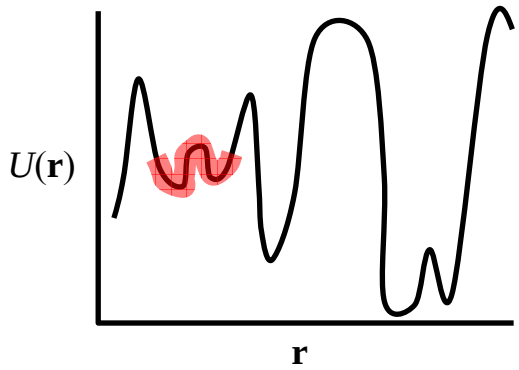
Exotic Monte Carlo

Andrew Torda, May 2008, 67.912

- 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

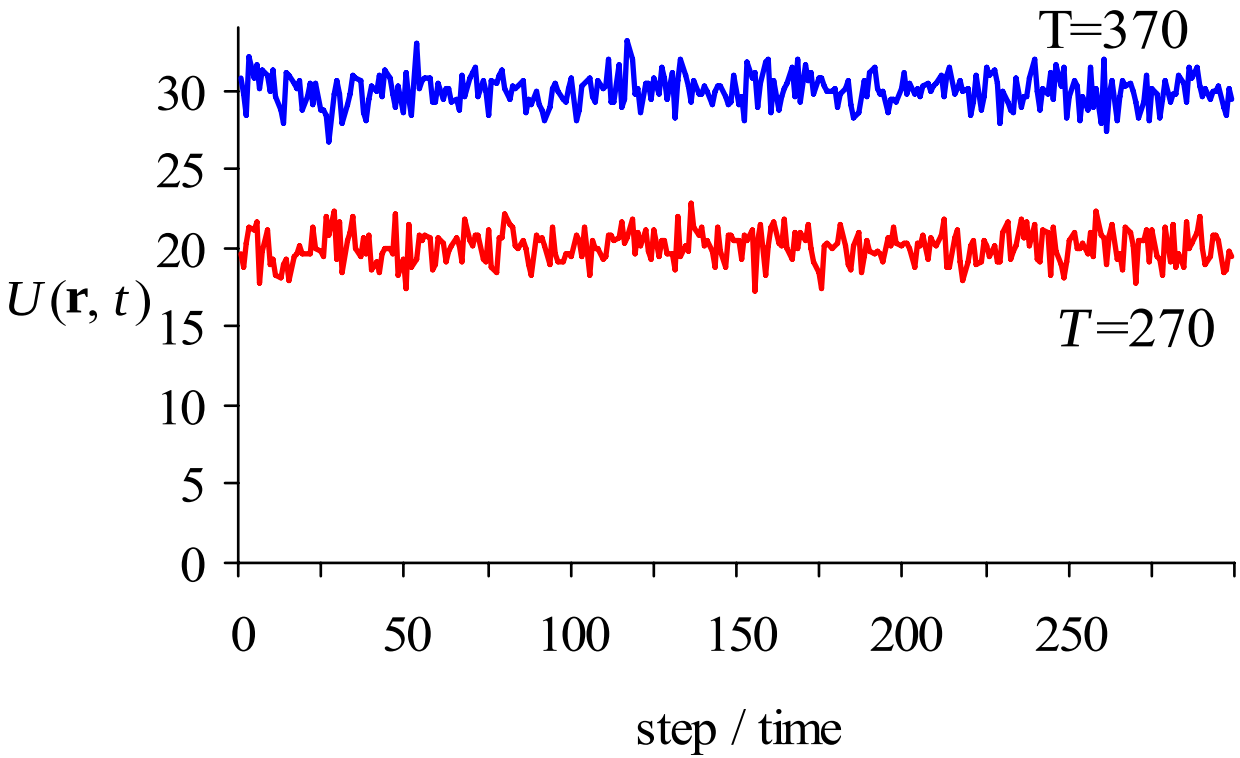
- two approaches
  - make barriers easier to pass
  - waste less time on failed moves

Restrictions

- must retain Boltzmann distribution
- must preserved detailed balance

Parallel Tempering / Replica exchange

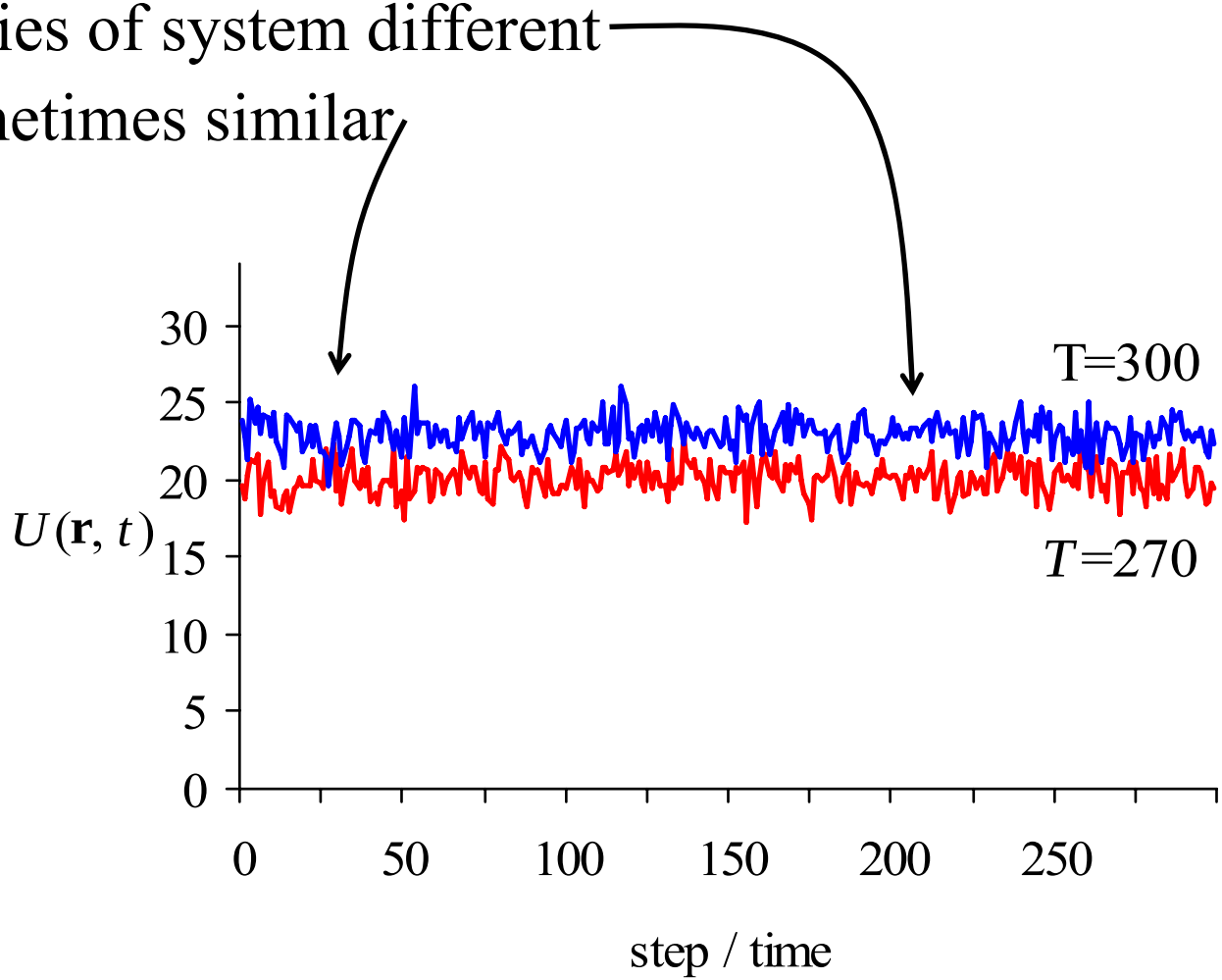
- two simulations, two temperatures



- hotter simulation moves faster, hops over barriers but
  - it does not give <*A*> for desired temperature (270)

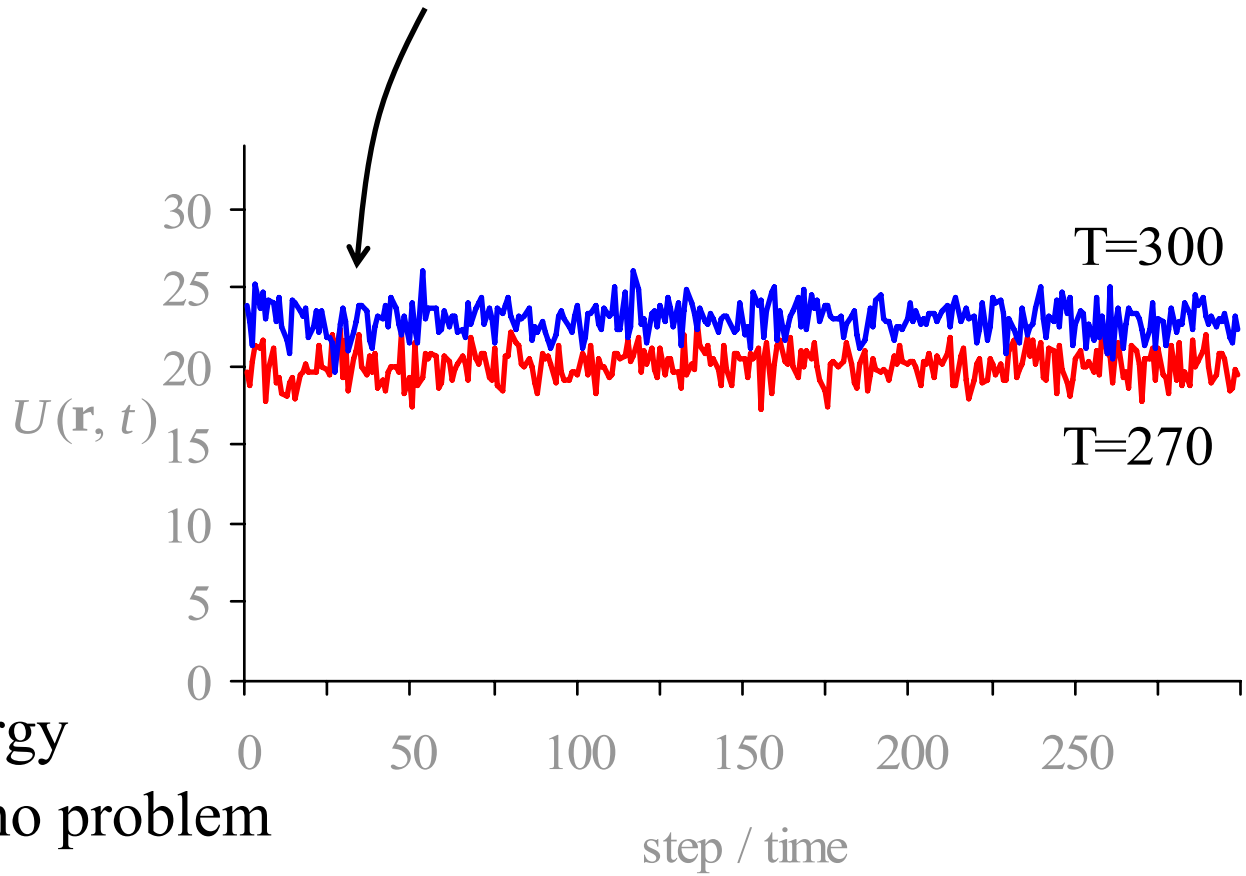
Closer temperatures

- copies of system different
- sometimes similar



swaps of copies

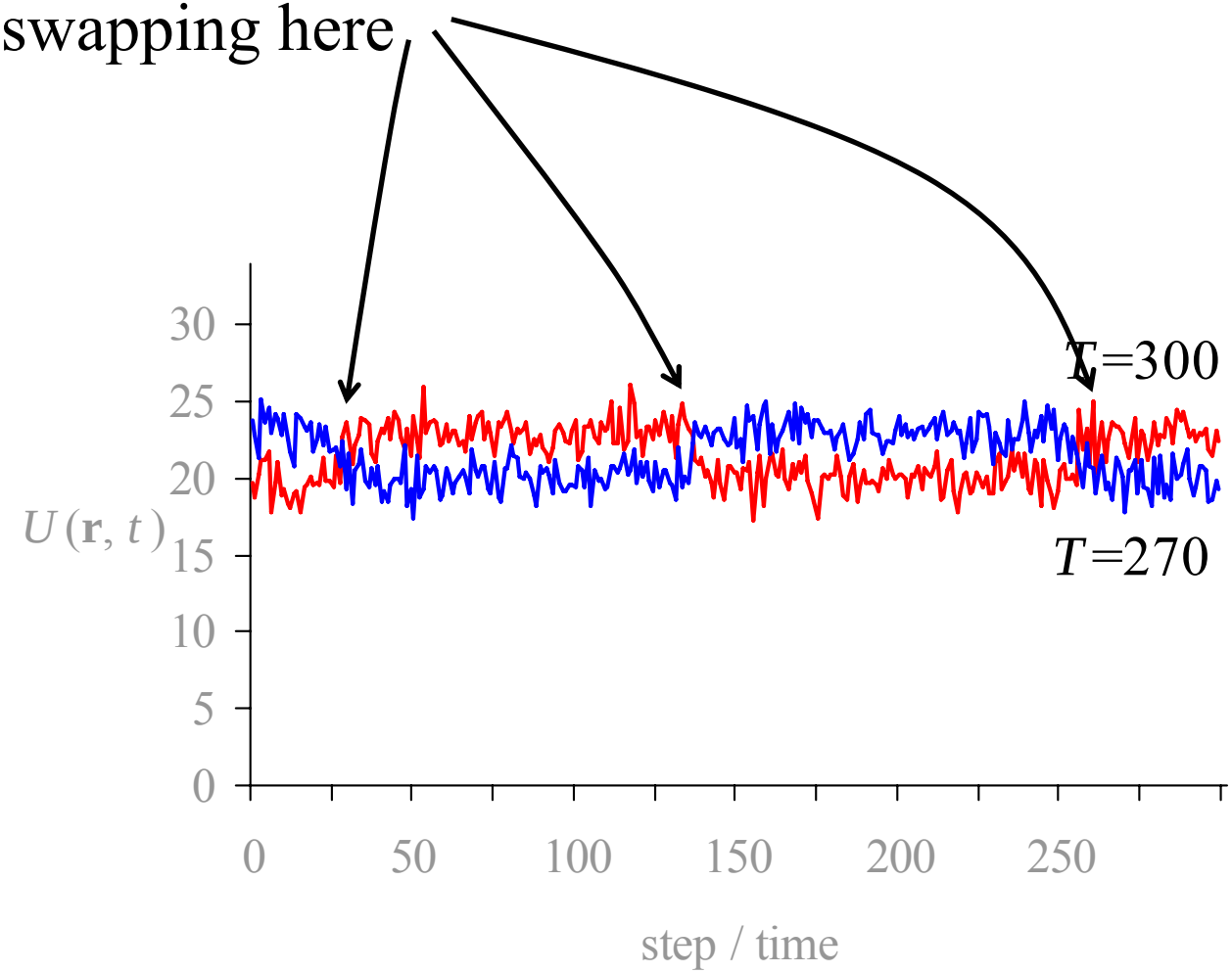
- try swapping here



- energy
  - no problem
- effect ?
  - we have correct energy of red system, but it has been hotter
    - more likely to cross barriers

easy swaps

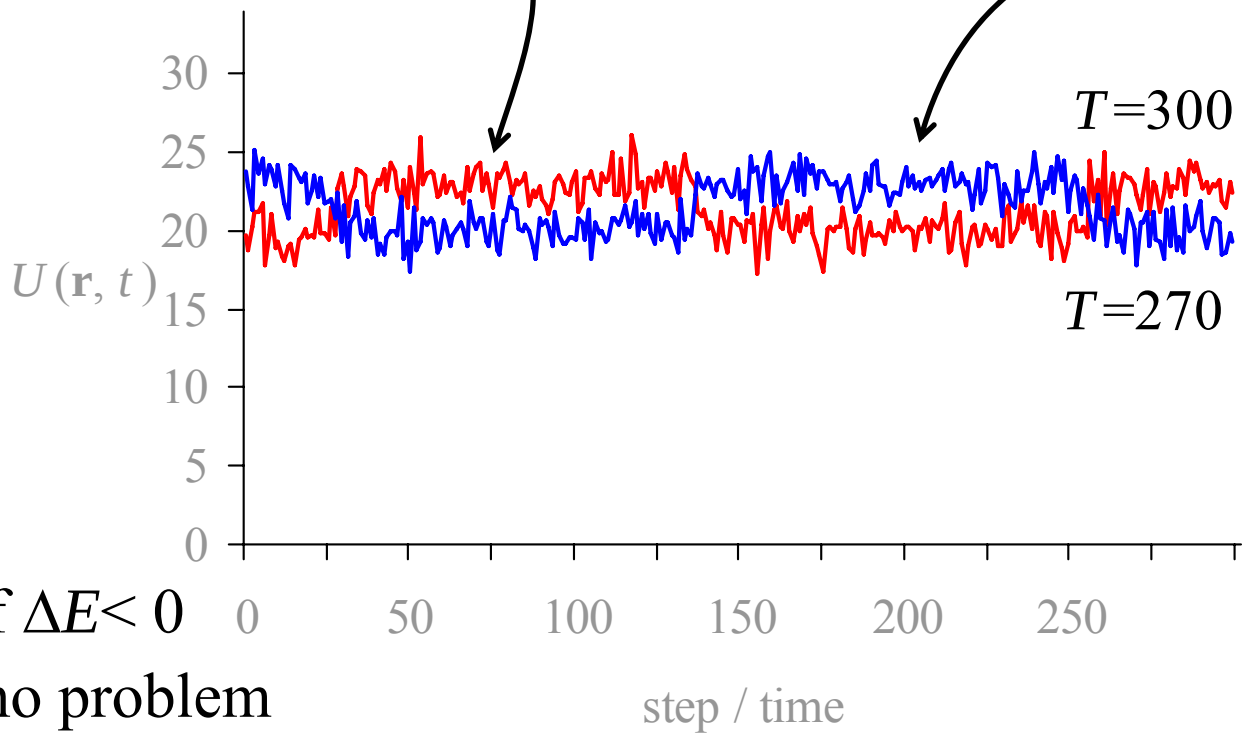
- try swapping here



- if $E_{hot} < E_{cold}$
  - no problem to swap copies

possible swaps

- $E_{blue} > E_{red}$ but not by much
  - swapping possible
- $E_{blue} >> E_{red}$
  - swapping not likely

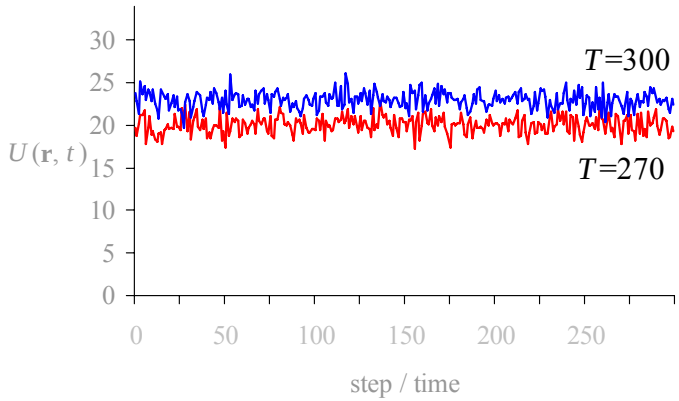


- so if $\Delta E < 0$
  - no problem
- if $\Delta E > 0$
  - small ? possible
  - big ? less likely

Exchange Probability

- Question
  - could the blue be part of the red ensemble ?
  - could the red be part of the blue ensemble ?
- Depends on temperatures, ΔE

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

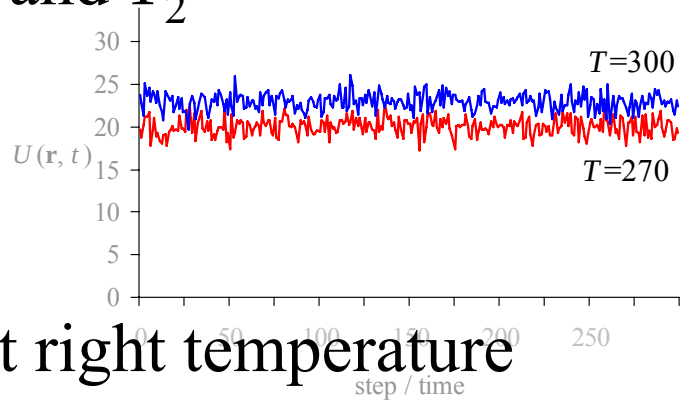


- if $p_{swap} > 1$
  - accept
- else use random number [0..1] and compare with p_{swap}
- consider $E_j \approx E_i$
  - blue bit higher than red (moves likely)
  - blue much higher than red (moves very unlikely)

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

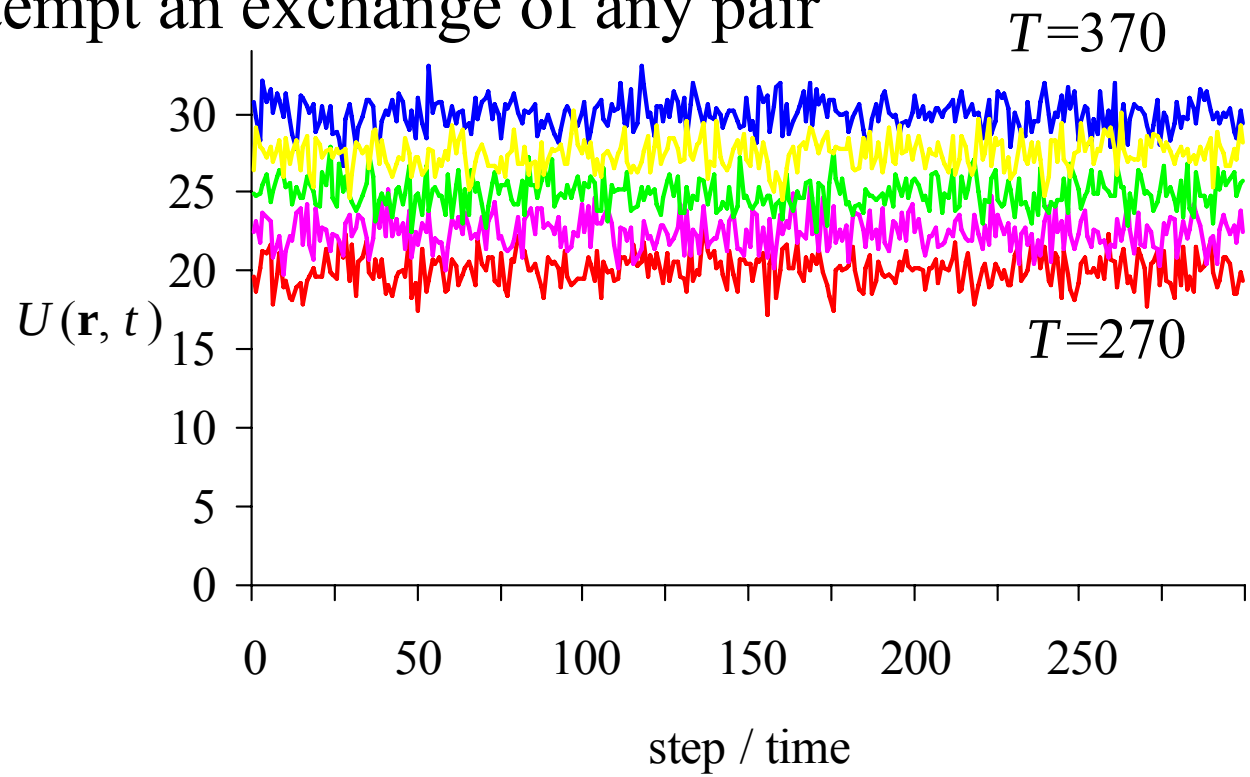


Result

- 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

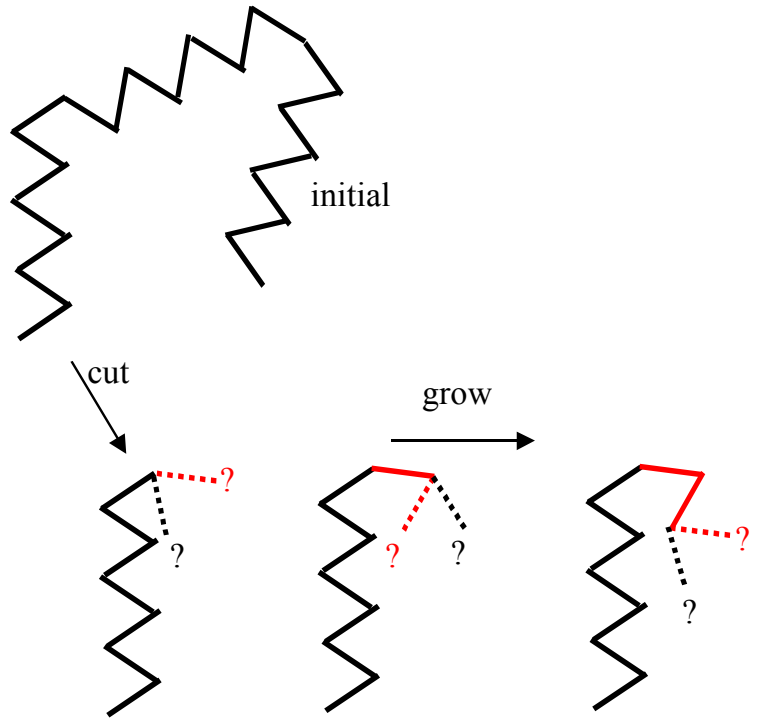
Configurational Bias Monte Carlo
Rosenbluth sampling

- 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

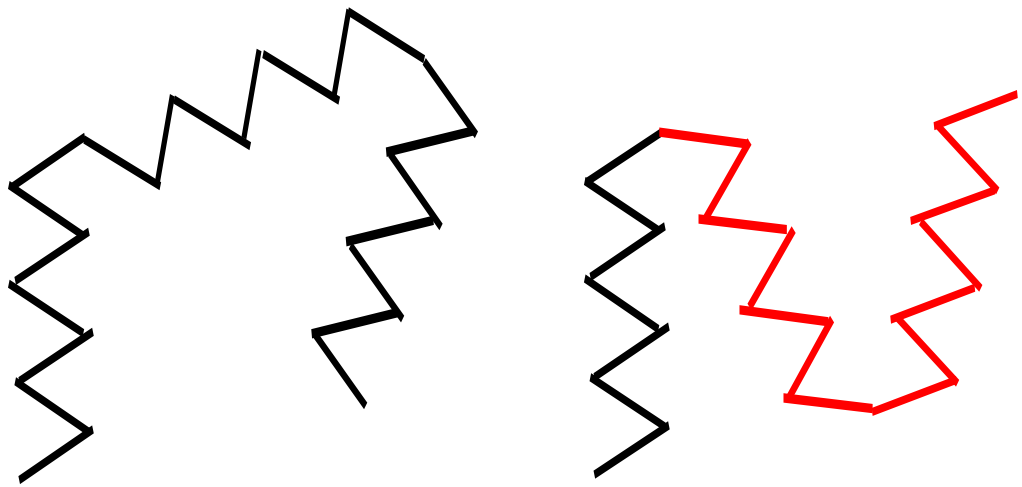
- Lattice / off-lattice often easier to deal with
  - particles only exist in certain places
  - can only occupy certain states
- Off-lattice discrete protein



- 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

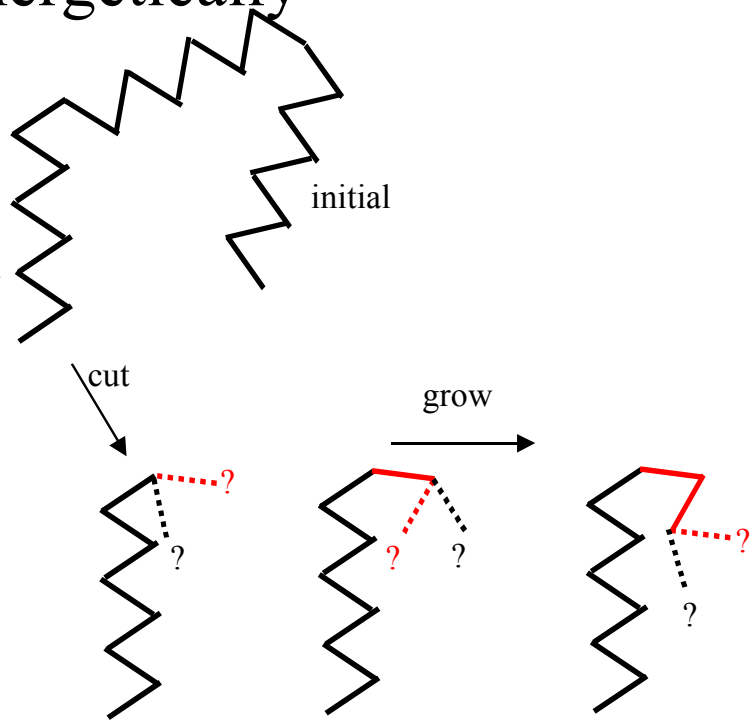


initial trial

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 $N_i \pi(i \to j) \neq N_j \pi(j \to i)$
    - detailed balance not preserved

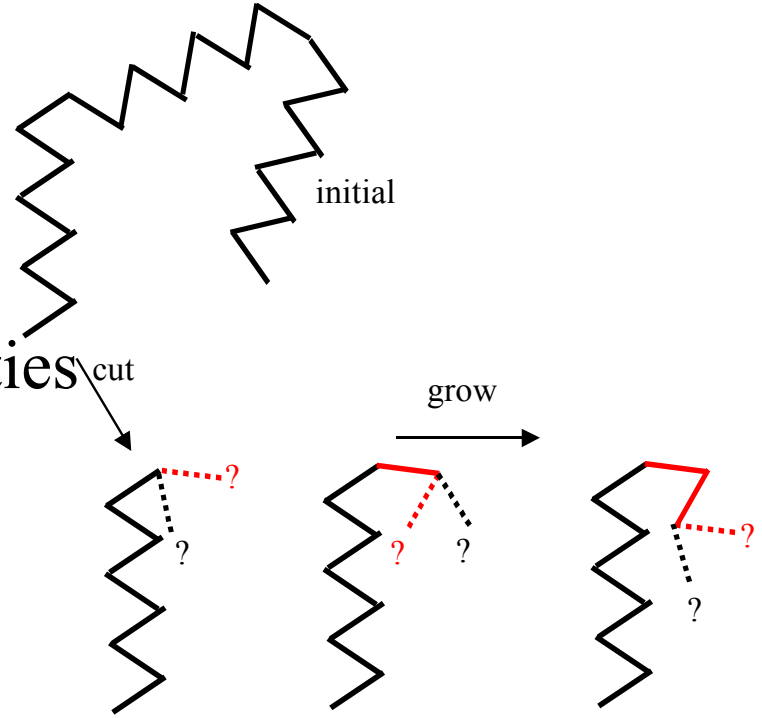


Bias

- 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_{black}, E_{red} and probabilities

$$p_{black} = \frac{e^{-E_{black}/kT}}{\sum_{red,black,\ldots i} e^{-E_i/kT}}$$

- $\sum_{i=1}^{N_{choices}} p_i = 1$

- 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

```
pick random number 0 ≤ x ≤ 1
if 0 ≤ x < 0.2      choose (1)
  elseif 0.2 ≤ x < 0.7  choose (2)
else choose (3)
```

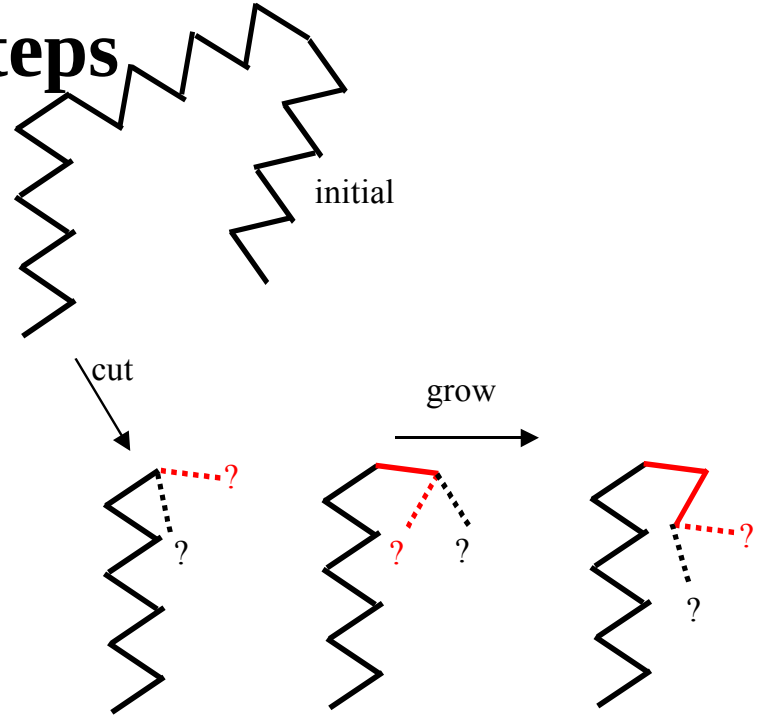
- what have we got now ? not much yet
- usually choose single steps and preserve Boltzmann distribution

formalism

- where we have N_{choice} possible directions $p_i = \frac{e^{-E_i/kT}}{w}$
- w will come back in a moment

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

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

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

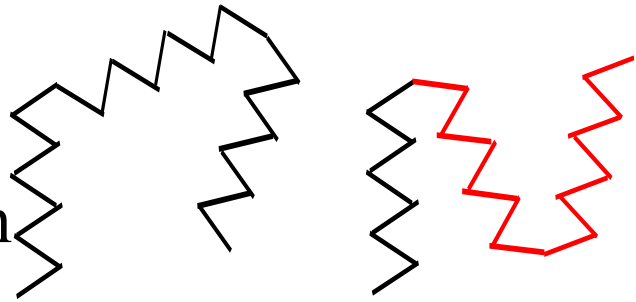
$$W = \prod_{m=1}^{N_{step}} w_m$$

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

- 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 $p = W_n/W_o$

Net result ?

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



initial trial

- 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 ?

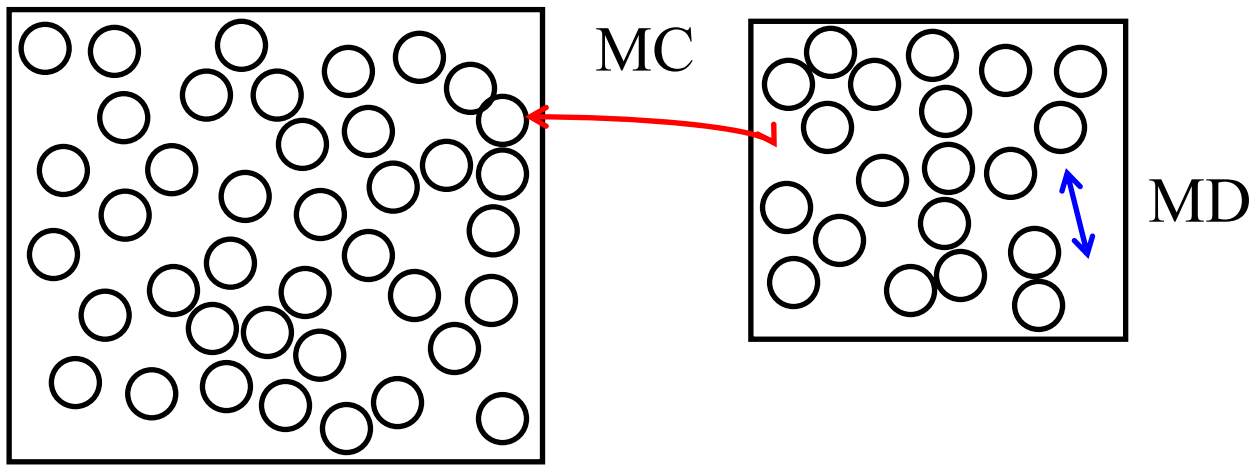
- proteins, polymers
- easiest when system is discrete
  - difficult to code in continuous systems

- typical of many methods (introduce bias and correct afterwards)

- putting techniques together

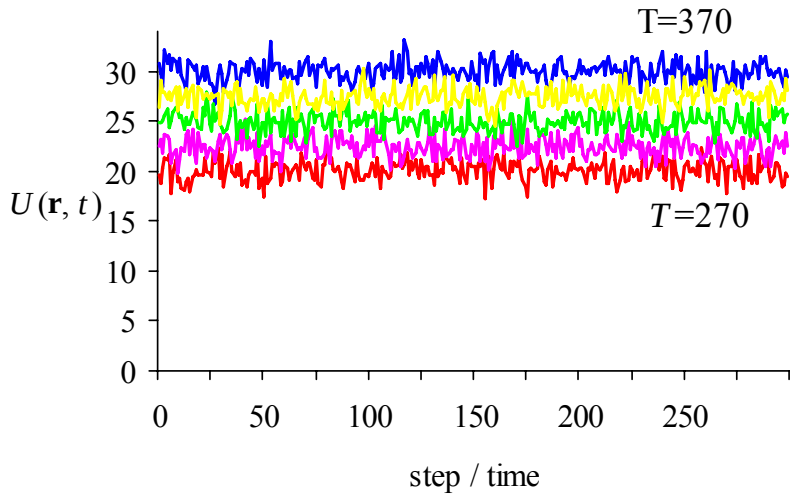
Combinations of techniques

- 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