy}}$
1	Cas sur-amorti espace infini	0.087 s	0.089 s	0.978
2	Cas inertiel espace infini	7.49 s	0.10 s	76.09
3	Cas sur-amorti mur	41.26 s	0.71 s	58.20
4	Cas inertiel mur	42.65 s	0.76 s	55.92

TABLE 1 – Comparatif des temps de calcul pour chaque cas physique étudié en fonction du langage utilisé : Python t_{Py} et Cython t_{Cy} , ainsi que le rapport entre ces temps $\frac{t_{Py}}{t_{Cy}}$. Paramètres fixés : $N_t = 10^6$ points, $R = 1.5 \cdot 10^{-6}$ m, $\rho_P = 1050$ kg.m³, $\rho_F = 1000$ kg.m³, $\eta =$ Pa.s, $T = 300$ K et dans les cas sur-amorti, $dt = \frac{1}{60}$ s et $\vec{x}_0 = (0, 0, 0)$, et inertiel, $dt = 10^{-6}$ s et $\vec{x}_0 = (0, 0, R)$.

Dans le cas 1, il n'y a pas d'accélération de temps de calcul notable puisqu'il n'y a pas de boucle `for`. Dans le cas 2 par contre, le temps de calcul en Cython est environ 80 fois plus rapide qu'en Python. Puis pour les cas 3 et 4, le temps de calcul en Cython y est environ 58 fois plus rapide en moyenne. Ainsi on voit bien que ce choix d'optimisation est adapté pour les cas où la trajectoire a besoin d'une boucle `for`. Mais cette optimisation n'est pas suffisante pour étudier le problème dans la globalité et nous allons voir pourquoi dans le paragraphe suivant.

3.2 Perspective de parallélisation

Pour simuler une particule inertielle proche d'une mur à l'aide de Cython, j'ai effectué la même méthode que dans la section 1. Cependant, afin d'étudier ce cas je me heurte à quelques problèmes numériques. Dans ce cas, on souhaite vérifier qu'il y a trois régimes de fonctionnement : le régime inertiel en sur-diffusion $t < \tau$, le régime de diffusion classique $\tau < t < \tau_{eq}$ et le régime d'équilibre thermodynamique $t > \tau_{eq}$ et qu'il n'y a pas d'effet inertiel aux temps longs $t > \tau$ - c'est ce que l'on attend comme résultat théorique. On voit alors qu'il faut choisir un pas de temps $dt < 10^{-8}$ pour être sûrs d'avoir assez de données aux temps courts afin de visualiser le régime inertiel et qu'il faut également que le nombre de points soit $N_t > 10^8$ pour être sûrs d'aller jusqu'à des temps $t \gg \tau_{eq} \approx 1$ s après l'équilibre thermodynamique (cf. figure 7.a)).

Par exemple, pour $dt = 10^{-8}$ s, il faudrait choisir au moins $N_t = 10^{11}$ points pour balayer les temps jusqu'à $t = 10^3$ s et être sûrs d'avoir assez de statistiques jusqu'à $t = 10^2$ s (la dernière décade est toujours sous-estimée par manque de statistiques). Or comme on peut le voir sur la figure 8, le temps de calcul est linéaire avec le nombre de points pour la simulation et de pente $7.4 \cdot 10^{-7}$ s. On observe une saturation pour $N_t < 10^3$ qui est certainement due en partie à la conversion du langage Python en Cython en plus de la réponse du processeur (qui

est plutôt de l'ordre de la nanoseconde). Ainsi, si on extrapole ce résultat pour une trajectoire avec $N_t = 10^{11}$, on trouve $t_c^{\text{estimé}} = 7.4 \cdot 10^4 \text{ s} = 20.5 \text{ heures}$. Ce temps de calcul reste envisageable. Cependant si on veut complexifier le problème physique, par exemple prendre une particule de masse plus faible, ou changer la viscosité de l'eau, prendre le cas d'un mur mou, etc, ce nombre de points peut-être amené à augmenter. De ce fait il est important de chercher à réduire le temps de calcul. La parallélisation semble être la solution la plus efficace. Théoriquement, la répartition du calcul sur 100 processeurs devrait permettre de réduire ce temps à $\frac{20}{100} = 0.2 \text{ heures} = 12 \text{ min}$.

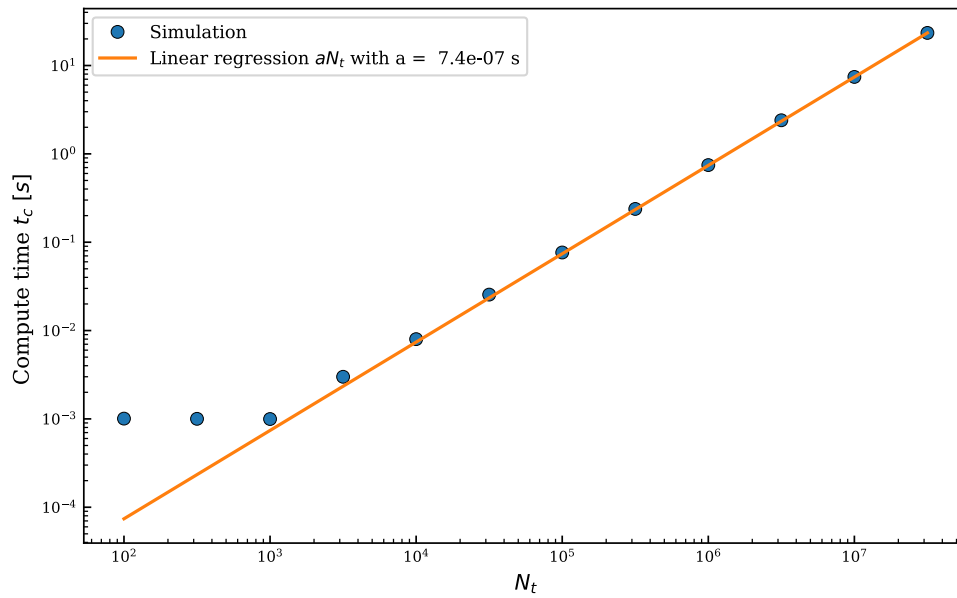


FIGURE 8 – Temps de calcul d'une trajectoire de la particule inertielle proche d'une paroi rigide en fonction du nombre de points de simulation. Calcul de l'ajustement linéaire $t_c = a N_t$ avec une pente de $a = 7.4 \cdot 10^{-7} \text{ s}$.

Il faut également considérer la taille des données et la place dans la mémoire vive ou la mémoire interne. À chaque simulation je dois avoir accès aux 3 dimensions de la trajectoire et au temps, de taille Nt , ce qui fait une matrice $(4 \times Nt)$ floats à stocker. Sans compter les variables créées pour calculer les MSD et les PDF qui doivent être stockées en mémoire vive également. Ainsi, pour pallier cela, il faudrait stocker N points sur les Nt de la simulation. Puis il faudrait faire une boucle `for` N fois en utilisant les points finaux de la simulation précédente comme point de départ pour la suivante; de telle sorte que le nombre de points enregistrés soit de $\sum N = N_t$; Puis d'enregistrer ces valeurs dans un fichier.

Conclusion

Lors de ce stage j'ai modélisé la trajectoire d'une particule brownienne dans un espace infini et proche d'un mur. Cette étude m'a permis de vérifier qu'en présence du mur dur, la particule se retrouve alors dans un gradient de coefficient de diffusion en \vec{e}_z , la direction perpendiculaire

au mur et que le déplacement de la particule n'est alors plus gaussien contrairement à son déplacement dans un milieu infini. Ce stage m'a aussi permis de découvrir le code en orienté objet ainsi que son optimisation en Cython. L'objectif pour la suite est de vérifier qu'en régime inertiel, la particule diffuse comme le carré du temps contrairement au régime de diffusion qui est linéaire avec le temps. De plus, il faudra vérifier que les déplacements aux temps courts n'influencent pas ce qui se passe à l'équilibre pour une particule proche d'un mur dur. C'est pourquoi il est important d'optimiser le code afin de réduire le temps de calcul pour pouvoir obtenir les résultats aux temps longs avec la théorie inertielle. Enfin il faudra étudier de la même manière l'influence d'une paroi molle qui n'est toujours pas connue à ce jour.

Références

- [1] Lydéric BOCQUET et Elisabeth CHARLAIX. "Nanofluidics, from bulk to interfaces". en. In : *Chemical Society Reviews* 39.3 (2010). Publisher : Royal Society of Chemistry, p. 1073-1095. DOI : 10.1039/B909366B. URL : <https://pubs.rsc.org/en/content/articlelanding/2010/cs/b909366b> (visité le 28/10/2020).
- [2] Howard BRENNER. "The slow motion of a sphere through a viscous fluid towards a plane surface". en. In : *Chemical Engineering Science* 16.3-4 (déc. 1961), p. 242-251. ISSN : 00092509. DOI : 10.1016/0009-2509(61)80035-3. URL : <https://linkinghub.elsevier.com/retrieve/pii/0009250961800353> (visité le 30/10/2020).
- [3] Aleksei V. CHECHKIN et al. "Brownian yet Non-Gaussian Diffusion : From Superstatistics to Subordination of Diffusing Diffusivities". In : *Physical Review X* 7.2 (avr. 2017). Publisher : American Physical Society, p. 021002. DOI : 10.1103/PhysRevX.7.021002. URL : <https://link.aps.org/doi/10.1103/PhysRevX.7.021002> (visité le 08/10/2020).
- [4] Mykyta V. CHUBYNSKY et Gary W. SLATER. "Diffusing Diffusivity : A Model for Anomalous, yet Brownian, Diffusion". In : *Physical Review Letters* 113.9 (août 2014). Publisher : American Physical Society, p. 098302. DOI : 10.1103/PhysRevLett.113.098302. URL : <https://link.aps.org/doi/10.1103/PhysRevLett.113.098302> (visité le 08/10/2020).
- [5] Oded FARAGO et Niels GRØNBECH-JENSEN. "Langevin dynamics in inhomogeneous media : Re-examining the It^o-Stratonovich dilemma". In : *Physical Review E* 89.1 (jan. 2014). Publisher : American Physical Society, p. 013301. DOI : 10.1103/PhysRevE.89.013301. URL : <https://link.aps.org/doi/10.1103/PhysRevE.89.013301> (visité le 08/10/2020).
- [6] Luc P. FAUCHEUX et Albert J. LIBCHABER. "Confined Brownian motion". In : *Physical Review E* 49.6 (juin 1994). Publisher : American Physical Society, p. 5158-5163. DOI : 10.1103/PhysRevE.49.5158. URL : <https://link.aps.org/doi/10.1103/PhysRevE.49.5158> (visité le 08/10/2020).
- [7] Jacob N. ISRAELACHVILI. *Intermolecular and Surface Forces*. en. Google-Books-ID : vgyBJbt-NOcoC. Academic Press, juil. 2011. ISBN : 978-0-12-391933-5.

- [8] Mpumelelo MATSE, Mykyta V. CHUBYNSKY et John BECHHOEFER. “Test of the diffusing-diffusivity mechanism using near-wall colloidal dynamics”. In : *Physical Review E* 96.4 (oct. 2017). Publisher : American Physical Society, p. 042604. DOI : 10.1103/PhysRevE.96.042604. URL : <https://link.aps.org/doi/10.1103/PhysRevE.96.042604> (visité le 08/10/2020).
- [9] Christian NGÔ et Hélène NGÔ. *Physique statistique - 3ème édition*. fr. Dunod, oct. 2008. ISBN : 978-2-10-053777-8.
- [10] Giovanni VOLPE et al. “Influence of Noise on Force Measurements”. In : *Physical Review Letters* 104.17 (avr. 2010). Publisher : American Physical Society, p. 170602. DOI : 10.1103/PhysRevLett.104.170602. URL : <https://link.aps.org/doi/10.1103/PhysRevLett.104.170602> (visité le 08/10/2020).