Random Walks

A&T 110-123 and F&S 3.1.2

- As we explained last time, it is very difficult to sample directly a general probability distribution.
  - If we sample from another distribution, **the overlap will be order exp(-aN)**, where N is the number of variables. The error bars will get exponentially larger as N increases.

- Today we will discuss Markov chains (random walks), detailed balance and transition rules.
  - These methods were introduced by *Metropolis et al*. in 1953 who applied it to a hard sphere liquid.
  - It is one of the most powerful and used algorithms.

Equation of State Calculations by Fast Computing Machines

NICHOLAS METROPOLIS, ARIANNA W. ROSENBLUTH, MARSHALL N. ROSENBLUTH, AND AUGUSTA H. TELLER,
Los Alamos Scientific Laboratory, Los Alamos, New Mexico

AND

EDWARD TELLER,* *Department of Physics, University of Chicago, Chicago, Illinois*

(Received March 6, 1953)

A general method, suitable for fast computing machines, for investigating such properties as equations of state for substances consisting of interacting individual molecules is described. The method consists of a modified Monte Carlo integration over configuration space. Results for the two-dimensional rigid-sphere system have been obtained on the Los Alamos MANIAC and are presented here. These results are compared to the free volume equation of state and to a four-term virial coefficient expansion.

I. INTRODUCTION

THE purpose of this paper is to describe a general method, suitable for fast electronic computing machines, of calculating the properties of any substance which may be considered as composed of interacting individual molecules. Classical statistics is assumed, only two-body forces are considered, and the potential field of a molecule is assumed spherically symmetric. These are the usual assumptions made in theories of liquids. Subject to the above assumptions, the method is not restricted to any range of temperature or density. This paper will also present results of a preliminary two-dimensional calculation for the rigid-sphere system. Work on the two-dimensional case with a Lennard-Jones potential is in progress and will be reported in a later paper. Also, the problem in three dimensions is being investigated.

* Now at the Radiation Laboratory of the University of California, Livermore, California.

II. THE GENERAL METHOD FOR AN ARBITRARY POTENTIAL BETWEEN THE PARTICLES

In order to reduce the problem to a feasible size for numerical work, we can, of course, consider only a finite number of particles. This number N may be as high as several hundred. Our system consists of a square† containing N particles. In order to minimize the surface effects we suppose the complete substance to be periodic, consisting of many such squares, each square containing N particles in the same configuration. Thus we define d_{AB}, the minimum distance between particles A and B, as the shortest distance between A and any of the particles B, of which there is one in each of the squares which comprise the complete substance. If we have a potential which falls off rapidly with distance, there will be at most one of the distances AB which can make a substantial contribution; hence we need consider only the minimum distance d_{AB}.

† We will use the two-dimensional nomenclature here since it is easier to visualize. The extension to three dimensions is obvious.

Markov chain or Random Walk

- Markov chain is a random walk through phase space:

$$s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow s_4 \rightarrow \ldots$$

  Here "*s*" is the state of the system.

- The *transition probability* is $P(s_n \rightarrow s_{n+1})$ a *stochastic matrix*

$$P(s \rightarrow s') \geq 0 \qquad \sum_{s'} P(s \rightarrow s') = 1$$

- In a Markov chain, the distribution of s_{n+1} depends only on s_n (by definition). *A drunkard has no memory!*

- Let $f_n(s)$ be the probability after "n" steps. It evolves according to a "master equation."

$$f_{n+1}(s') = \Sigma_s \, f_n(s) \, P(s \rightarrow s') \quad or \quad f_{n+1} = P \, f_n$$

- The stationary states are eigenvectors of *P:* $P\,\phi = \varepsilon\,\phi$

Properties of Random Walk

- Because P is positive, the eigenvalues have ε ≤ 1.

  An equilibrium state must have ε *=1.*

- How many equilibrium states are there?
  - If it is *ergodic*, then it will converge to a unique stationary distribution (only one eigenvalue = 1)

- (In contrast to MD) ergodicity can be proven if:
  - One can move everywhere in a finite number of steps with non-zero probability. *No barriers!*
  - Non-periodic transition rules (e.g. hopping on bi-partite lattice)
  - Average return time is finite. (No expanding universe.) Not a problem in a finite system.

- If ergodic, convergence is *geometrical and monotonic*.

$$f_n(s) = \pi(s) + \sum_{\lambda} \varepsilon_\lambda^n c_\lambda \phi_\lambda(s)$$

If ε < 1, then after "n" iterations, this is 0!
Hence, ε = 1 is the stationary state.

Random Walks Example

from *A&T 110-123*

- Markov chains, detailed balance and transition rules.
  - Consider two states S_m and S_n linked by a transition probability ***P(m → n)***.

- Suppose the reliability of your computer follows a certain pattern. If it is "up" and running on day, then it has a 60% chance of running on the next. If, however, it is "down", it has a 70% changes of being down on next.

$$P(s \to s') = \begin{matrix} \text{Up} & \text{Dn} \\ \begin{pmatrix} 0.6 & 0.4 \\ 0.3 & 0.7 \end{pmatrix} & \begin{matrix} \text{Up} \\ \text{Dn} \end{matrix} \end{matrix}$$

- Suppose the computer is initially equally likely to be "up" or "dn"

$$\pi(1) = (0.5 \quad 0.5)$$

- What is probability of being up on the second day?

$$\pi(2) = T(1 \to 2)\pi(1) \begin{pmatrix} 0.6 & 0.4 \\ 0.3 & 0.7 \end{pmatrix} \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix} = \begin{pmatrix} 0.45 \\ 0.55 \end{pmatrix}$$

$$\pi(3) = T(2 \to 3)\pi(2) = T(2 \to 3)T(1 \to 2)\pi(1) = \begin{pmatrix} 0.435 \\ 0.565 \end{pmatrix}$$

$$\pi = \lim_{n \to \infty} (T(1 \to 2))^n \, \pi$$

Or as a eigenvalue problem:

$$\pi = T\pi$$

Metropolis algorithm

Three key concepts:

1. Sample by using an ergodic random walk.
2. Determine equilibrium state by using detailed balance.
3. Achieve detailed balance by using rejections.

Detailed balance: $\pi(s)\, P(s \to s') = \pi(s') P(s' \to s)$.

Rate balance from s to s′.

Put $\pi(s)$ into the master equation. *(Or sum above Eq. on s.)*

$$\pi(s') = \sum_s \pi(s)\, P(s \to s') = \pi(s') \sum_s P(s' \to s) = \pi(s')$$

- Hence, $\pi(s)$ is an eigenvector.
- If $P(s \to s')$ is ergodic, $\pi(s)$ is unique steady state solution.
- Possible to stay in same state: $P(s \to s) = 1 - \sum_{s' \neq s} P(s \to s')$

Replace strong "Microscopic Reversibility" criterion

Detailed balance: $\pi(s)\,P(s \to s') = \pi(s')P(s' \to s)$.

$$P(s \to s') = T(s \to s') \qquad \pi(s') \geq \pi(s),\ s \neq s'$$

$$P(s \to s') = T(s \to s')\left(\frac{\pi(s')}{\pi(s)}\right) \qquad \pi(s') < \pi(s),\ s \neq s'$$

$$P(s \to s) = 1 - \sum_{s' \neq s} P(s \to s')$$

if $T(s \to s') = T(s' \to s)$ i.e., symmetric

Other choices are possible, e.g.

$$P(s \to s') = T(s \to s')\frac{\pi(s')}{\pi(s) + \pi(s')} \qquad s \neq s'$$

$$P(s \to s) = 1 - \sum_{s' \neq s} P(s \to s')$$

Choose a particular solution which minimizes the variance of $\langle A \rangle_{est}$ or maximizes the efficiency

Rejection Method

Metropolis achieves detailed balance by *rejecting* moves.

General Approach:

1. Choose distribution to sample, e.g., $\pi(s) = \exp[-\beta H(s)]/Z$
2. Impose detailed balance on transition: **$K(s \to s') = K(s' \to s)$**

   where $K(s \to s') = \pi(s)\, P(s \to s')$

   (*probability of being at s)* * (*transition probability of going to s'*).
3. Break up transition probability into sampling and acceptance:

   $P(s \to s') = T(s \to s')\, A(s \to s')$

   (*probability of generating s'* from s) * (*probability of accepting move)*

The optimal acceptance probability that gives detailed balance is:

$$A(s \to s') = \min\left\{1, \frac{T(s' \to s)\pi(s')}{T(s \to s')\pi(s)}\right\} = \min\left\{1, \frac{\pi(s')}{\pi(s)}\right\}$$

If T is symmetric!

IMPORTANTLY Normalization of π(s) is not needed or used!

The "Classic" Metropolis method

Metropolis-Rosenbluth[2] -Teller[2] (1953) method for sampling the Boltzmann distribution is:

- Move from s to s' with probability T(s→s')= constant
- Accept with move with probability:

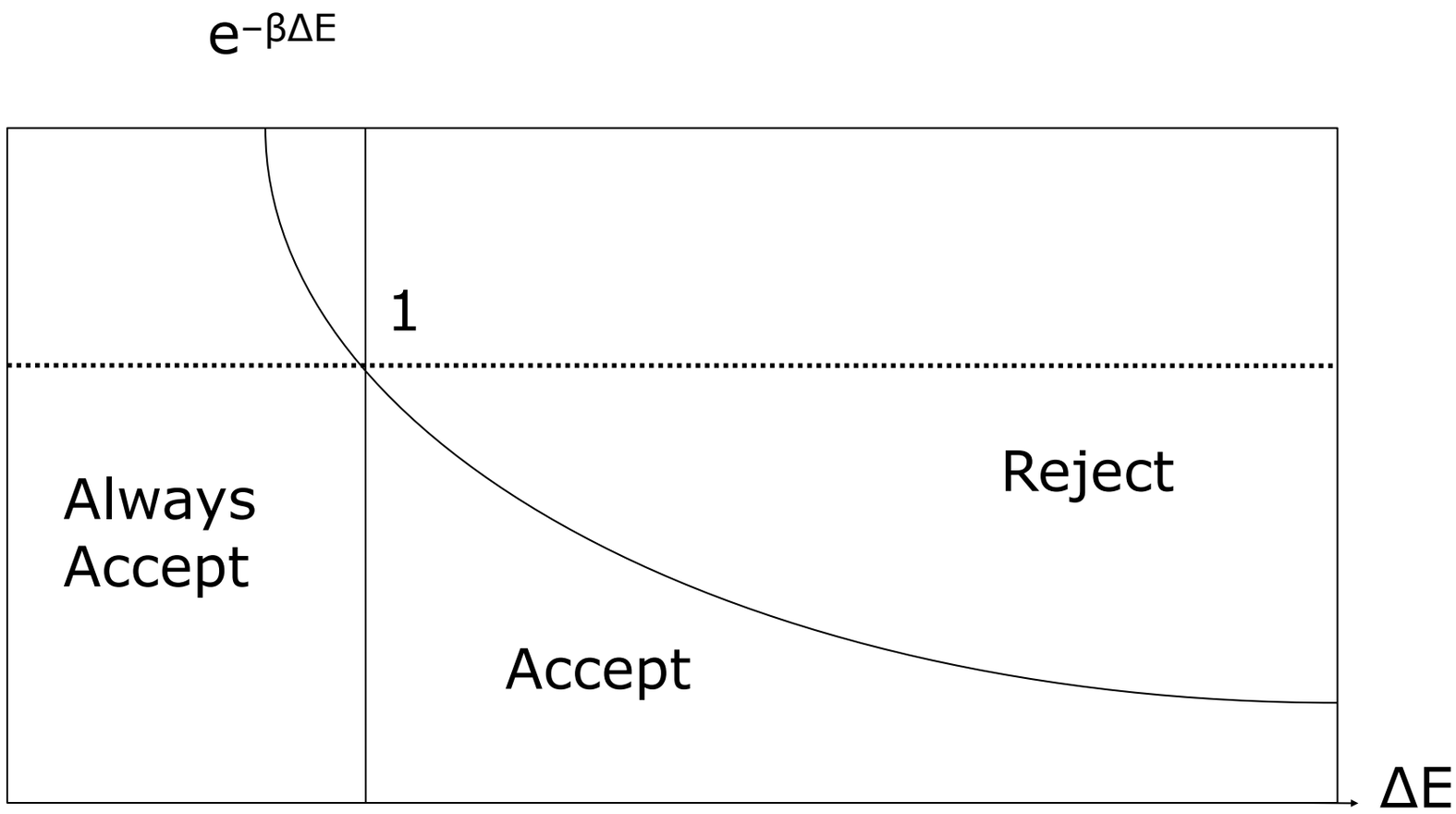
$$\mathbf{A(s \rightarrow s') = \min [\ 1\ ,\ \exp(-\beta[E(s')-E(s)])}$$

- Repeat many times

- Given ergodicity, the *distribution of s* will be the canonical distribution: $\pi(s) = \exp(-E(s)/k_BT)/Z$.

•Convergence is guaranteed but the rate is not!

Picture of Metropolis Rejection



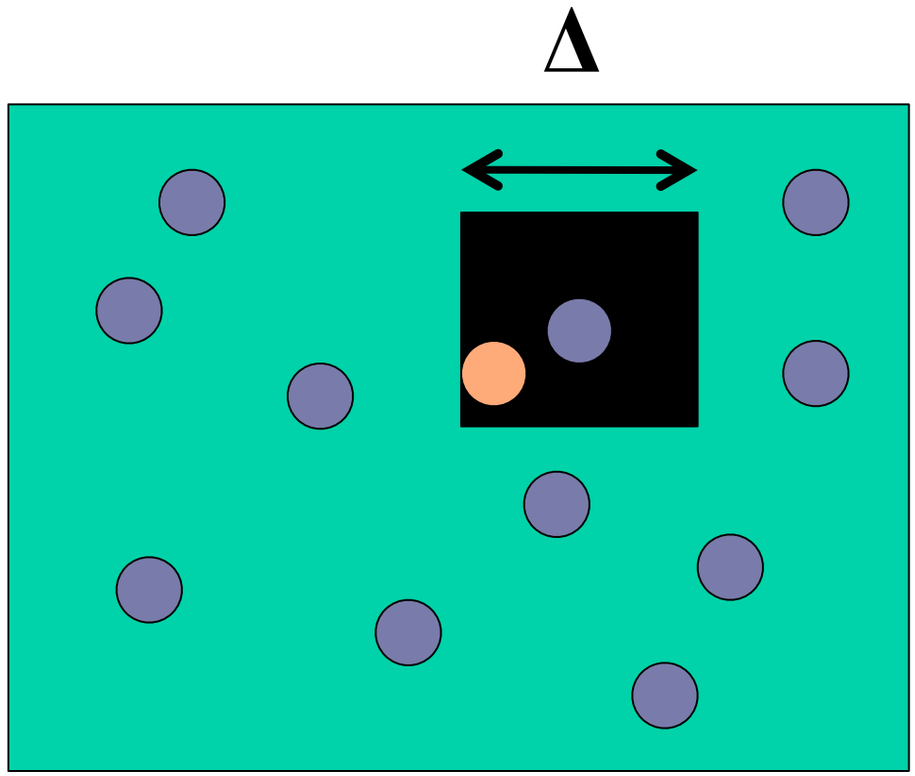
- If $\Delta E < 0$, it lowers the system energy → accept.
- Otherwise
  1. Generate UDRN u_n on (0,1)
  2. Compare u_n to $e^{-\beta\Delta E}$:

- If $u_n < e^{-\beta\Delta E}$, accept.
- If $u_n > e^{-\beta\Delta E}$, reject.

How to sample

$$S\_new = S\_old + \Delta \cdot (sprng - 0.5)$$

Uniform distribution in a cube of side "Δ".



Note: It is more efficient to move one particle at a time because only the energy of that particle comes in and the acceptance ratio will be larger.

$$A(s \to s') = \exp\left[-\beta(V(s') - V(s))\right]$$

$$= \exp\left[-\beta \sum_{j \neq i} \left(v(r_i' - r_j) - v(r_i - r_j)\right)\right]$$

For V with cut-off range, difference is local.

MONTE CARLO CODE

```
initstate(s_old)
E_old = action(s_old)
while True:
    sample(s_old,s_new,T_new,1)
    E_new = action(s_new)
    sample(s_new,s_old,T_old,0)
    A=exp(-E_new+E_old)(T_old/T_new)
    if A > sprng():
        s_old=s_new
        E_old=E_new
        naccept=naccept+1}
averages(s_old)
```

Initialize the state

Sample snew

Trial action

Find prob. of going backward

Acceptance prob.

Accept the move

Collect statistics

Overview of MCMC

- Decide how to move from state to state.
- Initialize the state
- Throw away first *k* states as being out of equilibrium.
- Then collect statistics but be careful about correlations.

Common errors:

1. **If you can move from *s* to *s'*, the reverse move must also be possible.** (You should check this.)
2. **Accepted and rejected states count the same!**

Exact: no time step error, no ergodic problems *in principle* but no dynamics either.

- *Always measure acceptance ratio.* RULE: 0.1 < a.r. < 0.9
  - ➢ Adjust ratio to *roughly* 0.5 by varying the "step size".

- A 20% acceptance ratio actually achieves better diffusion than a 50% acceptance ratio *in this example*.

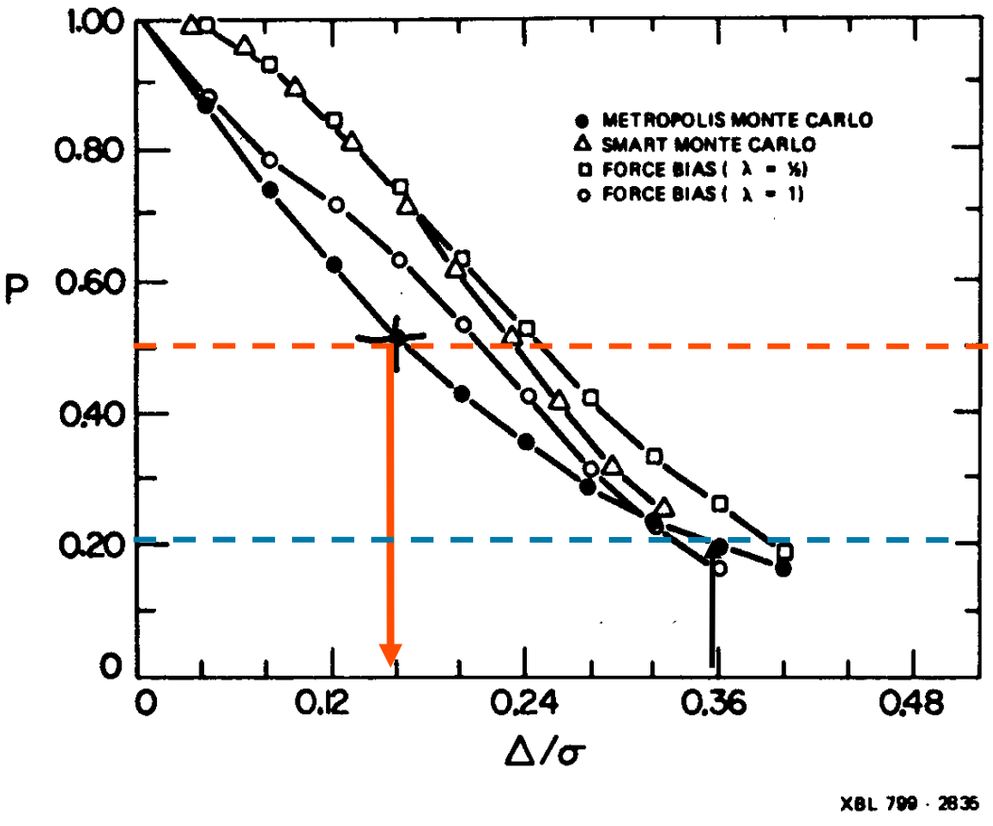


Fig. 1. Average acceptance probability.

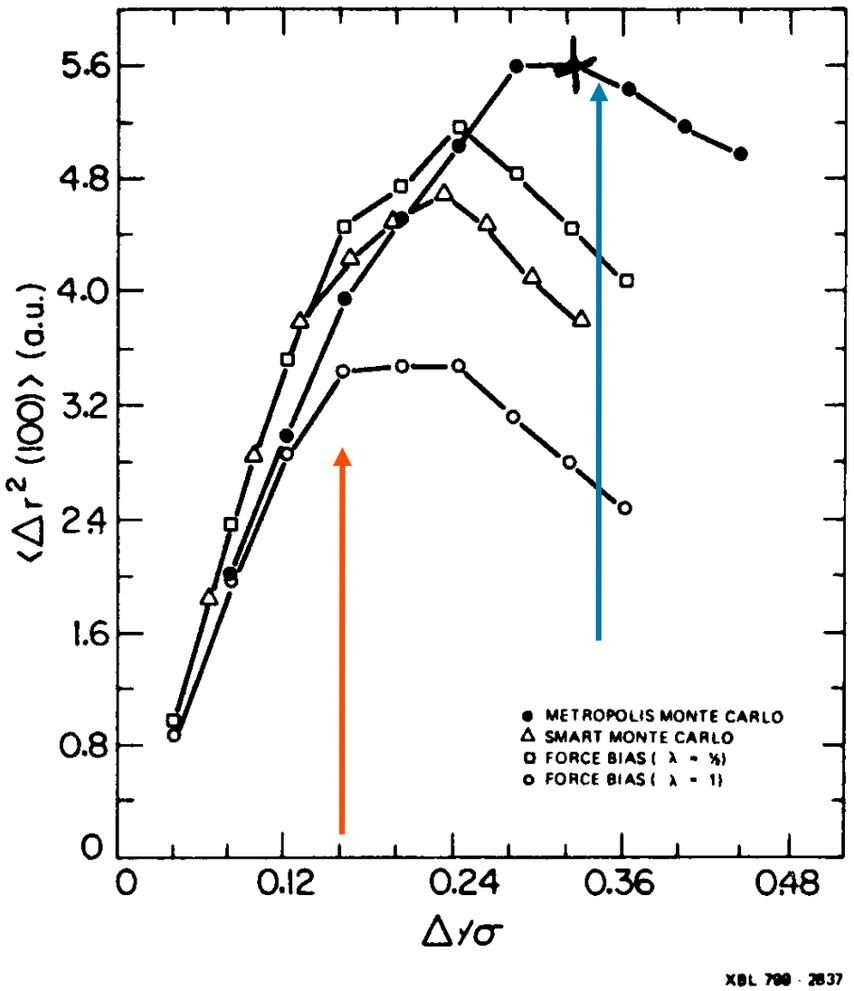


Fig. 3. $\langle(\vec{r}(i) - \vec{r}(i + 100))^2\rangle$

$\vec{r}(i)$ = 3n vector of argon positions at step i.

Variance of energy (local quantity) is not as sensitive to step size.

MC is a robust method! You don't need to fine tune things!

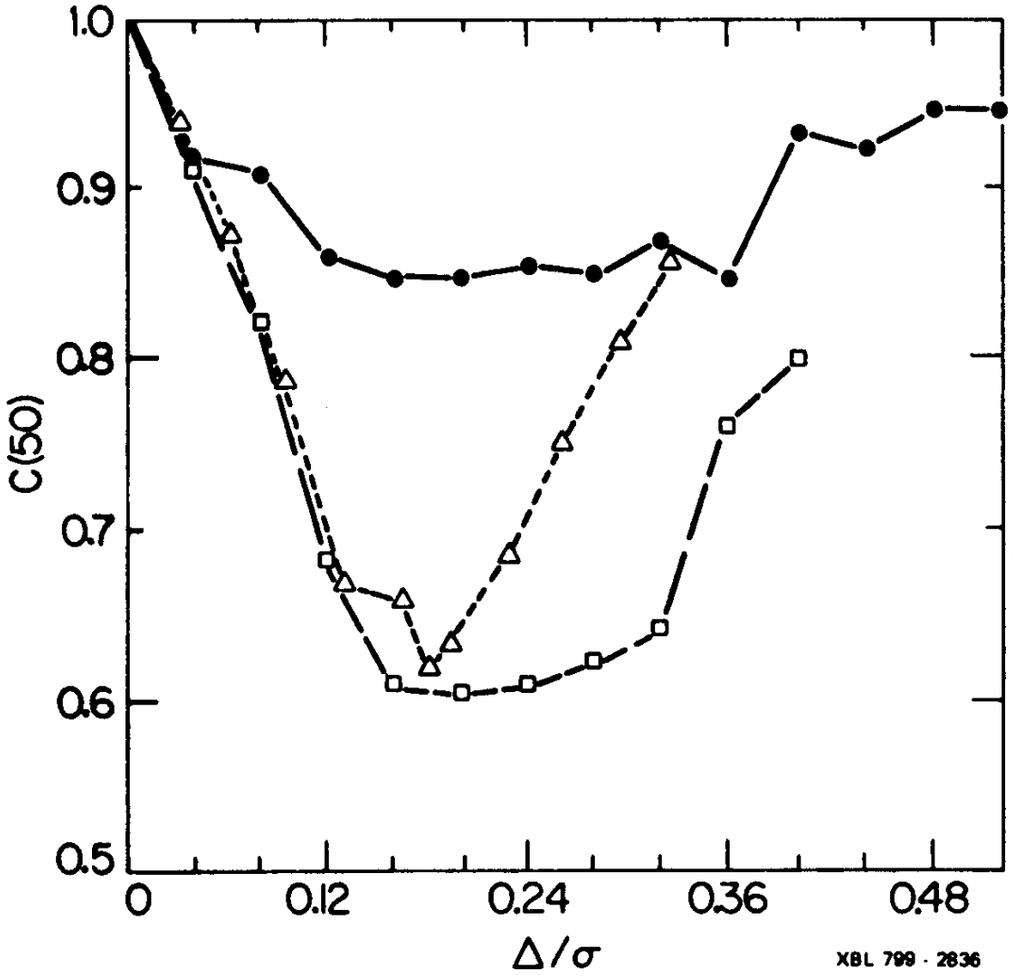


Fig. 2. $\frac{\langle \Delta V(i)\, \Delta V(i+50) \rangle}{\langle \Delta V(i)^2 \rangle}$

where i = step number and
Δ V is the deviation of potential energy from the mean.

Acceptable range of efficiency.

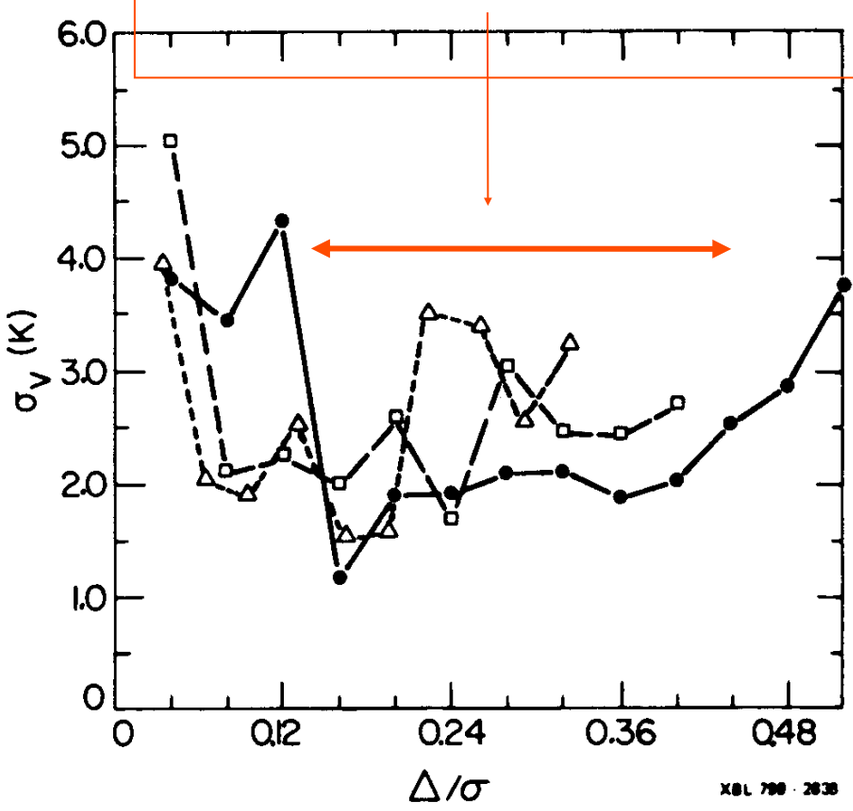


Fig. 4. The variance of the total potential energy for calculations with the same number of steps.

Optimizing the moves

- *Any transition rule is allowed* as long as you can go *anywhere in phase space with a finite number of steps*. (Ergodicity)

- Try to find a $T(s \rightarrow s') \sim \pi(s')/C$.
  - If you can, the acceptance ratio will be 1.

- Can use the forces to push the walk in the right direction.
  - Taylor expand about current point: $V(\mathbf{r})=V(\mathbf{r}_0)-F(\mathbf{r})(\mathbf{r}-\mathbf{r}_o)$
  - **Then set $T(s \rightarrow s') \sim \exp[-\beta(V(r_0)- F(r_0)(r-r_o))]$**
  - Leads to ***Force-Bias Monte Carlo.***
  - Related to Brownian motion (Smoluchowski Eq.)

next lecture: Force-biased and Smart Monte Carlo!

Comparison of MC and MD: Which is better?

- MD can compute dynamics.

  MC has a “kinetics" but dynamics is not necessarily physical. KMC dynamics is useful for studying long-term diffusive process.
- MC is simpler: No forces, no time-step errors and a direct simulation of the canonical ensemble.
  - In MC you can work on inventing better transition rules *to make CPUtime/physical-time faster*.
  - Ergodicity is less of a problem. However, MD is sometimes very effective in highly constrained systems.
- MC is more general.

  It can handle discrete degrees of freedom (e. g. spin models, quantum systems), grand canonical ensemble...

So you need both! The best is to have both in the same code so you can *use MC to warm up the dynamics (MD)* .