

Travaux pratiques n° 5 : Modèle d'Ising à température nulle

Le but de ce TP est d'étudier quelques propriétés du modèle d'Ising, un célèbre modèle de physique statistique qui permet de décrire des effets collectifs dans des systèmes de particules à deux états, et de comprendre par exemple le comportement des matériaux ferromagnétiques.

Pour visualiser les états du système simulé, nous apprendrons à utiliser une « bibliothèque externe de fonctions ». La bibliothèque que nous utiliserons dans ce but est la bibliothèque graphique « g2 », qui permet de faire très facilement des graphismes en C. Pour utiliser « g2 », il faut d'abord, en théorie, télécharger et installer la bibliothèque sur le système. Cette étape a déjà été effectuée sur les machines de TP. Néanmoins, il est probable que cette bibliothèque soit absente sur vos ordinateurs personnels¹.

Voici un exemple de programme :

```
#include<stdio.h>
#include <g2.h>
#include <g2_X11.h>

int main(void){
    int h = g2_open_X11(400, 400);

    g2_line(h, 20,20, 380,380);

    g2_pen(h, 3);          // Sélectionne la couleur no 3 (bleu)
    g2_circle(h, 200,200, 150);

    g2_pen(h, 25);        // Sélectionne la couleur no 25 (jaune)
    g2_filled_rectangle(h, 200,200, 150);

    g2_pen(h, 19);        // Sélectionne la couleur no 19 (rouge)
    g2_string(h, 250,175, "Hello");

    scanf("%*c");        // Attend que l' utilisateur appuie sur « Entrée »

    return 0;
}
```

Les `#include <g2.h>` et `#include <g2_X11.h>` servent à déclarer l'existence des fonctions `g2_open_X11()`, `g2_line()`, `g2_circle()`, etc. L'appel de fonction `g2_open_X11(400, 400)` ouvre une fenêtre de taille 400×400 pixels et renvoie un entier qui sert d'identificateur pour cette fenêtre. On a besoin de fournir cet entier pour tous les autres appels aux fonctions de la bibliothèque g2 pour indiquer la fenêtre où doivent être faits les dessins. L'appel `g2_line(...)` trace une ligne entre les points

¹On peut télécharger la bibliothèque g2 sur <http://g2.sourceforge.net/>

de coordonnées (20, 20) et (380, 380). L'appel à `g2_circle(...)` trace un cercle centré en (200, 200) et de rayon 150, et l'appel à `g2_string(...)` affiche la chaîne de caractères « Hello » à partir du point de coordonnées (250, 175). Les deux appels à `g2_pen` changent la couleur du crayon. (Des couleurs sont prédéfinies pour tous les entiers entre 0 et 26, et on peut définir d'autres couleurs².)

Ce programme se compile par la commande :

```
gcc -Wall -W essaig2.c -L/usr/local/lib -lg2 -L/usr/X11R6/lib -lX11 -lm -o essaig2
```

Le `-lg2` sert à charger la bibliothèque `g2` et le `-L/usr/local/lib` indique où la trouver. Cette bibliothèque utilise d'autres bibliothèques : `X11` (pour ouvrir des fenêtres) et `m` (pour les fonctions mathématiques) et il faut également les charger. Le `-L/usr/X11R6/lib` sert à indiquer au compilateur où trouver la bibliothèque `X11`.

1. Montrer que vous avez bien compris ce que fait la variable `h` dans le programme d'exemple en créant un programme qui ouvre deux fenêtres graphiques et qui dessine des choses différentes (des lignes, des rectangles pleins, des cercles vides, etc...) dans chacune des fenêtres.

Le modèle d'Ising

Dans un morceau de métal, les noyaux et les électrons forment de petits moments magnétiques. Tous ces moments sont en général orientés dans des directions aléatoires, et le morceau de métal n'a finalement aucune aimantation globale. Cependant, dans certains matériaux, les moments magnétiques s'orientent spontanément dans la même direction et le morceau de métal possède une aimantation globale. Ce sont les aimants. Si on chauffe un aimant, on constate que l'aimantation diminue jusqu'à disparaître à une certaine température critique dite température de Curie. Il s'agit d'une transition de phase entre une phase aimantée et une phase non-aimantée.

Le modèle d'Ising (1925) est un modèle très simple qui permet de décrire et de comprendre la transition de phase des aimants. On considère des moments magnétiques (ou « spins ») qui ne peuvent prendre que deux valeurs possibles : $+1$ ou -1 . Ces moments sont rangés sur une grille carrée de taille $L \times L$ et l'on s'intéresse à l'*aimantation spécifique* du système, définie comme la moyenne des $N = L^2$ moments magnétiques.

Dans le modèle d'Ising, chaque spin interagit avec ses quatre voisins (celui du haut, celui du bas, celui de gauche et celui de droite. Un spin sur un bord n'a que trois voisins, un spin dans un coin n'a que deux voisins.) À température nulle, la dynamique du système est la suivante :

Pendant un temps « 1 » (dans une unité arbitraire), on répète N fois l'opération suivante : on choisit un spin au hasard et on change sa valeur. Sa nouvelle valeur est égale à la valeur majoritaire parmi ses voisins. (S'il y a égalité, on tire au hasard.)

²Sur la page web http://g2.sourceforge.net/g2_ref/modules.html se trouve la documentation complète de « g2 ». Il faut surtout aller voir la section « graphical output » et, éventuellement, « color manipulation ». Oui, la documentation est écrite en anglais, bien sûr.

2. Faire un programme qui tire une configuration initiale complètement désordonnée (chaque spin a une valeur ± 1 aléatoire) pour le modèle d'Ising et puis l'affiche graphiquement à l'aide de `g2`. On représentera chaque spin par un pixel qui peut prendre deux couleurs différentes selon sa valeur. Comme d'habitude, des fonctions indépendantes se chargeront des différentes tâches.
3. L'affichage est trop lent ! Il est beaucoup plus efficace de dessiner tous les pixels d'un coup plutôt que de les dessiner un par un. Pour cela, on pourra appeler la fonction `g2_set_auto_flush(0)` au début du programme pour désactiver l'affichage automatique puis, à des endroits bien choisis (quand on veut afficher), appeler `g2_flush()` pour que la fenêtre graphique soit effectivement mise à jour. De plus, à chaque fois qu'on change un spin, il ne faut re-afficher que ce dernier.
4. Compléter le programme par une fonction `evolution()` pour simuler Ising à température nulle à partir de cette configuration désordonnée. Le programme doit afficher graphiquement l'état instantané du système ainsi que le temps passé et la magnétisation moyenne au fur et à mesure de son évolution. (L'affichage du temps et de la magnétisation se fera, au choix, dans le terminal, dans un fichier ou (plus dur) directement dans la fenêtre graphique.)

À température non nulle, les choses sont plus compliquées. On utilisera la règle suivante :

Quand on change la valeur d'un spin, on pose n_+ le nombre de voisins valant $+1$ et n_- le nombre de voisins valant -1 . La nouvelle valeur du spin considéré est alors $+1$ avec une probabilité $(1 + \tanh \frac{n_+ - n_-}{T})/2$ et -1 sinon.

(Nota : la température est donnée dans une unité arbitraire.)

5. Vérifier que si l'on met $T = 0$ dans la formule précédente, on retrouve bien la règle à température nulle.
6. Que devient la règle dans la limite $T = \infty$?
7. Modifier le programme pour gérer une température arbitraire. Observer le système pour plusieurs températures (inférieures à 5) et pour deux types de conditions initiales : complètement désordonnée (spins aléatoires) et complètement ordonnée (tous les spins égaux à $+1$).
8. Essayer d'identifier vers quelle température se fait la transition de phase, et décrire en quelques mots chacune des phases.