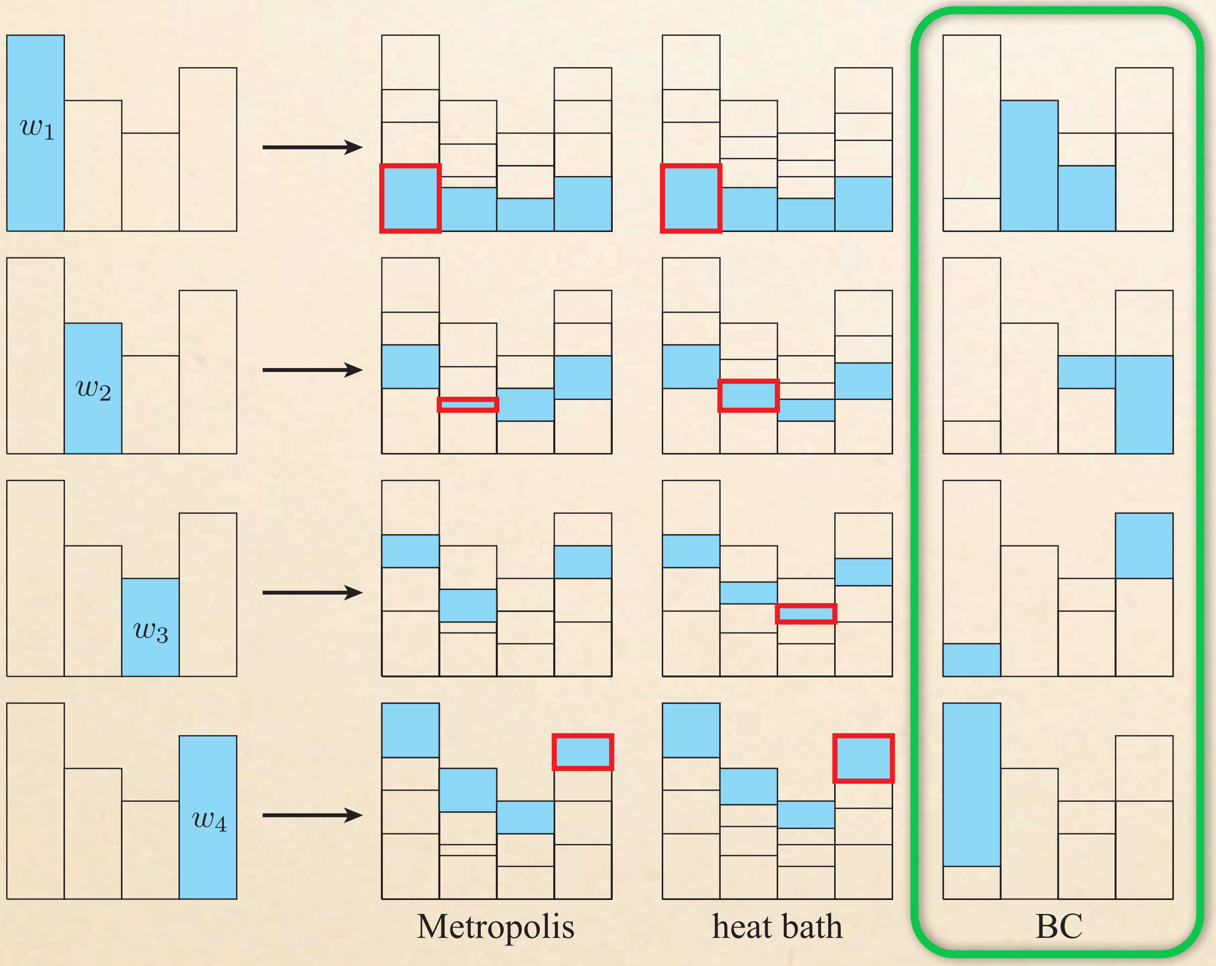
Monte Carlo Algorithms in Statistical Physics
@Univ. Melbourne Jul. 27, 2010

Markov Chain Monte Carlo without Detailed Balance and Bounce-free Worm Algorithm



Department of Applied Physics, University of Tokyo

Hidemaro Suwa ,

Synge Todo

Conditions to Equilibrium State

- The balance condition (BC)

$$W(c_i) = \sum_j W(c_i)P(c_i \to c_j) = \sum_j W(c_j)P(c_j \to c_i) \ \ \forall i$$

- The ergodicity $\exists k \in N \ \ P^k > 0$

- The MCMC with these two conditions ensures that any initial condition converges to the equilibrium distribution (Perron-Frobenius theorem).

Meyn and Tweedie (1994) Tierney (1994)

- In most practical implementations,

the detailed balance condition (DBC), the reversibility,

$$W(c_i)P(c_i \to c_j) = W(c_j)P(c_j \to c_i) \ \ \forall i, j$$

is imposed as a sufficient condition to the balance condition.

- Thanks to the DBC, it becomes easy to find qualified transition probabilities.

- Note that the DBC is generally not a necessary condition for the BC, that is, the invariance of the target distribution.

Sequential Update Breaks DBC

- Even in the single spin flip update, a sequential update where spins are swept in a fixed order breaks the DBC ! Menousiouthakis *et al.* (1999)

Example :
the sequential updates in 2 Ising spin system

Weights

$\uparrow\ \uparrow \quad p$

$\downarrow\ \uparrow \quad 1-p$

In a fixed order $\longrightarrow$

$$P(c_i \to c_j) = \frac{W(c_j)}{\sum_k W(c_k)}$$

Heat bath algorithm (Gibbs sampler)

$$\downarrow\ \uparrow \longrightarrow \uparrow\ \uparrow \longrightarrow \uparrow\ \uparrow$$

$$\uparrow\ \uparrow \longrightarrow \downarrow\ \uparrow \longrightarrow \downarrow\ \uparrow$$

$$(1-p) \times p \times p = p^2(1-p)$$

$$\neq$$

$$p \times (1-p) \times (1-p) = p(1-p)^2$$

$$P(c_i \to c_j) = \min\left(1, \frac{W(c_j)}{W(c_i)}\right)$$

Metropolis (Metropolis-Hastings) algorithm

$$\downarrow\ \uparrow \longrightarrow \uparrow\ \uparrow \longrightarrow \uparrow\ \uparrow$$

$$\uparrow\ \uparrow \longrightarrow \downarrow\ \uparrow \longrightarrow \downarrow\ \uparrow$$

In the case of $p > 1/2$

$$(1-p) \times 1 \times \left(1 - \frac{1-p}{p}\right) = \frac{(1-p)(2p-1)}{p}$$

$$\neq$$

$$p \times \frac{1-p}{p} \times 0 = 0$$

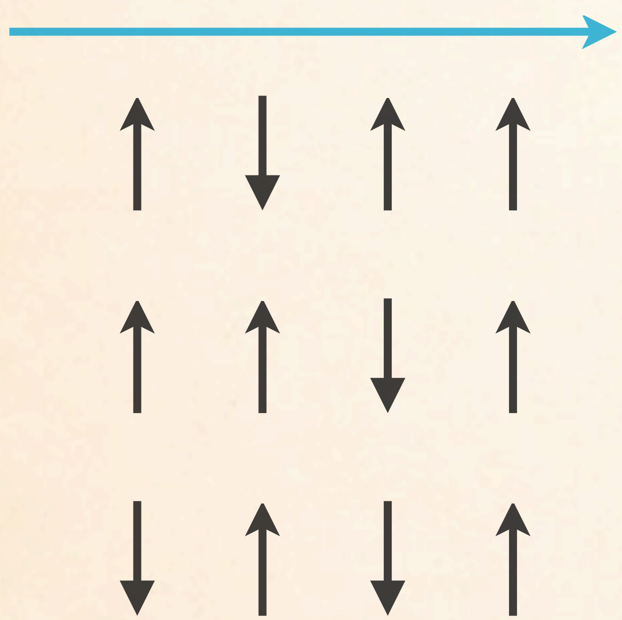
Local DBC in Sequential Update

- In the sequential update, the BC is fulfilled, which is demonstrated by satisfying *local* DBC.

Local DBC

$$W(c_i)P_t(c_i \to c_j) = W(c_j)P_t(c_j \to c_i) \quad \forall\, i,\, j$$

Update in a fixed order



$$\begin{aligned} P(c \to c') &= \sum_{\{c_1\}} \cdots \sum_{\{c_{N-1}\}} P_1(c \to c_1) P_2(c_1 \to c_2) \cdots P_N(c_{N-1} \to c') \\ &= \sum_{\{c_1 \cdots c_{N-1}\}} T^{(N)}_{N,N-1} \cdots T^{(2)}_{2,1}\, T^{(1)}_{1,0} \\ &= \left(T^{(N)} \cdots T^{(2)}\, T^{(1)} \right)_{N,0}, \end{aligned}$$

$$\begin{aligned} \underbrace{\sum_{\{c_N\}} W(c_N) P(c_N \to c_0)}_{\text{BC}} &= \sum_{\{c_1 \cdots c_N\}} T^{(N)}_{0,1} \cdots T^{(2)}_{N-2,N-1}\, \boxed{T^{(1)}_{N-1,N}\, W(c_N)} \\ &= W(c_0) \sum_{\{c_1 \cdots c_N\}} T^{(N)}_{1,0} \cdots T^{(2)}_{N-1,N-2}\, T^{(1)}_{N,N-1} \\ &\underline{= W(c_0),} \end{aligned}$$

$$\sum_{\{c_1 \cdots c_N\}} T^{(N)}_{1,0} \cdots T^{(2)}_{N-1,N-2}\, T^{(1)}_{N,N-1} = \sum_{\{c_N\}} \left(T^{(1)}\, T^{(2)} \cdots T^{(N)} \right)_{N,0} = 1.$$

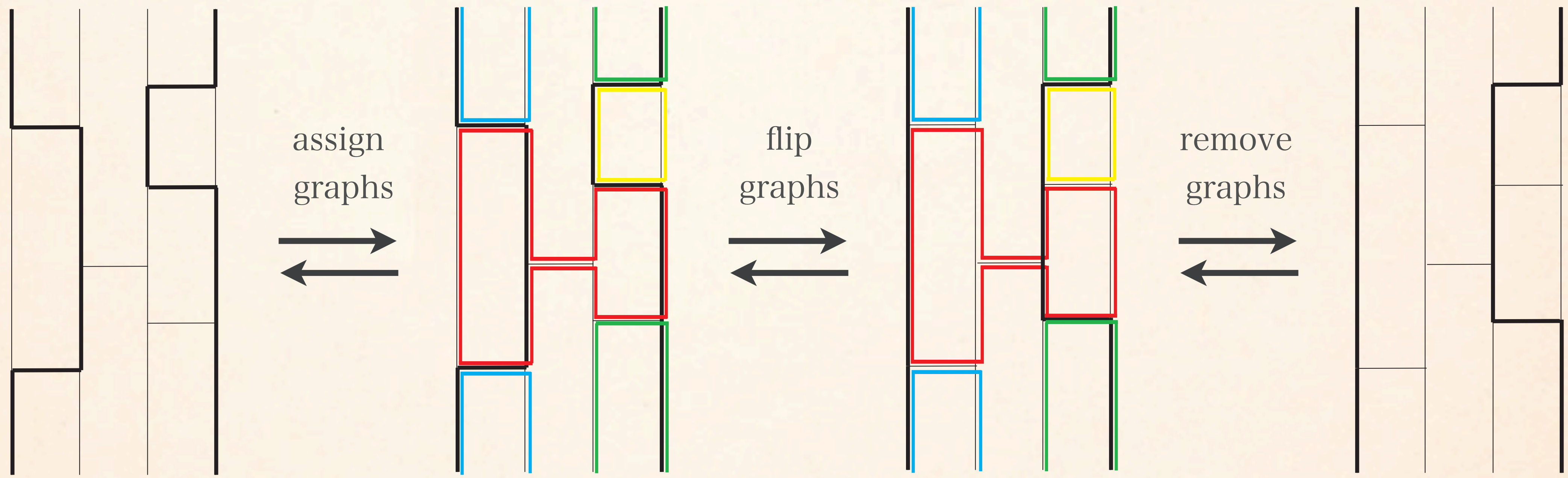
Menousiouthakis *et al.* (1999)

Variants of MCMC

- In order to shorten autocorrelation times, a number of efficient methods have been invented, such as the cluster algorithms, e.g.,

the Swendsen-Wang algorithm Swendsen & Wang (1987)

and the loop algorithm in quantum systems. Evertz *et al.* (1993)



Here the red one is flipped.

- The extended ensemble methods, for example,

the multicanonical method Berg at el. (1992), and

the exchange Monte Carlo method Hukushima & Nemoto (1996)

have been applied to various hardly-relaxing problems e.g., protein folding problems and spin glasses.

Optimization of Probabilities

Metropolized sampling Liu (1996)

$$T^{MG}_{ij} = \begin{bmatrix} 0 & \frac{\pi_2}{1-\pi_1} & \frac{\pi_3}{1-\pi_1} & \cdots & \frac{\pi_n}{1-\pi_1} \\ \frac{\pi_1}{1-\pi_1} & 1-\ldots & \frac{\pi_3}{1-\pi_2} & \cdots & \frac{\pi_n}{1-\pi_2} \\ \frac{\pi_1}{1-\pi_1} & \frac{\pi_2}{1-\pi_2} & 1-\ldots & \cdots & \frac{\pi_n}{1-\pi_3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\pi_1}{1-\pi_1} & \frac{\pi_2}{1-\pi_2} & \frac{\pi_3}{1-\pi_3} & \cdots & 1-\ldots \end{bmatrix}$$

Locally optimal update (LOU) Pollet *et al.* (2004)

$$T^{Opt}_{ij} = \begin{bmatrix} 0 & \frac{W_2}{W_1}y_1 & \frac{W_3}{W_1}y_1 & \cdots & \frac{W_n}{W_1}y_1 \\ y_1 & 0 & \frac{W_3}{W_2}y_2 & \cdots & \frac{W_n}{W_2}y_2 \\ y_1 & y_2 & 0 & \cdots & \frac{W_n}{W_3}y_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_1 & y_2 & y_3 & \cdots & 1-y_1-y_2-\ldots \end{bmatrix}$$

$$T_{ij} = P(c_i \rightarrow c_j)$$

$$T^{MG}_{ij} = \min\left(\frac{\pi_j}{1-\pi_i}, \frac{\pi_j}{1-\pi_j}\right)$$

$$\pi_i = \frac{Wi}{\sum_j W_j}$$

$$y_1 = \frac{\pi_1}{1-\pi_1}$$

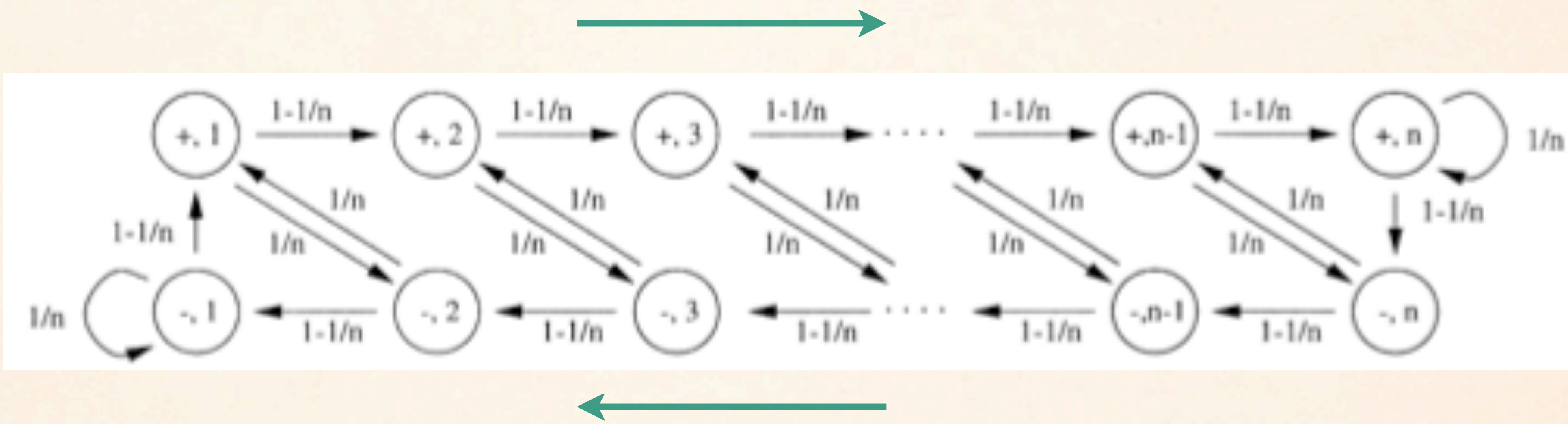
$$y_2 = \frac{(1-y_1)\pi_2}{1-\pi_1-\pi_2}$$

$$y_3 = \frac{(1-y_1-y_2)\pi_3}{1-\pi_1-\pi_2-\pi_3}$$

- Some optimizations of probabilities have been proposed within the DBC. However, some rejection remains in most cases.

Nonreversible Random Walk

・In the mean time, there was a report that a nonreversible one-dimensional random walk that breaks the DBC explored the distribution faster than the ordinary diffusive random walk. Diaconis *et al.* (2000)



Stochastic Flow

The total variation distance

$$\|P_a^N - W\|_{TV} = \frac{1}{2}\sum_{x\in X}|P_a^N(x) - W(x)| \to 0 \quad \sim C^{N/n} \qquad 0 < C < 1$$

The χ^2 distance

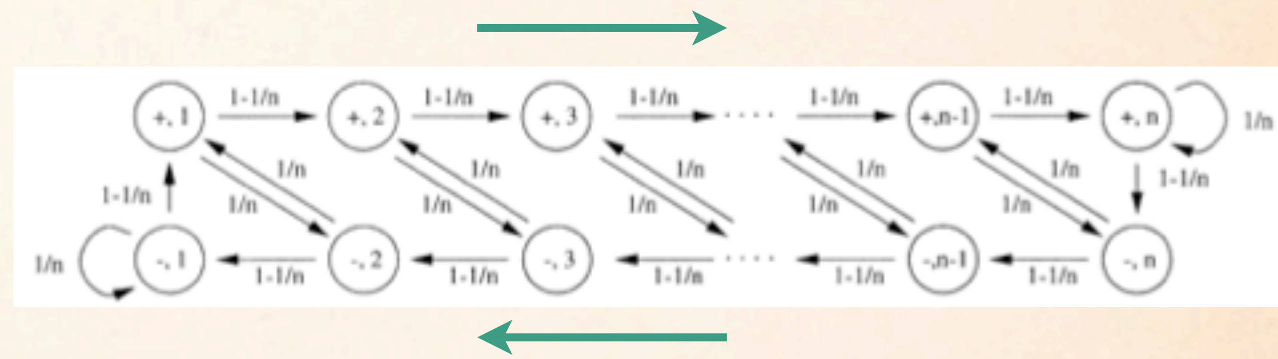
These distances converge faster than n² steps.

$$\chi^2(N) = \max_x \sum_y \frac{(P_x^N(y) - W(y))^2}{W(y)} = \max_x \left\| \frac{P_x^N}{W} - 1 \right\|_2^2 \to 0 \quad \sim C^{N/n \ln n}$$

Advantages of BC against DBC

- Existence of stochastic flow

When the DBC is broken, a net stochastic flow definitely exists and will push forward a Markov sequence to other configurations, as the example of nonreversible random walk.



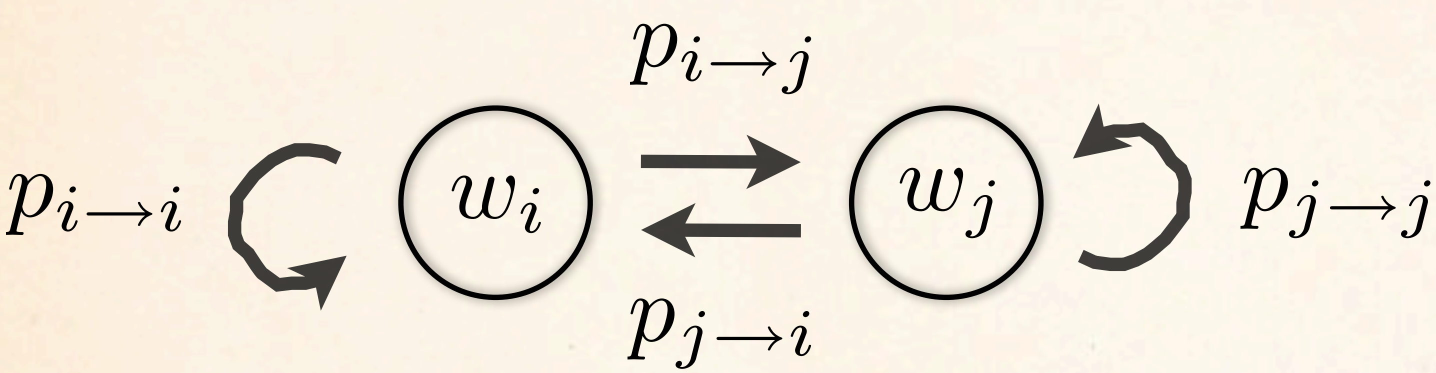
- Lower rejection rates

Since the BC is generally a weaker condition than the DBC, it becomes possible to reduce rejection rates by breaking the DBC.

$$T_{ij}^{Opt} = \begin{bmatrix} 0 & \frac{W_2}{W_1}y_1 & \frac{W_3}{W_1}y_1 & \cdots & \frac{W_n}{W_1}y_1 \\ y_1 & 0 & \frac{W_3}{W_2}y_2 & \cdots & \frac{W_n}{W_2}y_2 \\ y_1 & y_2 & 0 & \cdots & \frac{W_n}{W_3}y_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_1 & y_2 & y_3 & \cdots & 1-y_1-y_2-\ldots \end{bmatrix}$$

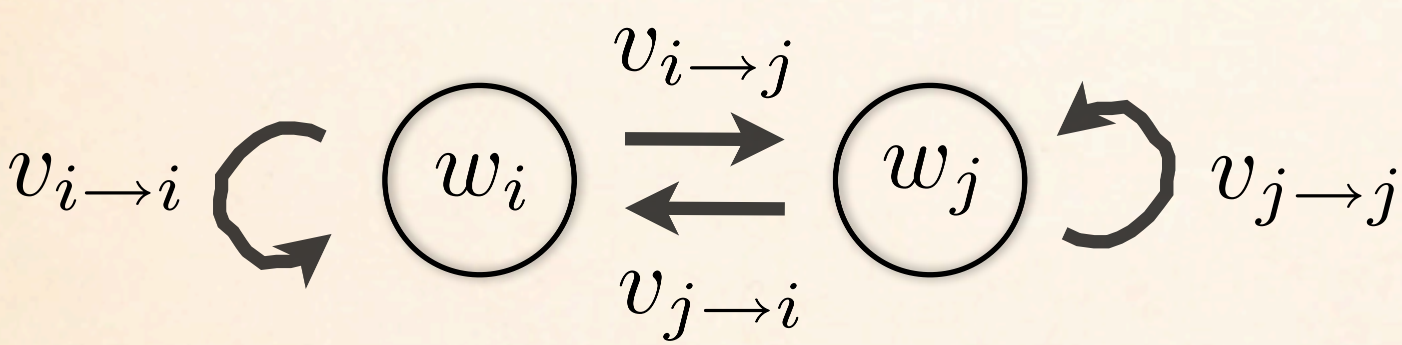
Stochastic Flow

We introduce a quantity $v_{i\to j} := w_i\, p_{i\to j}$, (raw) stochastic flow.



The BC in the conventional form

$$w_i = \sum_{j=1}^{n} w_i\, p_{i\to j} = \sum_{j=1}^{n} w_j\, p_{j\to i}$$



$$p_{i\to j} = v_{i\to j}/w_i$$

Law of probability conservation

$$w_i = \sum_{j=1}^{n} v_{i\to j} \quad \forall i \qquad (1)$$

BC expressed by stochastic flow

$$w_j = \sum_{i=1}^{n} v_{i\to j} \quad \forall j \qquad (2)$$

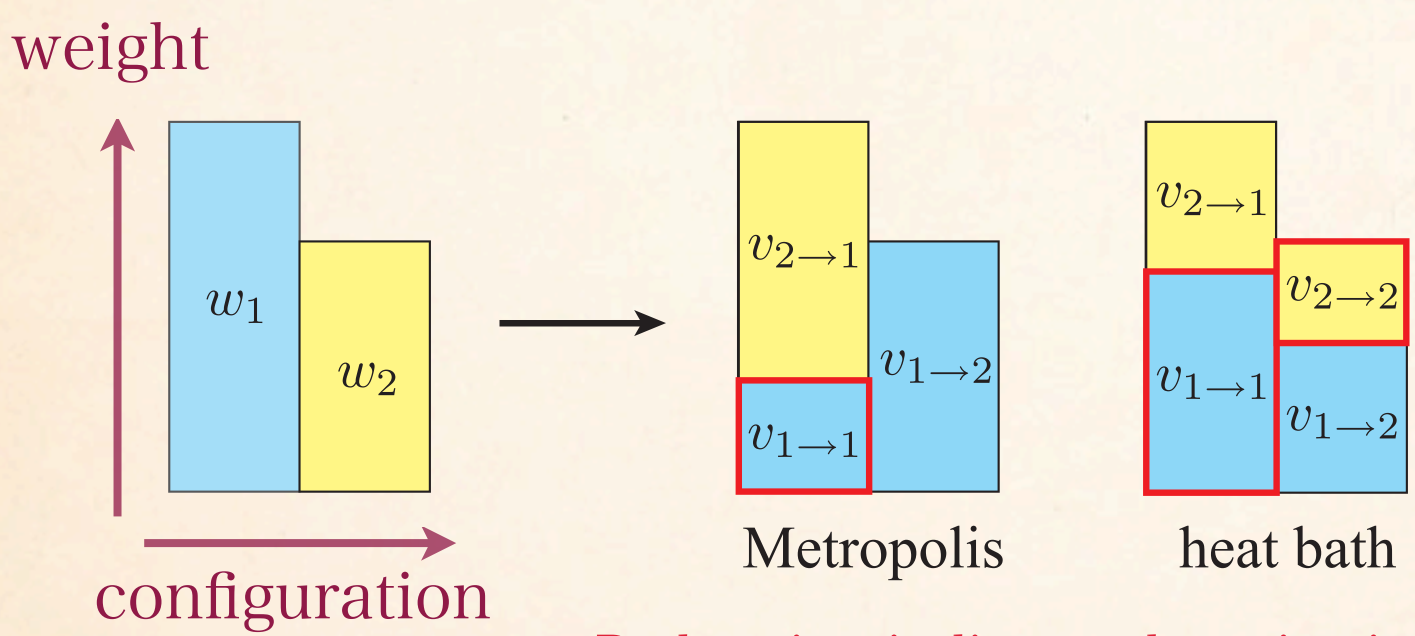
- Our task is to find a set $\{v_{i\to j}\}$ that minimizes the average rejection rate while satisfying Eqs. (1) and (2).

Weight Landfill

- We introduce a new picture, *weight landfill,* in order to visualize the concept of the BC.

Each weight is allocated to elsewhere (box).

Consider the case with n=2, e.g., Ising model.



Red region indicates the rejection rate.

Law of probability conservation

$$w_i = \sum_{j=1}^{n} v_{i\to j} \quad \forall i \qquad (1)$$

BC expressed by stochastic flow

$$w_j = \sum_{i=1}^{n} v_{i\to j} \quad \forall j \qquad (2)$$

Metropolis

$$v_{i\to j} = \frac{1}{n-1}\min(w_i, w_j) \qquad i \neq j$$

Heat bath

$$v_{i\to j} = \frac{w_i w_j}{\sum_{k=1}^{n} w_k} \qquad \forall i, j$$

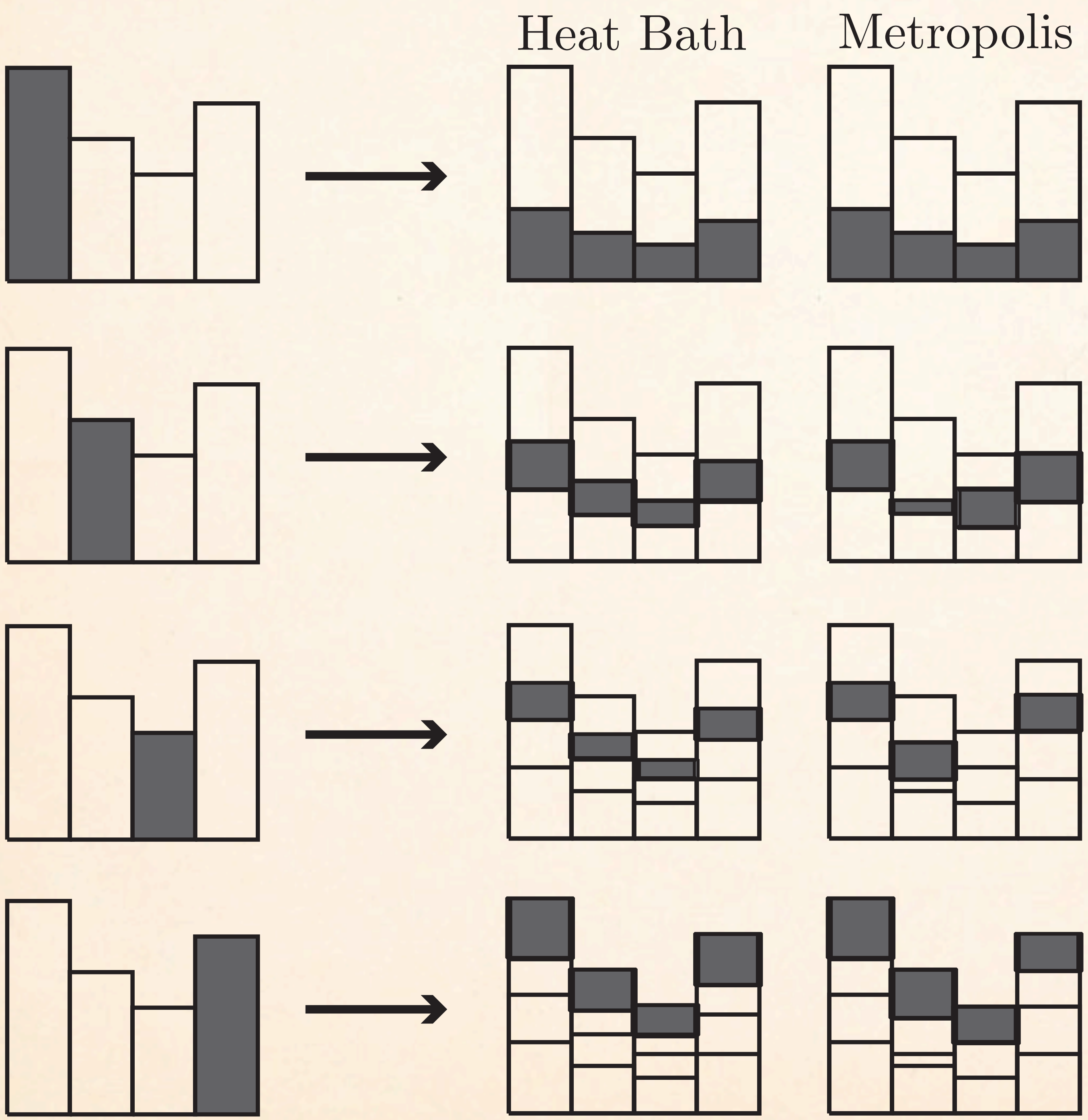
$$p_{i\to j} = v_{i\to j}/w_i$$

After allocating all weights (landfill), the total shape of boxes is not changed!

DBC : $v_{i\to j} = v_{j\to i}$

Landfill with n=4

- Let us landfill weights in a 4-candidate case, like 4-state Potts model, by the conventional algorithms.



How complicated these landfills are !

Is there any simpler algorithm?

Yes!

Metropolis

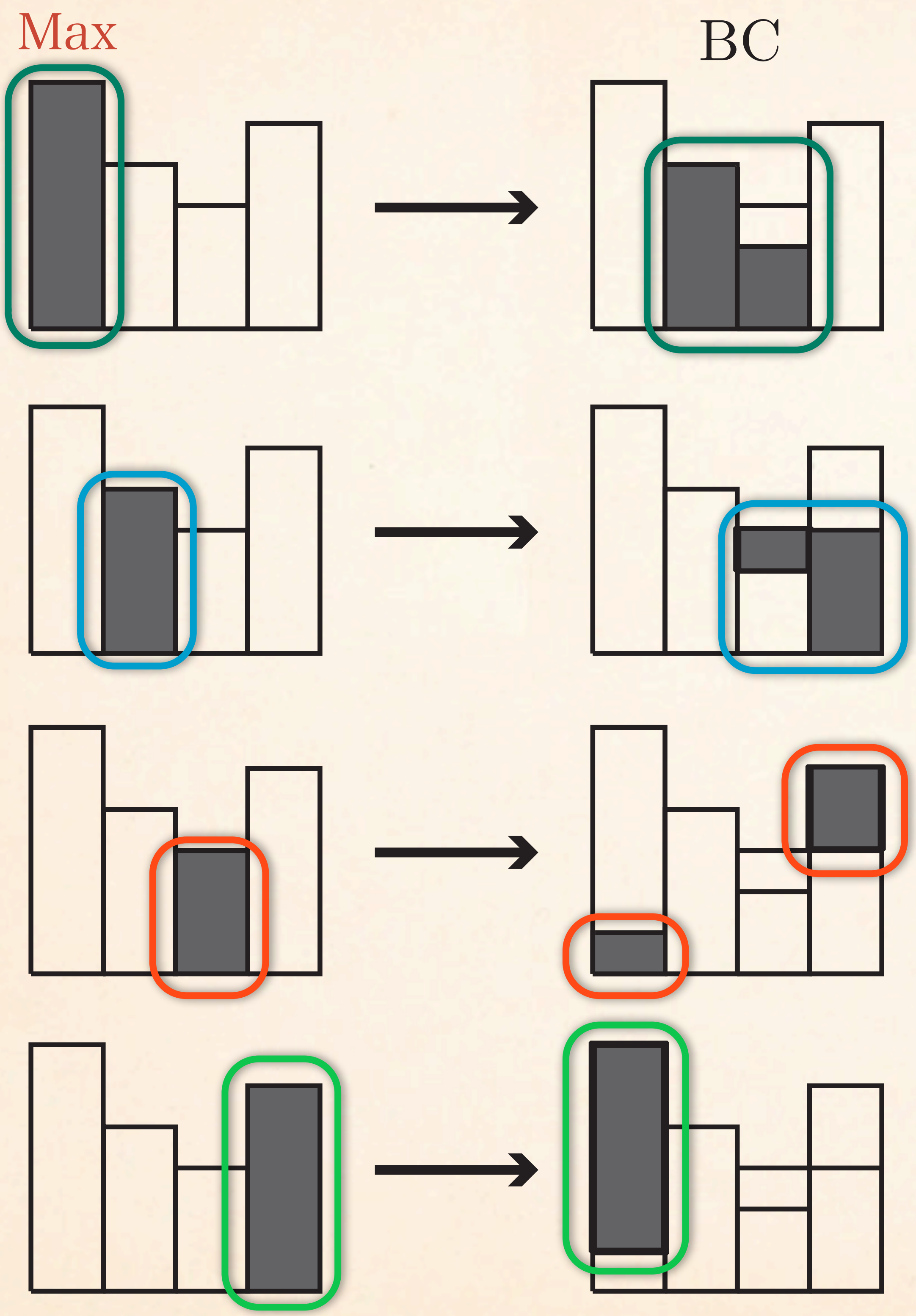
$$v_{i\to j} = \frac{1}{n-1}\min(w_i, w_j) \qquad i \neq j$$

Heat bath

$$v_{i\to j} = \frac{w_i w_j}{\sum_{k=1}^{n} w_k} \qquad \forall i,\, j$$

New Algorithm

H.S. and Synge Todo
arXiv:1007.2262



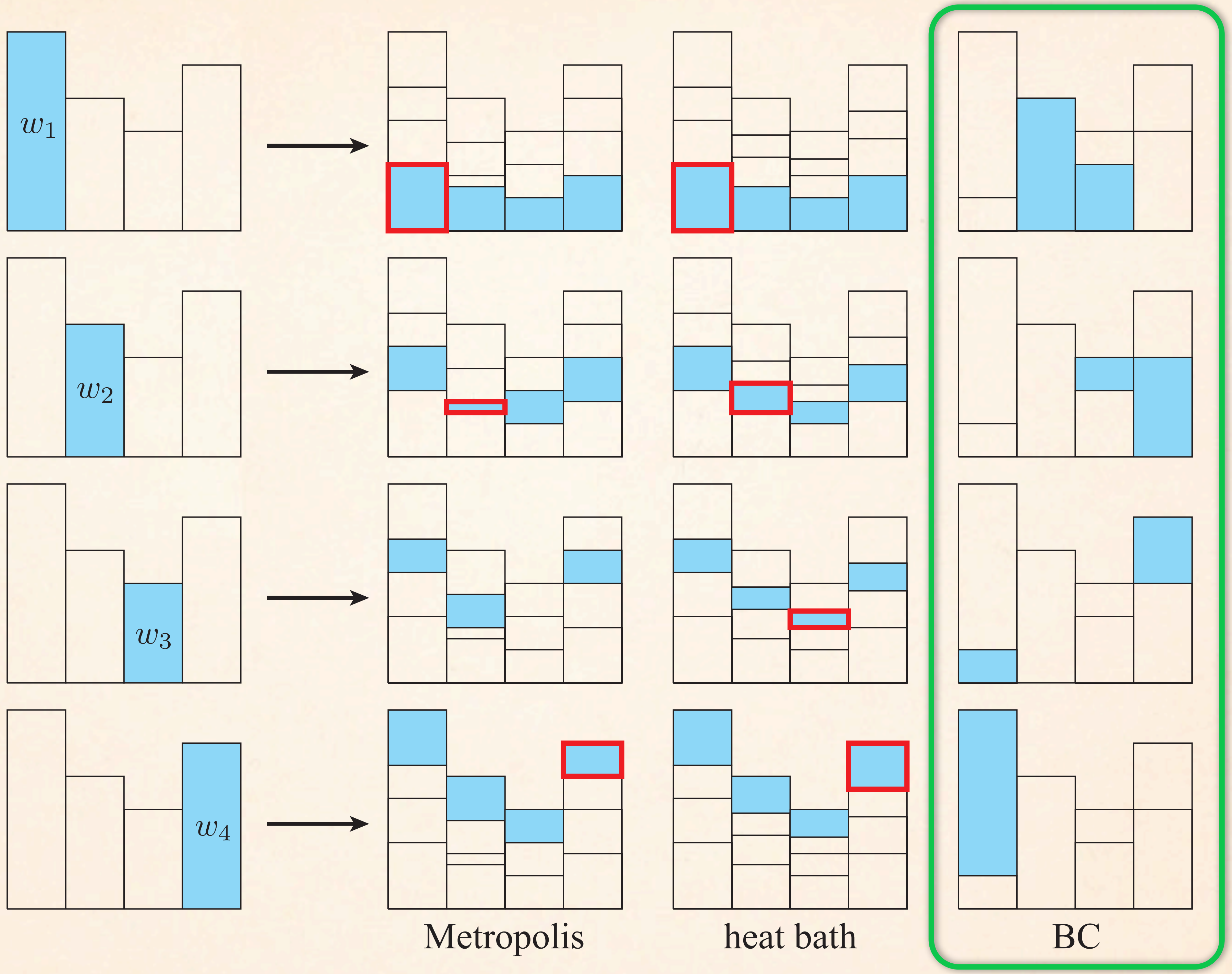
1. We start at a configuration which has a maxmum weight.
2. Allocate the weight to the next box. If some remain, reallocate it to the subsequent box.
3. Allocate the weight of the first landfilled box to the subsequent position of the last landfill. If some remain, reallocate it similarly.
4. Continue to allocate all weights likewise, keeping the starting order to the landfilled order.
5. Define transition probabilities as,

$$p_{i \to j} = v_{i \to j} / w_i$$

- There is NO self-allocated weight !

Rejection-free Algorithm!

Comparison with Conventional Methods



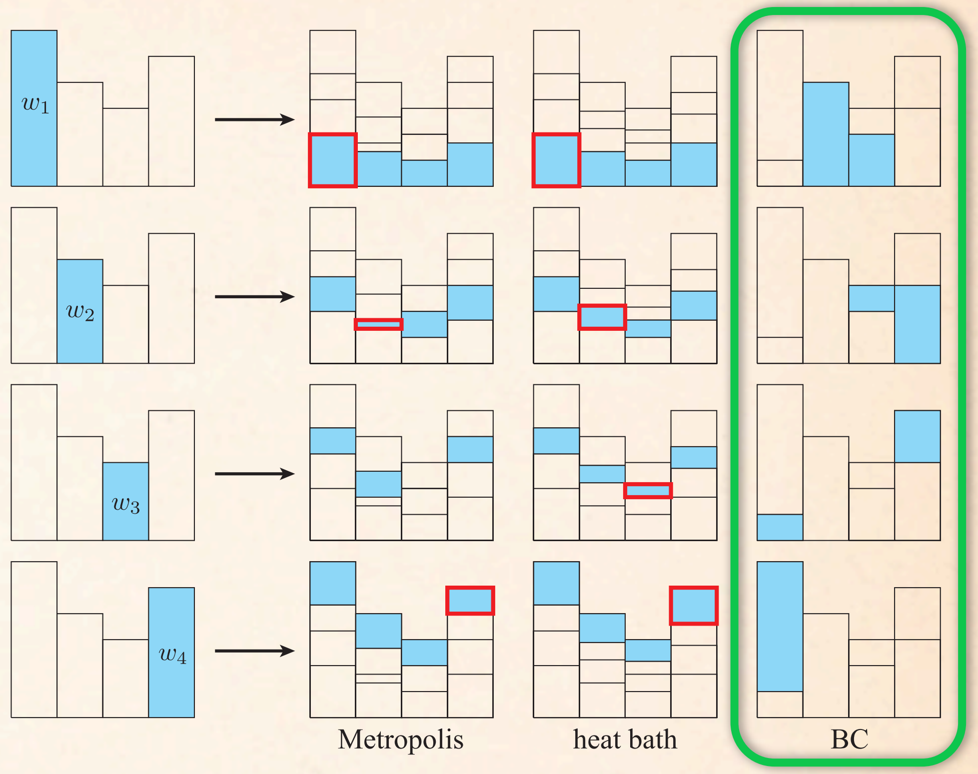
- It is clear that our algorithm is simpler and indeed accomplishes the rejection-free.
- It is possible to generate a random variable with given probability in *O(1)* CPU time, according to Walker's method of alias. *Fukui & Todo (2009)*

Formula of BC Algorithm

$$\Delta_{ij} := S_i - S_{j-1} + w_1 \qquad 1 \le i,\, j \le n$$

$$S_i := \sum_{k=1}^{i} w_k \qquad 1 \le i \le n$$

$$S_0 := S_n.$$



$$p_{i\to j} = v_{i\to j}/w_i$$

$$\boxed{v_{i\to j} = \max(0,\, \min(\Delta_{ij},\, w_i + w_j - \Delta_{ij},\, w_i,\, w_j)),}$$

$$v_{i\to i} = \begin{cases} \max(0,\, 2w_1 - S_n) & i = 1 \\ 0 & i \ge 2. \end{cases}$$

Rejection-free (bounce-free) condition $w_1 \le \frac{S_n}{2} \equiv \frac{1}{2}\sum_{k=1}^{n} w_k$

Acceleration in Potts Model

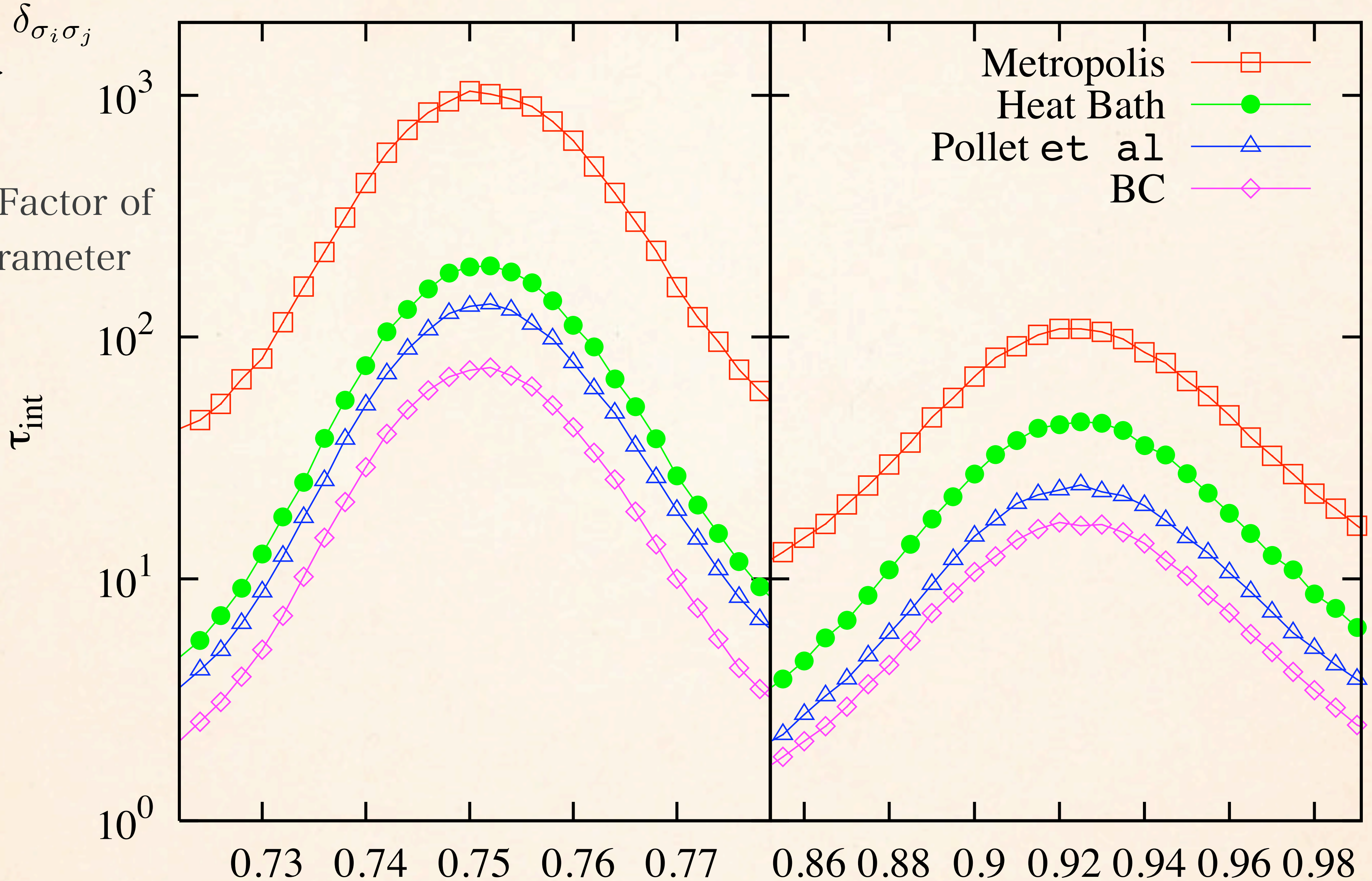
Potts model *Wu* (1982)

$$H = -\sum_{<i,j>} \delta_{\sigma_i \sigma_j}$$

Square lattice $L = 16$

Sequential update

- Structure Factor of Order Parameter



$T_c \simeq 0.744904$ $T_c \simeq 0.910239$

The BC is the fastest.

	8-state	4-state
Metropolis	× 14	× 6.4
Heat bath	× 2.6	× 2.7
LOU	× 1.8	× 1.4

Review of QMC

- The continuous imaginary time path integral representation is nothing but a perturbation expansion using the Feynman path integral, and is free from the decomposition error.

Beard *et al.* (1996)

$$H = H_0 + V \qquad E_i = \langle \alpha_i | H_0 | \alpha_i \rangle \qquad V = \sum_i V_{l_i}$$

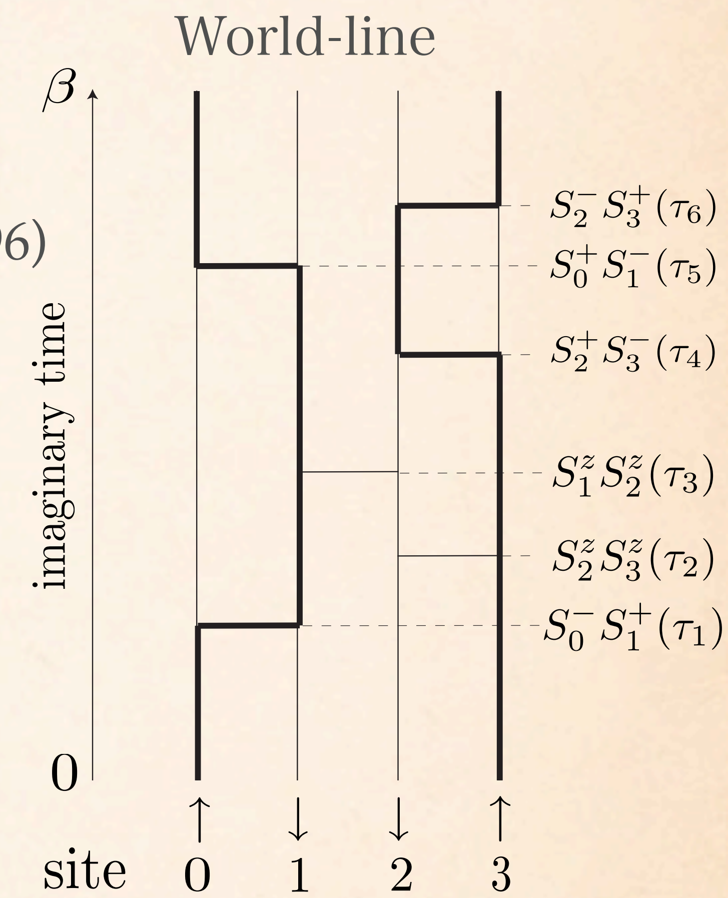
$$Z = \mathrm{tr} e^{-\beta H} = \mathrm{tr}\left[T_\tau \exp\left(-\int_0^\beta d\tau V(\tau) \right) e^{-\beta H_0} \right]$$

$$= \sum_{\alpha_1} \langle \alpha_1 | e^{-\beta E_1} | \alpha_1 \rangle$$

$$+ \sum_{n=1}^{\infty} \sum_{(\alpha_1,\ldots,\alpha_n)} \sum_{(l_1,\ldots,l_n)} \int_0^\beta d\tau_1 \int_{\tau_1}^\beta d\tau_2 \cdots \int_{\tau_{n-1}}^\beta d\tau_n$$

weight

$$\boxed{e^{-\tau_1 E_1} \langle \alpha_1 | -V_{l_1} | \alpha_2 \rangle e^{-(\tau_2 - \tau_1) E_2} \cdots \langle \alpha_n | -V_{l_n} | \alpha_1 \rangle e^{-(\beta - \tau_n) E_1}}$$



$$H_0 = 0$$

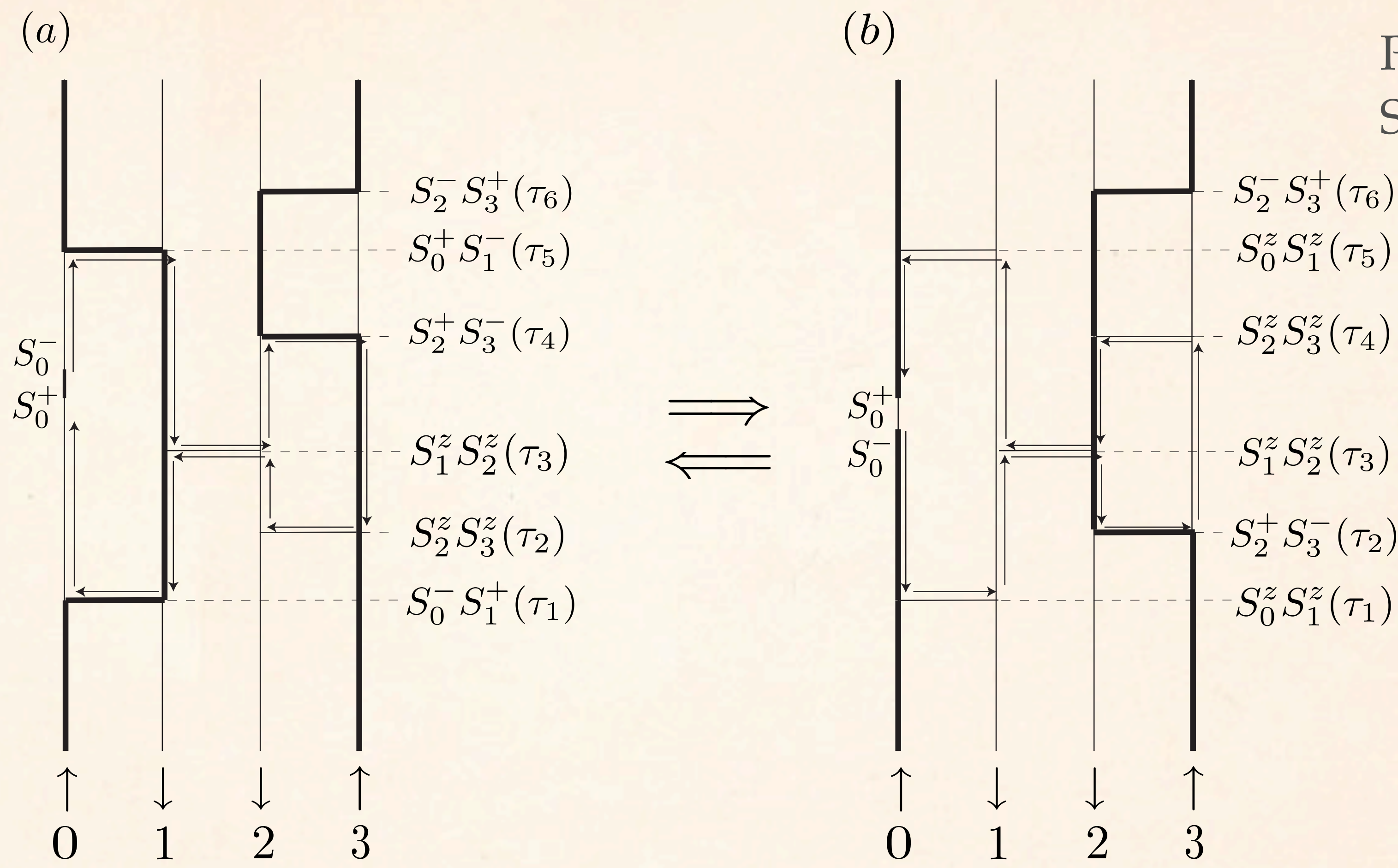
$$V = \sum_{\langle i,j \rangle} S_i \cdot S_j$$

S=1/2

- A *d*-dimensional quantum system can be mapped onto (*d*+1)-dimensional classical system.

These world lines are sampled in QMC.

Worm (Directed-Loop) Algorithm



Prokof'ev *et al.* (1998)
Syljuasen *et al.* (2002)

- The worm is a pair of operators S_i^+ and S_i^-
- The world-line configurations are updated by moving worms, as illustrated ((a) → (b) or (b) ← (a)).
- The worm algorithm can treat more general models than the loop algorithm, such as spin models with magnetic field and general soft-core bosonic models.
- The conventional worm algorithm satisfies the DBC in one loop.

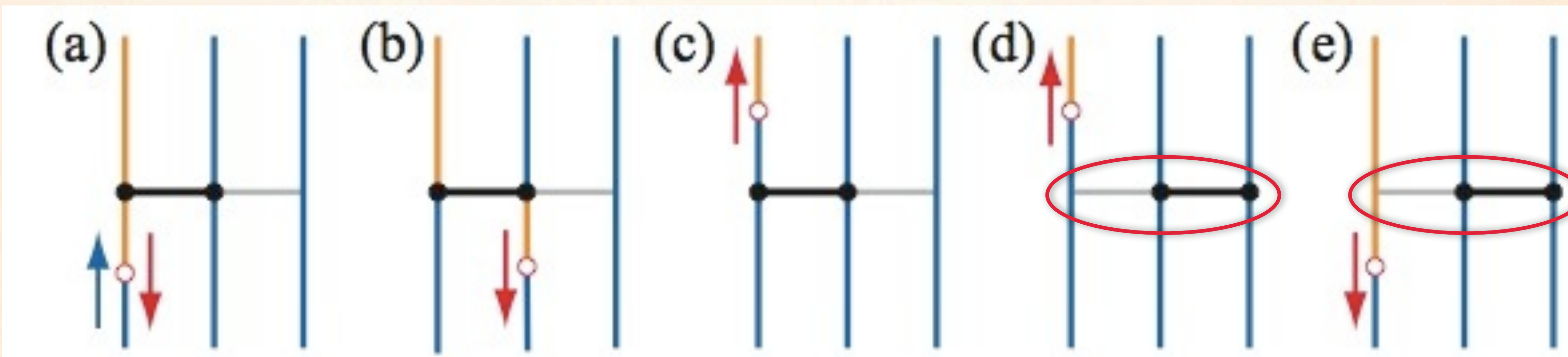
Operator Flip

- We have constructed a bounce-free worm by breaking the DBC and using a new update, the operator flip.

S=1/2, XXZ spin model with magnetic field

C needs to be greater than or equal to a value for avoiding negative signs.

$$H = \sum_{<i,j>} J\left(\frac{1}{2}(S_i^+ S_j^- + S_i^- S_j^+) + \Delta S_i^z S_j^z \underline{- C}\right) - h\sum_i S_i^z$$

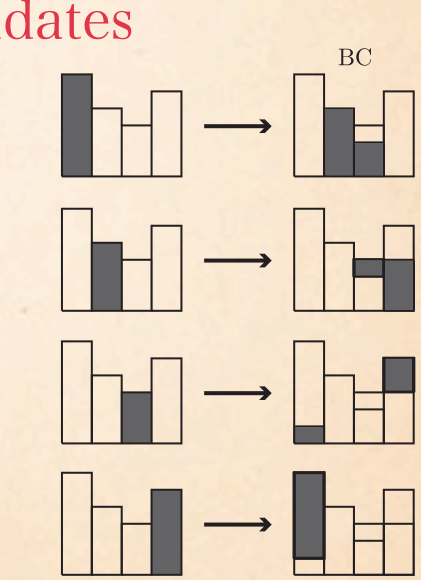


Flip

Conventional candidates

New candidates

$$C \geq \max\left(\frac{1}{4}(2\Delta + 3h - 1),\ \frac{1}{8}(\Delta + 3h + 1)\right) \qquad \Delta \geq 0$$



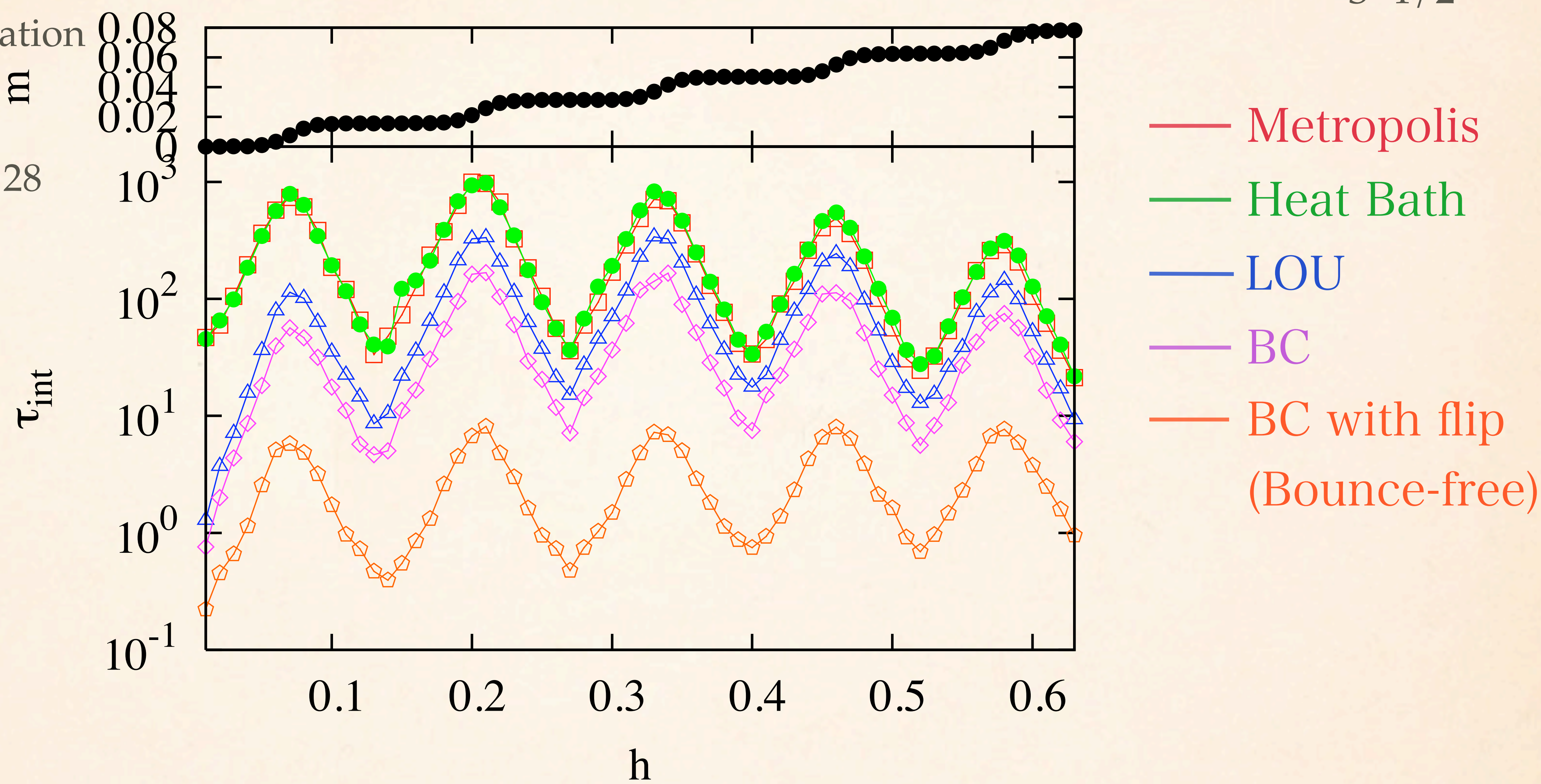
Rejection-free (bounce-free) condition $w_1 \leq \frac{S_n}{2} \equiv \frac{1}{2}\sum_{k=1}^{n} w_k$

Simple Strategy: Increase candidates for the next configuration !

Speed up in Heisenberg Chain

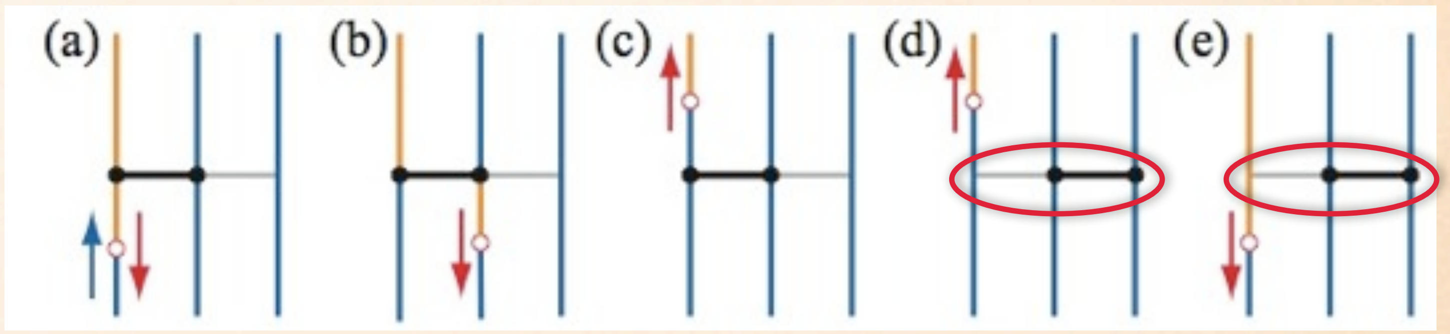
S=1/2

L=64, T=1/128



The BC is the fastest.

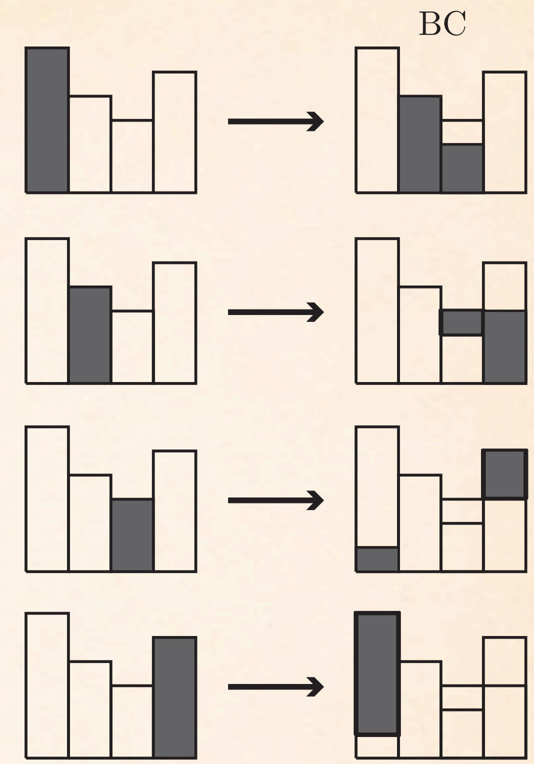
Metropolis	× 6
Heat bath	× 6
LOU	× 2



+ Bounce-free worms with operator flip

→ More than 100 times faster !

Summary

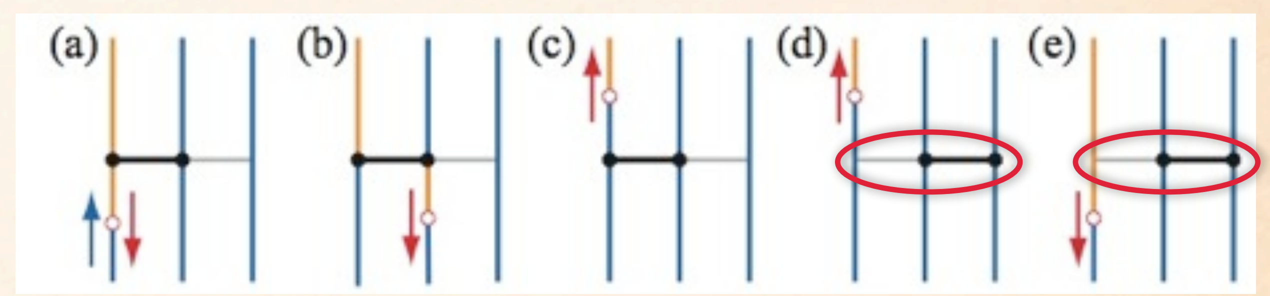


- We have presented a new algorithm that generally ensures the BC without the DBC.

$$v_{i\to j} = \max(0, \min(\Delta_{ij}, w_i + w_j - \Delta_{ij}, w_i, w_j)),$$

  - Existence of stochastic flow
  - Minimized rejection rates !

- We have constructed a bounce-free worm by using this algorithm and extending the worm-going pathways.



Strategy: Increase candidates for the next configuration !

$$w_1 \le \frac{S_n}{2} \equiv \frac{1}{2}\sum_{k=1}^{n} w_k$$

- Our algorithm can be universally applied to various Markov chain Monte Carlo methods, and will undoubtedly improve the relaxation.

(Metropolis *et al.*) 1953 ~ 2010 DBC
2010 ~ BC

H.S. and Synge Todo arXiv:1007.2262