.

Interpretations of Cluster Sampling by Swendsen-Wang

Abstract

Markov chain Monte Carlo (MCMC) methods have been used in many fields (physics, chemistry, biology, and computer science) for simulation, inference, and optimization. The essence of these methods is to simulate a Markov chain whose state $\mathbf{X}$ follows a target probability $\mathbf{X} \sim \pi(\mathbf{X})$. In many applications, $\pi(\mathbf{X})$ is defined on a graph $\mathbf{G}$ whose vertices represent elements in the system and whose edges represent the connectivity of the elements. $\mathbf{X}$ is a vector of variables on the vertices which often take discrete values called labels or colors. Designing rapid mixing Markov chain is a challenging task when the variables in the graph are strongly coupled. Methods, like the single-site Gibbs sampler, often experience long waiting time. A well-celebrated algorithm for sampling on graphs is the Swendsen-Wang (1987) (SW) method. The SW method finds a cluster of vertices as a connected component after turning off some edges probabilistically, and flips the color of the cluster as a whole. It is shown to mix rapidly under certain conditions. It has polynomial mixing time when the graph is a $O(1)$ connectivity, i.e. the number of neighbors of each vertex is constant and does not grow with the graph size.
In the literature, there are several ways for interpreting the SW-method which leads to various analyses or generalizations, including

1. A Metropolis-Hastings perspective: using auxiliary variables to propose the moves and accepting with probability 1.

2. Data augmentation: sampling a joint probability whose marginal probability is $\pi(\mathbf{X})$.

3. Slice sampling and partial decoupling.

Then we generalize SW from Potts model to arbitrary probabilities on graphs following the Metropolis-Hastings perspective, and derive a generalized Gibbs sampler.

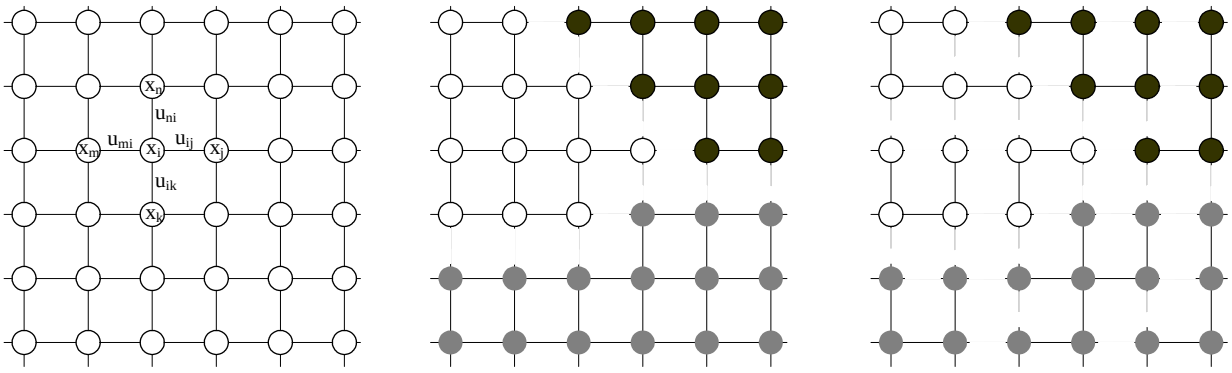
1 Introduction: Potts model and SW

In this section, we review the Potts model and the SW method.

Let $\mathbf{G} =< V, E >$ be an adjacency graph, such as a lattice with 4 nearest neighbor connections. Each vertex $v_i \in V$ has a state variable x_i with finite number of labels (or colors), $x_i \in \{1, 2, ..., \mathrm{L}\}$. The total number of label L is pre-defined. $\mathbf{X} = (x_1, x_2, ..., x_{|V|})$ is the labels of the graph and the Potts model for a homogeneous Markov field is,

$$\pi_{\mathrm{PTS}}(\mathbf{X}) = \frac{1}{Z} \exp\{\beta \sum_{<s,t>\in E} \mathbf{1}(x_s = x_t)\}. \tag{1}$$

$\mathbf{1}(x_s = x_t)$ is a Boolean function. It is equal to 1 if its condition $x_s = x_t$ is observed, and is 0 otherwise. In more general cases, $\beta = \beta_{st}$ may be position dependent. Usually we consider $\beta > 0$ for a ferro-magnetic system which prefers same colors for neighboring vertices. The Potts models and its extensions are used as *a priori* probabilities in many Bayesian inference tasks.



(a) G_0 (b) G (c) CCP

Figure 1: Illustating the SW method. (a) An adjacency graph $\mathbf{G}$ and each edge $e =< s, t >$ is augmented with a binary variable $\mu_e \in \{1, 0\}$. (b) A labeling of the Graph $\mathbf{G}$ where the edges connecting vertices of different colors are removed. (c). A number of connected component after turning off some edges in (b) probabilistically.

As Fig.1.(a) illustrates, the SW method introduces a set of auxiliary variables on the edges.

$$\mathbf{U} = \{\mu_e : \ \mu_e \in \{0, 1\}, \ \forall e \in E\}. \tag{2}$$

The edge e is disconnected (or turned off) if and only if $\mu_e = 0$. μ_e follows a Bernoulli distribution conditioning on x_s, x_t.

$$\mu_e|(x_s, x_t) \sim \text{Bernoulli}(q_e \mathbf{1}(x_s = x_t)), \quad \text{with } q_e = 1 - e^{-\beta}, \ \forall e \in E. \tag{3}$$

$\mu_e = 1$ with probability q_e if $x_s = x_t$, and $\mu_e = 0$ with probability 1 if $x_s \neq x_t$. The SW method iterates two steps.

1. **The clustering step**. Given the current state $\mathbf{X}$, it samples the auxiliary variables in $\mathbf{U}$ according to eqn. (3). It first turns off all edges e deterministically if $x_s \neq x_t$, as Fig.1.(b) shows.

$$E = E_{\text{on}}(\mathbf{X}) \cup E_{\text{off}}(\mathbf{X}). \tag{4}$$

Then it turns off the remain edges with probability ρ. The edge e is divided into the "on" and "off" sets respectively depending on $\mu_e = 1$ or 0. Therefore we further divide the edge set $E_{\text{on}}(\mathbf{X})$,

$$E_{\text{on}}(\mathbf{X}) = E_{\text{on}}(\mathbf{U}, \mathbf{X}) \cup E_{\text{off}}(\mathbf{U}, \mathbf{X}). \tag{5}$$

The edges in $E_{\text{on}}(\mathbf{U}, \mathbf{X})$ form a number of connected components shown in Fig. 1.(c). We denote all the connected components given $E_{\text{on}}(\mathbf{U}, \mathbf{X})$ by,

$$\text{CP}(\mathbf{U}, \mathbf{X}) = \{\text{cp}_i : \ i = 1, 2, ..., K, \ \text{with} \cup_{i=1}^{K} \text{cp}_i = V\}. \tag{6}$$

Vertices in each connected component cp_i have the same color.

2. **The flipping step**. It selects one connected component $V_o \in \text{CP}$ at random and assign a common color ℓ to all vertices in V_o. ℓ follows a uniform probability,

$$x_s = \ell \ \ \forall s \in V_o, \quad \ell \sim \text{uniform}\{1, 2, ..., \text{L}\}. \tag{7}$$

In this step, one may choose to repeat the random color flipping for all the connected components in $\text{CP}(\mathbf{U})$ independently, as they are decoupled given the edges in $E_{\text{on}}(\mathbf{U}, \mathbf{X})$. By doing so, all possible labels of the graph are connected in one step, just like one sweep of the Gibbs sampler.

In one modified version by Wolff (1989), one may choose a vertex $v \in V$ and grow a connected component following the Bernoulli trials on edges around v. This saves some computation in the clustering step, and thus bigger components have higher chance to be selected.

2 Interpretation 1: Metropolis-Hastings perspective

The SW algorithm can be interpreted as a Metropolis-Hastings step with acceptance probability 1.

Fig. 2 shows three partition states A, B and C which differ in the labels of the pixels in a connected component V_o (denoted by R in the figure). Suppose the current state is A

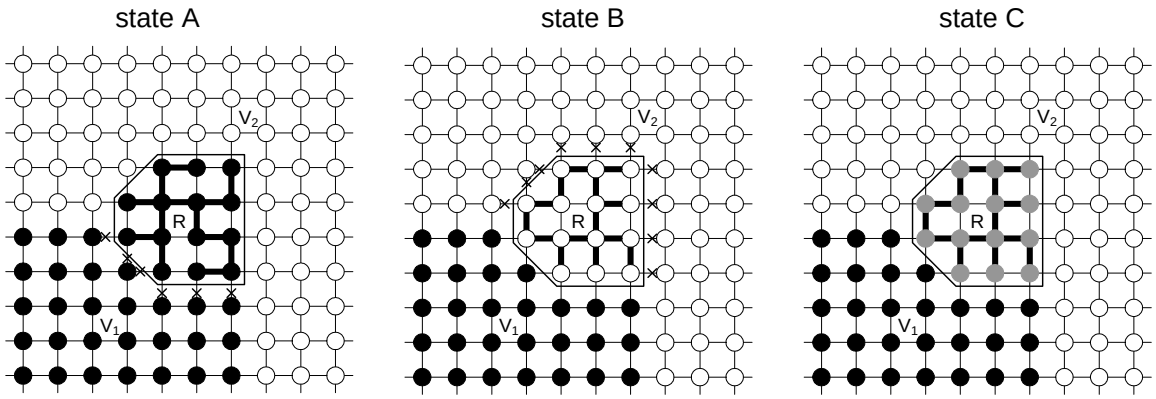


Figure 2: SW algorithm flips a patch of spins in one step for the Ising/Potts models.

in which V_o is connected to V_1 which are the remaining black vertices. The edges that are turned off probabilistically between V_o and V_1 is a cut

$$C_{01} = C(V_0, V_1) = \{e =< s, t >: \ \ s \in V_0, \ t \in V_1\}.$$

The cut is illustrated by the crosses in Figure 7.

Obviously there are many ways to arrive at a connected component V_o through the random steps. But they must share a common cut $C(V_0, V_1)$.

Similarly if the Markov chain is currently at state B, it also has a chance to select a connected component V_o in white. We denote the remaining white vertices as V_2, and the cut between V_0 and V_2 is

$$C_{02} = C(V_0, V_2) = \{e =< s, t >: \ \ s \in V_0, \ t \in V_2\}.$$

So far, we have a pair of states A and B who are different in the labels of V_o. A Metropolis-Hastings method is used to realize a reversible move between them. Though it is difficult to compute the proposal probabilities $Q(A \to B)$ and $Q(B \to A)$, one can compute their ratio easily through cancellation.

$$\frac{Q(A \to B)}{Q(B \to A)} = \frac{(1-q)^{|C_{01}|}}{(1-q)^{|C_{02}|}} = (1-q)^{|C_{01}|-|C_{02}|}. \tag{8}$$

In the above equation, $|\mathcal{C}|$ is the cardinality of the set. In other words, the probabilities for selecting V_0 in states A and B are the same, except that the cuts are different. Remarkably the probability ratio for $\pi(A)/\pi(B)$ is also decided by the cuts through cancellation.

$$\frac{\pi(A)}{\pi(B)} = \frac{e^{-\beta|C_{02}|}}{e^{-\beta|C_{01}|}} = e^{\beta(|C_{01}|-|C_{02}|)} \tag{9}$$

The acceptance probability for the move from A to B is,

$$\alpha(A \to B) = \min(1, \frac{Q(B \to A)}{Q(A \to B)} \cdot \frac{\pi(B)}{\pi(A)}) = (\frac{e^{-\beta}}{1-q})^{|C_{01}|-|C_{02}|}. \tag{10}$$

By a smart choice of the edge probability

$$q = 1 - e^{-\beta},$$

then the proposal from A to B is always accepted with

$$\alpha(A \to B) = 1.$$

As $\beta \propto \frac{1}{T}$ is proportional to the inverse temperature, thus $q \to 1$ at low temperature and SW flips a large patch at a time. So SW algorithm can mix very fast at even critical temperature.

Proof of eq.(8). I will give the proof on the blackboard. Here is the sketch of the proof.

Let $\mathbf{U}_A|\mathbf{X} = A$ and $\mathbf{U}_B|\mathbf{X} = B$ be the auxiliary variables in states A and B respectively. Following the Bernoulli probabilities in the flipping step, and they leads to two sets of connected components $\mathrm{CP}(\mathbf{U}_A|A)$ and $\mathrm{CP}(\mathbf{U}_A|B)$ respectively. We divide $\mathbf{U}_A$ into two sets for the on and off edges respectively,

$$\mathbf{U}_A = \mathbf{U}_{A,\mathrm{on}} \cap \mathbf{U}_{A,\mathrm{off}}. \tag{11}$$

$$\mathbf{U}_{A,\mathrm{on}} = \{\mu_e : \mu_e = 1\}, \;\; \mathbf{U}_{A,\mathrm{off}} = \{\mu_e : \mu_e = 0\}.$$

We are only interested in the $\mathbf{U}_A$'s (and thus $\mathrm{CP}(\mathbf{U}_A|A)$'s) which yield the connected component V_o. We collect all such $\mathbf{U}_A$ given A in a set,

$$\Omega(V_o|A) = \{\mathbf{U}_A : V_o \in \mathrm{CP}(\mathbf{U}_A|A)\}. \tag{12}$$

In order for V_o being a connected component in A, all edges between V_o and V_1 must be cut (turned off), otherwise V_o can not be a connected component. So, we denote the remaining "off" edges by ${}^{-}\mathbf{U}_{\mathrm{off}}$,

$$\mathbf{U}_{A,\mathrm{off}} = \mathcal{C}(V_o, V_1) \cup {}^{-}\mathbf{U}_{A,\mathrm{off}}, \quad \forall\, \mathbf{U}_A \in \Omega(V_o|A). \tag{13}$$

Similarly, we collect all $\mathbf{U}_B$ in state B which produce the connected component V_o,

$$\Omega(V_o|B) = \{\mathbf{U}_A : V_o \in \mathrm{CP}(\mathbf{U}_B|B)\}. \tag{14}$$

In order for V_o to be a connected component in $\mathbf{U}_B|B$, the clustering step must cut all the edges between V_o and V_2. Thus we have

$$\mathbf{U}_B = \mathbf{U}_{B,\mathrm{on}} \cup \mathbf{U}_{B,\mathrm{off}} \tag{15}$$

with

$$\mathbf{U}_{B,\mathrm{off}} = \mathcal{C}(V_o, V_2) \cup {}^{-}\mathbf{U}_{B,\mathrm{off}}, \quad \forall\, \mathbf{U}_B \in \Omega(V_o|B). \tag{16}$$

A key observation is that there is a one-to-one mapping between $\Omega(V_o|A)$ and $\Omega(V_o|B)$.

Proposition 1 *For any* $\mathbf{U}_A \in \Omega(V_o|A)$*, there exists one and only one* $\mathbf{U}_B \in \Omega(V_o|B)$ *such that*

$$\mathrm{CP}(\mathbf{U}_A|A) = \mathrm{CP}(\mathbf{U}_B|B) \tag{17}$$

and

$$\mathbf{U}_{A,\mathrm{on}} = \mathbf{U}_{B,on}, \quad {}^{-}\mathbf{U}_{A,\mathrm{off}} = {}^{-}\mathbf{U}_{B,\mathrm{off}}. \tag{18}$$

That is, $\mathbf{U}_A$ *and* $\mathbf{U}_B$ *differ only in the cuts* $\mathcal{C}(V_o, V_1)$ *and* $\mathcal{C}(V_o, V_2)$*.*

Suppose that we choose $V_o \in \mathrm{CP}(\mathbf{U}_A|A)$ with uniform probability, then the probability for choosing V_o at state A is

$$q(V_o|A) = \sum_{\mathbf{U}_A \in \Omega(V_o|A)} \frac{1}{|\mathrm{CP}(\mathbf{U}_A|A)|} \prod_{e \in \mathbf{U}_{A,\mathrm{on}}} q_e \prod_{e \in {}^{-}\mathbf{U}_{A,\mathrm{off}}} (1-q_e)] \prod_{e \in \mathcal{C}(V_o,V_1)} (1-q_e). \tag{19}$$

Similarly, the probability for choosing V_o at state B is

$$q(V_o|B) = \sum_{\mathbf{U}_B \in \Omega(V_o|B)} \frac{1}{|\mathrm{CP}(\mathbf{U}_B|B)|} \prod_{e \in \mathbf{U}_{B,\mathrm{on}}} q_e \prod_{e \in {}^{-}\mathbf{U}_{B,\mathrm{off}}} (1-q_e)] \prod_{e \in \mathcal{C}(V_o,V_2)} (1-q_e). \tag{20}$$

Dividing eqn. (20) by eqn. (20), we obtain the ratio in eqn. (44) due to cancellation following the observations in Proposition 1.

End of Proof.

In a special case when $\mathcal{C}(V_o, V_1) = \emptyset$, then $\prod_{e \in \mathcal{C}(V_o,V_1)}(1-q_e) = 1$. Note that the proof holds for arbitrary design of q_e.

There is a slight complication, when there are two pathes connecting the two states in Figure 3.

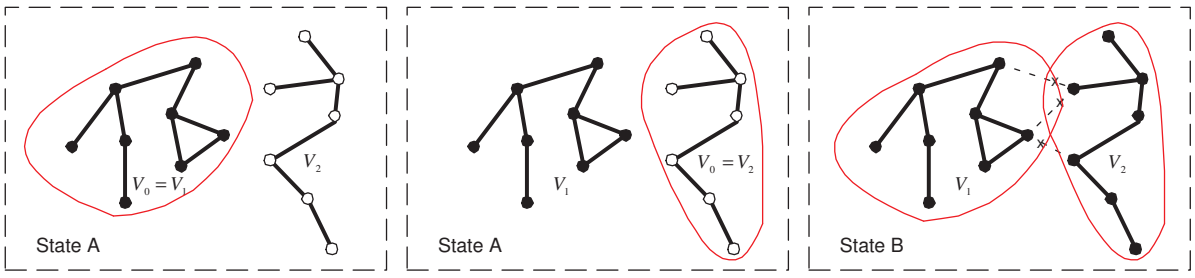


Figure 3: State A has two subgraphs V_1 and V_2 which are merged in state B.

Path 1. Choose $V_o = V_1$. In state A, choose new label $\ell = 2$, i.e. merge V_o to V_2, and reversely in state B, choose new label $\ell = 1$, i.e. split V_o from V_2.

Path 2. Choose $V_o = V_2$. In state A, choose new label $\ell = 1$, i.e. merge V_o to V_1, and reversely in state B, choose $\ell = 2$, i.e. split V_o from V_1. In such case, the proposal probability ratio is,

$$\frac{Q(B \to A)}{q(A \to B)} = \frac{q(V_o = V_1|B)q(\mathbf{X}_{V_o} = 2|V_o, B) + q(V_o = V_2|B)q(\mathbf{X}_{V_o} = 1|V_o, B}{q(V_o = V_1|A)q(\mathbf{X}_{V_o} = 1|V_o, A)) + q(V_o = V_2|A)q(\mathbf{X}_{V_o} = 2|V_o, A)}. \tag{21}$$

In state A, the SW-cut $\mathcal{C}(V_o, V_\ell \setminus V_o) = \emptyset$ for both paths, and in state B the cut is $\mathcal{C}(V_1, V_2)$ for both paths. Following theorem **??**, the probability ratios for choosing $V_o = V_1$ and $V_o = V_2$ are equal,

$$\frac{q(V_o = V_1|A)}{q(V_o = V_1|B)} = \frac{1}{\prod_{e\in\mathcal{C}(V_1,V_2)}(1-q_e)} = \frac{q(V_o = V_2|A)}{q(V_o = V_2|B}. \tag{22}$$

Once V_o is selected, either $V_o = V_1$ or $V_o = V_2$, then the remaining partition for both A and B are the same, and is denoted by $\mathbf{X}_{V\setminus V_o} = \mathbf{X}_{V\setminus V_o}$. In proposing the new label of V_o, we easily observe that

$$\frac{q(\mathbf{X}_{V_o} = 2|V_o = V_1, B)}{q(\mathbf{X}_{V_o} = 1|V_o = V_2, A)} = \frac{q(\mathbf{X}_{V_o} = 1|V_o = V_2, B)}{q(\mathbf{X}_{V_o} = 2|V_o = V_1, A)}. \tag{23}$$

Then the acceptance rate remains 1.

3 Interpretation 2: data augmentation

The SW method described above is far from what was presented in the original paper (Swendsen and Wang 1987). Instead our description follows the interpretation by Edward and Sokal (1988), who augmented the Potts model to a joint probability for both $\mathbf{X}$ and $\mathbf{U}$,

$$\begin{aligned} p_{\mathrm{ES}}(\mathbf{X}, \mathbf{U}) &= \frac{1}{Z}\prod_{e=<s,t>\in E}[(1-\rho)\mathbf{1}(\mu_e = 0) + \rho\mathbf{1}(\mu_e = 1)\cdot\mathbf{1}(x_s = x_t)] &(24)\\ &= \frac{1}{Z}[(1-\rho)^{|E_{\mathrm{off}}(\mathbf{U})|}\cdot\rho^{E_{\mathrm{on}}(\mathbf{U})}]\cdot\prod_{<s,t>\in E_{\mathrm{on}}(\mathbf{U})}\mathbf{1}(x_s = x_t). &(25)\end{aligned}$$

The second factor $\prod_{<s,t>\in E_{\mathrm{on}}(\mathbf{U})}\mathbf{1}(x_s = x_t)$ is in fact a hard constraint on $\mathbf{X}$ and $\mathbf{U}$. Let the space of $\mathbf{X}$ be

$$\Omega = \{1, 2, ..., \mathrm{L}\}^{|V|}. \tag{26}$$

Under this hard constraint, the labeling $\mathbf{X}$ is reduced to a quotient space $\frac{\Omega}{\mathrm{CP}(\mathbf{U})}$ where each connected component must have the same label,

$$\prod_{<s,t>\in E_{\mathrm{on}}(\mathbf{U})}\mathbf{1}(x_s = x_t) = \mathbf{1}(\mathbf{X}\in\frac{\Omega}{\mathrm{CP}(\mathbf{U})}). \tag{27}$$

The joint probability $p_{\mathrm{ES}}(\mathbf{X}, \mathbf{U})$ observes two nice properties, and both are easy to verify.

Proposition 2 *The Potts model is a marginal probability of the joint probability,*

$$\sum_{\mathbf{U}} p_{\mathrm{ES}}(\mathbf{X}, \mathbf{U}) = \pi_{\mathrm{PTS}}(\mathbf{X}). \tag{28}$$

The other marginal probability is the random cluster model $\pi_{\rm RCM}$,

$$\sum_{\mathbf{X}} p_{\rm ES}(\mathbf{X},\mathbf{U}) = \pi_{\rm RCM}(\mathbf{U}) = \frac{1}{Z}(1-\rho)^{|E_{\rm off}(\mathbf{U})|}\cdot \rho^{E_{\rm on}(\mathbf{U})}{\rm L}^{|{\rm CP}(\mathbf{U})|}. \quad (29)$$

Proposition 3 *The conditional probabilities of* $p_{\rm ES}(\mathbf{X},\mathbf{U})$ *are*

$$p_{\rm ES}(\mathbf{U}|\mathbf{X}) = \prod_{<s,t>\in E} p(\mu_e|x_s,x_t), \quad \text{with } p(\mu_e|x_s,x_t) = \text{Bernoulli}(\rho\mathbf{1}(x_s=x_t)), \quad (30)$$

$$p_{\rm ES}(\mathbf{X}|\mathbf{U}) = \text{unif}[\frac{\Omega}{{\rm CP}(\mathbf{U})}] = (\frac{1}{\rm L})^{|{\rm CP}(\mathbf{U})|} \text{ for } \mathbf{X}\in\frac{\Omega}{{\rm CP}(\mathbf{U})};\ =0 \text{ otherwise.} \quad (31)$$

Therefore the two SW steps can be viewed as sampling the two conditional probabilities.

1. Clustering step: $\mathbf{U}\sim p_{\rm ES}(\mathbf{U}|\mathbf{X})$, i.e. $\mu_e|(x_s,x_t)\sim$ Bernoulli$(\rho\mathbf{1}(x_s=x_t))$.
2. Flipping step: $\mathbf{X}\sim p_{\rm ES}(\mathbf{U}|\mathbf{X})$, i.e. $\mathbf{X}({\rm cp}_i)\sim$ Unif$\{1,2,...,{\rm L}\}$, $\forall {\rm cp}_i\in{\rm CP}(\mathbf{U})$.

As $(\mathbf{X},\mathbf{U})\sim p_{\rm ES}(\mathbf{X},\mathbf{U})$, discarding the auxiliary variables $\mathbf{U}$, we have $\mathbf{X}$ following the marginal of $p_{\rm ES}(\mathbf{X},\mathbf{U})$. The goal is achieved,

$$\mathbf{X}\sim\pi_{\rm PTS}(\mathbf{X}). \quad (32)$$

The beauty of this data augmentation method (Tanner and Wong 1987) is that the labeling of the connected components are completely decoupled (independent) given the auxiliary variables. As $\rho=1-e^{-\beta}$, it tends to choose smaller clusters if the temperature ($T\propto\frac{1}{\beta}$) in the Potts model is high, and in low temperature it chooses large clusters. So it can overcome the coupling problem with single site Gibbs sampler.

4 Some theoretical results

Let the Markov chain have kernel $\mathcal{K}$ and initial state $\mathbf{X}_o$, in t steps the Markov chain state follows probability $p_t=\delta(\mathbf{X}-\mathbf{X}_o)\mathcal{K}^t$ where $\delta(\mathbf{X}-\mathbf{X}_o)$ (for $\delta(\mathbf{X}-\mathbf{X}_o)=1$ for $\mathbf{X}=\mathbf{X}_o$ and 0 otherwise) is the initial probability. The convergence of the Markov chain is often measured by the total variation

$$||p_t-\pi||_{\rm TV} = \frac{1}{2}\sum_{\mathbf{X}}|p_t(\mathbf{X})-\pi(\mathbf{X})|. \quad (33)$$

The mixing time of the Markov chain is defined by

$$\tau = \max_{\mathbf{X}_o}\min\{t:\ ||p_t-\pi||_{\rm TV}\le\epsilon\}. \quad (34)$$

τ is a function of ϵ and the graph compexlity $M = |\mathbf{G}_o|$ in terms of the number of vertices and connectivity. The Markov chain is said to mix rapidly if $\tau(M)$ is polynomial or logarithmic.

Empirically, the SW method is found to mix rapidly. Recently some analytic results on its performance have surfaced. Cooper and Frieze (1999) proved using a path coupling technique that SW mixs rapidly on sparsely connected graphs.

Theorem 1 (Cooper and Frieze 1999) *Let $n = |V|$ and Δ be the maximum number of edges at any single vertex, and* L *the number of colors in Potts model. If* $\mathbf{G}$ *is a tree, then the SW mixing time is $O(n)$ for any β and* L*. If $\Delta = O(1)$, then there exists $\rho_o = \rho(\Delta)$ such that if $\rho \leq \rho_o$ (i.e. higher than a certain temperature), then SW has polynomial mixing time for all* L*.*

A negative case was constructed by Gore and Jerrum (1997) on complete graph.

Theorem 2 (Gore and Jerrum 1997) *If* $\mathbf{G}$ *is a complete graph and* $\mathrm{L} > 2$*, then for $\beta = \frac{2(\mathrm{L}-1)\ln(\mathrm{L}-1)}{n(\mathrm{L}-2)}$, the SW does not mix rapidly.*

In the image analysis applications, our graph often observes the Copper-Frieze condition and the graph is far from being complete.

Most recently an exact sampling technique was developed for SW on Potts by Huber (2002) for very high or very low temperatures. It designs a bounding chain which assumes that each vertex $s \in V$ has a set of colors S_s initialized with the full set $|S_s| = \mathrm{L}$, $\forall s$. The Bernoulli probability for the auxiliary variables μ_e is changed to

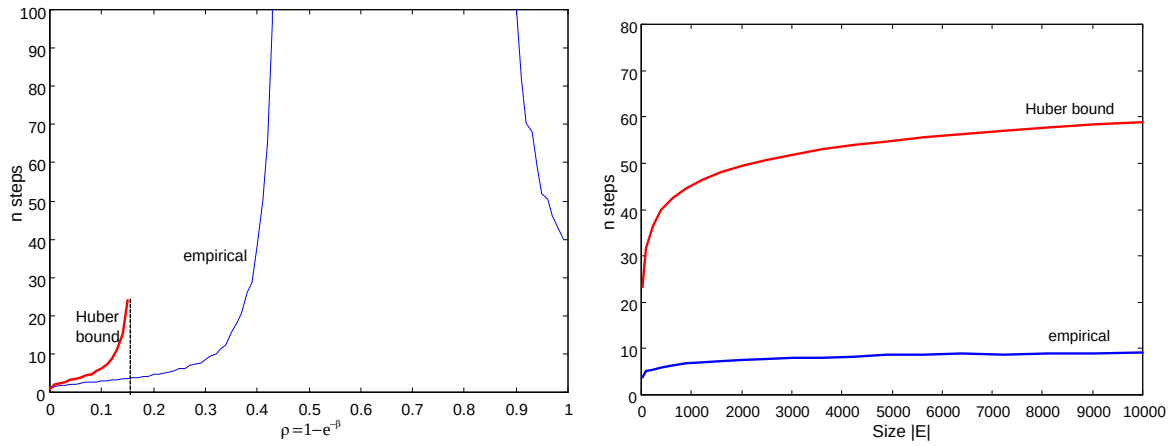
$$\mathbf{U}^{\mathrm{bd}} = \{\mu_e^{\mathrm{bd}} : \mu_e^{\mathrm{bd}} \in \{0,1\},\ \mu_e \sim \mathrm{Bernoulli}(\rho \mathbf{1}(S_s \cap S_t \neq \emptyset))\}. \tag{35}$$

Thus $\mathbf{U}^{\mathrm{bd}}$ has more edges than $\mathbf{U}$ in the original SW chain, i.e. $\mathbf{U} \subset \mathbf{U}^{\mathrm{bd}}$. When $\mathbf{U}^{\mathrm{bd}}$ collapses to $\mathbf{U}$, then all SW chains starting with arbitrary initial states have collapsed into the current single chain. Thus it must have converged (exact sampling). The step for collapsing is called the "coupling time".

Theorem 3 (Huber 2002) *Let $n = |V|$ and $m = |E|$, at high temperature, $\rho < \frac{1}{2(\Delta-1)}$, the bounding chain couples completely by time $O(\ln(2m))$ with probability at least $1/2$. At lower temperature, $\rho \geq 1 - \frac{1}{m\mathrm{L}}$, then the coupling time is $O((m\mathrm{L})^2)$ with probability at least $1/2$.*

In fact the Huber bound is not very tight as one may expect. Fig. 4(a) plots the results on a 5×5 lattice with torus boundary condition on the Ising model for the empirical coupling time against $\rho = 1 - e^{-\beta}$. The coupling time is large near the critical temperature

(didn't plot). The Huber bound for the high temperature starts with $\rho_o = 0.16$ and is plotted by the short curve. The bound for the low temperature starts with $\rho_o > 0.99$ which is not visible. Fig.4.(b) plots the coupling time at $\rho = 0.15$ against the graph size $m = |E|$ and the Huber bound.



(a) (b)

Figure 4: The coupling time empirical plots and the Huber bounds for Ising model.

Despite the encouraging success discussed above, the SW method is limited in two aspects.

Limit 1. It is only valid for the Ising and Potts models, and furthermore it requires that the number of colorings L is known. In many applications, such as image analysis, L is the number of objects (or image regions) which has to be inferred from the input data.

Limit 2. It slows down quickly in the presence of external field, i.e input data. For example, in the image analysis problem, our goal is to infer the label $\mathbf{X}$ from the input image $\mathbf{I}$ and the target probability is a Bayesian posterior probability where $\pi_{\text{PTS}}(\mathbf{X})$ is used as a prior model,

$$\pi(\mathbf{X}) = \pi(\mathbf{X}|\mathbf{I}) \propto \mathcal{L}(\mathbf{I}|\mathbf{X})\pi_{\text{PTS}}(\mathbf{X}). \tag{36}$$

$\mathcal{L}(\mathbf{I}|\mathbf{X})$ is the likelihood model, such as independent Gaussians $N(\bar{\mathbf{I}}_c, \sigma_c^2)$ for each coloring $c = 1, 2, ..., \text{L}$,

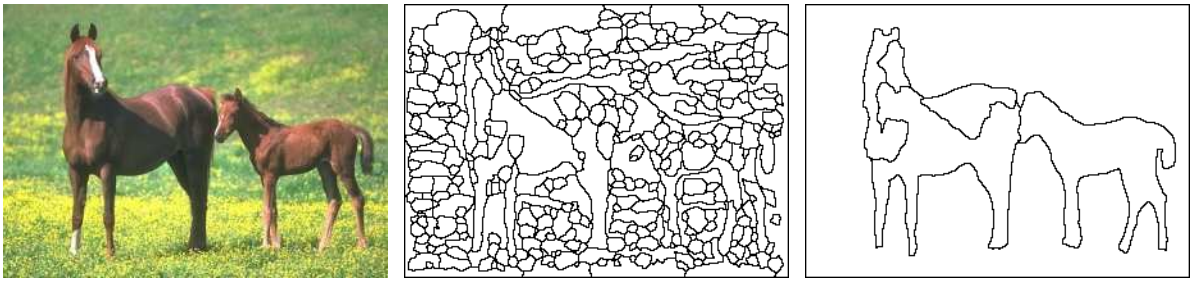
$$\mathcal{L}(\mathbf{I}|\mathbf{X}) \propto \prod_{c=1}^{\text{L}} \prod_{x_i=c} \frac{1}{\sqrt{2\pi}\sigma_c} \exp\{-\frac{(\mathbf{I}(v_i) - \bar{\mathbf{I}}_c)^2}{2\sigma_c^2}\}. \tag{37}$$

The slowing down is partially attributed to the fact that the Bernoulli probability $\rho = 1 - e^{-\beta}$ for the auxiliary variable is calculated independently of the input image.

5 Generalizing SW to arbitrary probabilities on graph

In this section, we generalize the SW to arbitrary probabilities from the perspective of Metropolis-Hastings method (Metropolis et al 1953, Hastings 1970). Our method iterates three steps: (i) a clustering step driven by data, (ii) a label flipping step which can introduce new labels, and (iii) an acceptance step for the proposed labelling. A key observation is the simple formula in calculating the acceptance probabilities.

We deliberate the three steps in the following three subsections, and then we show how it reduces to the original SW with Potts models.



(a). Input image (b). atomic regions (c). segmentation

Figure 5: Example of image segmentation. (a). Input image. (b). Atomic regions by edge detection followed by edge tracing and contour closing. each atomic region is a vertex in the graph $\mathbf{G}$. c. Segmentation (labeling) result where each closed region is assigned a color or label.

We illustrate the algorithm by an example on image segmentation shown in Fig. 5. Fig. 5.(a) is an input image $\mathbf{I}$ on a lattice Λ, which is decomposed into a number of "atomic regions" to reduce the graph size in a preprocessing stage. Each atomic region has nearly constant intensity and is a vertex in the graph $\mathbf{G}$. Two vertices are connected if their atomic regions are adjacent (i.e. sharing boundary). Fig. 5.(c) is a result by our algorithm optimizing a Bayesian probability $\pi(\mathbf{X}) = \pi(\mathbf{X}|\mathbf{I})$ (see section (7) for details). The result $\mathbf{X}$ assigns a uniform color to all vertices in each close region which hopefully corresponds to an object in the scene or a part of it. Note that the number of objects or colors L is unknown, and we do not distinguish the different permutations of the labels.

5.1 Step 1: data-driven clustering

We augment the adjacency graph $\mathbf{G}$ with a set of binary variables on the edges $\mathbf{U} = \{\mu_e : e =< s, t >\in E\}$, as in the original SW method. Each μ_e follows a Bernoulli probability depending on the current state of the two vertices x_s and x_t,

$$\mu_e|(x_s, x_t) \sim \text{Bernoulli}(q_e \mathbf{1}(x_s = x_t)), \quad \forall < s, t >\in E. \tag{38}$$

q_e is a probability on edge $e =< s, t >$ which tells how likely the two vertices s and t have the same label. In Bayesian inference where the target $\pi(\mathbf{X})$ is a posterior probability, then q_e can be better informed by the data.

For the image segmentation example, q_e is computed based on the similarity between image intensities at s and t (or their local neighborhood) and it may be an approximate to the marginal probability of $\pi(\mathbf{X}|\mathbf{I})$,

$$q_e = q(x_s = x_t|\mathbf{I}(s), \mathbf{I}(t)) \ \approx \ \pi(x_s = x_t|\mathbf{I}). \tag{39}$$

There are many ways for computing $q(x_s = x_t|\mathbf{I}(v_s), \mathbf{I}(v_t))$ using so called discriminative methods, and it is beyond this paper to discuss details.

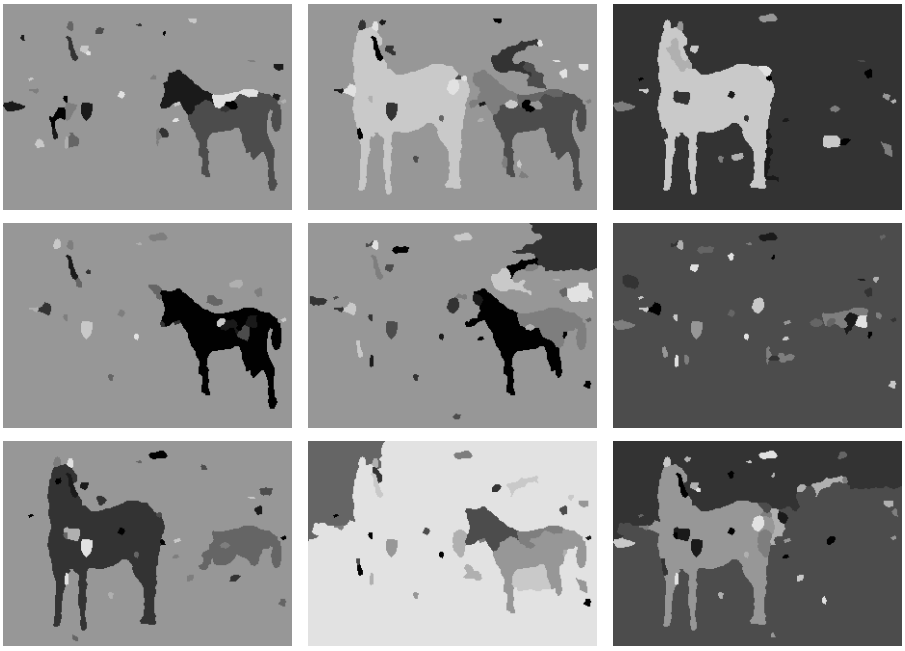


Figure 6: Nine examples of the connected components for the horse image computed using discriminative edge probabilities given that $\mathbf{X}$ is a uniform color $\mathbf{X} = c$ for all vertices.

Our method will work for any q_e, but a good approximation will inform the clustering step and achieve faster convergence empirically. Fig. 6 shows nine clustering examples of the horse image. In these examples, we set all vertices to the same color ($\mathbf{X} = c$) and sample the edge probability independently,

$$\mathbf{U}|\mathbf{X} = c \ \sim \prod_{<s,t>\in E} \text{Bernoulli}(q_e). \tag{40}$$

The connected components in $\text{CP}(\mathbf{U})$ are shown by different regions. We repeat the clustering step nine times. As we can see, the edge probabilities lead to "meaningful" clusters which correspond to distinct objects in the image. Such effects cannot be observed using constant edge probability.

5.2 Step 2: flipping of color

Let $\mathbf{X} = (x_1, V_2, ..., x_{|V|})$ be the current coloring state, and the edge variables $\mathbf{U}$ sampled conditional on $\mathbf{X}$ further decompose $\mathbf{X}$ into a number of connected components

$$\mathrm{CP}(\mathbf{U}|\mathbf{X}) = \{\mathrm{cp}_i : i = 1, 2, ..., N(\mathbf{U}|\mathbf{X})\}. \tag{41}$$

Suppose we select one connected component $V_o \in \mathrm{CP}(\mathbf{U}|\mathbf{X})$ with color $\mathbf{X}_{V_o} = \ell \in \{1, 2, ..., L\}$, and assign its color to $\ell' \in \{1, 2, ..., L, L+1\}$ with probability $q(l'|V_o, \mathbf{X})$ (to be designed shortly), obtaining new state $\mathbf{X}'$. There are three cases shown in Fig. 7.

1. The canonical case: $V_o \subset V_\ell$ and $\ell' \leq L$. That is, a portion of V_ℓ is re-grouped into an existing color $V_{\ell'}$, and the number of colors remains $\mathrm{L} = L$ in π'. The moves between $A \leftrightarrow B$ in Fig. 7 are examples.

2. The merge case: $V_o = V_\ell$ in $\mathbf{X}$ is the set of all vertices that have color ℓ and $\ell' \leq L,\ \ell \neq \ell'$. That is, color V_ℓ is merged to $V_{\ell'}$, and the number of distinct colors reduces to $L - 1$ in $\mathbf{X}'$. The moves $\mathbf{X}_C \to \mathbf{X}_A$ or $\mathbf{X}_C \to \mathbf{X}_B$ in Fig. 7 are examples.

3. The split case: $V_o \subset V_\ell$ and $\ell' = L + 1$. V_ℓ is split into two pieces and the number of distinct color increases to $L + 1$ in $\mathbf{X}'$. The moves $\mathbf{X}_A \to \mathbf{X}_C$ in Fig.7 are examples.

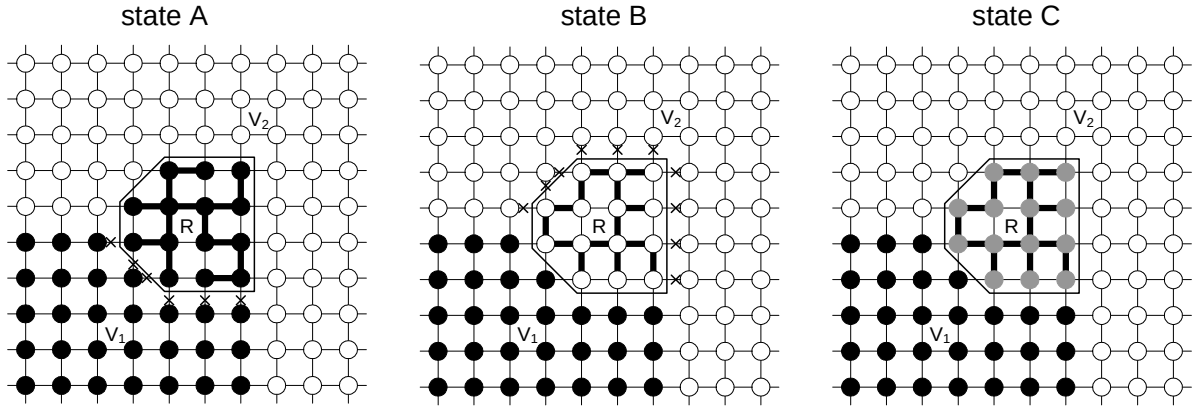


Figure 7: Three labeling states A, B, C which differ only in the color of a cluster R.

Note that this color flipping step is also different from the original SW with Potts model as we allow new colors in each step. The number of color L is not fixed.

5.3 Step 3: accepting the flipping

The previous two steps basically have proposed a move between two states $\mathbf{X}$ and $\mathbf{X}'$ which differ in coloring a connected component V_o. In the third step we accept the move with

probability,

$$\alpha(\mathbf{X} \to \mathbf{X}') = \min\{1, \frac{q(\mathbf{X}' \to \mathbf{X})}{q(\mathbf{X} \to \mathbf{X}')} \cdot \frac{\pi(\mathbf{X}')}{\pi(\mathbf{X})}\}. \tag{42}$$

$q(\mathbf{X}' \to \mathbf{X})$ and $q(\mathbf{X} \to \mathbf{X}')$ are the proposal probabilities between $\mathbf{X}$ and $\mathbf{X}'$. If the proposal is rejected, the Markov chain stays at state $\mathbf{X}$. The transition kernel is

$$\mathcal{K}(\mathbf{X} \to \mathbf{X}') = q(\mathbf{X} \to \mathbf{X}')\alpha(\mathbf{X} \to \mathbf{X}'), \quad \forall \mathbf{X} \neq \mathbf{X}'. \tag{43}$$

For the canonical case, there is a unique path for moving between bX and $\mathbf{X}'$ in one step – choosing V_o and changing its color. The proposal probability ratio is the product of two ratios decided by the clustering and flipping steps respectively: (i) the probability ratio for selecting V_o as the candidate in the clustering step in both states $\mathbf{X}$ and $\mathbf{X}'$, and (ii) the probability ratio for selecting the new labels for V_o in the flipping step.

$$\frac{q(\mathbf{X}' \to \mathbf{X})}{q(\mathbf{X} \to \mathbf{X}')} = \frac{q(V_o|\mathbf{X}')}{q(V_o|\mathbf{X})} \cdot \frac{q(\mathbf{X}_{V_o} = \ell|V_o, \mathbf{X}')}{q(\mathbf{X}_{V_o} = \ell'|V_o, \mathbf{X})}. \tag{44}$$

For the split and merge cases, there are two paths between $\mathbf{X}$ and $\mathbf{X}'$. But this does not change the conclusion (see Appendix B).

Now we compute the probability ratio $\frac{q(V_o|\mathbf{X}')}{q(V_o|\mathbf{X})}$ for proposing V_o.

Definition 1 *Let $\mathbf{X} = (V_1, V_2, ..., V_{\mathrm{L}})$ be a coloring state, and $V_o \in \mathrm{CP}(U|\mathbf{X})$ a connected component, the "cut" between V_o and V_k is a set of edges between V_o and $V_k \backslash V_o$,*

$$\mathcal{C}(V_o, V_k) = \{< s, t >: \ s \in V_o, t \in V_k \backslash V_o\}, \ \ \forall k.$$

One of our key observation is that this ratio only depends on the cuts between V_o and rest vertices.

Proposition 4 *In the above notation, we have*

$$\frac{q(V_o|\mathbf{X})}{q(V_o|\mathbf{X}')} = \frac{\prod_{<s,t> \in \mathcal{C}(V_o, V_\ell)}(1 - q_e)}{\prod_{<s,t> \in \mathcal{C}(V_o, V_{\ell'})}(1 - q_e)}. \tag{45}$$

q_e's are the edge probabilities.

Example. In image analysis, $\pi(\mathbf{X})$ is a Bayesian posterior $\pi(\mathbf{X}|\mathbf{I}) \propto \mathcal{L}(\mathbf{I}|\mathbf{X})p_o(\mathbf{X})$ with the prior probability $p_o(\mathbf{X})$ being a Markov model (like Potts in Eqn. (37)). Therefore one can compute the ratio of the target probabilities in the local neighborhood of V_o, i.e. ∂V_o.

$$\frac{\pi(\mathbf{X}')}{\pi(\mathbf{X})} = \frac{\mathcal{L}(\mathbf{I}_{V_o}|\mathbf{X}_{V_o} = \ell')}{\mathcal{L}(\mathbf{I}_{V_o}|\mathbf{X}_{V_o} = \ell)} \cdot \frac{p_o(\mathbf{X}_{V_o} = \ell'|\mathbf{X}_{\partial V_o})}{p_o(\mathbf{X}_{V_o} = \ell|\mathbf{X}_{\partial V_o})} \tag{46}$$

Note that $\mathbf{X}_{\partial V_o} = \mathbf{X}'_{\partial V_o}$ in the above equation.

The second ratio in eq.(**??**) is easy to design. For example, we can make it proportional to the likelihood,

$$q(\mathbf{X}_{V_o} = \ell | V_o, \mathbf{X}) = \mathcal{L}(\mathbf{I}_{V_o} | \mathbf{X}_{V_o} = \ell), \quad \forall \ell. \tag{47}$$

Therefore,

$$\frac{q(\mathbf{X}_{V_o} = \ell | V_o, \mathbf{X}')}{q(\mathbf{X}_{V_o} = \ell' | V_o, \mathbf{X})} = \frac{\mathcal{L}(\mathbf{I}_{V_o} | \mathbf{X}_{V_o} = \ell)}{\mathcal{L}(\mathbf{I}_{V_o} | \mathbf{X}_{V_o} = \ell')} \tag{48}$$

It cancels the likelihood ratio in eqn.(46). We get

Proposition 5 *The acceptance probability for the proposed cluster flipping using the proposal (47) is,*

$$\alpha(\mathbf{X} \to \mathbf{X}') = \min\{1, \frac{\prod_{<s,t> \in \mathcal{C}(R, V_{\ell'})}(1 - q_e)}{\prod_{e \in \mathcal{C}(V_o, V_\ell)}(1 - q_e)} \cdot \frac{p_o(\mathbf{X}_{V_o} = \ell' | \mathbf{X}_{\partial V_o})}{p_o(\mathbf{X}_{V_o} = \ell | \mathbf{X}_{\partial V_o})}\}. \tag{49}$$

The result above states that the computation is limited to a local neighborhood of V_o defined by the prior model. This result is also true if one changes the clustering step by growing V_o from a vertex, i.e. the Wolff modification.

In the experiments on image analysis, our cluster sampling method is empirically $O(100)$ times faster than the single site Gibbs sampler in terms of CPU time. We refer to plots and comparison in Figs.(10), (11) and (12) in section (7) for details.

6 Variants of the cluster sampling method

In this section, we briefly discuss two variants of the cluster sampling method.

6.1 Cluster Gibbs sampling — the "hit-and-run" perspective

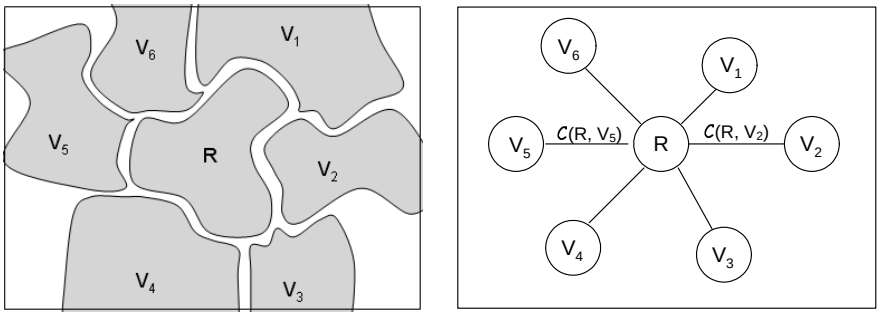
With a slight change, we can modify the cluster sampling method to a generalized Gibbs sampler.

Suppose that $V_o \in \text{CP}(U|\mathbf{X})$ is the candidate chosen in the clustering step, and Fig. 8 shows its cuts with adjacent sets

$$\mathcal{C}(V_o, V_k),\ k = 1, 2, ..., \text{L}(\mathbf{X}).$$

We compute the edge weight γ_k as the strength of connectivity between V_o and $V_k \backslash R$,

$$\gamma_k = \prod_{e \in \mathcal{C}(V_o, V_k)} (1 - q_e). \tag{50}$$



(a) (b)

Figure 8: Illustrating the cluster Gibbs sampler. (a) The cluster V_o has a number of neighboring components of uniform color. (b) The cuts between V_o and its neighboring colors. The sampler follows a conditional probability modified by the edge strength defined on the cuts.

Proposition 6 *Let $\pi(\mathbf{X})$ be the target probability, in the notation above. If R is relabelled probabilistically with*

$$q(\mathbf{X}_{V_o} = k|V_o, \mathbf{X}) \propto \gamma_k \pi(\mathbf{X}_{V_o} = k|\mathbf{X}_{\partial V_o}), \quad k = 1, 2, ..., N(\mathbf{X}), \tag{51}$$

then the acceptance probability is always 1 in the third step.

This yields a generalized Gibbs sampler which flips the color of a cluster according to a modified conditional probability.

Cluster Gibbs Sampler

1. Cluster step: choosing a vertex $v \in V$ and group a cluster V_o from v by the Bernoulli edge probability μ_e.

2. Flipping step: relabel V_o according to eqn. (51).

The traditional single site Gibbs sampler (Geman and Geman 1984) is a special case when $q_e = 0$ for all e and thus $V_o = \{v\}$ and $\gamma_k = 1$ for all k.

One may also view the above method from the perspective of hit-and-run. In continuous state space, a hit-and-run method (see Gilks etc 1996) chooses a new direction $\vec{e}$ (random ray) at time t and then sample on this direction by $a \sim \pi(x + a\vec{e})$. Liu and Wu (1999) extended it ray to any compact groups of actions. In finite state space Ω, one can choose any finite sets $\Omega_a \subset \Omega$ and then apply the Gibbs sampler within the set.

But it is difficult to choose good directions or subsets in hit-and-run methods. In the cluster Gibbs sampler presented above, the subset is selected by the auxiliary variables on the edges.

6.2 The multiple flipping scheme

Given a set of connected components $\mathrm{CP}(\mathbf{U}|\mathbf{X})$ (see eqn. (41)) after the clustering step, instead of flipping a single component R, we can flip all (or any chosen number of) connected components simultaneously. There is room for designing the proposal probabilities for labeling these connected components, independently or jointly. In what follows, we assume the labels are chosen independently for each connected component $\mathrm{cp} \in \mathrm{CP}(\mathbf{U}|\mathbf{X})$, by sampling from a proposal probability $q(\mathbf{X}_{\mathrm{cp}} = l|\mathrm{cp})$. Suppose we obtain a new label $\mathbf{X}'$ after flipping. Let $E_{\mathrm{on}}(\mathbf{X}) \subset E$ and $E_{\mathrm{on}}(\mathbf{X}') \subset E$ be the subsets of edges that connect the vertices of same color in $\mathbf{X}$ and $\mathbf{X}'$ respectively. We define two cuts by the differences of the sets

$$\mathcal{C}(\mathbf{X} \to \mathbf{X}') = E_{\mathrm{on}}(\mathbf{X}') - E_{\mathrm{on}}(\mathbf{X}), \text{ and } \mathcal{C}(\mathbf{X}' \to \mathbf{X}) = E_{\mathrm{on}}(\mathbf{X}) - E_{\mathrm{on}}(\mathbf{X}'), \tag{52}$$

We denote the set of connected components which have different colors before and after the flipping by $D(\mathbf{X}, \mathbf{X}') = \{\mathrm{cp} : \mathbf{X}_{\mathrm{cp}} \neq \mathbf{X}'_{\mathrm{cp}}\}$.

Proposition 7 *The acceptance probability of the multiple flipping scheme is*

$$\alpha(\mathbf{X} \to \mathbf{X}') = \min\{1, \frac{\prod_{e \in \mathcal{C}(\mathbf{X} \to \mathbf{X}')}(1-q_e)}{\prod_{e \in \mathcal{C}(\mathbf{X}' \to \mathbf{X})}(1-q_e)} \frac{\prod_{\mathrm{cp} \in D(\mathbf{X}, \mathbf{X}')} q(\mathbf{X}'_{\mathrm{cp}}|\mathrm{cp})}{\prod_{\mathrm{cp} \in D(\mathbf{X}, \mathbf{X}')} q(\mathbf{X}_{\mathrm{cp}}|\mathrm{cp})} \cdot \frac{p(\pi')}{p(\pi)}\} \tag{53}$$

Observe that when $D = \{V_o\}$ is a single connected component, this reduces to Theorem **??**.

It is worth mentioning that if we flip all connected components simultaneously, then the Markov transition graph of $\mathcal{K}(\mathbf{X}, \mathbf{X}')$ is fully connected, i.e.

$$\mathcal{K}(\mathbf{X}, \mathbf{X}') > 0, \quad \forall \mathbf{X}, \mathbf{X}' \in \Omega. \tag{54}$$

This means that the Markov chain can walk between any two partitions in a single step.

7 Experiment 1: image segmentation

Our first experiment tests the cluster sampling algorithm in an image segmentation task. The objective is to partition the image into a number of disjoint regions (as Figs.5 and 6 have shown) so that each region has consistent intensity in the sense of fitting to some image models. The final result should optimize a Bayesian posterior probability $\pi(\mathbf{X}) \propto \mathcal{L}(\mathbf{I}|\mathbf{X})p_o(\mathbf{X})$.

In such problem, $\mathbf{G}$ is an adjacency graph with vertices V being a set of atomic regions (see Figs.(5) and (6)). Usually $|V| = O(10^2)$. For each atomic region $v \in V$, we compute a 15-bin intensity histogram h normalized to 1. Thus the edge probability is calculated as

$$q_{ij} = p(\mu_e = \mathrm{on}|\mathbf{I}(v_i), \mathbf{I}(v_j)) = \exp\{-\frac{1}{2}(KL(h_i||h_j) + KL(h_j||h_i))\}, \tag{55}$$

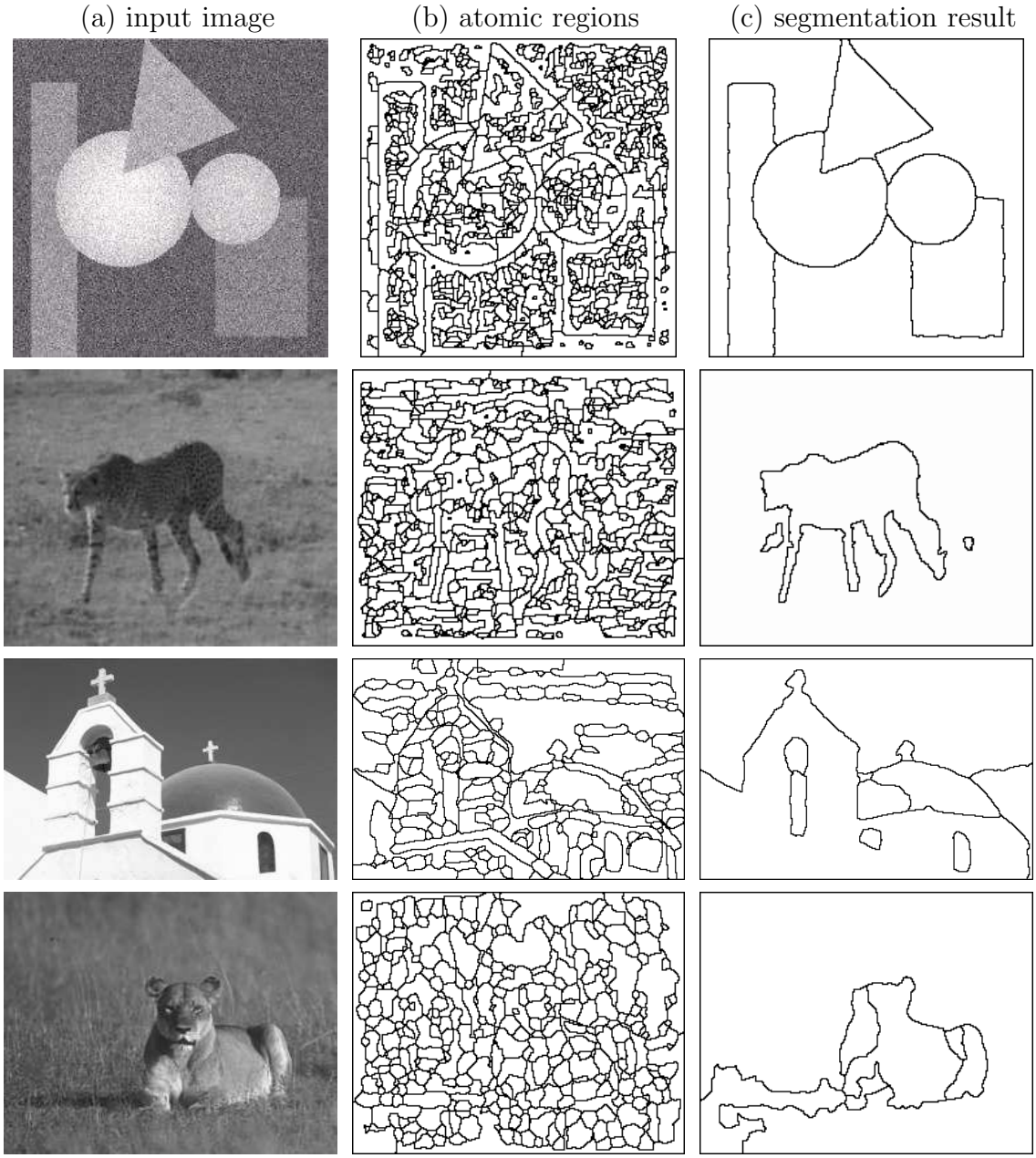


Figure 9: More results for image segmentation.

where $KL()$ is the Kullback-Leibler divergence between the two histograms. Usually q_e should be close to zero for e crossing object boundary. In our experiments, the edge probability leads to good clustering as Fig. 6 shows.

Now we briefly define the target probability in this experiment. Let $\mathbf{X} = (V_1, ..., V_{\mathrm{L}})$ be a coloring of the graph with L being a unknown variable, and the image intensities in each set V_k is consistent in terms of fitting to a model θ_k. Different colors are assumed to be

independent. Therefore, we have,

$$\pi(\mathbf{X}) = \pi(\mathbf{X}|\mathbf{I}) \propto \prod_{k=1}^{\mathrm{L}} [\mathcal{L}(\mathbf{I}(V_k); \theta_k) p_o(\theta_k)] p_o(\mathbf{X}). \tag{56}$$

We selected three types of simple models for the likelihood models to account for different image properties. The first model is a non-parametric histogram $\mathcal{H}$, which in practice is represented by a vector of B-bins $(\mathcal{H}_1, ..., \mathcal{H}_B)$ normalized to 1. It accounts for cluttered objects, like vegetation.

$$\mathbf{I}(x, y;\, \theta_0) \sim \mathcal{H} \text{ iid}, \ \forall (x, y) \in V_k. \tag{57}$$

The other two are regression models for the smooth change of intensities in the two-dimensional image plane (x, y), and the residues follow the empirical distribution $\mathcal{H}$ (i.e. the histogram).

$$\mathbf{I}(x, y;\, \theta_1) = \beta_0 + \beta_1 x + \beta_2 y + \mathcal{H} \text{ iid}, \ \forall (x, y) \in V_k. \tag{58}$$

$$\mathbf{I}(x, y;\, \theta_2) = \beta_0 + \beta_1 x + \beta_2 y + \beta_3 x^2 + \beta_4 xy + \beta_5 y^2 + \mathcal{H} \text{ iid}, \ \forall (x, y) \in V_k. \tag{59}$$

In all cases, the likelihood is expressed in terms of the entropy of the histogram $\mathcal{H}$

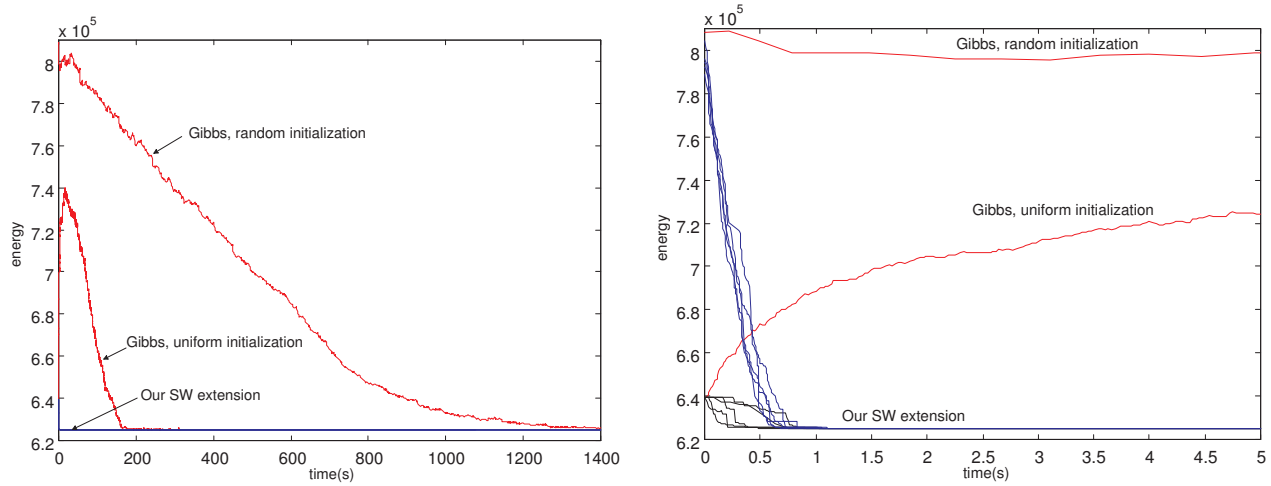
$$\mathcal{L}(\mathbf{I}(V_k); \theta_k) \propto \prod_{v \in V_k} \mathcal{H}(\mathbf{I}_v) = \prod_{j=1}^{B} \mathcal{H}_j^{n_j} = \exp(-|V_k| \text{entropy}(\mathcal{H})). \tag{60}$$

The model complexity is penalized by a prior probability $p_o(\theta_k)$ and the parameters θ in the above likelihoods are computed deterministically at each step as the best least square fit. The deterministic fitting could be replaced by the reversible jumps together with the flipping of color. This was done in (Tu and Zhu, 2002) and is beyond the scope of our experiments.

The prior model $p_o(\mathbf{X})$ encourages large and compact regions with small number of colors, as it was suggested in (Tu and Zhu 2002). Let $r_1, r_2, ..., r_m$, $m \geq \mathrm{L}$ be the connected components of all $V_k, k = 1, ..., \mathrm{L}$. Then the prior is

$$p_o(\mathbf{X}) \propto \exp\{-\alpha_0 \mathrm{L} - \alpha_1 m - \alpha_2 \sum_{k=1}^{m} \text{Area}(r_k)^{0.9}\}. \tag{61}$$

For the image segmentation example (horse) shown in Figs. 5 and 6, we compare the cluster sampling method with the single-site Gibbs sampler and the results are displayed in Fig. 10. Since our goal is to maximize the posterior probability $\pi(\mathbf{X})$, we must add an annealing scheme with a high initial temperature T_o and then decreases to a low temperature (0.05 in our experiments). We plot the $-\ln \pi(\mathbf{X})$ over CPU time in seconds with a Pentium IV PC. The Gibbs sampler needs to raise the initial temperature high (say $T_o \geq 100$)) and uses a slow annealing schedule to reach good solution. The cluster sampling method can



(a) convergence CPU time in seconds (b) Zoomed-in view of the first 5 seconds.

Figure 10: The plot of $-\ln \pi(X)$ over computing time for both the Gibbs sampler and our algorithm for the horse image. Both algorithms are measured by the CPU time in seconds using a Pentium IV PC. So they are comparable. (a). Plot in the first $1,400$ seconds. The Gibbs sampler needs a high initial temperature and slow annealing step to achieve the same energy level. (b). The zoomed-in view of the first 5 seconds.

run at low temperature. We usually raise the initial temperature to $T_o \leq 15$ and use a fast annealing scheme. Fig. 10.(a) plots the two algorithms at the first $1,400$ seconds, and Fig. 10.(b) is a zoomed-in view for the first 5 seconds.

We run the two algorithms with two initializations. One is a random labeling of the atomic regions and thus has higher $-\ln \pi(\mathbf{X})$, and the other initialization sets all vertices to the same color. The clustering methods are run five times on both cases. They all converged to one solution (see Fig.5.(c)) within 1 second, which is $O(10^2)$ times faster than the Gibbs sampler.

Fig.9 shows four more images. Using the sample comparison method as in the horse image, we plot $-\ln \pi(\mathbf{X})$ against running time in Figs. 11 and 12 for the images in the first and second row of Fig.9 respectively. In experiments, we also compared the effect of the edge probabilities. The clustering algorithm are $O(100)$ times slower if we use a constant edge probability $\mu_{ij} = c \in (0, 1)$ as the original SW method does. For example the single-site Gibbs sampler is an example with $q_{ij} = 0$, $\forall\, i, j$.

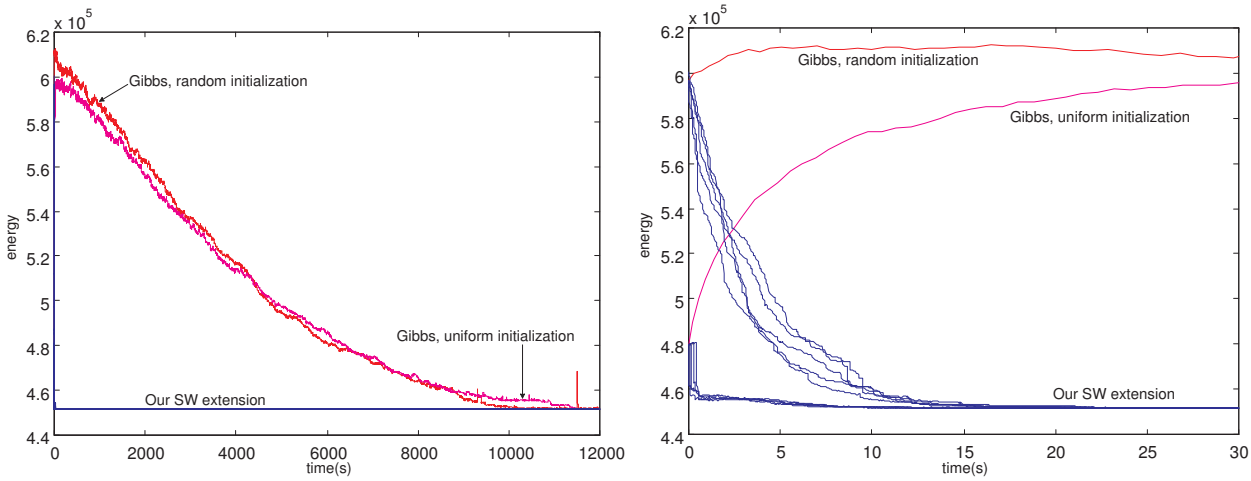


Figure 11: Convergence comparison between the clustering method and Gibbs sampler in CPU time (seconds) on the artificial image (circles, triangle and rectangles) in the first row of Fig.9. (a). The first 1,200 seconds. (Right) Zoomed-in view of the first 30 seconds. The clustering algorithm is run 5 trials for both the random and uniform initializations.

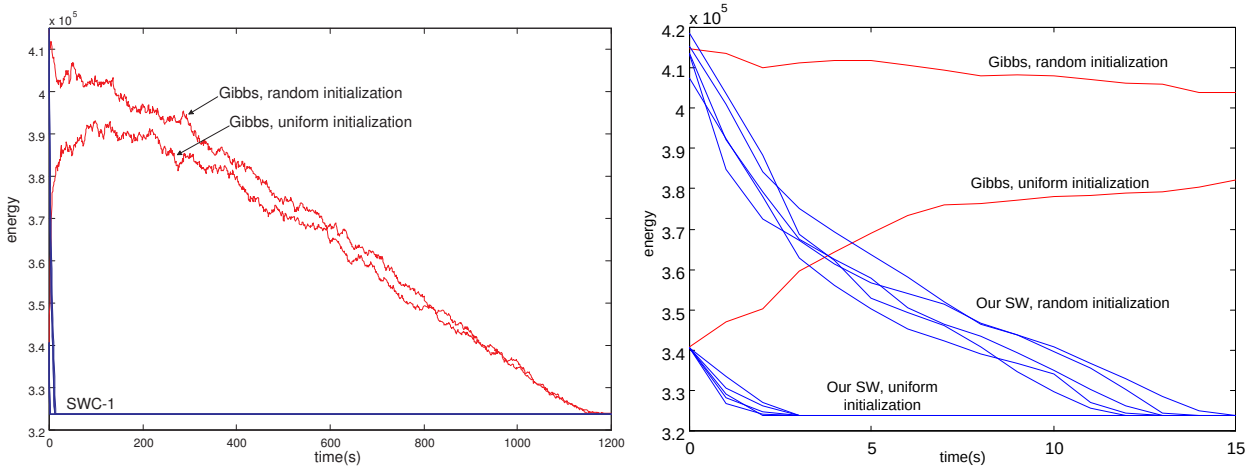


Figure 12: Convergence comparison between the clustering method and Gibbs sampler in CPU time (seconds) on the cheetah image. (Left) The first 1,200 seconds. (Right) Zoomed-in view of the first 15 seconds. The clustering algorithm is run 5 times for both the random and uniform initializations.

References

[1] Barbu, A. and Zhu, S.C. (2003). "Graph partition by Swendsen-Wang cuts", *Proc. Int'l Conf. on Computer Vision*, Nice, France.

[2] Barbu, A. and Zhu, S.C. (2004). "Multigrid and multi-level Swendsen-Wang cuts for

hierarchic graph partition", *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Washington DC, 2004.

[3] Cooper, C. and Frieze, A. (1999). "Mixing properties of the Swendsen-Wang process in classes of graphs", *Random Structures and Algorithms* **15**, no. 3-4, 242-261.

[4] Edwards, R.G. and Sokal, A.D. (1988). "Generalization of the Fortuin-Kasteleyn-Swendsen-Wang representation and Monte Carlo algorithm", *Phys. Rev. Lett.* **38**, 2009-2012.

[5] Geman, S. and Geman, D. (1984),"Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images", *IEEE Trans. on PAMI* **6**, 721-741.

[6] Gore, V. and Jerrum, M (1997). "The Swendsen-Wang process does not always mix rapidly", *Proc. 29th ACM Symp. on Theory of Computing* 674-681.

[7] Hastings, W.K. (1970). "Monte Carlo sampling methods using Markov chains and their applications", *Biometrika* **57**, 97-109.

[8] Higdon, D.M. (1998). "Auxiliary variable methods for Markov chain Monte Carlo with applications", *J. Am. Statist. Assoc.* **93**, 585-595.

[9] Huber, M. (2002). "A bounding chain for Swendsen-Wang." *Random Structures and Algorithms* **22**, no 1, 43-59.

[10] Liu, J.S. (2001). "Monte Carlo strategies in scientific computing", Springer, NY.

[11] Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A. and Teller, E. (1953). "Equations of the state calculations by fast computing machines", *J. Chem. Physics* **22**, 1087-1091.

[12] Potts, R.B. (1953) "Some generalized order-disorder transformations", *Proceedings of the Cambridge Philosophic Society* **48**, 106-109.

[13] Swendsen, R.H. and Wang, J.S. (1987), "Nonuniversal critical dynamics in Monte Carlo simulations", *Physical Review Letters* **58** no. 2, 86-88.

[14] Tanner, M. A. and Wong, W.H. (1987), "The calculation of posterior distributions by data augmentation (with discussion)", *J. Amer. Stat. Assoc.*, 82(398):528-540.

[15] Wolff, U. (1989). "Collective Monte Carlo updating for spin systems", *Physical Review Letters* **62**, no. 4, 361-364.