Course notes, CS664, 10/5/04

METROPOLIS AND BOLTZMANN, REDUX

- If your Markov Chain is ergodic, and you can generate a state vector where taking a single step results in the same state vector, you have found the stationary distribution. So we need to verify $K\pi = \pi$.

- We will make the simplifying assumption that at the stationary distribution the total amount of water shipped between any pair of states (i.e., the percentage of a very large ensemble) is the same.

- This is called the *detailed balance* constraint, and is sufficient but not necessary for the stationary distribution. (You might want to think about why it is sufficient, and why it's not necessary).

- Think about the Markov Chain corresponding to Metropolis. Consider two stats R, S, and wlog assume $E(R) > E(S)$. If we are in the Boltzmann distribution, $P(R) = C \exp(-E(R)/T)$ where C is some constant (which will turn out to be important). Similarly, $P(S) = C \exp(-E(S)/T)$.

- What is the probability that we move to state S if we start in state R by one step of Metropolis? This is a downhill move, so it is:

$$P(sample)P_r^{old} = P(sample) \cdot C \exp(-E(R)/T)$$

where $P(sample)$ is the chance we generate a given move in our sampling step, which we assume to be constant.

- What is the probability that we move to state R if we start in state S? This is an uphill move, so it is:

$$P(sample)P_s^{old} \exp(E(S) - E(R)/T) =$$

$$P(sample) \cdot C \exp(-E(S)/T) \exp(E(S) - E(R)/T)$$

- These are the same!

- What about the fact that the minimum energy is not necessarily zero?? Note that the probability of all states must sum to 1. This is what C is for!

- Also note that the acceptance probability is not normalized! Fortunately, it is bounded above by 1.

FROM METROPOLIS TO ANNEALING

- This suggests a great algorithm for minimizing an arbitrary energy function. Set T to be almost zero, and run Metropolis. Then you're done.

- What is wrong with this picture? Can it work even if P=NP?

- There are no real bounds on how fast Metropolis converges.

- A bad example: a well in the desert (or Utah). It can take arbitrarily long to ever find the place where you need to be most of them time.

- However, in general, Metropolis converges faster at higher temperature and slower at lower temperature. This is a primarily empirical fact (but note that at high T, we are approaching a uniform distribution).

- Also note that the general area of convergence proofs for annealing-style algorithms is quite hard, and there are many open problems. (Lots of smart people have made little progress...)

- The key idea of *simulated annealing* is to start at a high temperature, and slowly lower the temperature.

- At a given temperature, we run Metropolis for "a while".

- The starting point the next cycle is the end point the last cycle (this makes simulated annealing the simplest example of what is called a *continuation method*).

- There are some interesting facts about annealing. For example, you can prove that if you lower the temperature slowly enough, you will find the global minimum with arbitrarily high probability.

- These "optimal annealing schedules" are totally useless in practice. For example, you might have

$$T(t) \geq \frac{Nc}{\log t}$$

where c is a constant and N the number of pixels.

- To divide the temperature at the 100th iteration by 2, wait to 10,000th iteration. Unfortunately, you start at a really high temperature.

- This is actually a lot worse than exhaustive search, partly because it really requires both an inner loop (Metropolis) and an outer loop (annealing), each of which has to run forever to achieve optimality.

- There was also a grand early success with TSP, but for subtle reasons.

- Why is annealing so poorly suited to vision? Tons of local minima, large number of pixels, slow sampling (latter can be fixed).

MARKOV CHAIN MONTE CARLO METHODS

- You can use Markov Chains in lots of creative ways. We've just seen how to use them for optimization. But sometimes you don't want a single "best" answer (although it is usually what an application of vision is looking for).

- Classical example is hand tracking. You might want to hold onto several competing hypotheses, at least for a while, in order to disambiguate.

- You might want to generate a few solutions, with probability proportional to how good they are. Consider a bimodal distribution, for example. These approaches are called *sampling* (sometimes known as *particle methods*).

- You know that the Metropolis algorithm defines a Markov Chain, and that running it gives you outputs distributed according to Boltmann.

- Now you have some arbitrary distribution. How do we get outputs distributed according to this?

- Well, perhaps we can build an ergodic Markov Chain where this is our stationary distribution!

EXACT METHODS

- Ising model: exact solution is possible (amazing fact!)

- Potts Model is NP-hard, even on a grid