Physics 750: Exercise 18

Thursday, November 30, 2017

The Ising model describes a collection of spins having only two possible orientations $s = \pm 1$, each coupled ferromagnetically to its immediate neighbours. If the system is in contact with a heat reservoir at temperature $T = 1/k\beta$, its partition function is

$$Z = \sum_{\{s_i = \pm 1\}} \exp\left[\beta J \sum_{\langle ij \rangle} s_i s_j\right].$$

Here, $\langle ij \rangle$ indicates that the sum is over all pairs of adjacent sites.

On the square lattice, the Ising model has a phase transtition at a critical temperature $T_c = 2/\log(1 + \sqrt{2})$. Below T_c the spins are ferromagnetically aligned in one of the two magnetization directions. (In the thermodynamic limit, the tunnelling rate between the $s_i = +1$ and $s_i = -1$ states vanishes; this is an example of spontaneous symmetry breaking.) Above T_c the spins are magnetically disordered. Right at the transition, the system has critical magnetic fluctuations on all length scales. The structure of the magnetic domains is analogous to the percolation problem we looked at in Exercise 17.

A simple Monte Carlo simulation can be implemented using the so-called *Metropolis update scheme*, in which a single spin flip is attempted at a randomly chosen site, $s_i := -s_i$. The corresponding change in energy is

$$\Delta E = 2 s_i \sum_j \eta_{ij} s_j,$$

where $\eta_{ij} = 1$ if i and j are adjacent and vanishes otherwise. The move is accepted with probability $P = \min(1, e^{-\beta \Delta E})$.

1. Use the `curl` command to download from the class website everything you'll need for this exercise.

   ```
   $ WEBPATH=http://www.phy.olemiss.edu/~kbeach/
   $ curl $WEBPATH/courses/fall2017/phys750/src/exercise18.tgz -O
   $ tar xzf exercise18.tgz
   $ cd exercise18
   ```

   The `make` command creates an executable `MonteCarlo` and `MonteCarlo_openGL`.

2. Try running the simulation for a 60×60 square lattice.

   ```
   $ ./MonteCarlo_openGL 60
   ```

   (If the graphics are too slow on your machine, you might try a smaller lattice.) The red bar on the right shows the current temperature with respect to the critical T_c at which the ferromagnet disorders. You can adjust the temperature by holding down the mouse button and moving the mouse up or down. Try moving very slowly (adiabatically) through the transition. What happens if you heat up the system and then cool it very quickly (quenching)? Try to understand what's going on.

3. The nongraphical program `MonteCarlo.cpp` repeatedly measures the magnetization M of the system (at intervals of the autocorrelation time), accumulates the moments in bins of size 100, and dumps those values to a file `mag.bins.dat` in three-column format. It will help you to read over the class definition in `moments.hpp` and the function `measure` in `ising.cpp` to see how this works.

   Generate data with temperature sweeps across the transition for several values of the lattice size. Plot your Monte Carlo estimate $\langle M^2 \rangle$ alongside C. N. Yang's exact result

   $$M^2 = \left[\frac{1 + x^2}{(1 - x^2)^2}(1 - 6x^2 + x^4)^{1/2}\right]^{1/2},$$

   which is expressed in terms of $x = e^{-2J/kT}$.

*4. In the nongraphical program `MonteCarlo.cpp`, implement a measurement (setting $k = 1$) of the specific heat and spin susceptibility,

$$C = \frac{\langle E^2 \rangle - \langle E \rangle^2}{T^2} \text{ and } \chi = \frac{\langle M^2 \rangle - \langle M \rangle^2}{T}. \tag{1}$$

Try to find an efficient way to do this, one that doesn't require summing over the entire lattice each time you want to recompute the energy and magnetization. A good strategy is to keep track of the energy and magnetization changes after each successful update.

5. The update scheme as written consists of a single spin flip attempt. Implement a new update scheme that flips two randomly chosen spins at a time. Be careful of the special case when the two sites are adjacent!

6. Implement the analogous three-site update. Is this more efficient? What happens to the acceptance rate?

7. Eliminate "critical slowing down" with the Wolff cluster algorithm described in Phys. Rev. Lett. **62, 361 (1989) (and discussed in class).