

Projet 1

Sphères dures en dimension 1

sujet proposé par Lucas Gerin

lucas.gerin@polytechnique.edu

Outils de MAP361 : Probabilités conditionnelles, Combinatoire, Convergence de variables aléatoires...

Domaine d'application : Physique statistique

En physique statistique, les sphères dures sont un modèle d'interaction entre particules dans lequel ces dernières sont représentées par des sphères de taille identique disposées sur un réseau régulier (typiquement, \mathbb{Z}^d) et qui ne peuvent se toucher. Ce modèle d'apparence très simple est très difficile à analyser mathématiquement en dimension $d \geq 2$.

L'objectif du projet est d'analyser et simuler deux modèles aléatoires de sphères dures en dimension 1.

1.1 Modèle dynamique

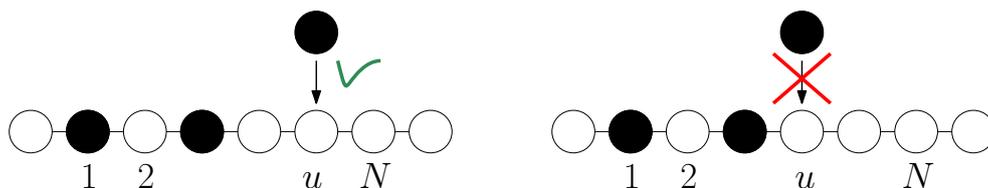
1.1.1 Définition et simulation du modèle

Soit $N \geq 2$ fixé, on considère l'ensemble $\mathcal{I}_N = \{0, 1, 2, \dots, N, N + 1\}$ qui est vu comme un milieu en dimension 1 sur lequel vont s'accrocher des particules appelées *sphères dures*.

Pour $t \in \mathbb{N}$ et $1 \leq k \leq N$ on va noter $M_t(k) = 0$ ou 1 selon que la position k est libre ou occupée par une sphère à l'instant t . Initialement tout est vide :

$$M_0(0) = M_0(1) = M_0(2) = M_0(3) = \dots = M_0(N) = M_0(N + 1) = 0.$$

Le processus $t \mapsto (M_t(k))_{t \geq 0, k \leq N}$ évolue de la façon suivante :



- Pour chaque $t \geq 1$ on tire une variable aléatoire U_t uniforme dans $\{1, 2, \dots, N\}$. Les (U_t) sont supposées indépendantes. Supposons que $U_t = u$, deux situations sont possibles (voir le schéma plus bas) :

- Si les positions $u - 1, u, u + 1$ sont libres (c'est-à-dire $M_{t-1}(u - 1) = M_{t-1}(u) = M_{t-1}(u + 1) = 0$) alors une sphère vient s'accrocher en u :

$$M_t(u) = 1$$

et les N autres coordonnées restent inchangées : $M_t(v) = M_{t-1}(v)$ pour tout $v \neq u$. C'est le cas de gauche dans le schéma.

- Si l'une au moins des positions $u - 1, u, u + 1$ est occupée alors rien ne se passe et $M_t(v) = M_{t-1}(v)$ pour tout v . C'est le cas de droite.

Le processus continue jusqu'à ce qu'il n'y ait plus dans la configuration M_t trois positions consécutives inoccupées. On dit que la configuration est saturée.

Remarques :

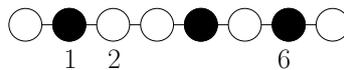
- Les positions 0 et $N + 1$ vont toujours rester libres au cours du temps : pour tout $t \geq 0$ on a $M_t(0) = M_t(N + 1) = 0$.
- On admet pour l'instant que le processus finit forcément par s'arrêter et atteindre une configuration saturée. Ceci sera démontré à la Question T5.

On note $M_t = (M_t(0), M_t(1), \dots, M_t(N + 1))$ la configuration à l'instant t et on note $M_\infty := (M_\infty(0), M_\infty(1), \dots, M_\infty(N + 1))$ la configuration finale après saturation. Les variables aléatoires M_t et M_∞ sont donc à valeurs dans $\{0, 1\}^{N+2}$.

Voici un exemple d'évolution du processus (M_t) dans le cas $N = 6$:

$$\begin{aligned} M_0 &= (0, 0, 0, 0, 0, 0, 0) \\ U_1 = 6, \quad M_1 &= (0, 0, 0, 0, 0, 0, 1, 0) \\ U_2 = 5, \quad M_2 &= (0, 0, 0, 0, 0, 0, 1, 0) \\ U_3 = 4, \quad M_3 &= (0, 0, 0, 0, 1, 0, 1, 0) \\ U_4 = 1, \quad M_4 &= (0, 1, 0, 0, 1, 0, 1, 0) \end{aligned}$$

et le processus s'arrête, on a donc $M_\infty = (0, 1, 0, 0, 1, 0, 1, 0)$. Sur cet exemple M_∞ peut se représenter schématiquement par :



S1. Afficher une trajectoire du processus $(M_t)_{0 \leq t \leq T}$. On utilisera la représentation suivante : la trajectoire va être codée par une matrice $(N + 2) \times T$ où le coefficient (k, t) est la variable $M_t(k)$. Pour afficher une matrice M , et donc la trajectoire, une façon simple avec `matplotlib` est d'utiliser `plt.matshow(M)`.

On choisira les paramètres suivants : $N = 40, T = 200$.

1.1.2 Densité de sphères

On note $X_t(N) \in \{0, 1, \dots, N\}$ (resp. $X_\infty(N)$) le nombre de positions occupées par une sphère après l'étape t (resp. à la fin du processus) c'est-à-dire

$$\begin{aligned} X_t(N) &= M_t(1) + M_t(2) + \dots + M_t(N) \\ X_\infty(N) &= M_\infty(1) + M_\infty(2) + \dots + M_\infty(N). \end{aligned}$$

T1. Pour le cas $N = 3$ déterminer la loi de M_∞ , la loi de $X_\infty(3)$ et en déduire $\mathbb{E}[X_\infty(3)]$.

T2. Démontrer qu'il existe $c > 0$ tel que pour tout N on a, presque sûrement,

$$X_\infty(N) \geq cN.$$

Cette borne inférieure suggère que $(X_\infty(N))_{N \geq 1}$ croît linéairement. C'est effectivement le cas et un résultat remarquable¹ est que

$$\frac{1}{N} \mathbb{E}[X_\infty(N)] \xrightarrow{N \rightarrow +\infty} \frac{1}{2}(1 - e^{-2}) \approx 0.43233\dots$$

Remarque : Cette valeur numérique semble compatible avec des résultats expérimentaux réalisés avec du méthylvinylcétone².

On souhaite illustrer expérimentalement ce résultat théorique de convergence.

S2. Écrire un programme qui prend en entrée $N \geq 1$ et simule une configuration saturée M_∞ de taille N . Simuler $K = 1000$ fois la variable $X_\infty(N)$ pour $N = 100$, afficher un histogramme des valeurs obtenues. Calculer également la moyenne de vos K valeurs et comparer avec $\frac{1}{2}(1 - e^{-2})$. Avec `matplotlib` un histogramme des valeurs `Donnees` avec comme classes $\{39, 40, \dots, 47\}$ s'affiche avec

```
plt.hist(Donnees, np.arange(38.5, 48.5), ec='black')
```

(l'argument `ec='black'` sert à dessiner les bords des bâtons).

On cherche maintenant à valider la convergence théorique vers $(1 - e^{-2})/2$ par un calcul numérique exact (et pas par simulation). On pose $e_{-1} = 0, e_0 = 0, e_1 = 1$ et pour $N \geq 2, e_N = \mathbb{E}[X_\infty(N)]$. On cherche un moyen de calculer numériquement e_N .

T3. Justifier que pour tout $N \geq 2$ on a

$$e_N = 1 + \frac{1}{N} \sum_{u=1}^N (e_{u-2} + e_{N-u-1}). \quad (\star)$$

(Indication : On peut conditionner par rapport à U_1 .)

S3. Écrire un script python qui permet de calculer e_N en utilisant l'équation (\star) . Donner la valeur de e_{100} .

1.1.3 Courbe limite du processus de densité X_t

À N fixé le processus $(X_t(N))_{t \geq 0}$ est croissant au cours du temps et finit par s'arrêter à une valeur aléatoire $X_\infty(N)$, dont l'espérance est proche de $N(1 - e^{-2})/2$ d'après les questions précédentes. On cherche maintenant à décrire plus précisément le processus $(X_t(N))_{t \geq 0}$.

On pose

$$\mathbf{x} : s \mapsto \frac{1}{2} (1 - \exp(-2(1 - e^{-s}))).$$

¹Une preuve accessible au niveau MAP361 est donnée ici : R.Pinsky. *Problems from the Discrete to the Continuous*. Springer (2014).

²C.S.Marvel, C.L.Levesque, *Journal of the American Chemical Society*, vol.60 (1938).

Il a été démontré³ que lorsque $N \rightarrow +\infty$ le processus $(X_t(N))_{t \geq 0}$ suit une courbe de plus en plus proche de celle de $N \times \mathbf{x}(t/N)$. Formellement, pour tout réel $S > 0$ et tout réel $\varepsilon > 0$

$$\mathbb{P} \left(\sup_{0 \leq s \leq S} \left| \frac{1}{N} X_{\lfloor sN \rfloor}(N) - \mathbf{x}(s) \right| > \varepsilon \right) \xrightarrow{N \rightarrow +\infty} 0, \quad (\oplus)$$

où $\lfloor z \rfloor$ désigne la partie entière d'un réel z .

S4. Afficher le graphique d'une simulation qui permet d'illustrer la convergence (\oplus) . (À vous de choisir le type de simulation, ainsi que N, S .)

1.1.4 Fin du processus

On cherche à montrer que le processus s'arrête forcément, et donner une estimation du temps nécessaire pour saturer. On note $F_N \in \{1, 2, \dots\} \cup \{+\infty\}$ la variable aléatoire donnée par le premier instant auquel la configuration est saturée.

T4. Justifier que

$$\{F_N > t\} \subset \bigcup_{u \in \{1, 2, \dots, N\}} \{ \text{Pour tout } s \leq t, U_s \neq u \}.$$

T5. En déduire que

$$\mathbb{P}(F_N > t) \leq N \left(1 - \frac{1}{N}\right)^t. \quad (\#)$$

et démontrer que le processus s'arrête forcément : $F_N < +\infty$ presque-sûrement.

1.2 Modèle uniforme

On note \mathcal{S}_N l'ensemble des configurations saturées sur \mathcal{I}_N :

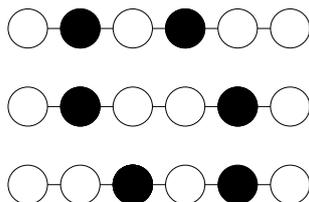
$$(c_0, \dots, c_{N+1}) \in \{0, 1\}^{N+2} \in \mathcal{S}_N \Leftrightarrow \begin{cases} \text{Bords vides :} & c_0 = c_{N+1} = 0, \\ \text{Contrainte de sphères dures :} & \forall 0 \leq i \leq N, c_i \times c_{i+1} = 0, \\ \text{Saturation :} & \forall 0 \leq i \leq N-1, c_i + c_{i+1} + c_{i+2} \geq 1. \end{cases}$$

On note $C^{(N)}$ une configuration uniforme dans \mathcal{S}_N . Comme l'était M_∞ précédemment, $C^{(N)}$ est une variable aléatoire dans $\{0, 1\}^{N+2}$. On écrit $C^{(N)} = (C^{(N)}(0), C^{(N)}(1), \dots, C^{(N)}(N+1))$.

Remarque : Le modèle uniforme est complètement statique, il n'y a plus de notion de temps.

On note $Z_N = C^{(N)}(1) + \dots + C^{(N)}(N)$ le nombre de sphères dans la configuration $C^{(N)}$. L'objectif est d'étudier numériquement le comportement asymptotique de $\mathbb{E}[Z_N]$.

On pose $s_0 = s_1 = 1, s_2 = 2$ et pour $N \geq 2$ on pose $s_N = \text{card}(\mathcal{S}_N)$, par exemple $s_4 = 3$:



³C'est une conséquence de l'éq.(19) dans : P.C.Hemmer. The random parking problem. *Journal of Statistical Physics*, vol.57, n.3, p. 865-869 (1989).

(Au passage on voit que pour $N = 4$ la variable Z_4 est constante et égale à 2.)

T6. Démontrer que pour tout $N \geq 3$,

$$s_N = s_{N-2} + s_{N-3}.$$

S5. En déduire une fonction `python` qui calcule s_N . Donner la valeur de s_{100} .

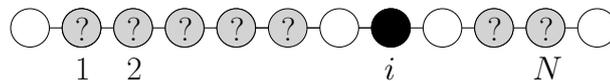
T7. On fixe $N \geq 3$.

- Montrer que pour $i = 1$ ou $i = N$,

$$\mathbb{P}(C^{(N)}(i) = 1) = \frac{s_{N-2}}{s_N}.$$

- Calculer $\mathbb{P}(C^{(N)}(i) = 1)$ pour tout $1 < i < N$.

Indication : Voici une représentation schématique de l'événement $\{C^{(N)}(i) = 1\}$:



S6. Pour $N = 30$ tracer la fonction $i \mapsto \mathbb{P}(C^{(N)}(i) = 1)$. Quelles sont les positions dans $\{1, \dots, 30\}$ qui ont le plus de chance d'être vides dans $C^{(30)}$?

T8. (Bonus, Hors-barème) En combinant les question T6 et T7, calculer de façon exacte la constante ρ telle que $\mathbb{E}[Z_N] \sim N\rho$.

1.2.1 Simulation de la loi uniforme

Il n'est pas évident du tout *a priori* de simuler la mesure uniforme sur \mathcal{S}_N . La méthode du rejet n'est pas envisageable : pour $N = 100$ par exemple un élément uniforme de $\{0, 1\}^{N+2}$ a seulement une probabilité $\simeq 10^{-18}$ d'être une configuration de sphères dures.

Une solution consiste à simuler successivement $C^{(N)}(1)$ puis $C^{(N)}(2), C^{(N)}(3), \dots$, jusqu'à $C^{(N)}(N)$. On a déjà déterminé la loi de $C^{(N)}(1)$ à la question T7, on peut déterminer la loi de chaque $C^{(N)}(i)$ conditionnellement aux $i - 1$ valeurs précédentes en utilisant la méthode de la question T7. Pour simplifier on ne va faire que le cas $i = 3$.

T9. Soit $N \geq 5$. En vous inspirant de la question T7, déterminez la loi jointe de $(C^{(N)}(1), C^{(N)}(2), C^{(N)}(3))$.

S7. Pour $N = 30$ simuler 10000 fois des triplets $(C^{(N)}(1), C^{(N)}(2), C^{(N)}(3))$ et afficher un graphique qui permet de comparer au résultat théorique de la question S6.