Monte Carlo methods for interacting spatial permutations

John Kerl

April 2, 2008

Abstract

These are lecture notes for a talk given to the Mathematical Physics Seminar at the University of Arizona Department of Mathematics on April 2, 2008.

This is a continuation of last week’s lecture given by Daniel Ueltschi. I sketch Monte Carlo methods which are used to estimate distribution of cycle length for the non-interacting case, the two-jump-interaction two-cycle case, and the general two-jump-interaction case.
### Contents

<table>
<thead>
<tr>
<th>Contents</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acknowledgements</td>
<td>4</td>
</tr>
<tr>
<td>2 Review of spatial random permutations</td>
<td>5</td>
</tr>
<tr>
<td>2.1 Hamiltonians for bosons and permutations</td>
<td>5</td>
</tr>
<tr>
<td>2.2 Models</td>
<td>5</td>
</tr>
<tr>
<td>2.3 Conceptualization</td>
<td>6</td>
</tr>
<tr>
<td>2.4 Context</td>
<td>6</td>
</tr>
<tr>
<td>2.5 Physics literature</td>
<td>7</td>
</tr>
<tr>
<td>2.6 Critical temperature</td>
<td>7</td>
</tr>
<tr>
<td>3 The computational project</td>
<td>9</td>
</tr>
<tr>
<td>3.1 Density of sites in infinite cycles</td>
<td>9</td>
</tr>
<tr>
<td>3.2 Tools</td>
<td>9</td>
</tr>
<tr>
<td>4 Visualization</td>
<td>11</td>
</tr>
<tr>
<td>4.1 H plots</td>
<td>11</td>
</tr>
<tr>
<td>4.2 Dot plots</td>
<td>11</td>
</tr>
<tr>
<td>4.3 Plots for the non-interacting model</td>
<td>12</td>
</tr>
<tr>
<td>4.4 Plots for the ( r^2 ) mode</td>
<td>13</td>
</tr>
<tr>
<td>4.5 Plots for the interacting model</td>
<td>14</td>
</tr>
<tr>
<td>5 Computation of ( V )</td>
<td>15</td>
</tr>
<tr>
<td>5.1 Brownian bridges</td>
<td>15</td>
</tr>
<tr>
<td>5.2 Software testing</td>
<td>15</td>
</tr>
<tr>
<td>5.3 Bridge results</td>
<td>16</td>
</tr>
<tr>
<td>5.4 Integral expression for ( V )</td>
<td>17</td>
</tr>
<tr>
<td>5.5 Argument reduction</td>
<td>17</td>
</tr>
<tr>
<td>5.6 Visualization</td>
<td>18</td>
</tr>
<tr>
<td>6 Metropolis-Hastings</td>
<td>19</td>
</tr>
</tbody>
</table>
1 Acknowledgements

We implement Monte Carlo techniques for the following:

- arXiv:cond-mat/0703315 (Gandolfo, Ruiz, Ueltschi): describes the non-interacting model. Referred to herein as the GRU paper. (Monte Carlo results were obtained by Gandolfo and Ruiz; we have reproduced their results.)

- arXiv:0711.1188 (Betz, Ueltschi): Describes the interacting model in detail. The U07 paper (next) summarizes much of the content of this longer paper.

- arXiv:0712.2443v3 (Ueltschi): Describes the interacting model. Referred to herein as the U07 paper.

This work (spring 2008) is supported by NSF grant DMS-0601075.

Many thanks to the following people for multiple insights: Daniel Ueltschi, Tom Kennedy, Janek Wehr, Ben Dyhr.
2 Review of spatial random permutations

2.1 Hamiltonians for bosons and permutations

As described in [BU] and [U07], the Hamiltonian for \( N \) interacting bosons in a domain \( \Lambda \) is

\[
H = -\sum_{i=1}^{N} \Delta_i + \sum_{i<j} U(x_i - x_j) \quad \text{in} \quad L^2_{\text{sym}}(\Lambda^N).
\]

Then, with inverse temperature \( \beta \),

\[
\text{Tr} e^{-\beta H} = \frac{1}{N!} \int \prod dx \int dx_N \int dW_{x_1 x_{\pi(1)}}(w_1) \cdots dW_{x_N x_{\pi(N)}}(w_N) \exp\left\{-\frac{1}{2} \sum_{i<j} \int_0^{2\beta} U(w_i(s) - w_j(s)) ds \right\}
\]

where \( w_i(s) \) is a Brownian bridge running from \( x_i \) to \( x_{\pi(i)} \) in time \( 2\beta \). Write this as

\[
\text{Tr} e^{-\beta H} = \frac{1}{N!} \int \prod dx \sum_{\pi} e^{-H(x,\pi)}
\]

where

\[
e^{-H(x,\pi)} = \left[ \prod_{i=1}^{N} dW_{x_i x_{\pi(i)}}(\omega_i) \right] \exp\left\{-\frac{1}{2} \sum_{i<j} \int_0^{2\beta} U(w_i(s) - w_j(s)) ds \right\}.
\]

After cluster expansion (a highly non-trivial step, as yet lacking rigorous justification), one obtains

\[
H(x, \pi) = \frac{1}{4\beta} \sum_{i=1}^{N} |x_i - x_{\pi(i)}|^2 + \sum_{i<j} V(x_i, x_{\pi(i)}, x_j, x_{\pi(j)}) + \text{higher orders}.
\]

Note that the bosonic Hamiltonian \( H \) has been converted to a Hamiltonian \( H \) on permutations.

The interaction between jumps \( x \mapsto y \) and \( x' \mapsto y' \) is

\[
V(x, y, x', y') = \int [1 - e^{-\frac{1}{4\beta} \int_0^{2\beta} U(\omega(s)) ds}] dW_{x-x', y-y'}(\omega).
\]

If \( U \) is a hard-core potential with radius \( a \) (i.e. \( U(r) = \infty \) for \( r < a \) and \( U(r) = 0 \) for \( r \geq a \)), then \( V(\cdot) \) is the probability that a Brownian bridge from \( x - x' \) to \( y - y' \) hits the ball of radius \( a \) centered at the origin.

Is there a simple expression involving special functions? Apparently not.

2.2 Models

We simulate three models for spatial random permutations. The first two have been completely coded; the third is in progress.

- The non-interacting model ([GRU]):

\[
H(x, \pi) = \frac{1}{4\beta} \sum_{i=1}^{N} |x_i - x_{\pi(i)}|^2.
\]
The $r_2$ interacting model (U07):

$$H(x, \pi) = \frac{1}{4\beta} \sum_{i=1}^{N} |x_i - x_{\pi(i)}|^2 + \alpha r_2(\pi).$$

The interacting model (U07):

$$H(x, \pi) = \frac{1}{4\beta} \sum_{i=1}^{N} |x_i - x_{\pi(i)}|^2 + \sum_{i<j} V(x_i, x_{\pi(i)}, x_j, x_{\pi(j)}).$$

### 2.3 Conceptualization

- The distance-dependent term $e^{-\frac{1}{4\beta} \sum_{i} \| x - \pi(x) \|^2}$ makes a permutation $\pi$ with a long jump (i.e. $\pi(x)$ far from $x$) less probable.
- The $e^{-\alpha r_2(\pi)}$ term discourages permutations with 2-cycles.
- The interacting term discourages permutations with $x_i$ close to $x_j$ and $\pi(x_i)$ close to $\pi(x_j)$, regardless of jump lengths $\|x_i - \pi(x_i)\|$ or $\|x_j - \pi(x_j)\|$. The permutation is favored even less if the two black arrows cross (i.e. larger $\theta$ as discussed below).

### 2.4 Context

The critical temperature $T_c$ for Bose-Einstein condensation is a (mostly unknown) function of scattering length $a$. Even the sign of the slope of $T_c(a)$ near zero is contested. It is believed that

$$\frac{T_c(a) - T_c(0)}{T_c(0)} = c\rho^{1/4}a + o(\rho^{1/3}a).$$

Currently, it is thought that $c \approx 1.3$. The Monte Carlo simulations described here will permit tighter estimation of $c$. 
2.5 Physics literature

- 1964: Huang: $\frac{\Delta T}{T_c} \sim (a\rho^{1/3})^{3/2}$, increases
- 1971: Fetter & Walecka: $\frac{\Delta T}{T_c}$ decreases
- 1982: Toyoda: $\frac{\Delta T}{T_c}$ decreases
- 1992: Stooft: $\frac{\Delta T}{T_c} = c a\rho^{1/3} + o(a\rho^{1/3}), \quad c > 0$
- 1996: Bijlsma & Stooft: $c = 4.66$
- 1997: Grütter, Ceperley, Laloë: $c = 0.34$
- 1999: Holzmann, Grütter, Laloë: $c = 0.7$: Holzmann, Krauth: $c = 2.3$;
- 1999: Baym et. al.: $c = 2.9$
- 2000: Reppy et. al.: $c = 5.1$
- 2001: Kashurnikov, Prokof'ev, Svistunov: $c = 1.29$
- 2001: Arnold, Moore: $c = 1.32$
- 2004: Kastening: $c = 1.27$
- 2004: Nho, Landau: $c = 1.32$

2.6 Critical temperature

We define $\phi(\alpha)$ to be the probability that the origin is in an infinite cycle. (Here, $\alpha = 1/4\beta$; this figure is from [GRU].) At the critical temperature $\alpha$, $\phi(\alpha)$ goes to zero. Monte Carlo simulations undertaken in this project will discover how this graph changes in the presence of interactions.
3 The computational project

3.1 Density of sites in infinite cycles

Given a random variable $\theta(\pi)$, compute its expected value. The random variable of interest for this project is the density of sites in cycles of specified length:

$$\rho_{mn}(\pi) = \frac{1}{V} \# \{ i = 1, \ldots, N : m \leq \ell_i(\pi) \leq n \}$$

The usual prescription in probability is

$$E[\rho_{mn}] = \sum_{\pi \in S_N} \rho_{mn}(\pi) P(\pi) = \sum_{\pi \in S_N} \rho_{mn}(\pi) \frac{e^{-H(x,\pi)}}{Y}.$$ 

The computational burden splits into three main components:

1. Finding $H$, especially its $V$ term. (For Metropolis, $\Delta H$ including $\Delta V$.)
2. Sampling (via Metropolis) from a non-uniform probability distribution on $N!$ permutations for $N$ as big as $50^3$.
3. Visualizing the results.

3.2 Tools

- Linux environment, although in principle everything should be portable to other operating systems.
- Optimizing compiler: gcc -O3.
- Build tool: make and automatic makefile generation.
- Performance analyzer: gprof. This shows where a program is spending most of its time.
- Error detector: valgrind. Finds many (but not all!) common errors, e.g. malloc without free.
- Code navigation: ctags. Allows a smart editor (vim, emacs) to jump directly to a subroutine body.
- Graphing utility: xgr. Nice plots in these slides were made in Matlab; quick-and-dirty plots without axis labels were done using xgr.

Sample gprof output:

<table>
<thead>
<tr>
<th>% cumulative</th>
<th>self time</th>
<th>seconds</th>
<th>calls</th>
<th>name</th>
</tr>
</thead>
<tbody>
<tr>
<td>64.74</td>
<td>1.67</td>
<td>1.67</td>
<td>3729508</td>
<td>pmt_send_x_to_y_n2_delta</td>
</tr>
<tr>
<td>10.08</td>
<td>1.93</td>
<td>0.26</td>
<td>13047273</td>
<td>get_distance_squared</td>
</tr>
<tr>
<td>7.37</td>
<td>2.12</td>
<td>0.19</td>
<td>3617278</td>
<td>get_Delta_H</td>
</tr>
<tr>
<td>5.81</td>
<td>2.27</td>
<td>0.15</td>
<td>3617278</td>
<td>x_to_uniform_y</td>
</tr>
<tr>
<td>4.26</td>
<td>2.38</td>
<td>0.11</td>
<td>3617278</td>
<td>metro_step</td>
</tr>
<tr>
<td>3.49</td>
<td>2.47</td>
<td>0.09</td>
<td>3617278</td>
<td>pmt_send_x_to_y</td>
</tr>
</tbody>
</table>
3.10 2.55 0.08 10545 pmt_get_cycle_counts
0.78 2.57 0.02 10000 vector_accumulate
0.39 2.58 0.01 10546 metro_sweep
0.00 2.58 0.00 10545 get_rho_L_pi
...
4 Visualization

There are three main plots:

1. Plots of the system energy \( H \).
2. Dot plots of the cycles.
3. \( E[\rho_0,k] \) as a function of \( k \) from 0 to \( N \).

4.1 \( H \) plots

Here is a plot of system energy \( H \) for \( L = 10, d = 3 \), no interactions, and \( \beta = 1 \):

This plot is typical for various parameter values; only one such plot is shown here.

- The horizontal axis counts Metropolis sweeps.
- The system was found to be thermalized (as described below) after 559 steps; \( \rho \) values were accumulated over 10,000 sweeps.
- The system energy \( H \) is shown in blue.
- In red is \( H \) smoothed out over a sliding window of 100 sweeps.
- In green is the same smoothed system energy, multiplied by 0 before thermalization and 1 after. Thus, the plot “goes green” when thermalization has occurred.

4.2 Dot plots

A dot plot of the points \( \{x_1, \ldots, x_N\} \) and a permutation \( \pi \) has a dot for each point \( x \), along with a line from \( x \) to \( \pi(x) \) for each point \( x \).

Key points:

- For infinite \( \beta \), the permutation weight \( e^{-\frac{1}{\beta} \sum \|x-\pi(x)\|^2} \) becomes uniform: individual permutation jumps can be arbitrarily long.
• For $\beta = 0$, only the identity permutation is possible.

• For moderate $\beta$, long jumps are discouraged. Nonetheless, a long cycle can occur when short jumps chain together.

4.3 Plots for the non-interacting model

Here is $L = 10$, $d = 3$, point positions uniformly distributed on the cube of width 10 but not metropolized, no interactions, varying $\beta$:

The $E[\rho]$ plots are much as in the GRU paper.

• The horizontal axis is $k/N$ for $k$ from 0 to $N$.

• In blue on the vertical axis is $\rho_{0,k}$ for the permutation realized on the last Metropolis sweep.

• In green on the vertical axis is $\rho_{k,k}$ for the permutation realized on the last Metropolis sweep.

• In red on the vertical axis is $E[\rho_{0,k}]$ over 10,000 Metropolis sweeps.

• In yellow on the vertical axis is $E[\rho_{k,k}]$.

Here are $E[\rho]$ plots for the same parameter values as the dot plots:
4.4 Plots for the $r_2$ model

Here we fix $\beta = 0.5$ and vary $\alpha$. Note that $\alpha = 0$ recovers the non-interacting case. The dot plots are indistinguishable from the non-interacting case. The $E[\rho_{0,k}]$ plots are similar, so they are superimposed. Blue is $\alpha = 0$, red is $\alpha = 5$, and green is $\alpha = 20$. 
4.5 Plots for the interacting model

This is recent work — more are to be obtained.

Here is $\beta = 0.15626$ (just below non-interacting critical temperature), with $\alpha = 0.0$ and $\alpha = 0.1$:

The value $\phi(\beta)$ is the probability that the origin is in an “infinite” cycle. It may be read off the $E[\rho]$ plots as the distance from the upper left corner of the $\rho$ plot to the first leftward lean of the red curve. Critical $\beta_c$ has $\phi(\beta) = 0$.

<table>
<thead>
<tr>
<th>$\beta$</th>
<th>$\phi_0(\beta)$</th>
<th>$\phi_{\alpha=0.0}(\beta)$</th>
<th>$\phi_{\alpha=0.01}(\beta)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.227273</td>
<td>0.5203</td>
<td>0.5824</td>
<td>0.8081</td>
</tr>
<tr>
<td>0.208333</td>
<td>0.4373</td>
<td>0.5114</td>
<td>0.8057</td>
</tr>
<tr>
<td>0.192308</td>
<td>0.3703</td>
<td>0.4440</td>
<td>0.7835</td>
</tr>
<tr>
<td>0.178571</td>
<td>0.2625</td>
<td>0.3097</td>
<td>0.7868</td>
</tr>
<tr>
<td>0.166667</td>
<td>0.1517</td>
<td>0.2148</td>
<td>0.7769</td>
</tr>
<tr>
<td>0.161290</td>
<td>0.1133</td>
<td>0.1637</td>
<td>0.7663</td>
</tr>
<tr>
<td>0.156250</td>
<td>0.0824</td>
<td>0.1220</td>
<td>0.7693</td>
</tr>
<tr>
<td>0.147059</td>
<td>0.0311</td>
<td>0.0351</td>
<td>0.7645</td>
</tr>
</tbody>
</table>

Conclusion: interactions lower critical $\beta$. More simulations are needed.
5 Computation of V

5.1 Brownian bridges

Write $\hat{x}$ and $\hat{y}$ for $x - x'$ and $y - y'$ respectively. Simply generate $N_b$ Brownian bridges from $\hat{x}$ to $\hat{y}$, with $N_p$ mesh points per bridge, and see what fraction of them intersect the ball of radius $a$ centered at the origin.

- Start with a unit-uniform pseudorandom number generator (RNG).
- Use a Box-Muller transform (cf. Numerical Recipes) to get standard-normal deviates.
- Brownian motion for $t$ from 0 to 1 in steps of $\Delta t$: $B_0 = 0$ and $B_{t+1} = B_t + \Delta B$ where $\Delta B$ is normal with mean zero and variance $\Delta t$.
- Brownian bridge from $\hat{x} = 0$ to $\hat{y} = 0$ for $t$ from 0 to 1: $R_t = B_t - tB_1$.
- Brownian bridge from $\hat{x}$ to $\hat{y}$ for $t$ from 0 to $T$: $\sqrt{T} R_t + \hat{x} + \frac{t}{T}(\hat{y} - \hat{x})$.

The plot on the left shows, for $d = 1$, $N_b = 20$ bridges with $N_p = 1000$ points per bridge, bridged from $x = -1$ to $y = 2$ ($d = 1$) in time $T = 1$, with $R_t$ plotted against $t$.

The plot on the right shows, for $d = 3$, the trajectory of a single bridge from $x = (-1,0,0)$ to $y = (2,0,0)$ in time $T = 1$, with the first two components of $R_t$ plotted.

5.2 Software testing

Incrementally test the subroutines for Brownian motion, zero-to-zero bridges, and general bridges.

- Generate $N_b$ bridges of $N_p$ points each, for $d = 1, 2, 3$.
- Select time slices $s$ and $t$.
- Compute sample means, sample variances, and sample covariances for those time slices and compare against theoretically expected results.
Expected results for Brownian motion:

\[ E[B_t] = 0, \quad \text{Var}[B_t] = t, \quad \text{Cov}[B_s, B_t] = s \land t. \]

Expected results for zero-to-zero Brownian bridge \((s < t)\) to simplify notation:

\[ E[R_t] = 0, \quad \text{Var}[R_t] = t(1 - t), \quad \text{Cov}[R_s, R_t] = s(1 - t). \]

Expected results for \(\hat{x}\)-to-\(\hat{y}\) Brownian bridge \((s < t)\):

\[ E[R_t] = \hat{x} + \frac{t}{T} (\hat{y} - \hat{x}), \quad \text{Var}[R_t] = \frac{t(T - t)}{T}, \quad \text{Cov}[R_s, R_t] = \frac{s(T - t)}{T}. \]

Example with \(T =, \hat{x} = (2, 0, 0), \hat{y} = (-2, 0, 0), N_b = 1000, N_p = 1000, s = 0.004\) (i.e. index 1 of 1000), \(t = 2.0\) (i.e. index 500 of 1000):

<table>
<thead>
<tr>
<th>Actual</th>
<th>E[b(s)]</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Expected</td>
<td>E[b(s)]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Difference</td>
<td>E[b(s)]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Actual</td>
<td>E[b(t)]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Expected</td>
<td>E[b(t)]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Difference</td>
<td>E[b(t)]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

... 

<table>
<thead>
<tr>
<th>Actual</th>
<th>Var[b(s)]</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Expected</td>
<td>Var[b(s)]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Difference</td>
<td>Var[b(s)]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Actual</td>
<td>Var[b(t)]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Expected</td>
<td>Var[b(t)]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Difference</td>
<td>Var[b(t)]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Actual</th>
<th>Cov[b(s),b(t)]</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Expected</td>
<td>Cov[b(s),b(t)]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Difference</td>
<td>Cov[b(s),b(t)]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

5.3 Bridge results

Experimental results are discouraging. Performance requirements are too stiff for generation of Brownian bridges during Metropolis steps. To help this, one can (1) compute a database of zero-to-zero \(N_b\) Brownian bridges of \(N_p\) points each, and re-use this database for different \(\hat{x}, \hat{y}\). (2) Tabulate \(V\) off-line and interpolate at runtime.

- Dependence on \(N_b\): Increasing \(N_b\) decreases sampling variability of \(V\).
• Dependence on $N_p$: For small $N_p$, increasing $N_b$ only decreases sampling variability, but non-zero bias remains (vs. the integral and exact expressions, shown next). For the test case $r_1 = 1, r_2 = 1, \theta = \pi$, one needs $N_p$ on the order of 500,000 before $V$ begins to stabilize.

Interpretation: Note that $\Delta t = T/N_p$. Standard deviation of bridge steps is on the order of $\sqrt{T/N_p}$. For smaller $N_p$, bridges are too “hoppy” and miss the $a$-ball at the origin.

5.4 Integral expression for $V$

Ueltschi and Betz have recently found an approximation which is valid to low order in $a$:

$$V_2(\hat{x}, \hat{y}) = \frac{a}{\sqrt{8\pi\beta}} e^{\frac{-\|\hat{x} - \hat{y}\|^2}{8\beta}} \int_0^1 \frac{1}{[s(1-s)]^{3/2}} e^{-\frac{\|\hat{x}\|^2}{8\beta s}} e^{-\frac{\|\hat{y}\|^2}{8\beta(1-s)}} ds.$$ 

where, for notational convenience, we write

$$\hat{x} = x - x', \quad \hat{y} = y - y', \quad V_2(\hat{x}, \hat{y}) = V(x, y, x', y').$$

If $\|\hat{x}\| = \|\hat{y}\|$ then we have the exact expression

$$V_2(\hat{x}, \hat{y}) = \frac{2a}{\|x\|} e^{\frac{-\|\hat{x}\|^2}{8\beta \|x\|^2}} e^{-\frac{\|\hat{y}\|^2}{8\beta \|x\|^2}}.$$

This can be written in terms of the five real variables $r_1 = \|x\|, r_2 = \|y\|, \theta = \cos^{-1}(\langle x, y \rangle / \|x\| \|y\|), \beta$, and $a$.

5.5 Argument reduction

The potential $V$ depends on $d^4$ real variables: we have $V(x, y, x', y')$ where $x, x', y, y' \in \mathbb{R}^d$. Since only $x - x'$ and $y - y'$ appear in the formula, we can reduce to $d^2$ real variables: we have $V_2(\hat{x}, \hat{y})$ as above.

Using rotation and translation invariance of $V$, we can write down $V$ in terms of $r_1 = \|\hat{x}\|, r_2 = \|\hat{y}\|$, and angle $\theta$. Using the Law of Cosines, we have

$$\|\hat{x} - \hat{y}\|^2 = r_1^2 + r_2^2 - 2r_1r_2 \cos(\theta).$$

In particular, when $r_1 = r_2 = r$, we have

$$\|\hat{x} - \hat{y}\|^2 = 2r^2(1 - \cos(\theta)) = 4r^2 \sin^2(\theta/2).$$

Now $V$ depends only on three real variables. The integral expression is

$$V_2(r_1, r_2, \theta) = \frac{a}{\sqrt{8\pi\beta}} e^{\frac{-r_1^2 + r_2^2 - 2r_1r_2 \cos(\theta)}{8\beta}} \int_0^1 \frac{1}{[s(1-s)]^{3/2}} e^{-\frac{r_1^2}{8\beta s}} e^{-\frac{r_2^2}{8\beta(1-s)}} ds$$

and the exact expression, for $r_1 = r_2 = r$, is

$$V_2(r, r, \theta) = \frac{2a}{r} e^{-\frac{r^2(1 - \sin^2(\theta/2))}{8\beta}}.$$
5.6 Visualization

Here is a surface plot of $V(r, r, \theta)$ for $r$ from 1 to 4, $\theta$ from 0 to $\pi$, $\beta = 1$, and $a = 0.1$. Note that probability of intersecting the $a$-sphere decays as $r$ increases, and grows as $\theta$ runs from $0^\circ$ to $180^\circ$, as expected.
6 Metropolis-Hastings

6.1 Overview

The Metropolis-Hastings algorithm is a special case of Monte Carlo Markov chain (MCMC). It is best introduced by example: consider the 1D $N$-point Ising model.

- One has a system with multiple possible configurations. In the Ising model, the configuration space is $\Omega = \{\pm 1\}^N$, i.e. $N$ particles which may be in either an up (filled) or a down (hollow) state.

- A state is described by $\omega = (\omega_1, \ldots, \omega_n)$. The configuration space $\Omega$ has $2^N$ possible configurations.

- The system is endowed with an energy function. For the 1D Ising model, one has

$$H(\omega) = \sum_{i=1}^{n} \sum_{j=1}^{n} S_{ij} \omega_i \omega_j + \sum_{i=1}^{n} h_i \omega_i.$$  

where the $S_{ij}$’s are interaction terms (non-interacting, nearest neighbor, mean-field, etc.) and the $h_i$’s are magnetization terms.

- One picks an initial configuration. Typically, there are three choices: (1) Start with all spins down, i.e. $\omega = (-1, \ldots, -1)$. (2) Start with all spins up, i.e. $\omega = (+1, \ldots, +1)$. (3) Start with $\omega$ selected from a uniform probability distribution on $\Omega$.

- There is a temperature-related parameter $\beta$.

- One selects a site $i$ and decides whether to flip $\omega_i$ to $-\omega_i$.

- This decision is made using the Metropolis prescription, namely:
  - One computes the change in energy $\Delta H = H(\omega') - H(\omega)$ which would be obtained if $\omega$ were sent to $\omega'$ by flipping $\omega_i$.
  - One may compute $\Delta H$ by separately computing $H(\omega')$ and $H(\omega)$ and subtracting the two. However, since the only change is at the site $i$, one may do some ad-hoc algebra to derive an expression for $\Delta H$ which is less computationally expensive.
  - One accepts the change with probability

$$P(\text{change}) = \min\{1, e^{-\Delta H}\}$$

This is called a Metropolis step.

- Looping through all $n$ sites from $i = 1$ to $i = n$, performing a Metropolis step at each site $i$, is called a Metropolis sweep.
If one realizes a random variable $\theta(\pi)$ at each of $M$ sweeps, averaging $\theta$ over the $M$ sweeps, one obtains an approximation $\bar{\theta}$ for the expectation $E[\theta]$.

One should first run $L$ Metropolis sweeps of the system, discarding the realizations of the random variable $X$, before running the $M$ sweeps in which data are accumulated. The $L$ sweeps are called the thermalization phase; the $M$ sweeps are called the accumulation phase.

There is no general method to determine whether the system has thermalized (Kennedy); the underlying concern is the convergence of the Metropolis probability distribution to the stable distribution of its implicit Markov chain.

### 6.2 Metropolis for the random-cycle model

The random-cycle model is metropolized in a manner analogous to the 1D Ising model:

- The state space is $S_N$, the permutations on $N$ elements. It has size $N!$.
- The energy function is $H$ as described above.
- The initial configuration is found in one of two ways: (1) Start with an identity permutation. (2) Start with a permutation $\pi$ selected from a uniform probability distribution on $S_N$.
- There is a temperature-related parameter $\beta$.
- The analogue of conditionally flipping one of $N$ Ising spins is the following. One selects two of the $N$ points $x$ and $y$ and decides whether to send the old permutation $\pi$ to new permutation $\pi'$ via

\[
\pi : \begin{pmatrix} x & \pi^{-1}(y) & u & \cdots \\
\downarrow & \downarrow & \downarrow & \cdots \\
\pi(x) & y & \pi(u) & \cdots 
\end{pmatrix} \quad \pi' : \begin{pmatrix} x & \pi^{-1}(y) & u & \cdots \\
\downarrow & \downarrow & \downarrow & \cdots \\
y & \pi(x) & \pi(u) & \cdots 
\end{pmatrix}
\]

where $\pi'(u) = \pi(u)$ for all $u \neq x, \pi^{-1}(y)$.

- The energy change $\Delta H$ may be expensively computed by finding $H(\pi')$ and $H(\pi)$ and subtracting the two, but again, one may do some algebra to take advantage of the fact that most of the terms are identical. This computation is shown below.

- A single Metropolis step selects a site $x$ uniformly from the lattice. The site $y$ is selected uniformly from a Metropolis window: consider only sites $y$ within the radius $r$ such that $e^{-\beta r^2} \approx e^{-10}$, i.e. $r \approx 6\sqrt{3}$. (Selecting $y$ uniformly from the entire lattice gives too many rejected Metropolis steps, harming performance.)

- Looping through all $N$ sites $x$, performing a Metropolis step at each site $i$, is a Metropolis sweep.

- The principal random variable of interest is $\rho_{mn}$ as described above.

- Thermalization is detected as follows: smooth out $H$ over a sliding window of 100 Metropolis sweeps. Consider the system thermalized when this smoothed $H$ has reached 30 turning points. Heuristically, this is overly conservative (which is fine). It is better to run too many thermalization sweeps than too few.
6.3 $\Delta H$ for the non-interacting model

Recall that

$$H(\pi) = \frac{1}{4\beta} \sum_{u \in \Lambda} \|u - \pi(u)\|^2.$$  

For an implementation of a Metropolis algorithm, one wishes to compute the change in potential energy when one sends $\pi$ to $\pi'$ via

$\pi : (x \quad \pi^{-1}(y) \quad u \quad \cdots) \quad \pi' : (x \quad \pi^{-1}(y) \quad u \quad \cdots)$

where $\pi'(u) = \pi(u)$ for all $u \neq x, \pi^{-1}(y)$.

We have

$$\Delta H = \frac{\|x - y\|^2 - \|x - \pi(x)\|^2 + \|\pi^{-1}(y) - \pi(x)\|^2 - \|\pi^{-1}(y) - y\|^2}{4\beta}.$$  

6.4 $\Delta H$ for the $r_2$ model

The distance-dependent term is the same as in the non-interacting case. Additionally, we need to compute $\Delta r_2 = r_2(\pi') - r_2(\pi)$. There are three cases:

(1) $x \neq y$ and $x, y$ are in different cycles;
(2) $x \neq y$ and $x, y$ are in the same cycle;
(3) $x = y$.

In each case, it is easy to track the change $\Delta r_2$, without having to compute $r_2(\pi)$ and $r_2(\pi')$.

6.5 $\Delta H$ for the interacting model

The distance-dependent term is again the same as in the non-interacting case. Additionally, we need to compute $\Delta V$. Recall that

$$V(\pi) = \sum_{u < v} V(u, \pi(u), v, \pi(v))$$

where $u, v$ are lattice sites. When we send $\pi$ to $\pi'$ as above ($x \mapsto y$ replaces $x \mapsto \pi(x)$ etc.), we have

$$V_{\pi'} - V_{\pi} = \sum_{u < v} V(u, \pi'(u), v, \pi'(v)) - \sum_{u < v} V(u, \pi(u), v, \pi(v)).$$  

21
Since $\pi'(u) = \pi(u)$ for all $u \neq x, \pi^{-1}(y)$, and since the interaction potential satisfies the symmetry condition

$$V(x, y, x', y') = V(x', y', x, y),$$

we have simply

$$\Delta V = \sum_{u \neq x, u \neq \pi^{-1}(y)} \left( V(u, \pi(u), x, y) - V(u, \pi(u), x, \pi(x)) \right)$$

$$+ \sum_{u \neq x, u \neq \pi^{-1}(y)} \left( V(u, \pi(u), \pi^{-1}(y), \pi(x)) - V(u, \pi(u), \pi^{-1}(y), y) \right)$$

$$+ V(x, y, \pi^{-1}(y), \pi(x)) - V(x, \pi(x), \pi^{-1}(y), y).$$

## 7 Conclusions and further directions

### 7.1 Conclusions

- The $r_2$ model is easy to simulate. The $r_2$ term raises the critical temperature. One can quantify this dependence and verify it against the result of Betz and Ueltschi.

- Preliminary results show that in the full-interaction model, the critical temperature is also raised. Software optimization is currently in progress, so that more simulations may be done in a timely manner. Then, $T_c(a)$ may be plotted with confidence.

### 7.2 Further directions

- The cluster expansion is non-rigorous and needs further justification, in particular for non-lattice point distributions where inter-particle spacing can be small.

- Examine random variables other than $\rho_{mn}$.

- Use non-Gaussian weights for $d = 2$.

- Place the points not on a cubic lattice but distributed according to a point process; metropolize point positions as well as permutations. The correct point process for Bose-Einstein condensation is not known; it is known not to be Poisson.

- We can greatly increase system size by using parallelization: on a multiprocessor system, partition $\Lambda$ into subcubes. When $x, y$ are in the same subcube, computation is local; when $x$ is in one subcube and $y$ is in a neighbor, use message-passing.

- See what people come up with as $T_c(a)$ becomes better known . . . . Stay tuned for this as well!
## A Various energy functions

Here we tabulate, for handy reference, various energy functions used in [GRU] and [U07].

<table>
<thead>
<tr>
<th>Energy Function</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Permutation energy</td>
<td>$H_\Lambda(\pi) = \sum_{x \in \Lambda}</td>
</tr>
<tr>
<td>Permutation/point probability contribution</td>
<td>$Q(\pi, x) = e^{-\alpha</td>
</tr>
<tr>
<td>Permutation probability numerator</td>
<td>$P_\Lambda^*(\pi) = \prod_{x \in \Lambda} Q(\pi, x)$</td>
</tr>
<tr>
<td></td>
<td>$= \prod_{x \in \Lambda} e^{-\alpha</td>
</tr>
<tr>
<td></td>
<td>$= e^{-\alpha \sum_{x \in \Lambda}</td>
</tr>
<tr>
<td></td>
<td>$= e^{-\alpha H_\Lambda(\pi)}$</td>
</tr>
<tr>
<td>Partition function</td>
<td>$Z_\Lambda = \sum_{\pi \in B_\Lambda} P_\Lambda^*(\pi)$</td>
</tr>
<tr>
<td>Permutation probability</td>
<td>$P_\Lambda(\pi) = \frac{P_\Lambda^*(\pi)}{Z_\Lambda} = \frac{e^{-\alpha H_\Lambda(\pi)}}{Z_\Lambda}$</td>
</tr>
<tr>
<td>Probability the origin is in a cycle of length $k$</td>
<td>$P_\Lambda(\ell_0 = k) = \sum_{\pi \in B_\Lambda: \ell_0 = k} P_\Lambda(\pi)$</td>
</tr>
<tr>
<td></td>
<td>$P(\ell_0 = k) = \lim_{\Lambda \to \mathbb{Z}^d} P_\Lambda(\ell_0 = k)$</td>
</tr>
<tr>
<td>Probability the origin is in an infinite cycle</td>
<td>$\phi(\alpha) = 1 - \sum_{k=1}^{\infty} P(\ell_0 = k)$</td>
</tr>
<tr>
<td>Thermodynamic potential</td>
<td>$f_\Lambda(\alpha) = \frac{\log(Z_\Lambda)}{</td>
</tr>
<tr>
<td></td>
<td>$f(\alpha) = \lim_{\Lambda \to \mathbb{Z}^d} f_\Lambda(\alpha)$</td>
</tr>
</tbody>
</table>
B An overview of the Metropolis-Hastings algorithm

Here we summarize the Metropolis-Hastings algorithm for handy reference, with no attempt to prove correctness. More thorough discussions of the algorithm may be found in (for example) GS, Hua, Law, and Mac.

Metropolis-Hastings is perhaps best introduced by example.

- One has a system with multiple possible configurations. Specifically, one may think of the one-dimensional Ising model. This is \( \Omega = \{ \pm 1 \}^n \), i.e. \( n \) particles which may be in either an up or a down state. A state is described by
  \[ \omega = (\omega_1, \ldots, \omega_n). \]
  Here, the state space \( \Omega \) has \( 2^n \) possible configurations.

- The system is endowed with an energy function. For the 1D Ising model, one has
  \[ E(\omega) = \sum_{i=1}^{n} \sum_{j=1}^{n} S_{ij} \omega_i \omega_j + \sum_{i=1}^{n} h_i \omega_i, \]
  where the \( S_{ij} \)'s are interaction terms and the \( h_i \)'s are magnetization terms.

- One picks an initial configuration. Typically, there are three choices: (1) Start with all spins down, i.e. \( \omega = (-1, \ldots, -1) \). (2) Start with all spins up, i.e. \( \omega = (+1, \ldots, +1) \). (3) Start with \( \omega \) selected from a uniform probability distribution on \( \Omega \).

- There is a system temperature \( \beta \).

- One selects a site \( i \) and decides whether to flip \( \omega_i \) to \(-\omega_i\). This decision is made using the Metropolis prescription, namely:
  - One computes the change in energy \( \Delta E = E(\omega') - E(\omega) \) which would be obtained if \( \omega \) were sent to \( \omega' \) by flipping \( \omega_i \).
  - One may compute \( \Delta E \) by separately computing \( E(\omega') \) and \( E(\omega) \) and subtracting the two. However, since the only change is at the site \( i \), one may do some algebra to derive an expression for \( \Delta E \) which is less computationally expensive.
  - One accepts the change with probability
    \[ \min\{1, e^{-\beta \Delta E}\}. \]
    This is called a Metropolis step.

- Looping through all \( n \) sites from \( i = 1 \) to \( i = n \), performing a Metropolis step at each site \( i \), is called a Metropolis sweep.

- If one realizes a random variable \( X(\omega) \) at each of \( M \) sweeps, averaging \( X \) over the \( M \) sweeps, one obtains an approximation \( \overline{X} \) for the expectation \( E[X] \).

- One should first run \( L \) Metropolis sweeps of the system, discarding the realizations of the random variable \( X \), before running the \( M \) sweeps in which data are accumulated. The \( L \) sweeps are called the thermalization phase; the \( M \) sweeps are called the accumulation phase. There is no general method to determine whether the system has thermalized (Ken); the underlying concern is the convergence of the Metropolis probability distribution to the stable distribution of its implicit Markov chain. Techniques for thermalization for the random-cycle model, which is the subject of this paper, are presented in section 6.
References


[BY] Burdzy, M. and Yor, M. Personal communication.


[Ken] Kennedy, T. Personal communication.


Index

A
accumulation phase ........................................ 24

C
configurations .................................................. 24

E
energy function .................................................. 24

I
initial configuration ............................................. 24
Ising model ....................................................... 24

M
Metropolis prescription ......................................... 24
Metropolis step ................................................... 24
Metropolis sweep ................................................ 24

S
site ................................................................. 24
step .............................................................. 19 20 24
sweep ........................................................... 19 20 24
system ............................................................ 24

T
temperature ....................................................... 24
thermalization phase ........................................... 24