OTHER CLASSICAL MODELS

Models with continuous variables

In classical models, going from discrete to continuous variables in MC schemes introduces minimal modifications. If the problem is to evaluate an integral

$$< A >= Z^{-1} \int \ldots \int dx_1 dx_2 \ldots dx_M A(\{x_i\}) W(\{x_i\})$$

$$Z = \int \ldots \int dx_1 dx_2 \ldots dx_M W(\{x_i\})$$

we simply define the configuration as a collection of real numbers, $\nu = \{x_1, x_2, \ldots, x_M\}$ and proceed in exactly the same manner as before. Although the configuration weight now involves differential measures

$$\text{weight} = W(\{x_i\}) dx_1 dx_2 \ldots dx_M \sim (dx)^M$$

they all cancel in the weight ratios in the balance Equation

$$\frac{(\text{weight})_{\nu_2}}{(\text{weight})_{\nu_1}} = \frac{W(\nu_2)}{W(\nu_1)}$$

In the simplest updating strategy, after choosing at random a variable to modify, e.g., x_I, we may suggest another value for this variable with equal probability density anywhere in the domain of definition of this variable. For example, if $x_I \in (x_I^{min}, x_I^{max})$, then

$$x_I' = x_I^{min} + r \left(x_I^{max} - x_I^{min} \right) , \qquad (*)$$

In other words, we seed the new value of the variable with the normalized probability density

$$u(x_I') = \frac{1}{x_I^{max} - x_I^{min}} = const .$$

Note that $U(x_I') = u(x_I') \, dx_I'$ is the probability to apply an update $U_{\to}(x_I \to x_I')$ and it has to be used in the balance equation! In the example considered

above it does not depend on the initial value of x_I, and thus is the same for the inverse update from x'_I to x_I. The balance Eq. for this update is then

$$\frac{P^{acc}_{\rightarrow}(x_I \rightarrow x'_I)}{P^{acc}_{\leftarrow}(x'_I \rightarrow x_I)} = \frac{W(\nu\,')(dx)^N}{W(\nu)(dx)^N} \frac{N}{N} \frac{u(x_I)dx}{u(x'_I)dx} = R$$

We see that all differential measures cancel each other and

$$R = \frac{W(\nu\,')}{W(\nu)} \frac{u(x_I)}{u(x'_I)} \underset{\text{for } u(x)=const}{\Longrightarrow} \frac{W(\nu\,')}{W(\nu)}$$

We have considered the simplest updating scheme. Depending on the model, one may develop more sophisticated algorithms; the only necessary requirements are ergodicity and the balance equation

At this point it is worth recalling the "heat bath" trick to enhance the acceptance ratio which we discussed for the Ising model. Formally, the procedure of selecting the new value for x_I is flexible, and any normalized distribution function $u(x)$ can be used. In the random number generator Sec. we discussed techniques of how one may generate random variables with arbitrary distributions, so it is not a problem at all to use $u(x)$ which makes R larger; in certain cases it is even possible to make $R = 1$ independent of x_I and x'_I !, i.e., every update is definitely accepted! One more reason to use the distribution $u(x) \neq const$ is the ability to handle cases where $x^{max}_I - x^{min}_I \rightarrow \infty$. Then, one simply *has to* use some convergent function $u(x)$.

As an illustration consider the following model (never mind it can be solved exactly)

$$E = J \sum_{<ij>} (x_i - x_j)^2 \; ; \qquad x_i \in (-\infty, +\infty)$$

Suppose we have to update the variable x_I. Evaluate first

$$a = z^{-1} \sum_{<Ij>} x_j \; , \quad \text{where} \quad z = \text{the number of n.n.}$$

The energy difference between the new and current configurations can be written as

$$\Delta E = Jz\left[x'_I - a\right]^2 - Jz\left[x_I - a\right]^2$$

and the ratio of the configuration weights as

$$\frac{W(\nu\,')}{W(\nu)} = \frac{e^{-\beta Jz[x'_I - a]^2}}{e^{-\beta Jz[x_I - a]^2}}$$

2

If we seed the new variable using the Gaussian probability density (as we have seen the extra CPU time for doing this is minimal)

$$u(y) = \sqrt{\frac{\beta Jz}{\pi}} e^{-\beta Jz(y-a)^2}$$

the acceptance ratio will become unity!

The idea seems clear here - try to seed a new variable as close as possible to the probability density set by the current *local* "environment" (in our example a is defined by the values of the neighboring variables). Of course, you should not do it "at all costs", since what counts at the end is the best product t_o(in updates) $\times$ CPU time per update. Still, almost any $u(x)$ which qualifies for the transformation method is worth considering.

An important continuous classical model, the so-called **XY-model**, is a straightforward generalization of the Ising model

$$E = -J \sum_{<ij>} \vec{n}_i \vec{n}_j \ . \tag{1}$$

Instead of discrete variable σ_i we now have two-dimensional unit vectors $\vec{n}_i = (\cos\theta_i, \sin\theta_i)$. We may actually use angles θ to write XY-model as

$$E = -J \sum_{<ij>} \cos(\theta_i - \theta_j) \ . \tag{2}$$

The difference in physics between the Ising and XY-models is (i) minimal, because they have similar high-T (spins are disordered and uncorrelated) and low-T (spins tend to be parallel and correlated) phases, and (ii) enormous, because phase transitions (and even transition scenarios) between nigh-T and low-T phases are different as well as the spectrum of low-energy excitations. For example, the Ising model is a prototype for the liquid-vapor transition (see also below), and the XY-model is a prototype for the superfluid-normal fluid transition. This correspondence derives from the theory of critical phenomena which states that the only "important features" in continuous phase transitions are
- the dimension of space
- whether interactions are short- or long-range , and, finally,
- number of components of the order parameter.

Ising spin order is characterized by a single number—spontaneous magnetization— which is the difference between the number of up and down spins.. Similarly,

3

the difference between the liquid and vapor phases is characterized by the volume per particle difference. Order among XY-spins is characterized by the magnetization vector $\vec{M} = \sum_i \vec{n}_i$ which has two components, modulus and direction in the XY-plane. Similarly, the superfluid order is characterized by the complex order parameter which is no different from the two-dimensional vector. Systems having the same "important features" have continuous phase transition of the same kind, i.e. the scenario of the transition, important fluctuations, large-scale behavior of correlation functions, temperature or parameter dependence of various quantities in the vicinity of the critical point—all are universal=model/system independent.

One may go further and consider the so-called O(N)-models by making $\vec{n}_i$ in Eq. (1) unit vectors in N-dimensions. What concerns us now, however, is not what these models represent, but how we can simulate them using MC techniques. The simplest approach would be to select at random some site, and to propose $\vec{n}_i'$ by generating new values for N-dimensional solid angles using appropriate distribution functions.

Problem. Figure out the single-spin heat bath algorithm for the O(3)-model (where $\vec{n}_i$ are familiar three-dimensional unit vectors) with the "perfect" acceptance ratio $R = 1$. By algorithm I mean an explicit set of instructions/equations for performing an update. [You may explain in words the part related to rotating the coordinate system.]

Potts models

Potts models are another generalization of the Ising model, now in the direction of extending the list of possible values for discrete lattice variables σ_i. It reads

$$E = -J \sum_{<ij>} \delta_{\sigma_i,\sigma_j} , \quad \sigma_i = 1, 2, \ldots, q . \tag{3}$$

The number of allowed values q is an important parameter of the q-state Potts model. For $q = 2$ it is identical to the Ising model with $J_{Ising} = J_{Potts}/2$. Indeed, if we count energy from the fully polarized state $E_0 = E(\sigma_i = const)$, then for every pair of neighbors with different spin values we pay an extra energy J_{Potts}. In the Ising model we pay energy $2J_{Ising}$ for antiparallel spins.

In the Potts model spins are also correlated (would like to have the same value) at low temperature. The transition temperature to the spontaneously

4

ordered state is decreasing with increasing q because there are more and more ways of making disordered states and thus the entropic contribution, $-TS$, decreases the free energy of the disordered state.

It takes seconds to adapt the single spin flip or heat bath algorithm for the Ising model to perform simulations of the Pott model. The only thing to note here is that for large q the heat bath algorithm is more efficient since chances are getting smaller to suggest the same value of the spin variable (see discussion in the previous Sect.).

Problem. Heat bath algorithm. If n_s, $s = 1, 2, \ldots, q$ is the number of n.n. spins with $\sigma = s$ then propose a new value σ'_k with the probability distribution $p_{sigma} =$

Glasses. First-order phase transitions.

Glasses are one of the toughest topics in all of physics. Formally, glasses represent systems with very strong degree of disorder and frustration such that there is no easy way to guess what the lowest free-energy state looks like, or even how many macroscopically distinguishable states may be realized at a given temperature. In fact, the appearance of many minima with comparable free-energies which are separated by macroscopically large barriers is considered to be the origin of the glass transition at T_g. Experimentally, the glass transition is observed as freezing of the system dynamics; the empirical law used in many cases is $\tau \sim e^{\tau_o T_g/(T_g - T)}$, which says that the relaxation time for some observed quantity (stress, magnetization, density, etc.) diverges at T_g. Below T_g the system response shows hysteresis, memory effects, and other features expected when the relaxation time is much longer than the experimental time scale.

The picture of multiple minima with large barriers separating them (the time required to cross the barrier is an exponential function of the barrier height, F_B, and diverges fast in the thermodynamic limit if the free-energy barrier scales with the system size, $F_B \sim L^{\alpha > 0}$) is consistent with the experimental observations.

configuration space
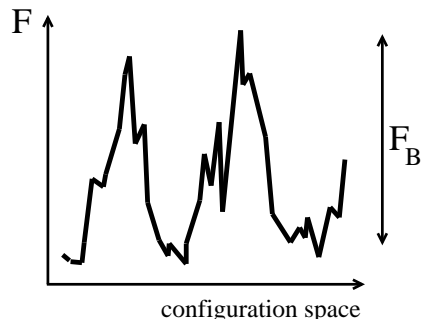
The simplest glass model is obtained by making coupling between spins ran-

dom

$$H/T = -\sum_{<ij>} K_{ij}\sigma_i\sigma_j \ . \tag{4}$$

The distribution of K_{ij} can be discrete with $K_{ij} = \pm K$ randomly on each bond, or continuous, e.g. with the Gaussian distribution $P(K) \sim e^{-K^2/2\sigma^2}$. This type of spin glass models are called Edwards-Anderson glasses.

Problem. Suppose that random couplings between spins are obtained from $K_{ij} = \xi_i\xi_j$ where $\xi_i = \pm 1$ on each site (Mattis, 1976). Show that the Mattis "glass" is not glass at all, and in fact its properties are identical to the regular Ising model.

The other model of interest is the random-field Ising model

$$H/T = -K\sum_{<ij>} \sigma_i\sigma_j + \sum_i h_i\sigma_i \ , \tag{5}$$

with the second term describing random local fields acting on spins. Again the distribution of local fields may be discrete or continuous with the zero mean. Strictly speaking, the random field Ising model is not a glass model, but simulating it is as hard as a true glass because its free-energy landscape also has large barriers. In the paramagnetic state $T > T_c$ of the regular Ising model spins are oriented randomly and typical fluctuations of the total magnetization are of order $N^{1/2}$. Close to T_c spins become correlated over a large correlation length ξ and form a complex structure of large fractal domains which, are free to evolve and change their shapes. When random fields are present, the domain structures are pinned to the lattice, and transforming them requires overcoming large energy barriers which scale with the correlation length.

Simulations of equilibrium properties of glasses are notoriously difficult for exactly the same reasons why experimental relaxation times diverge—large free-energy barriers. One may argue that simulating equilibrium properties of glasses is a useless exercise, since they are never observed in real life. True, but then there are also theories of glassy states which make predictions of equilibrium properties. In the long run, physics is about distilling our knowledge of Nature into the smallest number of basic concepts, ideas and theories. So far, there is no good method for simulating glasses. However, there are techniques designed specifically for problems with large energy barriers

6

which perform much better then the simple Metropolis spin-flip algorithm (next Sect.)

A related issue is how to simulate efficiently systems with first-order phase transitions. First-order phase transition, say between phases A and B, is discontinuous, which means that energies of phases A and B at the transition point satisfy $E_A(T_c) - E_B(T_c) \sim N$. In the equilibrium state of A/B the probability of having system energy E is given by the Gaussian (see previous Section) $P(E) \sim \exp\{-(E - E_{A/B})^2 \beta^2 / 2C_{A/B}\}$, where $C_{A/B}$ is the specific heat of the corresponding phase. When $|E_A - E_B|\beta \sim N \gg \sqrt{C_A + C_B} \sim \sqrt{N}$, the overlap between the energy distributions of phases A and B is so small that in a realistic MC simulation which generates configurations with probabilities proportional to their Gibbs weights one will never see both A and B at T_c. The simulation will simply stuck in one of the phases.

Moreover, even when T is slightly shifted to the B=phase side but the initial configuration is of the A-type it is still possible that it will never change to B-type unless $|T - T_c|$ is large enough. It is typical to see hysteresis in results when the transition is first-order.
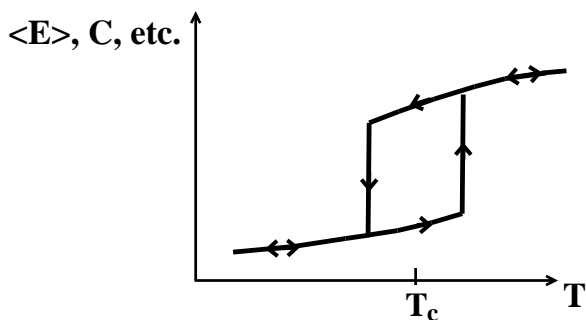
The size of the hysteresis loop is determined by the probability of the overlap between the two energy distributions times the simulation time.

Conserved order-parameter models

This is yet another version of the Ising model. The Hamiltonian itself is exactly the same, i.e. n.n. coupling between spins, but there is also an explicit condition that the total magnetization is conserved $\sum_i \sigma_i = M_0 = const.$ This model is interesting for two reasons (i) it describes the classical system of interacting particles on a lattice when the number of particles is fixed, and (ii) it models Ising spins in the presence of external magnetic field at high enough temperature, e.g. $T > T_c$, provided $M_{Ising}(H, T) = M_0$ since in the thermodynamic limit it really does not matter whether M is fluctuating a little or not.

7

The mapping to the lattice model of interacting particles is done by substituting $\sigma_i = 2n_i - 1$ where $n_i = 0, 1$, to the Ising model

$$H/T = -K \sum_{<ij>} (2n_i - 1)(2n_j - 1) = -\epsilon \sum_{<ij>} n_i n_j + \epsilon d \sum_i n_i - dN\epsilon/4 , \quad (6)$$

where $\epsilon = 4K$. The conserved magnetization condition transforms into the conserved particle number constraint $\sum_i n_i = N_p = (N + M_0)/2 = const.$ Given this condition, the second and third terms in the Hamiltonian (6) are constant $\epsilon d(N_p - N/4)$ and may be safely omitted. In this model particle interactions include the hard-core repulsion potential which does not allow to have more than one particle per site, and n.n. attraction ϵ which favors larger particle densities. In fact, this model, often called the conserved order parameter (COP) Ising model, is a pretty accurate representation of binary alloys.

As mentioned already, properties of the COP model are closely related to properties of the Ising model. Since states with $M_0 > 0$ and $M_0 < 0$ have identical properties we restrict ourselves to the case of positive M_0.

As long as M_0 is larger then the spontaneous magnetization of the conventional Ising model $M(H = 0, T) \equiv M_s$ shown by the solid line, the system state is homogeneous and in the thermodynamic limit corresponds to some state of the Ising model in an external magnetic field. If $M_0 < M_s$, the system state is phase separated.
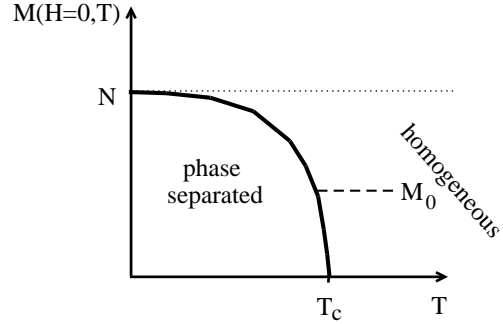
The reason for phase separation is the competition between the bulk and domain wall energies. If M is not conserved, then below T_c the best thermodynamic state with the lowest bulk free-energy corresponds to the state with spontaneous magnetization M_s. If $M_0 = M_s$ then the system is "happy" and homogeneous, but if $M_0 < M_s$ the best bulk energy is obtained by splitting the system into two parts $N_1 + N_2 = N$ such that in each part the magnetization density is exactly M_s/N but opposite in direction. We have then $M_1 + M_2 = M_s(N_1 - N_2)/N = M_0$ which defines

$$N_1/N = M_0/2M_s + 1/2 ; \quad N_2/N = M_0/2M_s - 1/2 .$$

Of course, there is an energy penalty for creating a surface between the up- and down-oriented spins (small and high density regions of the lattice gas) but this is only a tiny correction to the total free-energy with is dominated by the bulk in the limit of infinite system size. The study of surface tension (free-energy per unit area) is the most interesting application of the COP Ising model.

It is easy to suggest an algorithm for MC simulations of the COP Ising model. Obviously, the single spin-flip scheme is not applicable because it does not conserve M. The minimal effort should involve two spins, and if the selected pair has $\sigma_1 + \sigma_2 = 0$, i.e. spins are pointing in the opposite directions, we may suggest to flip them $\sigma_1 \to -\sigma_1$ and $\sigma_2 \to -\sigma_2$. Spins σ_1 and σ_2 can be selected at random anywhere in the lattice; if we select the same spin twice, or if $\sigma_1 + \sigma_2 \neq 0$, we simply try to select the pair again. The acceptance ratio is then simply the ratio of system Gibbs factors

$$R = \frac{P_{acc}^{\nu \to \nu'}}{P_{acc}^{\nu' \to \nu}} = e^{-\beta(E_{\nu'} - E_\nu)} \ .$$

Most of the time spins in the pair are far away from each other and R is nothing but the square of the acceptance ratio for the single spin-flip algorithm.

Polymers.

Formally, polymers are long molecules consisting of many connected monomers (monomers are small, compact, elementary units consisting of several atoms). In what follows we will consider polymers having only one type of monomers. Polymer models have different levels of complexity. More realistic ones consider spheres of radius R connected by elastic rods which are characterized by the corresponding spring and bending constants. In addition, there is also an interaction potential between the spheres which has short-range repulsive and long-range (e.g. Coulomb) components. If one is interested in large scale equilibrium properties, especially at the transition between the dilute and globule (dense) phases, then more simple lattice polymer models

are used.

The natural length-scale in lattice polymer models is the so-called persistence length, l; this is the length over which the directions between the n.n monomers are correlated. At distances smaller than the persistence length the polymer is basically a rigid elastic rod. In lattice models l is a natural unit of length equal to the lattice constant.
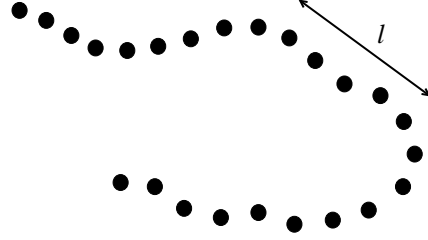
Consider a three dimensional simple cubic lattice of sites. The lattice polymer of length L is defined as an ordered list of lattice points $\mathbf{R}_1, \mathbf{R}_2, \ldots, \mathbf{R}_L$ such that $\mathbf{R}_{i+1}$ is the n.n. site of $\mathbf{R}_i$.

Basically, it is a continuous oriented line of length L. In realistic models compact monomers repel each other at short distances and the polymer line does not intersect itself. Since persistence length is typically much larger then the distance between monomers, it is less obvious whether self intersections should be allowed or prohibited in lattice models.
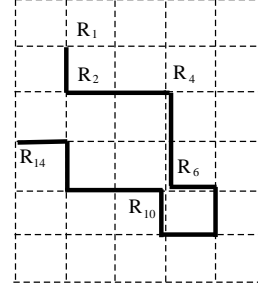
Both cases may be considered, and site list ordering is important to specify the polymer topology and to avoid ambiguous reading of the graph.

We are ready to write the lattice polymer model as

$$H/T = \sum_{(i \neq j)=1}^{L} V(\mathbf{R}_i - \mathbf{R}_j) \,. \tag{7}$$

Its crucial features are the ordered list of n.n. (a chain of sites) and nonlocal (along the chain index i) interaction potential $V(\mathbf{R})$. Note, that there is no energy penalty for changing the chain direction locally, i.e. it is free to bend. For $V(\mathbf{R}) = [V \to \infty]\delta_{\mathbf{R}=0}$ we obtain the so-called **self-avoided walk** model, which is identical to the conventional random walk plus the constraint of no self-intersection = all sites in the list $\{\mathbf{R_i}\}$ are different. Of course, one may reintroduce the notion of the persistence length by adding

10

energy terms equivalent to $\kappa \sum_{i=2}^{L-1} |2\mathbf{R}_i - \mathbf{R}_{i-1} - \mathbf{R}_{i+1}|$ which are non-zero if vectors $\mathbf{R}_i - \mathbf{R}_{i-1}$ and $\mathbf{R}_{i+1} - \mathbf{R}_i$ are different and the chain direction changes at site $\mathbf{R}_i$

In simulating polymers one is interested in such issues as the gyration radius $R_g^2 = \langle (\mathbf{R}_1 - \mathbf{R}_L)^2 \rangle$ and the end-points correlation function $P(R = |\mathbf{R}_1 - \mathbf{R}_L|)$ as a function of interaction potential and/or temperature. If V has and attractive, e.g. Van der Waals type, component than at some point the polymer undergoes a continuous transition to the globule state with much higher density. Finally, in the limit of $T \to 0$ polymers may form crystalline states.

So, what would be a simple MC way to simulate polymers?

Well, one may select at random any monomer $\mathbf{R}_I$ and suggest to shift it along the diagonal of the plaquette formed by $\mathbf{R}_i - \mathbf{R}_{i-1}$ and $\mathbf{R}_{i+1} - \mathbf{R}_i$ vectors (if vectors are parallel the update is not possible).
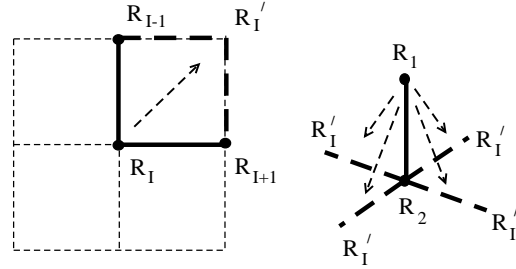

If $\mathbf{R}_I$ is the end point then it can be shifted to any of the points shown in the figure, i.e. along the diagonal of the plaquette formed by $\mathbf{R}_{I+1} - \mathbf{R}_I$ and one of the perpendicular lattice vectors. The update conserves chain ordering and its length, and its acceptance ratio is the ratio of Gibbs factors for the final and initial states. It is probably not the most efficient way of doing things, for example, if the initial state is a straight line, then only end-point updates will work at the beginning (but we address end-points only with probability 2/L). Of course, after the end-points have moved away from the straight line we may update their neighbors, etc. and the polymer will eventually evolve into a complex line shape.

In the **snake algorithm** one suggests to translate the polymer chain along itself by one monomer, i.e. the polymer "head" is shifted to the n.n. lattice point selected at random, $R_1 \to R_1'$, and all other sites follow it $R_{i>1}' = R_{i-1}$, or, going backwards, the "tail" is shifted to the n.n. lattice point selected at random, $R_L \to R_L'$, and all other sites follow it $R_{i<L}' = R_{i+1}$. The detailed balance equation for this simple algorithm involves only the ratio of configuration weights. This algorithm is not efficient since each move involves

at least $\sim L$ operations (we will see soon that a better handling of the data make the snake update entirely local), but changes the shape of the polymer only a little bit. However, its worm-algorithm version works better.

In the **pivot algorithm** one selects at random the monomer $\mathbf{R}_I$ and uses it as a pivot point for rotating either part of the polymer attached to $\mathbf{R}_I$. Let we select to rotate the higher index part. Then we choose $\mathbf{R}_{I+1}$ to be any n.n. site (assuming, for generality, that overlaps and selfintersections are allowed) but preserve the relative order of all sites with $i > I + 1$. The pivot algorithm is very efficient in the dilute phase but fails to work in the globule phase. The reason is clear—it is macroscopic in nature because most of the time we rotate large $\sim L$ parts of the system and create multiple overlaps between the monomers which are not favored by the potential energy, thus, small acceptance ratio.

Even in dilute phase, the number of overlaps scales roughly as $L^{0.6}$ according to some tests and theoretical estimates (Sokal). This scaling is not too bad given the possibility of radically changing the large-scale structure of the polymer in one update. Long range interactions further reduce the algorithm efficiency because its acceptance ratio involves macroscopically large Gibbs factors.
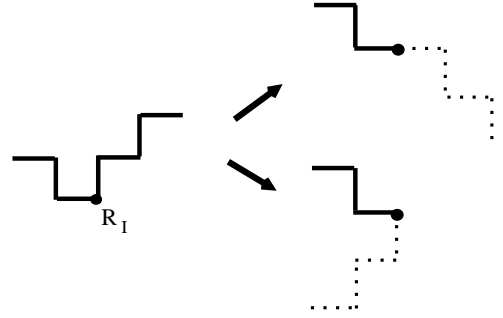
Kinetic Equations.

Formally kinetic equations are a set of differential Eqs. for the distribution function $f_i(t)$

$$\frac{df_i(t)}{dt} = \mathcal{F}[\{f_i\}] \,, \tag{8}$$

where $\mathcal{F}$ depends, in principle, on all f_i. The "i" index may represent particle coordinates, momenta, spins, energy levels, etc. If $\mathcal{F}$ is linear in f_i and the number of different i values is relatively small, then one may use linear algebra methods to solve (8). Otherwise, MC simulations may turn out to be more efficient.

12

I will consider here only the simplest case of linear $\mathcal{F}$:

$$\frac{df_i(t)}{dt} = \sum_j R_{ij}\, f_j \ . \tag{9}$$

Furthermore, I will assume that Eq. (9) describes an ensemble of particles which occupy states indexed by "i" and undergo transitions between them with rates R_{ij}. Since the number of particles is conserved (or the distribution function is normalized) we have

$$d\left[\sum_i f_i(t)\right]/dt \ = \ 0 \ = \ \sum_{ij} R_{ij}\, f_j$$

which for arbitrary distribution f_j requires that

$$R_{jj} \ = \ -\sum_{i \neq j} R_{ij} \ . \tag{10}$$

Now we switch the point of view from ensemble to single particle one, and say that if one particle is distributed initially over states "i" with the probability distribution $f_i(t = 0)$ and then evolves in time by changing it's current index from i to i' in time Δt with probability $\sim R_{ii'}\Delta t$, then an average over all possible *random* single-particle trajectories $i(t)$ is identical to solving Eq. (9). More precisely, if for each such trajectory we record it in the histogram of visited sites, $h_i(t_\alpha) \equiv h_{i\alpha}$, where index α refers to the time grid, $t_\alpha = \alpha \cdot \Delta t$, then

$$f_i(t_\alpha) = \frac{h_i(t_\alpha)}{\sum_{trajectories}} \ ; \quad \text{in the limit} \quad \Delta t \to 0 \ . \tag{11}$$

This correspondence is possible because Eq. (9) is linear and it does not matter whether we consider $N \to \infty$ independent particles and see their distribution evolving as they move around, or sum over independent single-particle trajectories.

The advantage of the second point of view is that it is ready for the MC simulation immediately. To begin with, initiate the histogram, $h_{i\alpha} = 0$, the time, $\alpha = 0$, and trajectory, $N = 1$, counters. Then do the following:

1. Select the initial value of i with the probability distribution $f_i(0)$.

13

2. Update $h_{i\alpha} = h_{i\alpha} + 1$.

3. If $rndm() < \Delta t \sum_{j \neq i} R_{ji} \equiv \Delta t |R_{ii}|$, change the value of i by selecting the new value according to the normalized distribution $P_{j \neq i} = R_{ji}/|R_{ii}|$.

4. Increase $\alpha = \alpha + 1$ and go to point 2, unless α has already reached the maximum value α_{max}; in this case reset $\alpha = 0$, increase $N = N + 1$, and go to point 1.

Sounds easy. The only tricky point is the value of Δt. Formally, it is arbitrarily small. In practice, it has to be finite to have non-vanishing probabilities of changing the state and being able to reach interesting time scales $t_{max} = \alpha_{max}\Delta t$. Large Δt are introducing systematic errors into the simulation which scale as Δt^2. Indeed, we ignore the possibility that the particle will change its state several times (most probably twice) during the time interval Δt. It seems reasonable to have $\Delta t R \sim 10^{-2}$ or similar, where R is the typical rate among R_{ii}. Unfortunately, in some kinetic problems there is no well defined notion of typical rates R_{ii} because they vary orders of magnitude depending on i. In such cases one may consider doing multiscale simulations with adapting time grids, but I will not touch this subject.

At the beginning of this course we discussed random walks and mentioned that they reproduce diffusion at large scales, In the previous version, the random walker was living on a time grid $t_\alpha = \alpha\tau$, and was forced to shift one step in space between temporal points. A better lattice version is given by the kinetic Equation

$$\frac{df_i}{dt} = -\frac{1}{\tau} \sum_{\pm\mu} \left(f_i - f_{i\pm l\mu} \right) . \tag{12}$$

When f_i is a smooth function of i we may reduce (12) to the conventional diffusion Equation by expanding small differences

$$f_{i+l\mu} + f_{i-l\mu} - 2f_i \approx l^2 \frac{d^2 f_i}{di_\mu^2} .$$

Then, Eq. (12) becomes (denoting $i \equiv \mathbf{r}$)

$$\frac{df(\mathbf{r}, t)}{dt} = -D \, \Delta f ; \quad D = l^2/\tau . \tag{13}$$

14

Problem. Doing MC simulation of model (12) in three dimensions $(d = 3)$ is the homework. Note, that selecting a new value of i if $rndm() < \Delta t\, 2d/\tau$ is easy—just shift it one step in any direction selected at random. Stop your simulation at $t_{max}/\tau = 100$. For initial distribution use $f_i = \delta_{i,0}$. At the end of the simulation compare your MC result with the Green function of the continuous diffusion Eq. (13)

$$G = \left(\frac{1}{4\pi Dt}\right)^{d/2} e^{-r^2/4Dt} \ .$$

There are models, models and more models. The Nature is complex in appearance and behavior and simple models may only provide a limiting view on some of its properties. We have to stop now ...

15