

Simulation du modèle XY par méthode Monté Carlo

Rapport de projet pour le cours de Computational Statistical Physics

Aurélien Valade

Introduction

La physique statistique est l'étude des comportements systèmes macroscopiques composés de systèmes microscopiques dont les propriétés sont connues. On va donc systématiquement s'intéresser à un grand nombre de particules en interactions, ce qui est complexe à représenter mathématiquement sans faire approximation de continuité, de champ moyen, etc. Ainsi, les comportements de nombreux modèles de physique statistique ne sont pas descriptibles analytiquement pour des cas pourtant pertinents. Si une résolution analytique n'est pas possible, il faut travailler avec des valeurs numériques, comme cela a par exemple été fait pendant la seconde guerre mondiale pour les calculs portant sur la bombe atomique, ou encore en astrophysique pour des calculs de trajectoires. Cependant la quantité de calcul nécessaire pour résoudre des systèmes réalistes est tout bonnement inhumaine.

L'informatique nous offre une capacité de calcul immensément plus grande que celle des calculatrices humaines, puisqu'on passe d'une dizaine d'opérations par minute à des milliards par seconde. Mais ce n'est toujours pas suffisant pour attaquer le problème de front, par exemple, calculer directement la fonction de partition du système reste souvent hors de portée. Il nous faut donc trouver des méthodes plus puissantes pour résoudre le système. Cette formalisation a cependant un prix puisque de telles méthodes sont alors plus spécifiques : une méthode de résolution est adaptée à une certaine question, mais ne permet pas toujours de bien répondre à d'autres. On pourra ainsi rapidement calculer des propriétés d'un système à l'équilibre sans pouvoir décrire la dynamique physique de la relaxation. Réciproquement, une autre méthode peut permettre de décrire la dynamique du système avec précision, mais demande une très grande quantité de calcul pour atteindre l'état d'équilibre.

Nous allons ici nous pencher sur l'étude du modèle XY, qui est un modèle d'interaction sur réseau se rapprochant du modèle d'Ising, à la différence que les états pris par les spins sont continus entre -1 et $+1$ et non restreints aux valeurs $\{-1, +1\}$. Il serait cependant faux de dire que le modèle XY est une généralisation du modèle d'Ising puisque pour un système équivalent, aucun choix de paramètres du modèle XY ne permet de reproduire des comportements semblables à ceux conjecturés par modèle d'Ising. Le modèle XY est donc adapté à d'autres systèmes que le modèle d'Ising : il permet notamment de décrire des systèmes dont le paramètre d'ordre ne correspond pas à une brisure de symétrie spatiale lors de la transition de phase, mais plutôt à une brisure de la symétrie interne, comme c'est par exemple le cas pour l'Helium II supercritique ou encore les cristaux liquides hexatiques¹.

Nous allons dans un premier temps présenter le problème dans sa formulation physique. Nous verrons ensuite les procédures et algorithmes utilisés et nous étudierons enfin quelques résultats produits.

1. La phase hexatique est une phase située entre les phases solide isotropique et liquide isotropique dans les systèmes de particules à deux dimensions (cristal liquide bidimensionnel). Il est caractérisé par deux paramètres d'ordre : un ordre moléculaire de positionnement à courte portée et un ordre orientationnel à longue portée [Wik17].

1 Modèle physique

1.1 Description du modèle

Soit un ensemble de particules arrangées sur une grille carrée de taille $N \times N$. On note la position dans l'espace \mathbf{r}_{ij} pour la particule de la ligne $0 \leq i < N$ et la colonne $0 \leq j < N$. On utilise aussi l'indexage des particules $0 \leq I < N \times N$ dans les cas où connaître la position dans l'espace n'est pas nécessaire bien qu'on puisse définir $I = i \times N + j$ et lier les deux systèmes d'indexage.

Ces particules ont une propriété appelée spin que l'on peut lier à l'angle entre la normale au plan et un axe intrinsèque à la particule. On note cet angle $\theta_{ij} \in [0, \pi]$. Dans notre modèle, le spin de chaque particule est alors tout simplement le cosinus de l'angle : $\sigma_{ij} = \cos(\theta_{ij})$. On considère que ces particules interagissent entre elles à travers leur spin, et qu'il existe un champ extérieur avec lequel ces spin tendent à s'aligner. Le Hamiltonien décrivant l'énergie d'un tel système s'écrit alors :

$$\begin{aligned} H &= -k \underbrace{\sum_{I, I'} J_{II'} \cos(\theta_I - \theta_{I'})}_{\text{interactions}} - h \underbrace{\sum_I \cos(\theta_I)}_{\text{champ extérieur}} \\ &= -kQ - hM \end{aligned} \quad (1)$$

avec

$$Q = \sum_{I, I'} J_{II'} \cos(\theta_I - \theta_{I'}) , \quad M = \sum_I \cos(\theta_I) \quad (2)$$

nommées respectivement le couplage et la magnétisation par analogie aux systèmes magnétiques.

La forme de la matrice J est centrale au problème. Le modèle XY préconise de prendre une fonction de la distance entre les deux particules : $J_{II'} = (\mathbf{r}_I - \mathbf{r}_{I'})^{-\alpha}$. On peut ensuite choisir de n'appliquer cette définition qu'aux n plus proches voisins et de négliger les interactions à longue distance :

$$J_{II'} = \begin{cases} (\mathbf{r}_I - \mathbf{r}_{I'})^{-\alpha} & \text{si } |i - i'| < n \text{ et } |j - j'| < n, \\ 0 & \text{sinon.} \end{cases} \quad (3)$$

Les paramètres h et k vont nous permettre d'adapter ce système à différents cas concrets. La valeur de h représente l'intensité du champ externe auquel les spins tendent à s'aligner, et k modélise la force du couplage entre les spins. Pour un modèle magnétique, $k > 0$ signifie que le matériau est ferromagnétique : les spins cherchent à pointer dans la même direction et $k < 0$ représente un matériau paramagnétique : les spins cherchent à pointer dans une direction opposée à celle de leurs voisins. Il faut cependant préciser que le cas paramagnétique de ce modèle n'a pas d'application physique et nous intéresse donc moins. Ces quantités ont été préalablement adimensionnées, et dérivent de termes de la forme $h = h' / (\mathcal{K}_B T)$, $k = k' / (\mathcal{K}_B T)$ où \mathcal{K}_B est la constante de Boltzmann et T la température. C'est donc à travers ces paramètres qu'on peut jouer sur la température pour observer les transitions de phase, etc.

1.2 Comportement prédit

A haute température les paramètres h et k du système sont très faibles, le Hamiltonien résultant est donc quasi-nul et n'a pas de minimum significatif. Il n'y a pas de configuration qui puisse réellement minimiser l'énergie, toutes sont donc équiprobables, il en résulte que la magnétisation est nulle en moyenne et que le couplage est à une valeur moyenne non toujours nulle discutée plus loin et dépendante du modèle.

À basse température, le comportement du modèle XY est différent du modèle d'Ising. Tout d'abord, lors de la transition de phase, on n'observe pas de brisure de symétrie spatiale. A champ magnétique externe nul $h = 0$, lors d'une trempe (*i.e.* lors qu'on abaisse progressivement la température du système), la magnétisation reste nulle. En revanche, en présence d'un champ externe, la symétrie du système étant déjà brisée, et les spins s'alignent effectivement avec ce champ.

On observe à la transition de phase un changement globale de comportement de la fonction de corrélation

$$C_{II'} = |\langle \sigma_I \sigma_{I'} \rangle|. \quad (4)$$

En effet, à haute température, la fonction de corrélation décroît exponentiellement :

$$C_{II'}^{HT} \propto e^{\Gamma(\beta)|\mathbf{r}_I - \mathbf{r}_{I'}|} \quad (5)$$

alors qu'à basse température, on s'attend à trouver une fonction qui décroît moins vite que

$$C_{II'}^{BT} \propto \frac{\Gamma(\beta)}{1 + |\mathbf{r}_I - \mathbf{r}_{I'}|}. \quad (6)$$

2 Implémentation

Le but de ce travail est de fournir un programme puissant, flexible, et aisément utilisable qui permette de simuler et d'analyser un modèle XY classique par une méthode Monté Carlo. Il doit notamment utiliser un système de production, de stockage et d'analyse des données cohérent qui permette de reproduire chaque expérience. Il faut donc notamment ne pas écraser les données au fur et à mesure qu'elles sont produites, mais il faut en plus garder des traces des paramètres associés à chaque expérience produite, ainsi que des conventions de nommage compréhensibles et cohérentes.

2.1 Structure du code et des scripts

Le code produit se décompose en deux parties distinctes :

- le code principal d'environ 600 lignes, écrit en C : il produit les données pour une expérience et les stocke dans un dossier ;
- et un code de scripting écrit en Python d'environ 1000 lignes : celui ci permet d'appeler le code C avec les bons paramètres puis d'exploiter les données qui en résultent.

2.2 Fonctionnement du code principal

2.2.1 Entrées et sorties

Chaque expérience est lancée avec la connaissance des paramètres suivant :

- N la taille de grille ;
- n le nombre de plus proches voisins à considérer pour les interactions ;
- a l'exposant de la fonction de décroissance pour les interactions ;
- h la force du champ extérieur ;
- k la force de couplage entre spins ;
- le type d'initialisation :
 - UP : $\theta_I = 0$ pour tout I ;
 - DOWN : $\theta_I = \pi$ pour tout I ;
 - CHECKERBOARD : $\theta_{ij} = \pi |\text{mod}(i, 2) - \text{mod}(j, 2)|$ pour tout i, j ;
 - RANDOM : $\theta_I = U([0, \pi])$ pour tout I avec U la loi uniforme.
- le type de Monté Carlo : METROPOLIS ou GLAUBER.
- Ainsi que le nombre de pas, la fréquence de sauvegarde des différentes données et le répertoire où les stocker.

En sortie, le programme fournit les données suivantes :

- Un fichier où sont stockées les valeurs des quantités M , Q et E toutes les `n_sca` itérations ;
- Un fichier toutes les `n_lat` itérations, représentant l'ensemble des angles de la grille ;
- Un fichier toutes les `n_lat` itérations, représentant l'ensemble des interactions Q_I de la grille.

2.2.2 Définition des quantités numériques

Les quantités physiques introduites précédemment sont définies numériquement comme suit :

Magnétisation. La définition de la magnétisation est similaire à sa définition physique :

$$M = \sum_I \cos(\theta_I).$$

On a donc $M \in [-N \times N, N \times N]$.

Couplage. On introduit la matrice de couplage normalisée telle que

$$\hat{J}_{II'} = \frac{J_{II'}}{\sum_{KK'} J_{KK'}}.$$

Cette définition permet de garder un couplage “indépendant” du nombre de voisins considérés. En effet, en absence de normalisation, pour une configuration donnée, l’énergie de couplage de chaque particule grandit fortement avec le nombre de plus proches voisins considérés, ce qui peut s’avérer gênant pour comparer des simulations à n différent. On introduit ensuite le couplage local

$$Q_I = \sum_{I'} \hat{J}_{II'} (\cos \theta_I \cos \theta_{I'} - \sin \theta_I \sin \theta_{I'})$$

tel que $Q = \sum_I Q_I$, et dont les valeurs qui sont stockées sur la grille des interactions introduite plus haut. Développer le cosinus en une différence de produits de cosinus/sinus permet de stocker les valeurs de cosinus et de sinus pour chaque angle et de calculer un minimum de fonctions trigonométriques à chaque mise à jour. Avec la normalisation de la matrice de couplage, on a aussi $Q \in [-N \times N, N \times N]$.

Énergie. On définit l’énergie du système

$$E = -kQ - hM.$$

Quantités réduites. Pour pouvoir comparer des expériences dont la taille de la grille diffère, on utilise les valeurs réduites :

$$m = M/N^2, \quad q = Q/N^2, \quad e = E/N^2.$$

Ces valeurs sont particulièrement stables puisqu’on a $m, q \in [-1, 1]$ et $e \in [-|h| - |k|, |h| + |k|]$ pour toute expérience, quelque soient ses paramètres. On n’utilisera uniquement ces valeurs dans la suite.

2.2.3 Algorithmes utilisés

Après initialisation des différentes parties du code, on procède à l’exploration de l’espace des phases par méthode de Monté Carlo comme vue en cours. À chaque itération, une cellule de la grille est choisie au hasard à laquelle on assigne un angle aléatoire. On recalcule l’énergie de la nouvelle configuration. On propose ensuite deux méthodes

- **METROPOLIS** : la modification est acceptée avec une probabilité de $\min(1, \exp(-\Delta E))$, sinon elle est rejetée ;
- **GLAUBER** : la modification est acceptée avec une probabilité de $1/(1 + \exp(\Delta E))$, sinon elle est rejetée.

```

i ← int_uniform_random(0, N − 1)                                ▷ ligne à modifier
j ← int_uniform_random(0, N − 1)                                ▷ colonne à modifier
a ← uniform_random(0,  $\pi$ )                                       ▷ nouvel angle
p ← uniform_random(0, 1)                                         ▷ probabilité d'acceptation

b ← get_lattice_value(lattice, i, j)                            ▷ sauvegarde de l'angle
E ← get_energy(lattice)                                          ▷ sauvegarde de l'énergie

set_lattice_value(lattice, i, j, a)                            ▷ assignation du nouvel angle
compute_energy(lattice)                                           ▷ calcul de l'énergie dans la nouvelle configuration
 $\Delta E \leftarrow \text{get\_energy}(\text{lattice}) - E$ 

if monte_carlo=METROPOLIS then
    if exp( $-\Delta E$ ) < p then                                ▷ si le pas est rejeté
        set_lattice_value(lattice, i, j, b)                ▷ réassignation de l'ancienne valeur d'angle
        set_energy(lattice, E)                               ▷ réassignation de l'ancienne valeur d'énergie
    end if
else if monte_carlo=GLAUBER then
    if 1/(1 + exp( $\Delta E$ )) < p then                            ▷ si le pas est rejeté
        set_lattice_value(lattice, i, j, b)                ▷ réassignation de l'ancienne valeur d'angle
        set_energy(lattice, E)                               ▷ réassignation de l'ancienne valeur d'énergie
    end if
end if

```

FIGURE 1 – Procédure de mise à jour de la grille en pseudo code.

La procédure est décrite en pseudo code dans la figure 1.

Une fois le spin d'une particule mis à jour, il faut mettre à jour toutes les quantités scalaires, à savoir M , Q et E . Il s'agit de la partie du code qui demande le plus de calcul. Plus précisément, c'est le calcul du couplage qui est le plus lourd, et l'est d'autant plus que l'on considère les actions à longue distance.

Une approche naïve serait de recalculer la magnétisation et le couplage après chaque modification sur toute la grille ; une telle approche serait de complexité $\mathcal{O}(n^2 \times N^2)$. Une meilleur algorithme, celui-ci en $\mathcal{O}(n^2)$, est décrit en figure 2. L'idée consiste à ne recalculer que la partie de grille affectée par le changement et de reporter directement les modifications sur les scalaires M et Q . Le poids final de chaque opération est reporté en figure 3.

2.3 Fonctionnement des scripts

2.3.1 Script de lancement du code principal

Ce script est un “wrapper” du code principal. Il simplifie l'appel par le nommage des variables, par le passage des valeurs par défaut, etc².

Il permet aussi de lancer plusieurs types d'expériences :

- **Simple sampling.** Le script permet alors de nommer automatiquement les dossiers de stockage, lance bien sûr automatiquement les N_{simple} expériences et compile tous les fichiers de scalaires en un seul pour une analyse postérieure. Le programme principal est appelé avec un nombre d'itérations à faire nul. Il s'initialise donc avec une grille aléatoire, calcul son énergie, enregistre la grille ainsi que ses valeurs scalaire et quitte.
- **Sweep.** Il s'agit de parcourir un ensemble de paramètres et de lancer une simulation pour chaque valeur. Contrairement à ce qui était prévu initialement, on se concentre sur des en-

2. On utilise le package python `argparse` pour le passage des arguments, plus simple à gérer que les fonctions POSIX en C.

Require: a, b la nouvelle/ancienne valeur de l'angle modifié	
Require: i, j la position de l'angle modifié	
Require: M, Q, E la magnétisation, le couplage et l'énergie	
$M \leftarrow M - \cos(b) + \cos(a)$	▷ Mise à jour de la magnétisation
$p \leftarrow 0, q \leftarrow 0$	▷ Ancienne/nouvelle valeur du couplage local
for i', j' in the neighborhood of i, j do	▷ Boucle sur les particules en interaction
$p \leftarrow p + \text{get_interaction_value}(i', j')$	▷ Ancienne valeur
$\text{compute_interaction_one_particule}(i', j')$	▷ Mise à jour couplage local
$q \leftarrow q + \text{get_interaction_value}(i', j')$	▷ Nouvelle valeur
end for	
$Q \leftarrow Q - p + q$	▷ Mise à jour du couplage
$E \leftarrow -kQ - hM$	▷ Calcul de l'énergie

FIGURE 2 – Procédure de mise à jour de l'énergie en pseudo code.

Children	Self	...	Symbol
...			
+ 81.17%	0.41%	...	update
+ 75.87%	2.28%	...	calc_all_one_change
+ 67.41%	22.76%	...	calc_inter_one_spin
+ 14.56%	14.56%	...	get_cos
+ 12.19%	12.19%	...	get_sin
+ 11.88%	11.88%	...	add_inter
+ 8.29%	8.29%	...	periodic_conditions
+ 7.77%	7.77%	...	get_dist
...			

FIGURE 3 – Resultats de **perf** sur le surcoût de calcul pour le programme principal. La fonction **update** est décrite en figure 1 et la fonction **calc_all_one_change** est décrite en 2. Les fonctions **get_*** ne servent qu'à lire les valeurs des différents tableaux, **add_inter** ajoute une valeur donnée sur une partie de la grille des couplages locaux, et **periodic_conditions** fait la transformation d'indice qui simule les conditions périodiques.

sembles de couples (h, k) : on peut donc lancer automatiquement $n_h \times n_k$ simulation avec $\{(h_i, k_j)\}_{h_0 < \dots < h_{n_h}, k_0 < \dots < k_{n_k}}$. En plus de lancer les expériences automatiquement, le script nomme automatiquement les dossiers de stockage.

- **Parallel sweep.** Même chose pour que pour le sweep, mais les instances du code principale s'exécutent en parallèle. On lance les processus en parallèle par grappe de n_{par} , on attend que tous aient fini, et on lance la grappe suivante.

2.3.2 Scripts d'analyse des données

Ces scripts prennent des dossiers de données formatés par les scripts de lancement présentés précédemment. Pour une seule expérience, plusieurs visualisations sont possibles :

- Tracé des scalaires m , q et e en fonction du nombre d'itération ;
- Tracé de q en fonction de m ;
- Tracé de grilles θ_{ij} (une seule, un ensemble, toutes, la dernière), si plusieurs grilles sont demandées, elles sont affichées sous forme d'une film à 5 fps par défaut ;
- Tracé de la fonction de d'auto-corrélation radiale $C(r)$.

Fonction d'auto-corrélation. La fonction de d'auto-corrélation radiale peut être vue comme la moyenne des corrélations

$$C(r) = \langle \sigma_i \sigma_j \rangle_{i^2+j^2=r^2} = \sum_{\substack{i,j \\ i^2+j^2=r^2}} \sigma_i \sigma_j \quad (7)$$

Le processus utilisé pour la calculer est décrit en figure 4. Le calcul de cette fonction est relativement lourd, il est cependant fondamental pour observer la transition de phase. D'autres méthodes, plus rapides, basées sur des transformées de Fourier ont été essayées, sans succès cependant.

On rappelle qu'on s'attend à trouver une fonction de la forme

$$C_h(r) = be^{-ar} \iff \log(C_h(r)) = -ar + b \quad (8)$$

pour des températures au delà de la transition de phase, et

$$C_l(r) = \frac{1}{b+ar} \iff \frac{1}{C_l(r)} = ar + b \quad (9)$$

pour des températures inférieure à la transition de phase. Comme on peut le voir, il est facile de transformer ces fonctions en équations linéaires, ce qui peut s'avérer utile pour reconnaître l'un ou l'autre des régimes. On va donc chercher à *fitter* ces deux droites affines et noter les R^2 pour essayer d'observer une transition de comportement.

Stabilisation. Une première méthode qui permet de détecter la stabilisation est mise en place. Il s'agit de moyenner les m derniers points de $E(t)$, et de noter la première fois que cette valeur est atteinte. Cette heuristique se base sur le fait que l'énergie est globalement décroissante si (h, k) , et qu'elle converge donc vers sa valeur finale, moyennant fluctuations. Pour éviter les problèmes d'égalités entre nombres flottants et pour tenir compte des dites fluctuations, on considère :

$$t_{\text{stab}} = \min t \text{ t.q. } \left(\frac{\langle E \rangle_{\text{end}} - E(t)}{E(0) - \langle E \rangle_{\text{end}}} \right) < 1.$$

Erreur statistique. On peut ensuite se poser la question de la corrélation temporelle : une longue expérience doit avoir un poids statistique plus fort qu'une expérience plus courte, le lier à l'erreur statistique est plus difficile. Une méthode pour estimer l'erreur est présentée en figure 5. Elle consiste à découper récursivement une expérience en plusieurs sous-expériences de taille un demi, et de choisir l'ensemble de sous-expériences pour laquelle l'erreur statistique

$$\Sigma_N^* = \frac{\sigma}{\sqrt{N}}$$

```

Require: prod_shift(lattice, i, j) qui retourne un tableau tel que  $A_{kl} = B_{kl}B_{(k+i)(j+l)}$  avec des condi-
tions periodiques
Require: sum(lattice) qui ajoute toutes les valeurs d'une grille
Require: abs_sort( $X, Y$ ) tri les valeurs des listes  $X$  et  $Y$  rapport à  $X$ 
 $X \leftarrow \emptyset, Y \leftarrow \emptyset$ 
for  $i, j$  in  $[0, N/4] \times [0, N/4]$  do                                ▷ Boucle sur le voisinage à  $N/4$  dans une direction
    push_item( $X, \sqrt{i^2 + j^2}$ )                                       ▷ Calcul du rayon
     $S^{(++)} = \text{prod\_shift}(\text{lattice}, i, j)$                         ▷ corrélations par direction
     $S^{(+-)} = \text{prod\_shift}(\text{lattice}, i, -j)$ 
     $S^{(-+)} = \text{prod\_shift}(\text{lattice}, -i, j)$ 
     $S^{(--)} = \text{prod\_shift}(\text{lattice}, -i, -j)$ 
    push_item( $Y, \text{sum}\left(\frac{S^{(++)} + S^{(+-)} + S^{(-+)} + S^{(--)} }{4N^2}\right)$ )  ▷ Moyenne des corrélations directionnelles
end for
 $X, Y \leftarrow \text{abs\_sort}(X, Y)$ 

```

FIGURE 4 – Procédure de calcul de la fonction d'auto-corrélation en pseudo code.

est la plus grande. Cela nous permet de savoir combien de sous-expériences “indépendantes” sont présentes dans chaque longue expérience.

On définit aussi l'erreur statistique sur N expériences indépendantes comme

$$\Sigma_N = \frac{\sigma(\{\bar{x}\})}{\sqrt{N}}, \quad x \in \{m, q, e\},$$

c'est à dire l'écart type sur l'ensemble des moyennes obtenues divisée par la racine carrée du nombre d'expériences indépendantes. On pourrait combiner ces deux définitions en décomposant chaque expérience réelle en plusieurs sous-expériences *a priori* décorréliées. Cependant, comme on le verra plus loin, la définition de l'erreur statistique ne produit pas de bons résultats³.

3 Résultats

3.1 Test du code sur des configurations simples

On teste le code sur des configurations connues :

— UP :

On trouve bien $m = q = 1$. Il s'agit d'une configuration stable à basse température, pour $h > 0, k > 0$.

— DOWN :

On trouve bien $m = -1, q = 1$. Il s'agit d'une configuration stable à basse température, pour $h < 0, k > 0$.

— CHECKERBOARD :

On trouve bien $m = 0, q = -0.17$. Il s'agit d'une configuration stable à haute température, pour $k = 0, k < 0$.

3.2 Simple sampling

Une fois qu'on est sûr que les valeurs scalaires sont bien celles voulues, on tente d'explorer l'espace des phases par simple sampling. La figure ?? montre les valeurs de m, q obtenues pour une telle expérience. On obtient des valeurs centrées en $m = 5 \cdot 10^{-5}$, $q = 0.40$ avec un étalement $\sigma_m = \sigma_q = 10^{-1}$ pour 10000 expériences sur une grille de $N = 128$. On ne couvre donc qu'une très faible

3. Il s'agit plutôt de résultats que je ne comprends pas...

Require: L_M, L_Q, L_E les listes des valeurs de M, Q, E après stabilisation

Require: $\text{average_2}(A)$ une fonction qui renvoie une liste de taille $N/2$ dont chaque point vaut

$$B_i = 1/2 (A_{2i} + A_{2i+1})$$

Require: $\text{standard}(A)$ une fonction qui calcule l'écart type

$S \leftarrow (\emptyset, \emptyset, \emptyset)$

for $i \in \{M, Q, E\}$ **do**

$L_n = \emptyset$

$n \leftarrow \text{len}(L_i)$

while $n > 1$ **do**

$L_i = \text{average_2}(L_i)$

$\text{append_value}(\text{get_sublist}(S, i), \text{standard}(L_i))$

$n \leftarrow \text{len}(L_i)$

end while

end for

$n_M \leftarrow \text{argmax get_sublist}(S, 1)$

$n_Q \leftarrow \text{argmax get_sublist}(S, 2)$

$n_E \leftarrow \text{argmax get_sublist}(S, 3)$

$n \leftarrow \min(n_M, n_Q, n_E)$

$e_M \leftarrow \text{get_value}(S(1), n)$

$e_Q \leftarrow \text{get_value}(S(2), n)$

$e_E \leftarrow \text{get_value}(S(3), n)$

FIGURE 5 – Procédure de calcul de l'erreur statistique sur une longue expérience.

portion des configurations accessibles : la partie précédente nous a montré qu'on pouvait accéder à des configurations telles que $m, q = 1$, etc. Ceci était tout à fait prévisible, sachant que pour un modèle d'Ising de taille équivalente il y a environ 10^{4900} configurations possibles et que le modèle XY étant continu, le nombre de d'état accessible est bien plus grand⁴.

On note que les états initiaux utilisés dans la suite pour l'importance sampling sont issus des mêmes procédés aléatoires, la distribution présentée ci dessus est donc la distribution des états initiaux. Contrairement à ce que l'on pourrait penser donc, utiliser des configurations aléatoires comme configurations initiales ne permet de démarrer des simulations à partir de points distants de l'espace des phases, au contraire.

3.3 Expérience à température infinie, sans polarité favorisée

Pour simuler une température infinie, on prend $(h, k) = (0, 0)$. Comme le montre la figure 7, l'énergie relative reste donc nulle tout au long de l'expérience, toute modification est acceptée. La moyenne des spins et les couplages sont donc aléatoires. Ce comportement est cohérent avec la physique du problème : à température infinie, chaque particule a une énergie infinie, et donc est indifférente à sa configuration.

Pour un ensemble de 1000 expériences, la moyenne se stabilise autour de $m = 3 \cdot 10^{-5}$, $q = 0.405$, avec une erreur statistique de $\Sigma_m = 5 \cdot 10^{-4}$, $\Sigma_q = 3 \cdot 10^{-4}$.

Comme attendu, la magnétisation moyenne est aux alentours de 0, cependant le couplage oscille lui vers 0.4. Ceci s'explique par le fait que l'on ne considère dans les plus proches voisins pas que les points $\{(i+1, j), (i, j+1), (i-1, j), (i, j-1)\}$, mais aussi les diagonales $\{(i+1, j+1), (i-1, j-1), (i+1, j-1), (i-1, j+1)\}$. Avec ces interactions, la valeur moyenne de couplage est non nulle. On note

4. Infiniment plus grand... Il faudrait plutôt passer par une approche continue pour pouvoir comparer, mais une telle analyse n'a pas été faite ici.

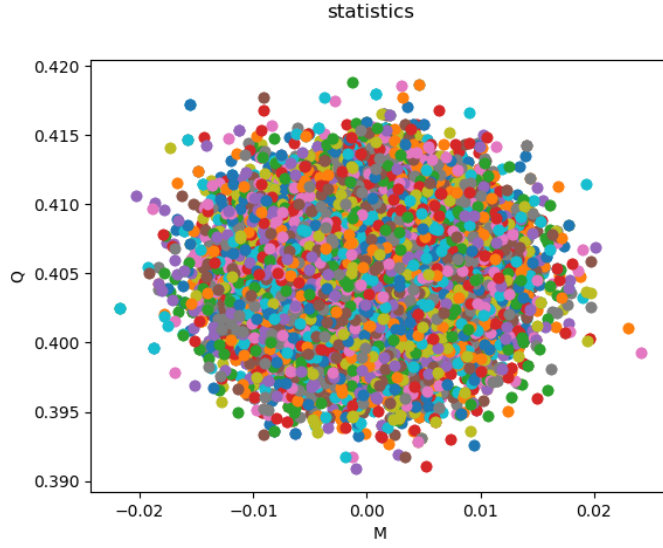


FIGURE 6 – Étalement des valeurs m, q pour 10000 expériences, pour une grille de $N = 128$, avec prise en compte des $n = 1$ plus proches voisins et une décroissance en $a = 1$.

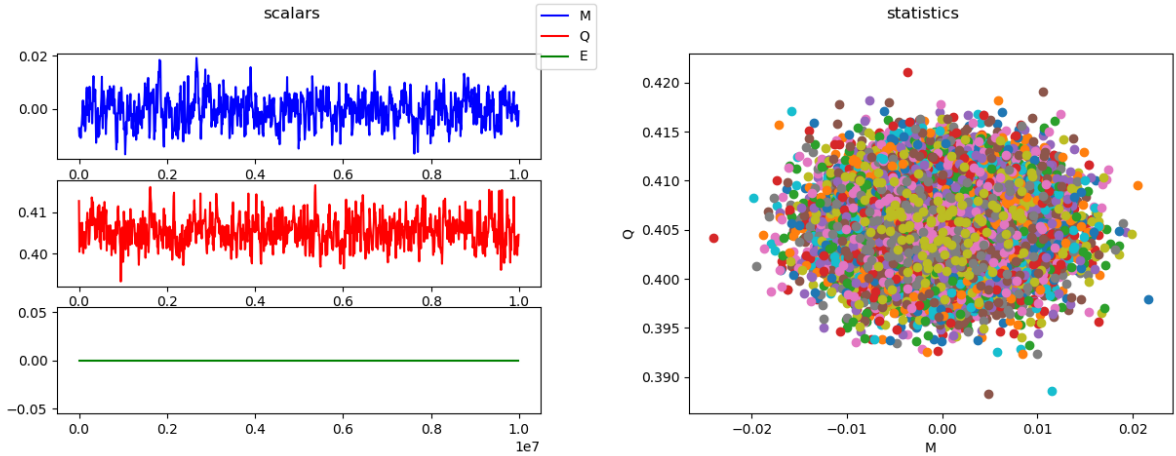


FIGURE 7 – Scalaires pour une expérience à température infinie $(h, k) = (0, 0)$ pour une grille de $N = 128$, avec prise en compte des $n = 1$ plus proches voisins et une décroissance en $a = 1$. On a pour cette réalisation $\Sigma_m^* = 3 \cdot 10^{-4}$, $\Sigma_q^* = 1.5 \cdot 10^{-4}$.

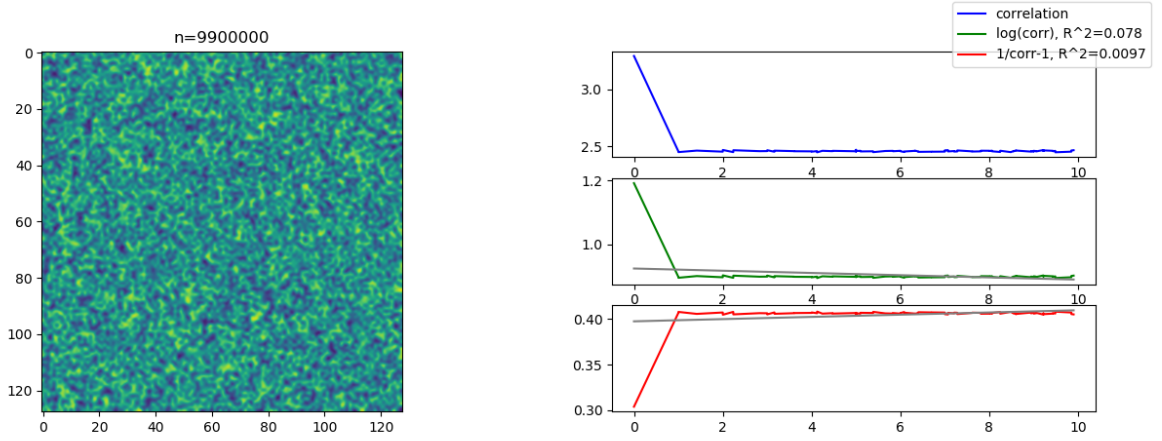


FIGURE 8 – Grille finale et fonction d’auto-corrélation radiale, une expérience à température infinie $(h, k) = (0, 0)$ pour une grille de $N = 128$, avec prise en compte des $n = 1$ plus proches voisins et une décroissance en $a = 1$.

d’ailleurs qu’aucune configuration ne permet d’avoir un couplage $q = -1$, comme on pouvait l’avoir avec la configuration **CHECKERBOARD** sans les interactions diagonales.

La fonction de corrélation radiale est présentée en figure 8. On remarque une décroissance très rapide de la corrélation comme prévue. En revanche celle-ci ne semble pas suivre une loi exponentielle comme prédit par la théorie. Cela peut être lié au fait qu’à température infinie, il n’y a *a priori* aucune corrélation, comme on peut le voir sur les courbes, la fonction d’auto-corrélation tombe à sa valeur minimale dès $r = 1$.

3.3.1 Expérience à haute température, sans polarité favorisée

On choisit arbitrairement $(h, k) = (0, 1)$ pour simuler une expérience à “haute” température sans polarité favorisée. Comme le montre la figure 9, on observe une convergence du couplage vers une valeur autre que sa valeur initiale : $m = -3 \cdot 10^{-5}$, $q = 0.61$, avec $\Sigma_m = 2 \cdot 10^{-4}$, $\Sigma_q = 1 \cdot 10^{-4}$. Cependant on peut remarquer une oscillation moyenne de q notable autour de sa valeur moyenne, signe du fait qu’il y a à cette températures des fluctuations assez important (de l’ordre du pour-cent). La forme du nuage de points de la figure 9 nous permet de voir que q est plus contraint que m puisque le nuage est plus contracté verticalement qu’horizontalement, et c’est effectivement ce que nous disent les statistiques : $\Sigma_q/\Sigma_m \approx 2$. La magnétisation semble elle toujours non contrainte, puisque Σ_m reste le même qu’à température infinie, mais on observe aussi que les variations sont plus lentes, il y donc bien une certaine contrainte.

La fonction d’auto-corrélation radiale nous montre qu’il y a un peu de corrélation : sur un ou deux points. Cependant, il ne s’agit toujours pas d’une forme exponentielle comme attendu, la décroissance semble en effet trop rapide, puisque la courbe reste convexe même en échelle semi-logarithmique.

3.3.2 Expérience à basse température, sans polarité favorisée

On prend $(h, k) = (0, 100)$ pour simuler cette expérience à basse température. La figure ?? présente les courbes pour les scalaires. On observe une magnétisation moyenne de $m = 3 \cdot 10^{-4}$ et un couplage de $q = 0.995$ avec des erreurs statistiques de $\Sigma_m = 4 \cdot 10^{-4}$, $\Sigma_q = 9 \cdot 10^{-7}$. On remarque que le couplage est stabilisé et oscille très peu. La magnétisation, qui n’est elle toujours pas directement contrainte, semble toujours prendre des valeurs aléatoires, mais on remarque que ses variations sont très lentes. Cela se voit aussi très bien sur le graphe de droite où on observe que les valeurs de m prises par chaque expérience dépendent beaucoup des conditions initiales et qu’elles ne durent pas assez longtemps pour se recouvrir les unes les autres.

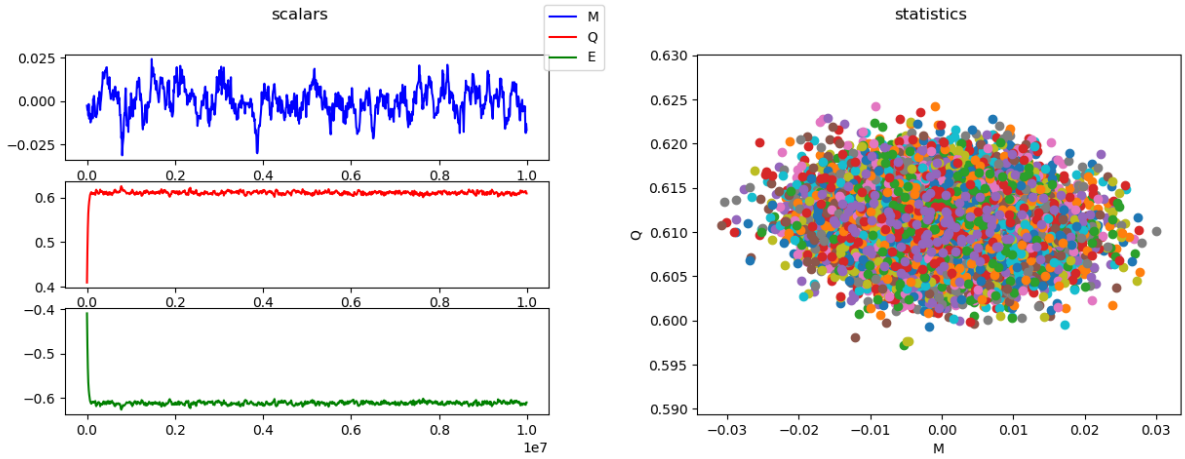


FIGURE 9 – Scalaires d’une expérience à haute température $(h, k) = (0, 1)$ pour une grille de $N = 128$, avec prise en compte des $n = 1$ plus proches voisins et une décroissance en $a = 1$. On a pour cette réalisation $\Sigma_m^* = 1 \cdot 10^{-3}$, $\Sigma_q^* = 3 \cdot 10^{-4}$.

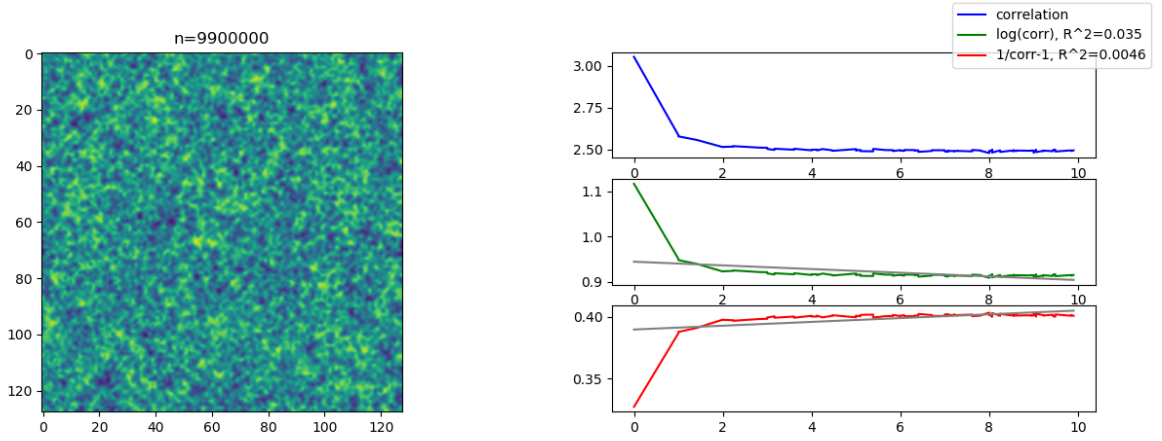


FIGURE 10 – Grille finale et fonction d’auto-corrélation radiale, une expérience à haute température $(h, k) = (0, 1)$ pour une grille de $N = 128$, avec prise en compte des $n = 1$ plus proches voisins et une décroissance en $a = 1$.

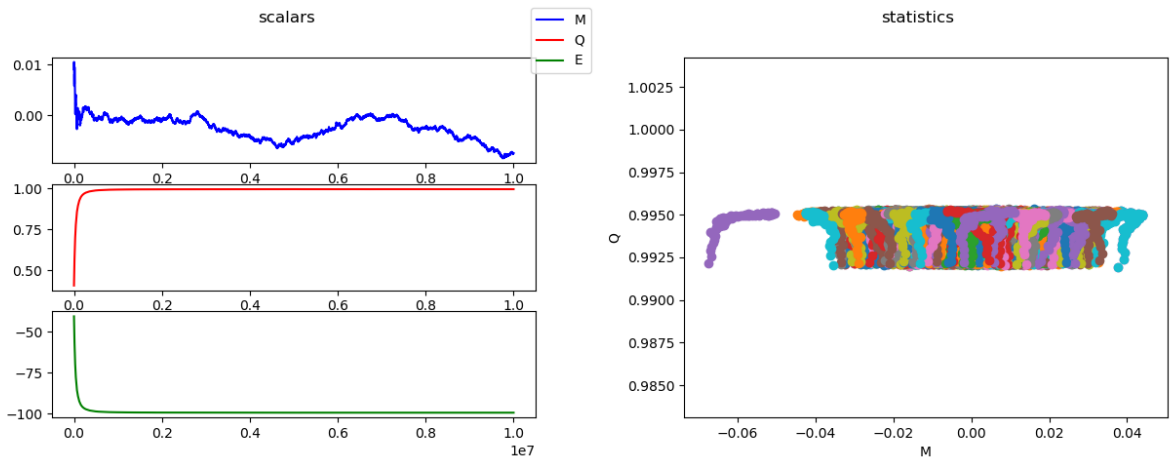


FIGURE 11 – Scalaires pour une expérience à basse température $(h, k) = (0, 100)$ pour une grille de $N = 128$, avec prise en compte des $n = 1$ plus proches voisins et une décroissance en $a = 1$. On a pour cette réalisation $\Sigma_m^* = 1 \cdot 10^{-3}$, $\Sigma_q^* = 3 \cdot 10^{-4}$.

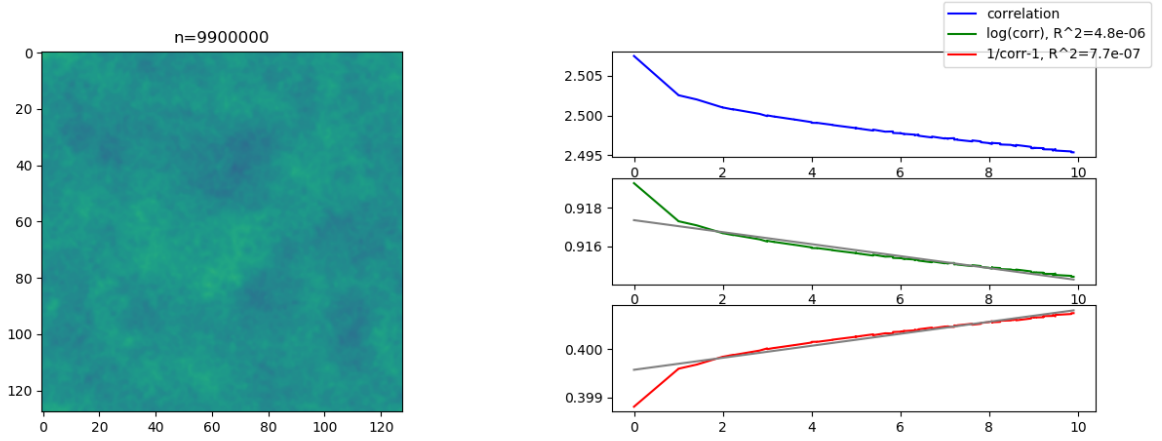


FIGURE 12 – Grille finale et fonction d’auto-corrélation radiale, pour une expérience à basse température $(h, k) = (0, 100)$ pour une grille de $N = 128$, avec prise en compte des $n = 1$ plus proches voisins et une décroissance en $a = 1$.

Sur la figure ??, on remarque directement sur la grille que les valeurs sont beaucoup plus uniformes qu’à haute température, et qu’elles sont relativement centrées autour de $\theta = \pi/2$, contrairement aux expériences à plus haute température où on pouvait voir des variations en amplitudes beaucoup plus grandes ainsi qu’une corrélation spatiale bien plus courte. En effet, on voit sur la courbe de droite que les points dans un rayon de 10 sont corrélés, bien que faiblement. La courbe semble cependant ne suivre ni loi exponentielle, ni loi affine inverse. Elle est toujours convexe, bien qu’on puisse obtenir une forme concave en prenant k avec des valeurs encore plus grandes.

3.4 Parcours de l’espace des configurations par les paramètres h et k

Dans un premier temps, on peut se poser la question de l’influence d’un champ extérieur sur l’état stationnaire du système. Pour cela on se propose d’afficher les valeurs des scalaires m , q , e à l’équilibre en fonction de h (*resp.* k) pour des isovaleurs du k (*resp.* h). Un tel graphe est présenté en figure 13. On observe les comportements attendus, *i.e.* on a une convergence continue plus ou moins forte vers une magnétisation du même signe de h . Réciproquement on observe un couplage qui tend vers $q = 1$ quand k va croissant, cependant la valeur minimale atteinte par q est aux alentours de -0.3 . On impute ce comportement au fait que les diagonales sont considérées parmi les plus proches voisins. On remarque aussi que cette valeur est plus basse que celle donnée par la configuration du **CHECKERBOARD**, poussant à se demander quelle forme non triviale prend la grille pour minimiser cette valeur. La réponse est apportée dans la partie suivante.

On se propose maintenant de parcourir toute la surface des observables possibles par modifications des paramètres h et k . La figure 14 montre le résultat d’une telle approche. On observe globalement une forme de l’ensemble des configurations similaire à ce qu’on peut trouver avec un modèle d’Ising. On remarque cependant que les états de très bas couplages ne sont pas accessibles, puisqu’on n’atteint un minimum au environs de $q = -0.3$. On remarque aussi des états de fort couplage $q \approx 1$ avec des m moyens semblent difficiles d’accès. En effet, il est impossible d’atteindre une configuration avec $q = 1$ avec $m \neq \pm 1$, mais la limite exacte n’est pas nette sur ce graphe. Si ils existent, il faudrait utiliser des valeurs de h très faibles pour les atteindre, mais il faudrait alors aussi faire attention à bien être en présence de systèmes stabilisés et non en transition vers $m = \pm 1$.

On observe aussi que les îlots sont relativement séparés au milieu, et concentrés sur les bords. Cet effet peut avoir deux sources qui se combinent, soit on a sur-échantillonné les paramètres menant à ces états, soit la densité d’état est effectivement plus basse au centre du graphe.

Enfin on remarque que de nombreux nuages de points ne sont pas complètement circulaires, notamment vers $m = 0$, $q = 1$, ou aux extrémités. Cela nous pousse à penser que les systèmes qu’ils

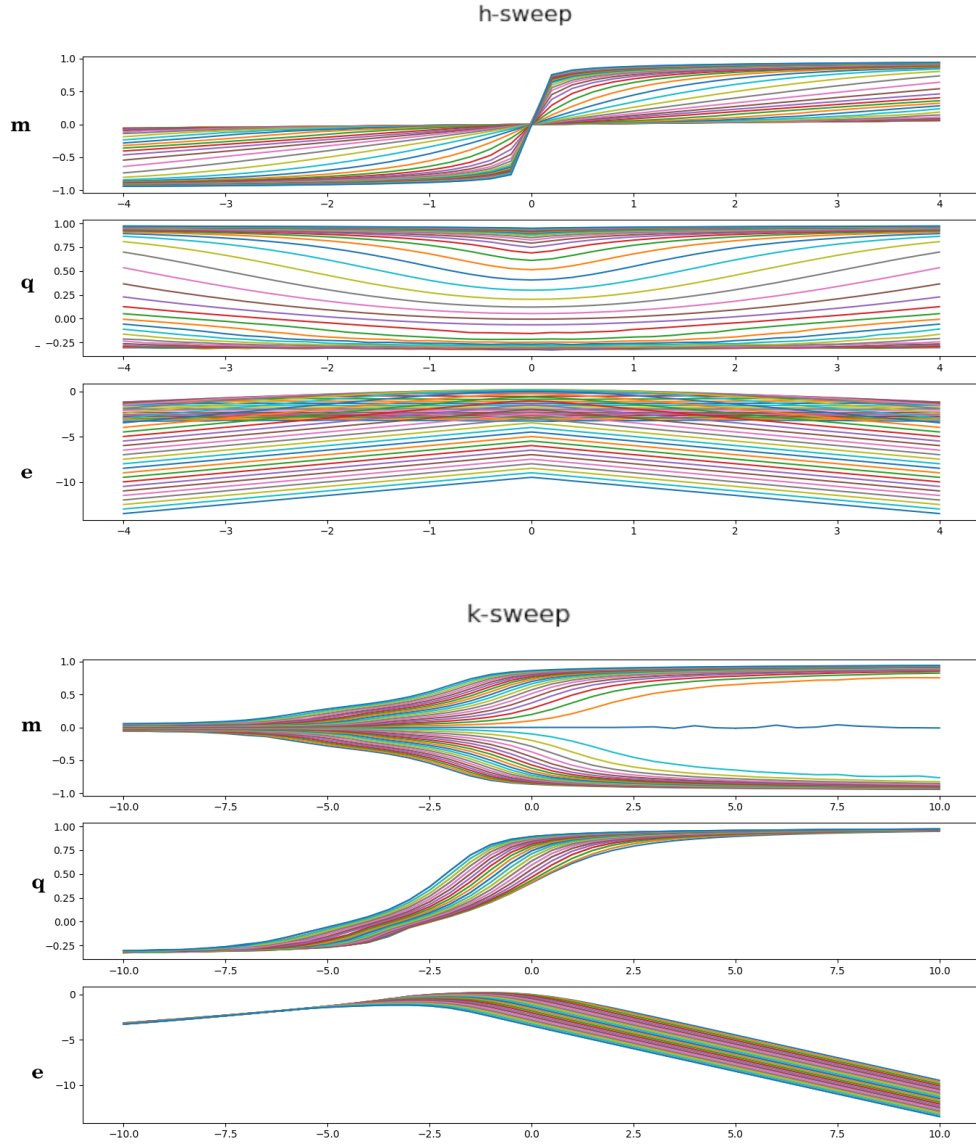


FIGURE 13 – Sweeps pour k et h en fonction de h (*resp.* k) pour des isovaleurs du k (*resp.* h). On a pris une grille de $N = 128$, avec prise en compte des $n = 1$ plus proches voisins et une décroissance en $a = 1$.

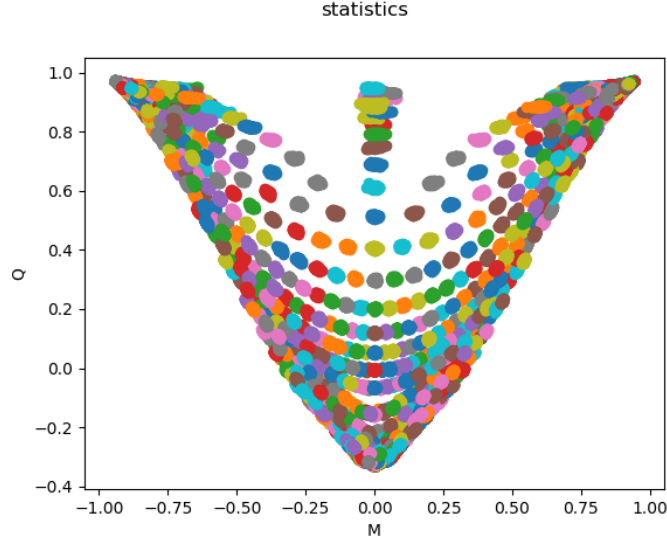


FIGURE 14 – Étalement des valeurs m, q pour des valeurs de $-4 < h <$ et $-10 < k < 10$. On prend une grille de $N = 128$, avec prise en compte des $n = 1$ plus proches voisins et une décroissance en $a = 1$.

représentent ne sont pas totalement stabilisés. En effet, dans un souci de temps de calcul, les nuages de points présents sur ce graphe ne représentent qu’une expérience chacun, et non un millier comme on a pu le voir jusqu’à présent. Certains nuages de points souffrent donc d’une variation forte et lentes de m ou q dans des cas où l’une des deux quantités est extrêmement contrainte et l’autre est “libre”. Utiliser des ensembles statistiques plutôt que des expériences uniques pour construire ce graphe aurait peut-être aussi changer le problème visuel de recouvrement des parties centrales du graphe mentionné dans le paragraphe précédent.

3.4.1 Cas paramagnétique

On prend $(h, k) = (0, -100)$ pour simuler cette expérience à basse température dans un cas paramagnétique. L’analyse de cas n’est pas très pertinente puisque [Wik18] précise qu’il n’a pas d’application, mais la grille qui en résulte est digne d’intérêt intéressante par la forme non triviale des structures qui apparaissent comme on peut le voir dans la figure 15.

4 Conclusion et perspectives

Nous avons donc pu simuler dans ce projet un grand nombre de configurations à l’équilibre pour le modèle XY, dont seuls quelques exemples sont présentés dans ce rapport. On a réussi à automatiser une large majorité des procédures, à garder un stockage cohérent permettant une analyse rapide et automatique des résultats produits.

Le code principal, codé en C, a permis de faire des calculs sur des grilles relativement grandes en des temps tout à fait acceptables : on ne présente pas d’analyse de scalabilité dans ce rapport, mais on peut évoquer que le code permet de faire 10 000 000⁵ itérations sur une grille de $N = 128$ en environ 30 secondes.

Cependant, bien que nous avons pu observer quelques comportements intéressants, nous n’avons pas pu accéder à la transition de phase, puisque la fonction d’auto-corrélation radiale calculée ne semble pas se comporter comme le prédit la théorie. On observe donc bien que la corrélation croît quand la température décroît, et qu’elle passe d’un comportement convexe à concave, mais elle ne

5. Il s’agit des valeurs utilisées dans ce rapport comme on peut le voir sur les graphes.

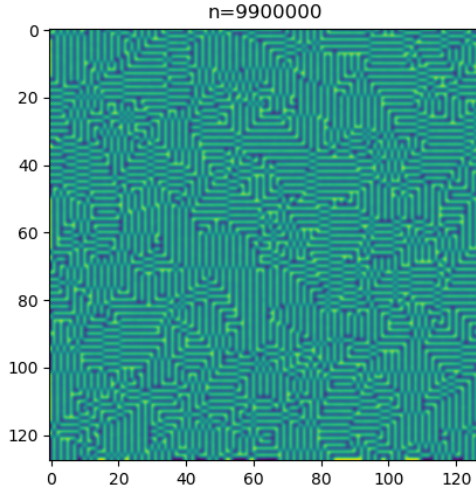


FIGURE 15 – Grille finale pour une expérience à basse température $(h, k) = (0, -100)$ dans un cas paramagnétique, pour une grille de $N = 128$, avec prise en compte des $n = 1$ plus proches voisins et une décroissance en $a = 1$.

semble jamais ressembler à un exponentielle ni ne jamais s’approcher d’une forme affine inverse. Il faudrait donc essayer de trouver d’autres paramètres d’ordre pour observer la transition de phase, ou bien remettre en cause les calculs faits dans ce travail.

Enfin, je n’ai pas pu aller aussi loin que prévu dans le projet, puisque j’aurais notamment aimé faire varier le nombre de plus proches voisins, la taille de la grille, le type de Monté Carlo, ainsi que le paramètre de décroissance spatiale qui semble est fondamental dans le modèle. Toute la machinerie est prête, mais je n’ai pas eu le temps d’analyser les données... J’aurais aussi aimé prendre le temps de mieux comprendre le calcul de l’erreur statistique sur une seule expérience.

Pour aller plus loin, il aurait été intéressant d’appliquer toutes les méthodes vues en cours, mais aussi par exemple de modifier le code pour que les valeurs d’angles attribuées soit discrétisées sur $\in \mathcal{N}$ valeurs et pouvoir ainsi observer l’apport du passage au continu.

De manière plus anecdotique, j’aurais aimé tenter d’automatiser l’initialisation les simulations en partant de points “opposés” dans l’espace des phases pour prouver qu’il ne s’agit pas d’un état métastable auquel on accède en partant du nuage de point resserré présenté dans la partie sur le simple sampling.

Références

- [Wik17] Wikipédia. Phase hexatique — wikipédia, l’encyclopédie libre, 2017. [En ligne ; Page disponible le 7-avril-2017].
- [Wik18] Wikipedia contributors. Classical xy model — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Classical_XY_model&oldid=843750453, 2018. [Online ; accessed 16-October-2018].