Ripulitura del modello di Ising in due dimensioni

- Invece di una tabella di spin voglio un array monodimensionale
- uso le seguenti convenzioni
  1. i primi N elementi corrispondono alla prima riga, gli N successivi alla seconda riga, e così via;
  2. per ottenere l'elemento della riga i, colonna j prendo l'elemento $N \cdot (j-1) + i - 1$
- mi libero delle routine poco professionali per la generazione dei numeri pseudo casuali e uso invece *gsl* (vedi esempi con la grafica messi in rete);
- raggruppo in un'unico programma i due algoritmi (Metropolis e Wolff)in modo da poter scegliere senza cambiare programma;
- adatto la grafica ai nuovi programmi (Attenzione!! la grafica rallenta MOLTO i programmi, usare #ifdef)

- configuro la shell per la programmazione in modo che le librerie gsl, gslcblas e plot siano linkate automaticamente

Modello di Potts

Simile al modello di Ising, ma ogni spin può assumere q valori diversi.

$$H = -\sum_{<ij>} \delta_{s_i,s_j} \qquad i\,,j\,\text{primi vicini}$$

Posso anche scrivere

$$H = -\frac{1}{2}\sum_{<ij>} 2\cdot(\delta_{s_i,s_j}-1) - \sum_{<ij>} \delta_{s_i,s_j}$$

- Per $q = 2$ questo è il modello di Ising con β scambiato con 2β, più un termine costante ininfluente
- Si può usare il metodo di Metropolis a temperatura grande;
- per $q = 4$ si può fare una prova, visualizzando anche graficamente il modello;

Bagno termico

- per q grande il metodo di Metropolis diventa molto inefficiente per via delle molte scelte possibili che lasciano invariata l'energia: per $q = 100$ uno spin ha quattro vicini e non più di una volta su 25 l'energia cambia;

- si può usare l'algoritmo del *bagno termico*: scelgo uno spin a caso e prendo un nuovo valore indipendentemente dal vecchio, con probabilità:

$$e^{-\beta E_n} / \sum_k e^{-\beta E_k}$$

  dove E_n è l'energia del nuovo stato scelto e la somma è su tutti i nuovi stati possibili;

- in questo modo spin-flip che corrispondono a piccole variazioni di energia sono meno probabili;

- ogni transizione può dare un semplice spin-flip, come per Metropolis, quindi è ergodico;

- il bilancio dettagliato è ovviamente soddisfatto:

$$\begin{aligned} P(n \to p) &= e^{-\beta E_p} / \sum_k e^{-\beta E_k} \\ P(p \to n) &= e^{-\beta E_n} / \sum_k e^{-\beta E_k} \\ P(n \to p) &/ P(p \to n) = e^{-\beta (E_p - E_n)} \end{aligned}$$

- per calcolare la probabilità basta memorizzare un numero finito di esponenziali.

Algoritmo di Wolff

- Anche qui vicino alla temperatura critica Metropolis funziona male;
- Uso l'algoritmo di Wolff in modo identico al modello di Ising;
- mi ricordo però la corrispondenza

$$2 \cdot \beta_{Ising} \leftrightarrow \beta_{Potts}$$

- la probabilità di aggiungere un elemento al cluster sarà quindi

$$P_{add} = (1 - e^{-\beta})$$