# Markov-chain Monte Carlo: A modern primer

## Lecture 2: Surprises
## Part 2.2: Perfect sampling

Werner Krauth
Département de physique, Ecole normale supérieure
Paris, France

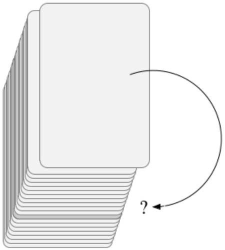
A Set of Lectures
University of Kent
Canterbury, Great Britain

14-17 November 2022

# References

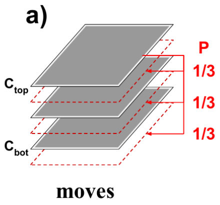

- J. G. Propp, D. B. Wilson **Exact sampling with coupled Markov chains and applications to statistical mechanics** `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.27.1022`
- W. Krauth "**Statistical Mechanics: Algorithms and Computations**" (Oxford University Press, 2006)

# Shuffling of cards 1/5



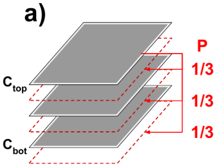

- $\Omega_n^{\text{shuffle}} = \{\text{Permutations of } \{1, \ldots, n\}\}$
- For $n = 3$: $\Omega_3^{\text{shuffle}} = \{1 \equiv \{1,2,3\}, 2 \equiv \{1,3,2\}, 3 \equiv \{2,1,3\}, 4 \equiv \{2,3,1\}, 5 \equiv \{3,1,2\}, 6 \equiv \{3,2,1\}\}$.
- $\pi^{t=0} = \delta(\{1, , \ldots, n\})$ (perfectly ordered set)

# Shuffling of cards 2/5




- $\Omega_3^{\text{shuffle}} = \{1 \equiv \{1,2,3\}, 2 \equiv \{1,3,2\}, 3 \equiv \{2,1,3\}, 4 \equiv \{2,3,1\}, 5 \equiv \{3,1,2\}, 6 \equiv \{3,2,1\}\}$.
- $$P = \frac{1}{3}\begin{pmatrix} 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}$$

# Shuffling of cards 3/5



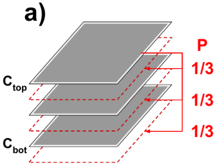

**moves**

**procedure** `top-to-random`
**input** $\{c_1, \ldots, c_n\}$
$i \leftarrow \texttt{choice}(\{1, \ldots, n\})$
$\{\hat{c}_1, \ldots, \hat{c}_n\} \leftarrow \{c_2, \ldots, c_i, c_1, c_{i+1}, \ldots, c_n\}$
**output** $\{\hat{c}_1, \ldots, \hat{c}_n\}$

---

- Insert upper card ($c_1$) after card $i$ and before card $i+1$
- NB: if $i = 1$, put it back on top.

# Shuffling of cards 4/5

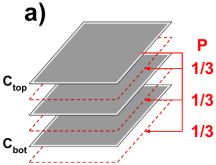



**moves**

```
procedure top2random-stop
input {c_1, ..., c_n}
c_first-n ← c_n
for t = 1, 2, ... do
    c̃_1 ← c_1
    {c_1, ..., c_n} ← top2random({c_1, ..., c_n})
    if (c̃_1 = c_first-n) break
output {c_1, ..., c_n, t}
```

- Perfect sample (!).
- Expected running time: $n \log n$.

# Shuffling of cards 5/5

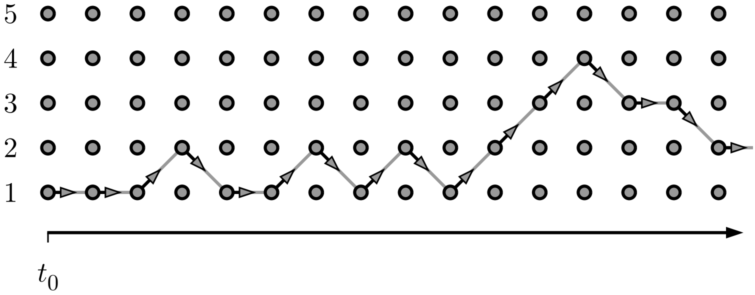



**moves**

**procedure** direct-shuffle
**input** $\{c_1, \ldots, c_n\}$
**for** $t = 1, \ldots, n$ **do**
$$\begin{cases} i \leftarrow \texttt{choice}(\{n-t+1, \ldots, n\}) \\ \{c_1, \ldots, c_n\} \leftarrow \{c_2, \ldots, c_i, c_1, c_{i+1}, \ldots, c_n\} \end{cases}$$
**output** $\{c_1, \ldots, c_n\}$

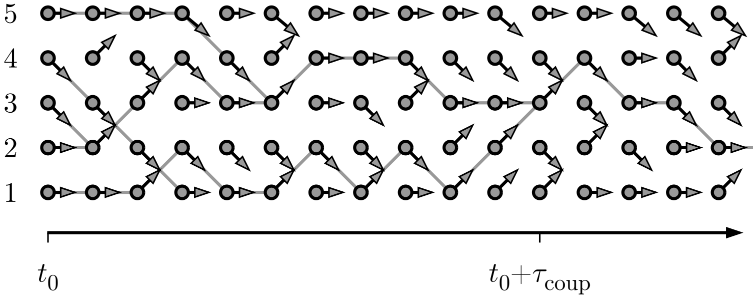
- Running time: *n*.
- Running time: *n*.
- Standard algorithm for generating random permutations.

# Markov chain (traditional view)




- Configuration $c_t$, move $\delta_t$.
- Set $t_0 = 0$.

# Markov chain (random maps), coupling 1/4



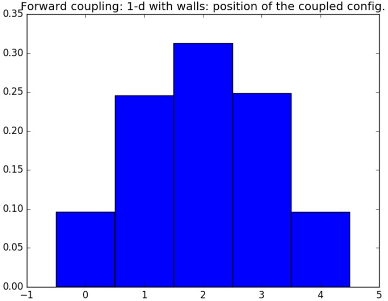

- Each configuration has its move at each time step.
- Coupling (Doeblin, 1930s).

# Markov chain (random maps), coupling 2/4

```
pos=[]
for stat in range(10000):
   posit=set(range(N))
   for t in range(1000000):
      posit = set([min(max(b + random.randint(-1, 1), 0), N - 1) for b in posit])
      if len(posit) == 1: break
   pos.append(posit.pop())
```

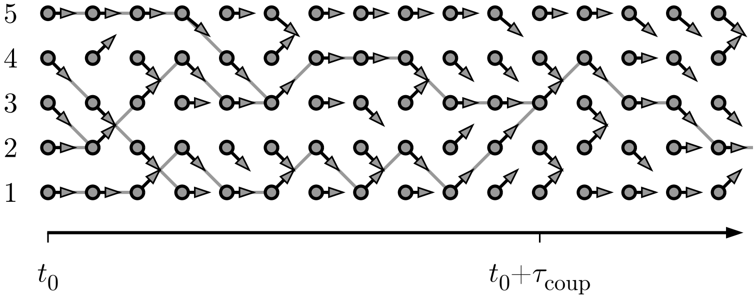
- Position of coupling not uniform.
- Coupling time larger than mixing time.

# Markov chain (random maps), coupling 3/4



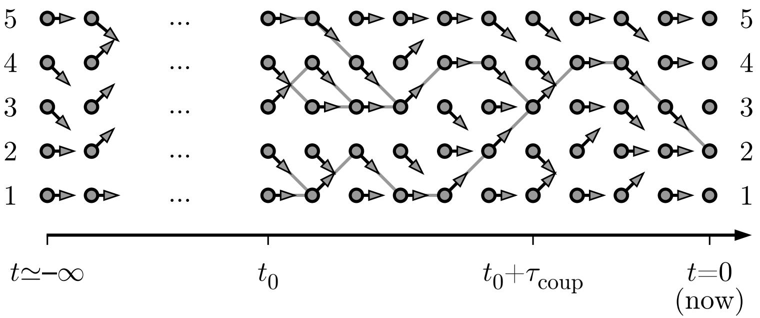

- Histogram of coupling position.

# Markov chain (random maps), coupling 4/4




- Each configuration has its move at each time step.
- Coupling (Doeblin, 1930s).

# Coupling from the past 1/8



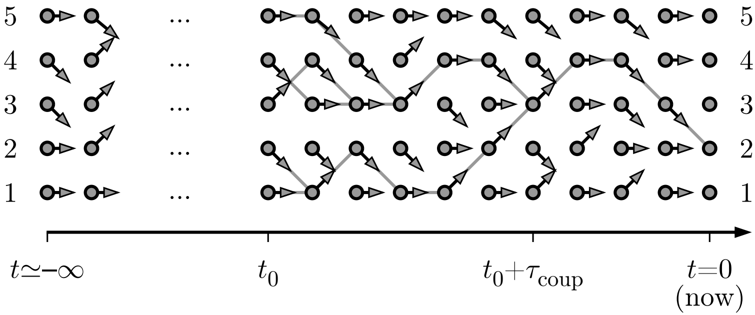

- Starting an MCMC simulation at $t = -\infty$
- Propp & Wilson (1997)

# Coupling from the past 2/8

```
pos = []
for statistic in range(100000):
    all_arrows = {}
    time_tot = 0
    while True:
        time_tot -= 1
        arrows = [random.randint(-1, 1) for i in range(N)]
        if arrows[0] == -1: arrows[0] = 0
        if arrows[N - 1] == 1: arrows[N - 1] = 0
        all_arrows[time_tot]=arrows
        positions=set(range(0, N))
        for t in range(time_tot, 0):
            positions = set([b + all_arrows[t][b] for b in positions])
            if len(positions) == 1: break
        if len(positions) == 1: break
```

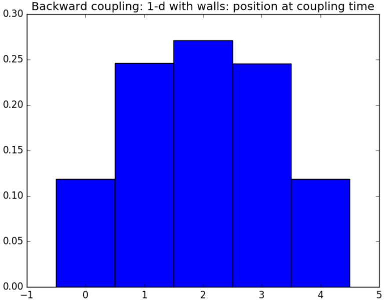
- Starting an MCMC simulation at $t = -\infty$
- Propp & Wilson (1997)

# Coupling from the past 3/8




- Starting an MCMC simulation at $t = -\infty$
- Propp & Wilson (1997)

# Coupling from the past 4/8



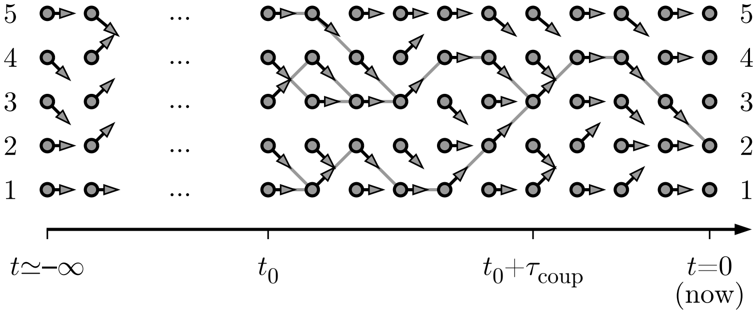

- Coupling position (in the past) non-uniform)

# Coupling from the past 5/8

```
for statistic in range(10000):
   all_arrows = {}
   time_tot = 0
   while True:
      time_tot -= 1
      old_pos = set(range(0, N))
      arrows = [random.randint(-1, 1) for i in range(N)]
      if arrows[0] == -1: arrows[0] = 0
      if arrows[N - 1] == 1: arrows[N - 1] = 0
      all_arrows[time_tot] = arrows
      positions = set(range(N))
      for t in range(time_tot, 0):
         positions = set([b + all_arrows[t][b] for b in positions])
      if len(positions) == 1: break
   a=positions.pop()
   pos.append(a)
```

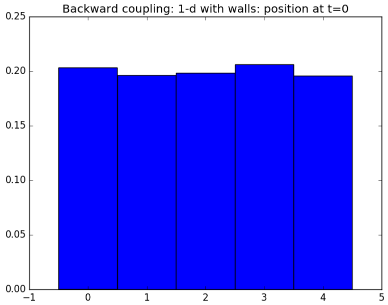
- Dictionary of random maps going back in time.

# Coupling from the past 6/8

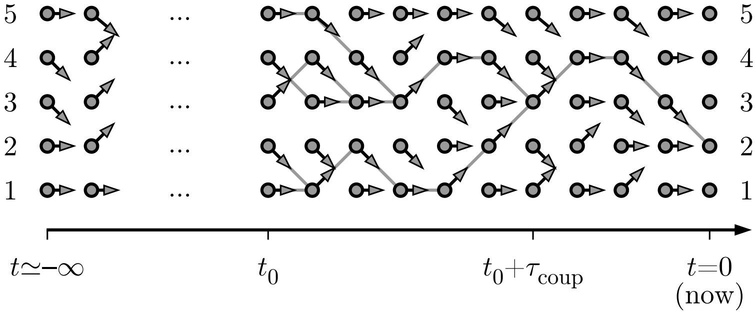



- Starting an MCMC simulation at $t = -\infty$
- Propp & Wilson (1997)

# Coupling from the past 7/8

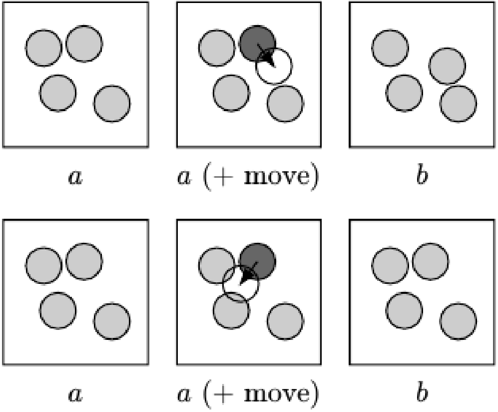



- Perfect sample at $t = 0$, starting from $t = -\infty$
- Propp & Wilson (1997)

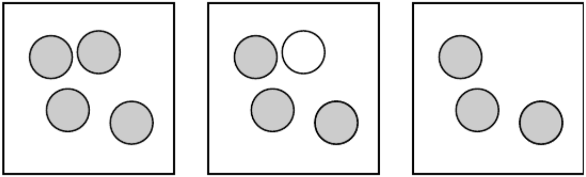



- Try it yourself!

# Hard-sphere simulation (traditional)



# Hard-sphere simulation (birth-and-death)




$$Z = \sum_{N=0}^{\infty} \lambda^N \int \cdots \int \mathrm{d}x_1 \ldots \mathrm{d}x_N \pi(x_1, \ldots, x_N)$$

- $\pi(a) = \lambda \pi(b)$
- Death probability (per particle, per time interval): $1 \mathrm{d}t$
- Birth probability (per unit square): $\lambda \mathrm{d}t$

# Poisson distribution

Poisson distribution (number $n$ of events per unit time):

$$\pi_{\Delta t=1}(n) = \frac{\lambda^n \mathrm{e}^{-\lambda}}{n!}$$

Poisson distribution (number $n$ of events per time $\mathrm{d}t$):

$$\pi_{\mathrm{d}t}(n) = \frac{(\lambda \mathrm{d}t)^n \mathrm{e}^{-\lambda \mathrm{d}t}}{n!} \implies \pi_{\mathrm{d}t}(1) = \lambda \mathrm{d}t, \pi_{\mathrm{d}t}(2) = 0$$

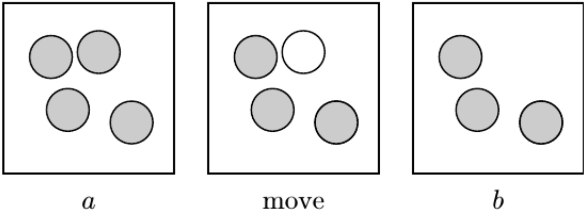
Poisson waiting time: Probability that next event after time $t$:

$$\mathbb{P}(t) = (1 - \lambda \mathrm{d}t), \ldots, (1 - \lambda \mathrm{d}t)\lambda \mathrm{d}t$$

$$\mathbb{P}(t) = \underbrace{\overbrace{(1 - \lambda \mathrm{d}t) \to (1 - \lambda \mathrm{d}t)}^{\sum \mathrm{d}t = t}}_{\mathrm{e}^{-\lambda t}} \lambda \mathrm{d}t$$

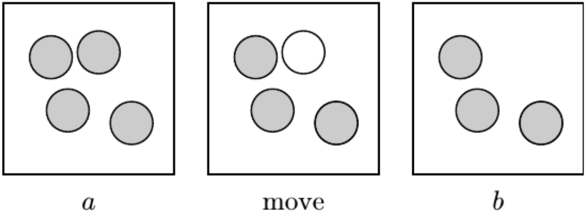
...can be sampled with $t = (-\log \mathrm{ran}[0, 1])/\lambda$

# Birth-and-death (principle 1)




- $N$ spheres, each of them may die.
- a new sphere may be born (but there may be problems).
- rate for next event: $N + \lambda$.
- $\mathbb{P}$(death) $\propto N$ and $\mathbb{P}$(birth) $\propto \lambda$, reject if overlap.

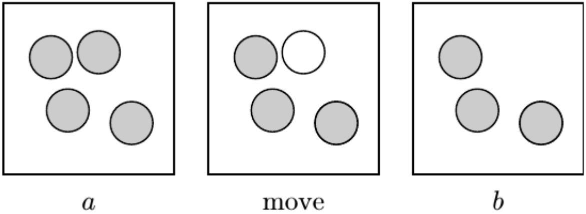
# Birth-and-death (implementation 1)




- start with $N = 0$ spheres
- Go to next-event time : $-\log \text{ran} /(N + \lambda)$ (in steps of 1)
- sample random number $\text{ran}[0, 1]$: if smaller than $\lambda/(\lambda + N)$: add a disk (reject if overlap), otherwise delete a disk.

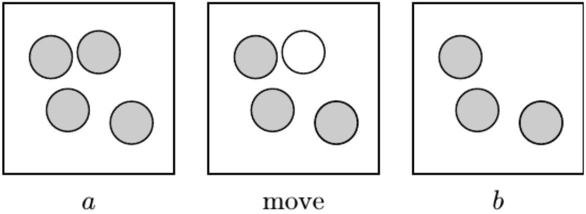
NB: Check configuration at integer time steps, for sampling.

# Birth-and-death (principle 2)



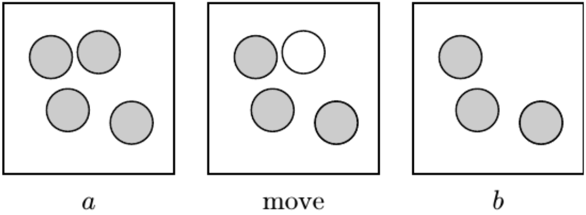

- $N$ spheres, each of them knows when it will die (sad) rate=1.
- a new sphere may be born (but there may be problems) rate $= \lambda$.

# Birth-and-death (implementation 2)




- start with $N = 0$ spheres.
- Advance to next birth time : $-\log \text{ran}[0, 1]/\lambda$ (in steps of 1).
- If no rejection, install death time $-\log \text{ran}[0, 1]$

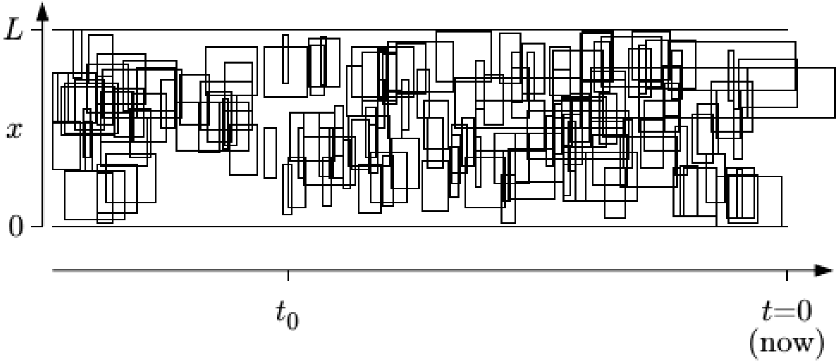
# Birth-and-death (principle 3)



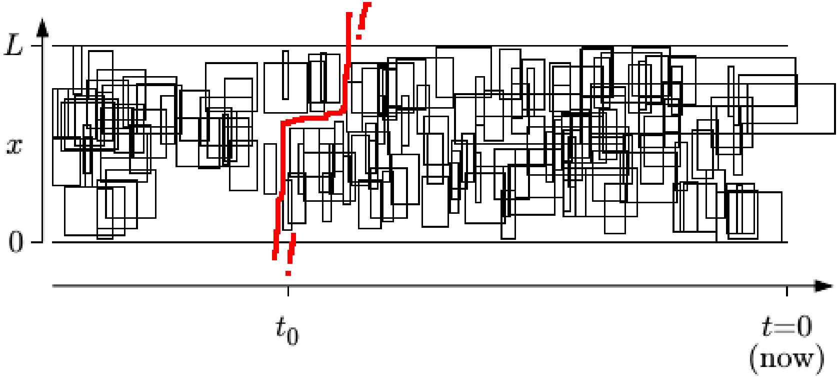

- Hyptothetical spheres are born with rate $= \lambda$, and they die with rate 1.

Check later whether all this pans out correctly.

# Birth-and-death (implementation 3)

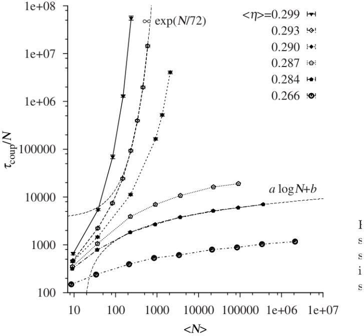


# Birth-and-death (implementation 3)



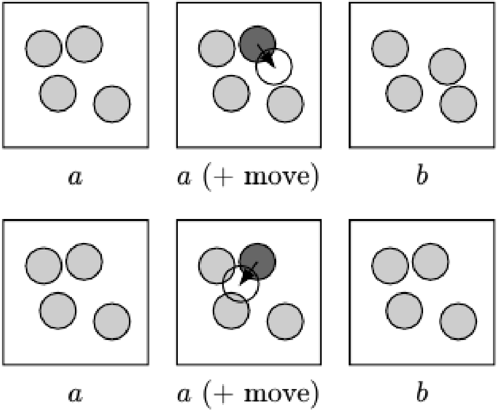

- Can be made into a perfect sampling algorithm

# Birth-and-death (implementation 3)



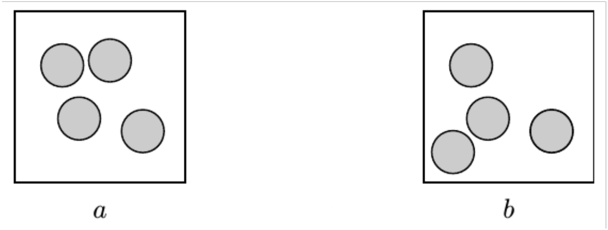

- Bernard et al. (2010)
- Dynamical phase transition

# Hard-sphere simulation (traditional)



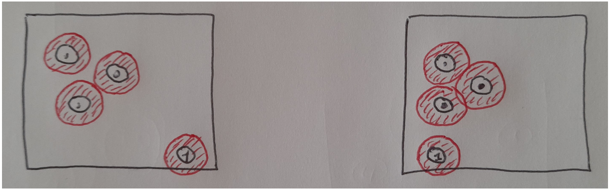

Algorithm remains correct if displacement random in box.

# Path coupling 1/4




- At low density, any two configurations of spheres $a$ and $z$ can be connected through a path of length $< 2N$ as follows: $a \to b \to c \to ..... \to z$, where any two neighbors differ only in 1 sphere.
- MC algorithm: Take random sphere, place it at random position anywhere in the box.

# Path coupling 2/4



- MC algorithm: Take random sphere, place it at the same random position for both copies.
- $p(1 \to 0)$: Pick 1, move to where it fits in both copies

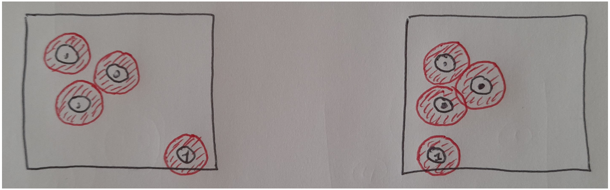
$$p(1 \to 0) \geq \frac{1}{N}\left[1 - \frac{N-1}{N} 4\eta\right]$$

- $p(1 \to 2)$: Pick $2 \ldots N$ move near to $1_A$ or $1_B$.

$$p(1 \to 2) \leq \frac{N-1}{N}\left[\frac{8}{N}\eta\right]$$

- $\implies$ for $\eta < 1/12$: further coupling likely.

# Path coupling 3/4



- MC algorithm: Take random sphere, place it at the same random position for both copies.
- $p(1 \to 0)$: Pick 1, move to where it fits in both copies

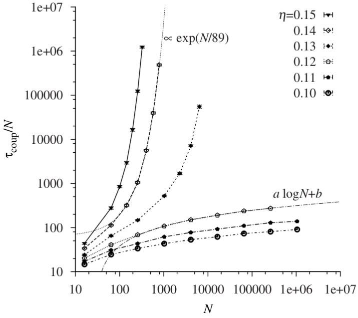
$$p(1 \to 0) \geq \frac{1}{N}\left[1 - \frac{N-1}{N} 4\eta\right]$$

- $p(1 \to 2)$: Pick $2 \ldots N$ move near to $1_A$ or $1_B$.

$$p(1 \to 2) \leq \frac{N-1}{N}\left[\frac{8}{N}\eta\right]$$

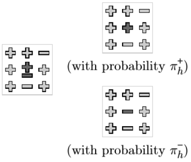
- $\Longrightarrow$ for $\eta < 1/12$: further coupling likely.

# Path coupling 4/4




- Bernard et al. (2010)
- Damage-spreading dynamical phase transition

# Ising model—perfect sampling (1/4)

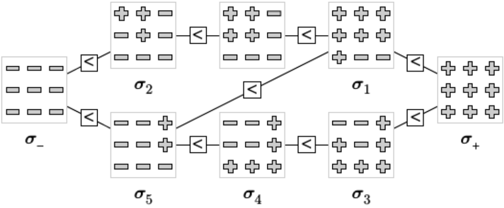



$$\pi_h^+ = \frac{e^{-\beta E^+}}{e^{-\beta E^+} + e^{-\beta E^-}} = \frac{1}{1 + e^{-2\beta h}},$$

$$\pi_h^- = \frac{e^{-\beta E^-}}{e^{-\beta E^+} + e^{-\beta E^-}} = \frac{1}{1 + e^{+2\beta h}}.$$

- Heatbath algorithm

# Ising model—perfect sampling (2/4)

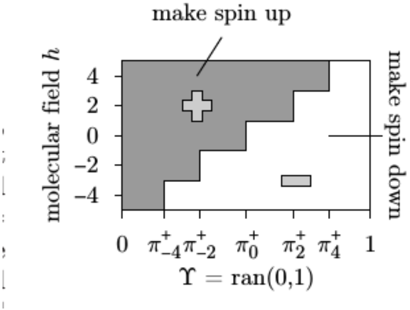



$$\pi_h^+ = \frac{e^{-\beta E^+}}{e^{-\beta E^+} + e^{-\beta E^-}} = \frac{1}{1 + e^{-2\beta h}},$$

$$\pi_h^- = \frac{e^{-\beta E^-}}{e^{-\beta E^+} + e^{-\beta E^-}} = \frac{1}{1 + e^{+2\beta h}}.$$

- Half order among all configurations in the Ising model

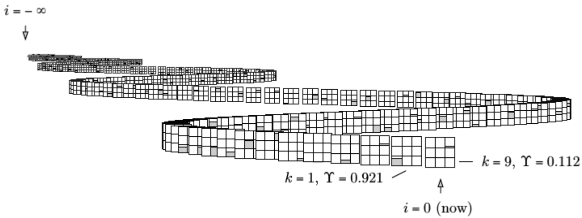
# Ising model—perfect sampling (3/4)




$$\pi_h^+ = \frac{e^{-\beta E^+}}{e^{-\beta E^+} + e^{-\beta E^-}} = \frac{1}{1 + e^{-2\beta h}},$$

$$\pi_h^- = \frac{e^{-\beta E^-}}{e^{-\beta E^+} + e^{-\beta E^-}} = \frac{1}{1 + e^{+2\beta h}}.$$

# Ising model—perfect sampling (4/4)




- Ising-model simulation that has run since time $i = -\infty$.
- Allows us to produce perfect samples.