Modern Monte Carlo

Problems

• we are interested in properties at room temperature
• at room temperature, processes are slow
  • phase transitions, protein structure re-arrangement...
  • system can be trapped
• most large moves are rejected (wasted cpu time)
Goals

Speed simulation
  • two approaches
    • make barriers easier to pass
    • waste less time on failed moves

Restrictions
  • must retain Boltzmann distribution
  • must preserved detailed balance
Parallel Tempering / Replica exchange

Two simulations, two temperatures

Hotter simulation moves faster, hops over barriers but
• it does not give $\langle A \rangle$ for desired temperature (270 K)
Closer temperatures

- copies of system different
- sometimes similar
Try swapping here

Energy
• no problem

Effect?
• we have correct energy of red system, but it has been hotter
  • more likely to cross barriers
if $E_{\text{hot}} < E_{\text{cold}}$
- no problem to swap copies
\( E_{\text{blue}} < E_{\text{red}} \) but not by much
- swapping possible

\( E_{\text{blue}} \gg E_{\text{red}} \)
- swapping not likely

so if \( \Delta E < 0 \)
- no problem

if \( \Delta E > 0 \)
- small ? possible
- big ? less likely
Probability of a total system

- probability of one system $i$  
  \[ p_i = \frac{e^{-E_i/kT_i}}{Z_i} \]

- probability of whole system
  \[
  p_{old} = p_ip_j = \frac{e^{-E_i/kT_i} e^{-E_j/kT_j}}{Z_i Z_j} = \frac{e^{(-E_i+E_j)/(kT_i+kT_j)}}{Z_iZ_j}
  \]

- probability of system before and after swap $\frac{p_{new}}{p_{old}}$
- $Z$'s will cancel
Exchange Probability

Question
- could the blue be part of the red ensemble?
- could the red be part of the blue ensemble?

Depends on temperatures, $\Delta E$

$$p_{\text{swap}} = \exp\left(\frac{E_j - E_i}{k(T_i - T_j)}\right)$$

if $p_{\text{swap}} > 1$
- accept
else use random number $[0..1]$ and compare with $p_{\text{swap}}$
- consider $E_j \approx E_i$
- blue bit higher than red (moves likely)
- blue much higher than red (moves very unlikely)
Implementing

Example

• try 100 moves normal MC of each system
• try 1 exchange / swap of systems
• swap means:
  • in MC steps \( e^{-\Delta E/kT} \) change \( T_1 \) and \( T_2 \)

Result

• two simulations
• each has Boltzmann distribution at right temperature
• cooler system has visited high temperatures / moved faster
• generalising
  • ...
Many replicas

- run many copies, similar temperatures
- every N moves, attempt an exchange of any pair

• normally blue would never exchange with red
• now possible in several steps
• red simulation is a valid ensemble at $T_{red}$
implementation

Any set of exchange attempts OK
  • may not be efficient
Detail balance preserved

Easy to implement

  • set up $N$ simulations at different temperatures
  • whenever a swap is successful, set $T_i$ to $T_j$ and $T_j$ to $T_i$

Alternative perspective
  • like simulated annealing but
    • annealing schedule (cooling) is automatic
Configurational Bias Monte Carlo
Rosenbluth sampling

Many Monte Carlo methods
• do not take random step
• find a low energy direction
• trial move more likely in that direction
• make acceptance probability less likely

Result
• less time spent generating unlikely moves + energy calculation

Rule
• must maintain detailed balance
• must finish with a Boltzmann distribution

Example – discrete system
Discrete Models / Chain growth moves

Lattice / off-lattice often easier to deal with
- particles only exist in certain places
- can only occupy certain states

Off-lattice discrete protein

Typical moves set
- pick random site in chain
- discard one half
- re-grow each site

- look at new configuration, accept/reject
- big reorganisation possible
Chain regrowth methods

Moves are big, but
- in a dense system, most will be rejected

We have big moves, but consider each step
Looking at sub-moves

at first step
• one possible direction is more likely

what if we move in the more likely region?
• we will tend to move downhill energetically
  • no Boltzmann distribution
• move $N_i \pi(i \rightarrow j) \neq N_j \pi(j \rightarrow i)$
  • detailed balance not preserved
Bias

Make downhill moves more likely
• make them more difficult to accept
Sometimes try uphill moves
• gain
  • fewer attempts at uphill moves
  • keep detailed balance + Boltzmann distribution

Next step
• do several biased moves
  • set of (probably) downhill moves
One step

Look at red and black choices

- calculate $E_{\text{black}}$, $E_{\text{red}}$ and probabilities

$$p_{\text{black}} = \frac{e^{-E_{\text{black}}}}{e^{\frac{-E_{\text{black}}}{kT}} \sum_{i}^{\text{red, black, ...}} e^{-\frac{-E_{i}}{kT}}}$$

$$N_{\text{choices}} \sum_{i=1} p_{i} = 1$$

- pick a direction according to $p_{i}$
- example...
direction picking

We have three possible directions

- \( p_1 = 0.2, p_2 = 0.5, p_3 = 0.3 \) from Boltzmann weights

挑一个随机数 \( 0 \leq x \leq 1 \)

if \( 0 \leq x < 0.2 \) choose (1)

elseif \( x < 0.7 \) choose (2)

else choose (3)

- what have we got now? not much yet
- usually choose single steps and preserve Boltzmann distribution
formalism

Where we have $N_{\text{choice}}$ possible directions

$$p_i = \frac{e^{-E_i}}{w}$$

$$w = \sum_{j=1}^{N_{\text{choice}}} e^{-E_j}$$

$w$ will come back in a moment
Several Bias steps

- break chain
- pick first step with bias
- second step with bias
- ...
- chain complete

- heavily biased
  - series of $N_{step}$ steps – usually favourable
    - without accept / reject along the way
- how to correct?
  - introduce "Rosenbluth factor"
    - $W_o$ (old), $W_n$ (new / trial)
Rosenbluth factor

Rosenbluth factor $W_n$

$$W = \prod_{m=1}^{N_{step}} w_m$$

Rosenbluth factor $W_o$

- pretend that the chain was chopped and calculate $w_m$ for each step

Accept reject

- if $W_n/W_o > 1$ accept
- else accept with $p = W_n/W_o$

$$p_i = \frac{e^{-E_i/kT}}{W}$$

$$w = \sum_{j=1}^{N_{choice}} e^{-E_j/kT}$$
Net result?

- take $N_{step}$ biased moves
- fix up distribution via acceptance criterion

Practical explanation (dense protein)
- each step we put atoms in a likely place (not on top of other atoms)
- after $N_{steps}$ we have a chain which is probably physically likely (unlikely to waste time on crazy moves)

Compare with normal Monte Carlo
- to go from black to red would have required a very specific set of random moves (unlikely to be found)
Who uses configurational biased MC?

- proteins, polymers
- easiest when system is discrete
  - difficult to code in continuous systems
- typical of many methods (introduce bias and correct afterwards)
- putting techniques together
Combinations of techniques

Goal
- finish with a Boltzmann distribution
- dynamics? maybe

Combinations
- Molecular dynamics and Monte Carlo?
- Monte Carlo good for non-physical systems
More combinations

Replica Exchange method

- MC or MD
- both will give ensemble / distribution at desired temperature

Imagine

- MC is good for complete re-arrangement of chain
- MD explores local (nearby) configurations
- could combine biased MC with MD
Comparison with other methods

• classic minimisation method – genetic algorithm
• basic idea
  • 100 or 1000 copies of system (protein, travelling salesman routes)

• make 100 copies of system
  while (not happy)
    find 50 worst copies (highest energy) throw away
    copy 50 best
    for (i = 0; i < 50; i++)
      apply random changes, combine copies
• system will gradually improve – fittest copies are kept
Comparing to MC

- Methods like genetic algorithm work with unknown distribution
- No theory to fall back on
  - No defined temperature
  - No defined probabilities

Summary of everything

Methods like molecular dynamics / Monte Carlo
- Infinite number of variations possible / legal
- Best may be system dependent
- Not restricted to molecular / atomic systems

Arbitrary decisions
- Temperature, move types
Questions

• how happy are we with replica exchange?
  • I simulate with four replicas – 4 × the time
  • Could I just simulate 4 times?

• some questions
  • could we use Monte Carlo for sequence optimisation?
  • could we use Monte Carlo for sequence alignments?
    • why would you want to?
    • what might be the problems?

• why I hate genetic algorithms