

Thermal Physics: Problem Set #9

Interacting Systems: Phase Diagrams and Monte Carlo Methods

Due: Sunday Mar. 29 by 4 pm

Reading Assignment:

- | | |
|---|-------------------------------------|
| for Mon, 5.3 [9 pgs] | (phase behavior of the vdW model) |
| Frenkel-Smit handout [18 pgs] | (Monte Carlo simulations) |
| for Wed, 8.2 [15 pgs] | (Ising model and MC methods) |
| for Fri, Baierlein handout Secs. 1-4 [16 pgs] | (Ising model and mean-field theory) |

Overview: Having studied in some detail both classical and quantum ideal systems we now turn our attention to non-ideal (interacting) systems. We begin by considering phase transitions, something that does not occur in ideal systems. Most pure materials can exist in three phases: solid, liquid, and vapor. At a given T and P the phase which is stable is the one with the lowest Gibbs free energy. The coexistence of two phases (i.e., ice in water) requires an equality of the intensive thermodynamic parameters T , P , μ (and thus the Gibbs free energy). A full and rigorous statistical mechanical description of transitions between different phases remains a major outstanding problem. We will first consider some of the thermodynamic aspects of phase transitions and carry out a detailed analysis of the van der Waals model of a fluid. This simple model displays a liquid-gas phase transition and much can be learned about the thermodynamics of phase transitions from the vdW model. From a statistical mechanics point of view this model is of the "mean-field" type. Remarkably enough, all mean-field models give exactly the same (incorrect!) results in the neighborhood of a critical point. The exact analysis of more realistic models of a liquid is an intractable many-body problem and thus, to make progress, we usually resort to computational methods such as Monte Carlo (MC) simulations.

In an MC simulation one attempts to generate a large number of configurations of a system and compute thermodynamic averages over this sample space. The volume of phase space one can actually explore in a simulation is only a minute fraction of the total phase space (even for the small systems we are able to simulate) so one attempts to concentrate on the "important" regions by weighting states according to a Boltzmann factor. Such "importance sampling" is realized by the Metropolis algorithm which is THE standard method for MC simulations. The Frenkel-Smit handout provides a general introduction to the MC approach with some emphasis on applications to liquid systems while Schroeder's Section 8.2 gives a clear description of the MC method applied to a magnetic system (the Ising model). Since generating "random" numbers is at the core of an MC simulation, in problem A4 we test some standard algorithms for random number generation.

Our first look at an actual Monte Carlo simulation will be to work with some Matlab code for the 2D Ising model (a simple model for ferromagnetism). Although the Ising model may, at first glance, appear to be just another two-state spin system it is actually quite special and historically, has been extremely important in the development of a rigorous theory of phase transitions. What's new here is that the spins in the Ising model interact with each other and thus, in the partition function each spin is coupled to its neighbors. In two and three dimensions this model undergoes a phase transition: the spins line up when the system is cooled below a critical temperature. Note that this is *spontaneous* alignment which occurs in the absence of an external magnetic field, something the non-interacting

system does not do. The Ising model can be solved exactly in one dimension (which we will do next week) and in two dimensions (which Onsager did and which we will not do). It has yet to be solved exactly in three dimensions! Onsager's solution for the 2D Ising model really initiated the modern theory of phase transitions. We will study the Ising phase transition using both Monte Carlo methods and mean-field theory.

Problem Assignment: (6 problems total)

- 5.35** (vapor pressure equation ... Use this equation to predict the boiling temperature of water at the following locations: Hiram, OH, elevation = 382m , $P \approx 0.95$ atm
Mt. Whitney, CA, elevation = 4420m , $P \approx 0.58$ atm
Mt Everest, Nepal, elevation = 8840m , $P \approx 0.34$ atm)

***A1. The Physics of Ice Skating [Andrew]**

Conventional wisdom has it that ice skating is possible due the pressure induced melting of ice by the skate blade which generates a lubricating water layer. (This melting phenomena is a consequence of the volume expansion of water on freezing, which leads to a P - T liquid-solid phase boundary with negative slope ... see Schroeder Fig. 5.11). If the ice is too cold your weight will not produce enough pressure to cause melting. Derive an estimate for this minimum temperature. Use the Clausius-Clapeyron equation and assume that the liquid-solid P - T coexistence curve is linear. The latent heat of melting for ice is 334 J/g and the densities of ice and water are 0.92 g/cm³ and 1.00 g/cm³, respectively. (You will need to make some assumptions about the skater's weight and the area of a skate blade). Comment on the conventional wisdom.

A2. Isotherms, Compressibility, and Spinodal for the vdW Fluid

a) Show that the van der Waals equation of state can be written in the following reduced form:

$$\tilde{P} = \frac{8\tilde{T}}{(3\tilde{v}-1)} - \frac{3}{\tilde{v}^2} \quad \text{where } \tilde{v} = V/V_c, \tilde{P} = P/P_c, \text{ and } \tilde{T} = T/T_c.$$

b) Plot the \tilde{P} vs. \tilde{v} isotherms for $\tilde{T} = 1.5, 1.1, 1.0,$ and 0.9 . (See Schroeder Fig. 5.20)

c) The isothermal compressibility is defined as $\kappa_T = -(1/V)(\partial V/\partial P)_T$. We can define a reduced compressibility as follows $\kappa_T^* = P_c \kappa_T = -(1/\tilde{v})(\partial \tilde{v}/\partial \tilde{P})_T$. Compute κ_T^* for the vdW model and plot κ_T^* vs. \tilde{v} for $\tilde{T} = 5.0, 2.0, 1.5, 1.1,$ and 1.01 .

d) Notice that the subcritical (i.e., $T < T_c$) isotherm from part (b) displays a region with $dP/dV > 0$ (and thus $\kappa_T^* < 0$). Since $\kappa_T^* > 0$ is required for mechanical stability (why?) this region is unphysical. The loci of points separating the unphysical region from the stable region is known as the spinodal and is located by the extremum condition $dP/dV = 0$ (i.e., $\kappa_T^* \rightarrow \infty$). Using the reduced form of the van der Waals equation: determine the vdW spinodal curve (\tilde{T} vs. \tilde{v}) and plot it (\tilde{T} on the y-axis, \tilde{v} on the x-axis) for the temperature range $0.6 < \tilde{T} < 1.0$. Label the critical point and the areas corresponding to the liquid and gas phases.

By the way, the actual existence of a spinodal is the subject of some controversy ... many experimentalist claim to measure this boundary while most theorists scoff, realizing the spinodal concept is simply an artifact of our approximate (so called mean-field) theories.

***A3. Binodal for the vdW fluid [Richard]**

For $T < T_c$ the vdW equation of state displays a liquid-gas phase transition. The thermodynamic boundaries, or binodal, for this transition are determined by the condition: $\mu_\ell(P, T) = \mu_g(P, T)$. Using the so-called Maxwell construction this condition can be expressed as follows:

$$\int_{v_\ell}^{v_g} P(v, T) dv = (v_g - v_\ell) P_{\text{transition}}$$

where $P_{\text{transition}} = P(v_\ell, T) = P(v_g, T)$ is the transition pressure at temperature T , and v_ℓ and v_g are the volumes of coexisting gas and liquid phases, respectively. [Schroeder's Fig. 5.21 gives a demonstration of this equality]. By combining this result with the condition $P(v_g, T) = P(v_\ell, T)$ one has two equations for the two unknowns, v_g and v_ℓ , at a fixed T . Use the reduced vdW equation from A2 (dropping the twiddles) to set up these two nonlinear equations. Solve the equations numerically (e.g., use the Maple `fsolve()` routine) for enough T values to allow you to construct a T - V phase diagram. (It's useful to plot the spinodal and binodal on the same figure).

A4. Pseudo-Random Number Generators

Random (or pseudo-random) number generators are at the heart of all Monte Carlo simulations. The most common rng's in use today are "modulo generators" based the following recursion relation:

$$x_i = (a x_{i-1} + c)_{\text{mod } m}$$

where the sequence of numbers $\{x_i/m\}$ are (hopefully) uniformly and "randomly" distributed across the interval $[0, 1)$. The success of this type of generator depends crucially on the choice of the integer constants a , c , and m (and unfortunately, these generators have a very bad reputation due to some bad choices once upon a time by some folks at IBM!). The Matlab program `modulo_rng.m` implements a modulo random number generator.

- a) Run the `modulo_rng.m` code for a few "good" (a, c, m) -generators and compare with the infamous $(65539, 0, 2^{31})$ IBM generator. How "random" do the numbers look?

One simple test of a rng is to see how uniformly a k -dimensional space is filled by k -tuples of random numbers (i.e., if we take our random numbers two at a time and plot these as x - y coordinates we should fill the unit square "uniformly").

- b) Modify the `modulo_rng.m` code to perform the two-dimensional test on the (a, c, m) -generators used in part (a). How "randomly" is the 2-d space filled?
- c) Modify the `modulo_rng.m` code to perform the three-dimensional test on the (a, c, m) -generators used in part (a). Use the Matlab function `plot3()`. In the plot display window you can use the rotate tool to rotate your 3-d plot and look for any regularities
- d) Write Matlab code to do parts a-c using Matlab's built-in random number generator `rand()`.

Typical code at start of program to initialize `rand()`: `rand('state', sum(100*clock));`

Code to get next random number: `val = rand;`

A5. MC Simulation of the 2D Ising Model - Introduction

The two Matlab programs `ising1.m` and `ising2.m` are provided to give you a feel for the 2D Ising model. These programs are neither fast nor efficient but they do work. Both programs carry out a simple Monte Carlo simulation of the 2D Ising model in zero magnetic field.

Ising1.m is meant to give you a qualitative look at the running simulation and it displays the average magnetization per spin ($\langle M \rangle / N\mu = (1/N) \sum_i s_i$) vs MC cycle at the end. This final plot should give you an idea of how many MC cycles are required to equilibrate the system (which is an important issue here for $T < T_c$). (Note: one MC cycle equals N attempted spin flips). Run this program for $T^* = kT/J = 1.5, 2.0, 2.5, 3.0, 5.0$ and print out the plots of $\langle M \rangle$ vs MC cycle.

Ising2.m runs the same algorithm, but without the configuration snapshots and for a series of temperatures. It computes the average magnetization at each temperature and plots these results. You will need to set the starting temperature, final temperature, and temperature step size (and you may want to adjust the number of equilibration cycles). The critical temperature for this system is $T_c^* \approx 2.3$ (Baierlein, Table 16.3, 2-dim square lattice) so you should generate results in a range which brackets this temperature (say $T^* = 1.5 \rightarrow 3.0$). For each temperature the numerical value of $\langle M \rangle / N\mu$ is printed to the "console" window. Make a plot of $|\langle M \rangle / N\mu|$ vs T^* which includes both your simulation results and the following exact result due to Onsager:

$$|\langle M \rangle / N\mu| = [1 - \{\sinh(2J/kT)\}^{-4}]^{1/8} \quad T \leq T_c$$

$$= 0 \quad T > T_c \quad [\text{exact } T_c \text{ is given by } \sinh(2J/kT_c) = 1].$$