Beyond Windability: An FPRAS for The Six-Vertex Model

Zhiguo Fu
School of Information Science and Technology and KLAS, Northeast Normal University, Changchun, China

Junda Li
School of Mathematics and Statistics, Northeast Normal University, Changchun, China

Xiongxin Yang
School of Information Science and Technology, Northeast Normal University, Changchun, China

Abstract

The six-vertex model is an important model in statistical physics and has deep connections with counting problems. There have been some fully polynomial randomized approximation schemes (FPRAS) for the six-vertex model [30, 10], which all require that the constraint functions are windable. In the present paper, we give an FPRAS for the six-vertex model with an unwindable constraint function by Markov Chain Monte Carlo method (MCMC). Different from [10], we use the Glauber dynamics to design the Markov Chain depending on a circuit decomposition of the underlying graph. Moreover, we prove the rapid mixing of the Markov Chain by coupling, instead of canonical paths in [10].

2012 ACM Subject Classification Theory of computation → Design and analysis of algorithms

Keywords and phrases The Six-Vertex Model, MCMC, Windability, Coupling

Category Track A: Approximation and Online Algorithms

Acknowledgements We want to thank anonymous reviewers for their helpful comments.

1 Introduction

The six-vertex model originates in statistical mechanics for crystal lattices with hydrogen bonds. A state of the model consists of an arrow on each edge such that the number of arrows pointing inwards at each vertex is exactly two. This 2-in-2-out law on the arrow configurations is called the ice rule (It is also called the ice-type model). Thus there are six permitted types of local configurations around a vertex, hence the name six-vertex model (See Figure 1). The six configurations 1 to 6 are associated with six possible weights $\omega_1, \omega_2, \cdots, \omega_6$. 

![Figure 1: Valid configurations of the six-vertex model](image)

On a 4-regular graph $G$, the partition function of the six-vertex model is

$$Z(G; \omega_1, \omega_2, \cdots, \omega_6) = \sum_{\sigma} \prod_{i=1}^{6} \omega_i^{n_i},$$
where $\sigma$ is the set of the orientation of $G$ such that the incident edges of each vertex are 2-in-2-out, and $n_i$ is the number of vertices in type $i$ ($1 \leq i \leq 6$). Since the six-vertex model was introduced by Linus Pauling in 1935 to describe the properties of ice [31], it has attracted considerable attentions in physics, chemistry and mathematics.

Note that the partition function of the six-vertex model can be considered as a sum-of-product computation. Thus it is a counting problem naturally. For example, if all the weight are 1, then the value of the partition function is the number of the Eulerian orientations of the underlying graph. In the 1990s, Zeilberger [38], Kuperberg [25] showed the connection of the alternating sign matrix and the six-vertex model. It is also known that the six-vertex model are related to the Tutte polynomial [35, 37, 15]. In general, the computational complexity of counting problems was studied in two classical frameworks: Graph homomorphisms(GH), Counting Constraint Satisfaction Problems(#CSP). Holant problems is a new framework which is expressive enough to contain GH and #CSP as special cases [36, 12]. A Holant instance is a graph equipped with some local constraint functions. The six-vertex model is a Holant problem and the constraint function is determined by the parameters $\omega_1, \omega_2, \ldots, \omega_6$. In particular, we note that the six-vertex model can not be expressed by GH and #CSP. A series theorems of complexity classifications were built for GH and #CSP for exact computation [16, 4, 3, 19, 11, 6, 9, 21, 15, 7, 26, 1] and approximation complexity [22, 34, 32, 27, 18, 23, 17]. But the results are very limited for Holant problems, in particular for approximation complexity.

The six-vertex model is an important base case to study Holant problems with asymmetric constraint functions. And the computational complexity of the six-vertex model was investigated in the context of Holant problems. For the exact computation, there are the complexity classification of the six-vertex model on general graphs and planar graphs respectively [7, 8]. They proved that the six-vertex model can be divided into three categories: (1) tractable on general graphs; (2) tractable on planar graphs but #P-hard on general graphs; (3) #P-hard even on planar graphs. For the approximate complexity, Mihail and Winkler gave the FPRAS for the unweighted case, i.e., counting the number of the Eulerian orientations of the underlying graph [30]. For the weighted case, Cai etc. showed that the approximate complexity of the six-vertex model is dramatically different on the two sides separated by the phase transition threshold from physics [10]. They showed that there is no FPRAS for the six-vertex model in anti-ferroelectric phase if RP$\neq$NP. But for the remaining area, they just gave FPRAS by MCMC when the constraint function is “windable” and left a gap to the phase transition threshold.

Winding is proposed by McQuillan to design the canonical paths in a systematic way in [29]. Canonical path is an important tool to prove rapid mixing of the Markov Chain when designing MCMC. However, it is a difficult task to design the canonical paths that provide the routing as low congestion as possible for all state pairs of Markov Chain. McQuillan reduced the task of designing canonical paths to solving a set of linear equations, which makes design of canonical paths easier. In details, for a Holant problem, if all the constraint functions are “windable”, then we can find the canonical paths automatically. But proving windable property concisely for the constraint functions of counting problems is still a difficult problem. Therefore, Huang etc. simplified the conditions to check if a function is windable in [20]. From then on, the application of windability has been further extended. The FPRAS in [10, 11] are all for Holant problems with windable constraint functions.

In the present paper, we give an FPRAS for the six-vertex model with unwindable constraint functions. We design a Markov Chain depending on a circuit decomposition instead of the directed-loop algorithm in [10], and use path coupling to prove the rapid
mixing of the Markov Chain instead of canonical paths in [10]. In details, we consider the six-vertex model with the parameters $\omega_1 = 1$, $\omega_3 = \omega_4 = b > 0$ and $\omega_5 = \omega_6 = 0$, which produce an unwindable constraint function. According to the constraint function, the underlying graph can be decomposed into a set of circuits $\mathcal{C}$, and for each circuit in $\mathcal{C}$ we can define assignments which one-to-one corresponds to the valid configurations of the six-vertex model. Then we define a Markov Chain on the state space consisted of the assignments of $\mathcal{C}$ by the Glauber dynamics. We proved the Markov Chain can be mixing in $O(n \log \frac{n}{\delta})$ for $b \leq \frac{1}{\delta}$, where both $n$ and $\delta$ are less than the number of the vertices of the underlying graph. For some technical difficulties, we just prove the rapid mixing of the Markov Chain when the underlying graph is two-by-two-intersection free. We believe this condition is unnecessary and will resolve it in the future.

The paper is organized as follows:

- In Section 2, we formalize our model and prove that its constraint function is unwindable.
- In Section 3, we design the Markov Chain whose state space $\Omega$ is the set of all valid configurations to sample from the six-vertex model, which is closely related to computing the partition function [24] [34].
- In Section 4, we show that Markov Chain rapidly converges to its stationary distribution by path coupling, i.e., we find an efficient method to sample from the six-vertex model, and as an FPRAS to approximate the partition function.

## 2 Preliminaries

### 2.1 The six-vertex model

The six-vertex model is naturally expressed as a Holant problem, which we define as follows. A function $f : \{0, 1\}^k \rightarrow \mathbb{C}$ is called a constraint function of arity $k$. In this paper we restrict $f$ to take non-negative values in $\mathbb{Q}^+$. Fix a set of constraint functions $\mathcal{F}$. A function grid $\Gamma = (G, \xi)$ is a tuple, where $G = (V, E)$, $\xi$ labels each $v \in V$ with a function $f_v \in \mathcal{F}$ of arity $\deg(v)$, and the incident edges $E(v)$ at $v$ are identified as input variables to $f_v$, also labeled by $\xi$. Every assignment $\sigma : E \rightarrow \{0, 1\}$ gives an evaluation $\prod_{v \in V} f_v (\sigma|_{E(v)})$, where $\sigma|_{E(v)}$ denotes the restriction of $\sigma$ to $E(v)$. The problem Holant($\mathcal{F}$) on an instance $\Gamma$ is to compute

$$\text{Holant} (\Gamma; \mathcal{F}) = \sum_{\sigma : E \rightarrow \{0, 1\}} \prod_{v \in V} f_v (\sigma|_{E(v)}).$$

We use Holant ($\mathcal{F}$|$\mathcal{G}$) for Holant problems over function grids with a bipartite graph $(U, V, E)$ where each vertex in $U$ (or $V$) is assigned a function in $\mathcal{F}$ (or $\mathcal{G}$, respectively).

To write the six-vertex model on a 4-regular graph $G = (V, E)$ as a Holant problem, consider the edge-vertex incidence graph $G' = (U_E, U_V, E')$ of $G$. We model the orientation of an edge in $G$ by putting the Disequality function (\not=2) (which outputs 1 on inputs 01, 10 and outputs 0 on 00, 11) on $U_E$ in $G'$. We say an orientation on edge $e = (w, v) \in E$ is going out $w$ and into $v$ in $G$ if the edge $(u_{w}, u_{v})$ in $G'$ takes value 1 (and $(u_{v}, u_{w}) \in E'$ takes the value 0). In the following, we consider $G'$ as the underlying graph of the six-vertex model. We write an 4-ary function $f(x_1, x_2, x_3, x_4)$ as a matrix $M(f) = M_{x_1,x_2,x_3,x_4}(f) =

$$
\begin{pmatrix}
0000 & f_{0001} & f_{0010} & f_{0011} \\
0100 & f_{0101} & f_{0110} & f_{0111} \\
1000 & f_{1001} & f_{1010} & f_{1011} \\
1100 & f_{1101} & f_{1110} & f_{1111}
\end{pmatrix}.
$$

Then the constraint function for the six-vertex model can
be expressed as a function with \( M(f) = \begin{pmatrix} 0 & 0 & 0 & w_1 \\ 0 & w_2 & w_3 & 0 \\ 0 & w_4 & w_5 & 0 \\ w_6 & 0 & 0 & 0 \end{pmatrix} \). If \( w_1 = w_6 = a, w_3 = w_4 = b, w_2 = \omega_5 = c \), then we say the six-vertex model has arrow reversal symmetry. In [10], they proved there is no FPRAS if \( a + b > c \) (anti-ferroelectric) unless \( \text{RP}=\text{NP} \), and gave an FPRAS for \( a^2 + b^2 < c^2 \). In the present paper, we consider the six-vertex model with the constraint function

\[
M(f^*) = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & b & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}.
\]

Note that \( f^* \) has no arrow reversal symmetry. Moreover, the exact computation of the six-vertex model with \( f^* \) is \( \#P \)-hard [5].

2.2 Windable constraint function

\textbf{Definition 1.} (Windable). For any finite set \( J \) and any configuration \( x \in \{0, 1\}^J \) define \( \mathcal{M}_x \) to be the set of partitions of \( \{i \mid x_i = 1\} \) into pairs and at most one singleton. A function \( f : \{0, 1\}^J \to \mathbb{Q}^+ \) is windable if there exist values \( B(x, y, M) \geq 0 \) for all \( x, y \in \{0, 1\}^J \) and all \( M \in \mathcal{M}_x \) satisfying:

1. \( f(x) f(y) = \sum_{M \in \mathcal{M}_x} B(x, y, M) \) for all \( x, y \in \{0, 1\}^J \).
2. \( B(x, y, M) = B(x \oplus S, y \oplus S, M) \) for all \( x, y \in \{0, 1\}^J \) and \( S \in M \in \mathcal{M}_x \).

Here \( x \oplus S \) denotes the vector obtained by changing \( x \) to \( 1 - x \) for the one or two elements \( i \) in \( S \).

Note that for the six-vertex model with arrow reversal symmetry, the function is windable if \( a^2 + b^2 < c^2 \), i.e., [10] gave an FPRAS for the six-vertex model when the function is windable. We will prove that the function in Equation \( \text{[1]} \) is unwindable in the following lemma, i.e., we will give an FPRAS for the six-vertex model without windability.

\textbf{Lemma 2.} The constraint function \( f^* \) with \( b \neq 0 \) in [7] is unwindable.

\textbf{Proof.} Suppose that the constraint function \( f^* \) is windable. Then for \( x=0110 \) and \( y=1001 \), we have

\[ \mathcal{M}_{x \oplus y} = \{ M_1 : \{(x_1, x_2), (x_3, x_4)\}, M_2 : \{(x_1, x_3), (x_2, x_4)\}, M_3 : \{(x_1, x_4), (x_2, x_3)\} \}, \]

and there exist \( B(x, y, M) \geq 0 \) such that \( f^*(0110) \cdot f^*(1001) = \sum_{M \in \mathcal{M}_x \oplus y} B(x, y, M) = b^2 \) by Definition [7]. Moreover, we have \( B(x, y, M) = B(x \oplus S, y \oplus S, M) \) for \( S = \{x_1, x_2\} \) by Definition [1], i.e.,

\[ B(0110, 1001, M_1) = B(0110 \oplus \{x_1, x_2\}, 1001 \oplus \{x_1, x_2\}, M_1) = B(1010, 0101, M_1). \]

Since \( f^*(0110) \cdot f^*(0101) = \sum_{M \in \mathcal{M}_x \oplus y} B(x, y, M) = 0 \), we have

\[ B(0110, 1001, M_1) = B(1010, 0101, M_1) = 0, \]

(2)

Similarly, for \( S = \{x_1, x_3\} \) and \( S = \{x_1, x_4\} \), we have

\[ B(0110, 1001, M_2) = B(1100, 0011, M_2) = 0 \]

(3)
and
\[ B(0110, 1001, M_3) = B(1111, 0000, M_3) = 0. \]  
(4)

The equations (2), (3) and (4) imply that \( f^*(0110) \cdot f^*(1001) = b^2 = 0 \). This contradicts that \( b \neq 0 \).

\[ \Box \]

3 Machinery

A valid configuration of the six-vertex model is an assignment to all edges which produces a non-zero evaluation. Let \( \Omega \) be the set of all valid configurations. In this section, we design a Markov Chain for the six-vertex model with the state space \( \Omega \). And then we introduce coupling to prove the rapid mixing of the Markov Chain.

3.1 Markov chain of The six-vertex model

Dividing the four variables of the constraint function \( f^* \) into two pairs \((x_1, x_3)\) and \((x_2, x_4)\), note that the constraint function \( f^* \) forces the variables in the same pair take opposite values in each valid configuration, i.e. the corresponding edges have to be incoming and outgoing pairwisely. Inspired by this fact, we define a circuit decomposition for the underlying graph (See Figure 2 for an example)

- Taking any vertex as the initial point, the path crosses the vertex of degree 2 directly;
- for the vertices of degree 4, the path travels the edges corresponding to the variables in the same pair through the vertices of degree 4, i.e., if a path enters a vertex in the edge \( x_1 \) or \( x_3 \) (\( x_2 \) or \( x_4 \)), then it leaves the vertex in the edge \( x_3 \) or \( x_1 \) (\( x_4 \) or \( x_2 \)) respectively;
- when the path travels back to the initial point, it forms a circuit. Delete this circuit from the graph and repeat the process until the graph is empty.

![Figure 2](image)

Figure 2 An example of circuit decomposition: the red edges and the black edges form two circuits \( C_1 (v_1 \to u_1 \to v_2 \to u_3 \to v_1) \) and \( C_2 (v_2 \to u_2 \to v_1 \to u_4 \to v_2) \). In the same circuit, the dashed edges and the solid edges take opposite assignments.

We denote the circuit decomposition by \( \mathcal{C} \) in the following and assume that \( n = |\mathcal{C}| \). Now we redefine the valid configuration of the six-vertex model in terms of the circuit decomposition \( \mathcal{C} \) before constructing the Markov Chain. Note that the constraint function \( f^* \) forces the edges in the same circuit in \( \mathcal{C} \) take the values \( \{0, 1\} \) alternately. For each circuit \( C_i \), we set an initial edge \( e_i \) which corresponds to the variable \( x_1 \) or \( x_2 \) (note that each circuit in \( \mathcal{C} \) travels at least one edge corresponding to the variable \( x_1 \) or \( x_2 \)). The assignments of all the edges in \( C_i \) are determined by the assignment of \( e_i \) in a valid configuration. Thus we can define the assignment of the circuit \( C_i \) as the value of the initial edge \( e_i \), i.e., a valid
configuration assigns a value for each circuit. Conversely, if we assign a value for each $C_i \in \mathcal{C}$, then it gives a configuration for the six-vertex model. In the following, we set $\sigma$ is a 0-1 string $(C_1 C_2 \ldots C_{|\mathcal{C}|})$ where each $C_i \in \mathcal{C}$ takes the value 0 or 1 and denote it by $\sigma(C_i) = 0$ or 1. We say that $\sigma$ is a configuration since it assigns a value for each edge in fact. If two circuits $C_i$ and $C_j$ share a common vertex, then we say that $C_i$ is a neighbor of $C_j$. Note that $f^*(1100) = 0$. Thus for two adjacent circuits $C_i, C_j$, there is at most one which takes the value 1 in a valid configuration. We denote the set of neighbors of $C_i$ by $\Gamma(C_i)$ and set

$$\delta_{C_i} = |\Gamma(C_i)|$$

and

$$\delta = \max_{C_i \in \mathcal{C}} \delta_{C_i}.$$

Now we define the Markov Chain $\{Z_t\}$ on the state space $\Omega$ by the similar idea with [28]. For any configuration $\sigma \in \Omega$, the transition probability matrix $P$ of $\{Z_t\}$ is as follows:

- Choose an arbitrary circuit $C_i \in \mathcal{C}$ at random.
- Let

$$\sigma' = \begin{cases} 
\text{only move } C_i \text{ to 1 in } \sigma \text{ with the probability } \frac{b^{C_i}}{1+b^{C_i}}, \\
\text{only move } C_i \text{ to 0 in } \sigma \text{ with the probability } \frac{1}{1+b^{C_i}}.
\end{cases}$$
- If $\sigma'$ is a valid configuration, move to state $\sigma'$, otherwise remain at state $\sigma$.

**Proposition 3.** The transition probability matrix $P$ of $\{Z_t\}$ has the following properties:

- **aperiodicity:** $\gcd\{t : P_{ij}^t > 0\} = 1$ for all $\sigma_i \in \Omega$;
- **irreducibility:** there exists $t$ such that there is a positive probability of going from state $\sigma_i$ to state $\sigma_j$ after $t$ steps, $P_{ij}^t > 0$, for all $\sigma_i, \sigma_j \in \Omega$.

**Proof.** For each state $\sigma_i \in \Omega$, there is a positive probability such that it stays the current state. Thus we have $\gcd\{t : P_{ii}^t > 0\} = 1$ and aperiodicity is proved. For each state $\sigma \in \Omega$, by moving the circuits which take the value 1 in $\sigma$ to 0 step by step, there is a path to the state $\sigma_0$ in which all the circuits take the value 0 and vice versa, i.e., for any two states, there is at least one path connected them by crossing $\sigma_0$. Irreducibility is proved.

Based on the aperiodicity and irreducibility, we know that the chain $\{Z_t\}$ has a unique limiting distribution which is called the stationary distribution $\pi$, i.e.

$$\lim_{t \to \infty} P_{ij}^t = \pi_j, \text{ for all } \sigma_i, \sigma_j \in \Omega.$$ 

The distribution of the partition function $\mu$ is as follows

$$\mu(\sigma) = \frac{\prod_{C_i : \sigma(C_i)=1} b^{C_i}}{\sum_{\sigma \in \Omega} \prod_{C_i : \sigma(C_i)=1} b^{C_i}}.$$ 

Note that $\sum_{\sigma \in \Omega} \prod_{C_i : \sigma(C_i)=1} b^{C_i}$ is the partition function of the six-vertex model with the constraint function $f^*$. Moreover, we emphasize that the distribution $\mu$ is the unique stationary distribution of $\{Z_t\}$, because $\mu$ satisfies the detailed balance conditions

$$\mu_i P_{ij}^t = \mu_j P_{ji}^t,$$

for all $\sigma_i, \sigma_j \in \Omega$. Our goal is to bound the time when the chain is close to the stationary distribution, i.e., we are interested in the mixing time $\tau$:

$$\tau(\varepsilon) = \max \min \{t : \Delta_i(t') \leq \varepsilon \text{ for all } t' \geq t\}$$

for all $\sigma_i$. The Markov chain $\{Z_t\}$ is FPRAS for the six-vertex model when $\tau(\varepsilon)$ is small enough.
where $\Delta_i(t)$ is the total variation distance between $P_t^i$ and $\pi$, i.e.,

$$\Delta_i(t) = \frac{1}{2} \sum_{\sigma_j \in \Omega} |P_t^i - \pi_j|.$$ 

### 3.2 Coupling

We use coupling to prove the rapid mixing of $\{Z_t\}$. Coupling constructs a stochastic process $(X_t, Y_t)$ such that $\{X_t\}$ and $\{Y_t\}$ are copies of the $\{Z_t\}$ and if $X_t = Y_t$, then $X_{t+1} = Y_{t+1}$. To reduce the difficulty of constructing the coupling, we use path coupling in [2] as a tool in the following analysis.

Firstly, we define the notion of neighbors and paths in $\Omega$. For a pair of states $\sigma, \sigma_{C_i} \in \Omega$, if $\sigma(C_i) = 0$, $\sigma_{C_i}(C_i) = 1$ and $\sigma(C_j) = \sigma_{C_i}(C_j)$ for $j \neq i$, then $\sigma$ and $\sigma_{C_i}$ are neighbors and denote it by $\sigma \sim \sigma_{C_i}$. We call $\tau = (\tau_0, \tau_1, \cdots, \tau_k)$ a simple path if all $\tau_i$ are distinct and $\tau_0 \sim \tau_1 \sim \cdots \sim \tau_k$. And for each $\sigma, \eta \in \Omega$, we denote the set of the simple paths connected them by

$$\rho(\sigma, \eta) = \{ \tau : \sigma = \tau_0, \eta = \tau_k, \tau \text{ is a simple path} \}.$$ 

We assume that $X_t = \sigma$ and $Y_t = \sigma_{C_i}$, and denoted the next state by $X_{t+1} = \sigma'$ and $Y_{t+1} = \sigma'_{C_i}$ respectively. Note that both Markov Chains $\{X_t\}$, $\{Y_t\}$ use the same random walk at each step in path coupling, even if the walk cannot produce a valid state (if the produced state is not valid, the chain will stay the current state). The following lemma about path coupling follows from [13].

**Lemma 4.** Let $\Phi$ be an integer-valued metric defined on $\Omega \times \Omega$ which takes values in $\{0, \ldots, D\}$ and for all $\sigma, \eta \in \Omega$, then there exists $\tau \in \rho(\sigma, \eta)$ such that

$$\Phi(\sigma, \eta) = \sum_i \Phi(\tau_i, \tau_{i+1}).$$

Suppose there exists $\beta < 1$ and a coupling of the Markov Chain such that for all $\sigma, \sigma_{C_i} \in \Omega$:

$$E[\Phi(\sigma', \sigma'_{C_i})] \leq \beta \cdot \Phi(\sigma, \sigma_{C_i}).$$

Then the mixing time is

$$\tau(\varepsilon) \leq \frac{\log(D\varepsilon^{-1})}{1 - \beta}.$$ 

### 4 Analysis

For the circuit decomposition $C$ of the underlying graph of the six-vertex model, if there are no three circuits intersecting with one another, then we say the underlying graph is two-by-two-intersection free. In this section, we analyse the mixing speed of $\{Z_t\}$ on two-by-two-intersection free graphs.

#### 4.1 Potential Function

We will define the potential function as the distance of states associated with the path coupling at step $t$, such that if the potential function is zero, then the two chains will eventually reach the same state. According to path coupling, we consider a pair of states
Beyond Windability: An FPRAS for The Six-Vertex Model

If the next move is on the “bad neighbors” of \( C_i \), then the move will not work in one of chains and causes more differences between two chains. Thus we will define a potential function associated with the number of the “bad neighbors” of \( C_j \).

For a pair of adjacent state \( \sigma \sim \sigma_C \) of \( \{X_t\} \) and \( \{Y_t\} \), the circuit \( C_j \) is blocked if \( C_j \) has at least one neighbor which take the value 1 in both \( \sigma \) and \( \sigma_C \). Denote the set of blocked neighbors of \( C_j \) by

\[
B(\sigma, C_i) = \{ C_j : C_j \in \Gamma(C_i), \text{there exists } C_k \in \Gamma(C_j) \text{ s.t. } \sigma(C_k) = 1 \}.
\]

And the set of unblocked circuits is \( \Gamma(C_i) \setminus B(\sigma, C_i) \) for \( \sigma \) and \( \sigma_C \). For the current state \( (\sigma, \sigma_C) \) of the coupling \( (X_t, Y_t) \), the move on an unblocked circuit only works in \( \sigma \), and it makes more difference between \( \{X_t\} \) and \( \{Y_t\} \). The move on a blocked circuit will keep the difference between the states of \( \{X_t\} \) and \( \{Y_t\} \). Note that unblocked circuits are “bad neighbors” we mentioned above. For some technical reasons, we define the potential function as the number of the neighbors of \( C_i \) minus a positive constant \( w < 1 \) times the number of the blocked neighbors of \( C_i \), instead of the number of the unblocked neighbors of \( C_i \).

Specifically, for \( w = \frac{b}{b+2} \) (note that \( w < 1 \) if \( b \leq \frac{1}{2} \)), the potential function \( \Phi \) is as follows,

\[
\Phi(\sigma, \sigma_C) = \delta_{C_i} - w |B(\sigma, C_i)|. \tag{5}
\]

For any states \( \sigma \) and \( \eta \), the potential function is

\[
\Phi(\sigma, \eta) = \min_{\tau \in \rho(\sigma, \eta)} \sum_1 \Phi(\tau, \tau+1).
\]

This potential function \( \Phi \) clearly satisfies the following conditions for any \( \sigma, \eta \in \Omega \):

\[
\begin{align*}
&\Phi(\sigma, \eta) \leq \Phi(\sigma, \varsigma) + \Phi(\varsigma, \eta), \text{ for any } \varsigma \in \Omega; \\
&\Phi(\sigma, \eta) = \Phi(\eta, \sigma); \\
&\Phi(\sigma, \eta) \geq 0; \\
&\Phi(\sigma, \eta) = 0 \iff \sigma = \eta.
\end{align*}
\]

4.2 Analysis on the Potential Function

Recall that we set \( X_t = \sigma \) and \( Y_t = \sigma_C \), where \( \sigma \) and \( \sigma_C \) are neighbors, \( \sigma_C(C_i) = 1 \) and \( \sigma(C_j) = \sigma(C_j) \) for \( j \neq i \), and the next states are \( X_{t+1} = \sigma' \) and \( Y_{t+1} = \sigma_C' \). Let \( \Phi = \Phi(\sigma, \sigma_C) \) and \( E[|\Delta \Phi|] = E[|\Phi(\sigma', \sigma_C') - \Phi(\sigma, \sigma_C)|] \). We now analyze \( E[|\Delta \Phi|] \) to find the constant \( \beta \) in Lemma 4.

We classify the circuits in \( \mathcal{C} \) into four types: \( C_i, C_j \) as a neighbor of \( C_i, C_k \) as a neighbor of the neighbors of \( C_i, \) and \( C_t \) as others. In the following analysis, we will only discuss the effect of moves on \( C_i, C_j \) and \( C_k \), because the moves on \( C_t \) have no effect on \( \Phi \). For any circuit \( C_x \), let

\[
E[\Delta^{+C_x} \Phi] = E[|\Delta \Phi| \text{ Markov Chain attempts to move } C_x \text{ to } 1],
\]

\[
E[\Delta^{-C_x} \Phi] = E[|\Delta \Phi| \text{ Markov Chain attempts to move } C_x \text{ to } 0],
\]

\[
E[\Delta^{C_x} \Phi] = E[|\Delta \Phi| \text{ Markov Chain attempts to move } C_x],
\]

then

\[
E[\Delta^{C_x}_\Phi] = \frac{b^{C_x}}{1 + b^{C_x}} E[\Delta^{+C_x}_\Phi] + \frac{1}{1 + b^{C_x}} E[\Delta^{-C_x}_\Phi].
\]
By the above notations, we can rewrite $E[\Delta \Phi]$ as follows:

$$E[\Delta \Phi] = \frac{1}{n} \left[ E[\Delta C_i \Phi] + \sum_{C_j \in \Gamma(C_i)} E[\Delta C_j \Phi] + \sum_{C_k \in \Gamma(\Gamma(C_i))} E[\Delta C_k \Phi] \right].$$

(6)

Here, $\Gamma(\Gamma(C_i))$ is the set of the neighbors of the neighbors of $C_i$.

Now we consider the moves on $C_i$, $C_j$ or $C_k$ separately.

1. **Move on $C_i$:**
   In this case, a move on $C_i$ will work in both $\{X_t\}$ and $\{Y_t\}$. After the move, $\{X_t\}$ and $\{Y_t\}$ will reach the same state which is $\sigma$ or $\sigma_{C_i}$. Thus, we have
   $$E[\Delta^{+C_i} \Phi] = -\delta_{C_i} + w |B(\sigma, C_i)|,$$
   i.e.,
   $$E[\Delta C_i \Phi] = -\delta_{C_i} + w |B(\sigma, C_i)|.$$

(7)

2. **Move on $C_j$:**
   Since $C_j$ is a neighbor of $C_i$, $C_j$ must take the value 0 in $\sigma_{C_i}$. Note that $\sigma$ and $\sigma_{C_i}$ differ only at $C_i$, so $C_j$ takes the value 0 in $\sigma$. Thus, $E[\Delta^{-C_j} \Phi] = 0$.

   Consider the move which attempts to move $C_j$ to 1. If $C_j$ is unblocked, then this move will only work in $\{X_t\}$, and we have
   $$\Delta^{+C_j} \Phi = \Phi(\sigma', \sigma'_{C_j}) - \Phi(\sigma, \sigma_{C_j})$$
   $$= \Phi(\sigma_{C_j}, \sigma_{C_j}) - \Phi(\sigma, \sigma_{C_j})$$
   $$\leq \Phi(\sigma_{C_j}, \sigma) + \Phi(\sigma, \sigma_{C_j}) - \Phi(\sigma, \sigma_{C_j})$$
   $$= \delta_{C_j} - w |B(\sigma, C_j)|.$$

   If $C_j$ is blocked, then this move will work neither in $\{X_t\}$ nor in $\{Y_t\}$. And both chains will stay the current state. Thus $\Delta^{+C_j} \Phi = 0$. Thus we have
   $$E[\Delta C_j \Phi] \leq \begin{cases} \frac{\delta_{C_j}}{1+b^{\frac{\alpha_{C_j}}{r}}} \cdot (\delta_{C_j} - w |B(\sigma, C_j)|) & \text{if } C_j \notin B(\sigma, C_i), \\ 0 & \text{otherwise.} \end{cases}$$

(8)

3. **Move on $C_k$:**
   Firstly, we assume that $C_k$ takes the value 1 in $\sigma$, then $C_k$ also takes the value 1 in $\sigma_{C_i}$. The move which attempts to move $C_k$ to 1 will work in both $\{X_t\}$ and $\{Y_t\}$. And both of them will stay the current state. Thus
   $$\Delta^{+C_k} \Phi = 0.$$  

(9)

Then we consider the move which attempts to move $C_k$ to 0. This move will work in both $\{X_t\}$ and $\{Y_t\}$ and it might change some $C_j$ from blocked to unblocked. The set of such $C_j$ is defined as follows:

$$\alpha_{C_k} = \{C_j : C_j \in B(\sigma, C_i), \sigma(C' \cdot C_k) = 0 \text{ for any } C' \in \Gamma(C_j)\}.$$
We have
\[ \Delta^{-C_k} \Phi = \Phi(\sigma', \sigma_{C_i}) - \Phi(\sigma, \sigma_{C_i}) = \delta_{C_i} - w \left| B \left( (\sigma_{C_i})_{C_k}, C_i \right) \right| - (\delta_{C_i} - w |B(\sigma, C_i)|) \]
\[ = w \left( |B(\sigma, C_i)| - |B(\sigma_{C_i})_{C_k}, C_i| \right) \]
\[ = w |\alpha_{C_k}|, \]

where \((\sigma_{C_i})_{C_k}\) denotes the configuration in which all the circuits take the same value as \(\sigma_{C_i}\) except for \((\sigma_{C_i})_{C_k}(C_k) = 0\) and \(\sigma_{C_i}(C_k) = 1\). Thus

\[ \Delta^{-C_k} \Phi = \begin{cases} w |\alpha_{C_k}| & \text{if } \sigma(C_k) = 1, \\ 0 & \text{otherwise.} \end{cases} \] (10)

Secondly, we assume that \(C_k\) takes the value 0 in \(\sigma\). The move which attempts to move \(C_k\) to 0 will work in both \(\{X_t\}\) and \(\{Y_t\}\). And both will stay the current state. Thus

\[ \Delta^{-C_k} \Phi = 0. \] (11)

Then consider the move which attempts to move \(C_k\) to 1. Recall that the graph is two-by-two-intersection free, so \(C_k\) and \(C_i\) are not adjacent. If there exists a neighbor of \(C_k\) which takes the value 1 in \(\sigma\), then this move will not work in neither of two chains. Thus both chains stay the current states and \(\Delta^{+C_k} \Phi = 0\); if all the neighbors of \(C_k\) take the value 0 in \(\sigma\) and denote the fact by \(\sigma(\Gamma(C_k)) = 0\), then this move will work in both chains. Note that moving \(C_k\) to 1 which works in both chains effects the potential function by changing \(C_j\) from unblocked to blocked. The set of such \(C_j\) is defined as follows:

\[ \beta_{C_k} = \{ C_j : C_j \notin B(\sigma, C_i), C_j \in \Gamma(C_i) \cap \Gamma(C_k) \}. \]

We have
\[ \Delta^{+C_k} \Phi = \Phi(\sigma', \sigma'_{C_i}) - \Phi(\sigma, \sigma_{C_i}) = w \left( |B(\sigma, C_i)| - |B(\sigma_{C_i}, C_i)| \right) = -w |\beta_{C_k}|. \]

Thus
\[ \Delta^{+C_k} \Phi = \begin{cases} -w |\beta_{C_k}| & \text{if } \sigma(\Gamma(C_k)) = 0 \text{ and } \sigma(C_k) = 0, \\ 0 & \text{otherwise.} \end{cases} \] (12)

Combining Equation (10) and Equation (11), we have
\[ \Delta^{-C_k} \Phi = \begin{cases} 0 & \text{if } \sigma(C_k) = 0, \\ w |\alpha_{C_k}| & \text{if } \sigma(C_k) = 1. \end{cases} \] (13)

Combining Equation (9) and Equation (12), we have
\[ \Delta^{+C_k} \Phi = \begin{cases} 0 & \text{if } \sigma(C_k) = 1, \\ -w |\beta_{C_k}| & \text{if } \sigma(\Gamma(C_k)) = 0 \text{ and } \sigma(C_k) = 0, \\ 0 & \text{otherwise.} \end{cases} \] (14)
Combining Equation (13) and Equation (14), for any neighbor of neighbors of $C_i$, i.e., $C_k$, we have

$$E[\Delta C_k\Phi] = \begin{cases} \frac{|\alpha_{C_k}|}{1+b|\alpha_{C_k}|}w & \text{if } \sigma(C_k) = 1, \\ -\frac{b|\alpha_{C_k}|}{1+b|\alpha_{C_k}|}w & \text{if } \sigma(\Gamma(C_k)) = 0 \text{ and } \sigma(C_k) = 0, \\ 0 & \text{otherwise.} \end{cases} \quad (15)$$

After analyzing the moves on $C_i$, $C_j$ and $C_k$ separately, we are ready to bound $E[\Delta \Phi]$. It seemed that we can compute $E[\Delta \Phi]$ by Equation (6). But it cannot work in practice, since we do not know the number of $C_k$. Thus, we cannot give a sharp bound for $E[\Delta \Phi]$ by Equation (6). One way to handle this problem is “packing”: Each $C_j$ has a set of $C_k$ which change $C_j$ from unblocked to blocked (or from blocked to unblocked), and we pack such $C_k$ into a set. This idea will give a “holistic” property which tends to give a sharper bound. In the following analysis, we will formalize this idea. For a blocked circuit $C_j \in \Gamma(C_i)$, we define the set

$$\Gamma'(C_j) = \{C_k : C_k \in \Gamma(C_j) \setminus \{C_i\}, C_j \in \alpha_{C_k}\},$$

and

$$E[\Delta^* C_j \Phi] = E[\Delta C_j \Phi] + \sum_{C_k \in \Gamma'(C_j)} \frac{1}{|\alpha_{C_k}|} E[\Delta C_k \Phi].$$

For an unblocked circuit $C_j \in \Gamma(C_i)$, we define

$$\Gamma'(C_j) = \{C_k : C_k \in \Gamma(C_j) \setminus \{C_i\}, C_j \in \beta_{C_k}\}$$

and

$$E[\Delta^* C_j \Phi] = E[\Delta C_j \Phi] + \sum_{C_k \in \Gamma'(C_j)} \frac{1}{|\beta_{C_k}|} E[\Delta C_k \Phi].$$

We can rewrite Equation (6) as

$$E[\Delta \Phi] = \frac{1}{n} \left[ E[\Delta C_i \Phi] + \sum_{C_j \in \Gamma(C_i)} E[\Delta^* C_j \Phi] \right].$$

Firstly, we bound $E[\Delta^* C_j \Phi]$.

- If $C_j$ is blocked.

  By Equation (6), we know that $E[\Delta C_j \Phi] = 0$. According to the definition of $\alpha_{C_k}$, if $C_j \in \alpha_{C_k}$, then there exists $C_k$ such that $C_k \in (\Gamma(C_j))$ and $\sigma(C_k) = 1$, which implies
that \(|\Gamma'(C_j)| \leq 1\). Thus we have

\[
E[\Delta^{+C_j} \Phi] = E[\Delta^{C_j} \Phi] + \sum_{C_k \in \Gamma'(C_j)} \frac{1}{|\alpha_{C_k}|} E[\Delta^{C_k} \Phi]
\]

\[
= 0 + \sum_{C_k \in \Gamma'(C_j)} \frac{1}{|\alpha_{C_k}|} \frac{|w|}{1 + b^\delta_{C_k}}
\]

\[
= \sum_{C_k \in \Gamma'(C_j)} \frac{w}{1 + b^\delta_{C_k}} \leq \sum_{C_k \in \Gamma'(C_j)} \frac{w}{1 + b^\delta}
\]

\[
= |\Gamma'(C_j)| \cdot \frac{w}{1 + b^\delta}
\]

\[
\leq \frac{w}{1 + b^\delta}.
\]

- If \(C_j\) is unblocked.

By Equation (8), we have \(E[\Delta^{C_j} \Phi] \leq \frac{b^\delta_{C_j}}{1 + b^\delta_{C_j}} (\delta_{C_j} - w |B(\sigma, C_j)|)\). Moreover, we have

\[
E[\Delta^{+C_j} \Phi] = E[\Delta^{C_j} \Phi] + \sum_{C_k \in \Gamma'(C_j)} \frac{1}{|\beta_{C_k}|} E[\Delta^{C_k} \Phi]
\]

\[
\leq \frac{b^\delta_{C_j}}{1 + b^\delta_{C_j}} (\delta_{C_j} - w |B(\sigma, C_j)|) - \sum_{C_k \in \Gamma'(C_j), \sigma(C_k) = 0} \frac{1}{|\beta_{C_k}|} \frac{b^\delta_{C_k}}{1 + b^\delta_{C_k}} w
\]

\[
\leq \frac{b^\delta_{C_j}}{1 + b^\delta_{C_j}} (\delta_{C_j} - w |B(\sigma, C_j)|)
\]

\[
\leq \frac{b^\delta}{1 + b^\delta}.
\]

Note that \(w = \frac{b^\delta}{b^\delta + 2}\). Thus \(E[\Delta \Phi]\) can be bounded as follows.

\[
nE[\Delta \Phi] = \left[ E[\Delta^{C_j} \Phi] + \sum_{C_j \in \Gamma'(C_j)} E[\Delta^{+C_j} \Phi] \right]
\]

\[
= \left[ -\delta_{C_j} + w |B(\sigma, C_j)| + \sum_{C_j \in \Gamma'(C_j), C_j \in B(\sigma, C_j)} E[\Delta^{+C_j} \Phi] + \sum_{C_j \in \Gamma'(C_j), C_j \notin B(\sigma, C_j)} E[\Delta^{+C_j} \Phi] \right]
\]

\[
\leq \left[ -\delta_{C_j} + \sum_{C_j \in \Gamma'(C_j), C_j \in B(\sigma, C_j)} \frac{2 + b^\delta}{1 + b^\delta} w + \sum_{C_j \in \Gamma'(C_j), C_j \notin B(\sigma, C_j)} \frac{b^\delta}{1 + b^\delta} \right]
\]

\[
= \frac{1}{(1 + b^\delta)} \left[ -(2 + b^\delta) \delta_{C_j} + |B(\sigma, C_j) (2 + b^\delta) + (\delta_{C_j} - |B(\sigma, C_j)|) b^\delta \right]
\]

\[
= \frac{\delta_{C_j}}{(1 + b^\delta)} (b^\delta - 1 - b^\delta).
\]

(16)

Notice that \(E[\Delta \Phi] \leq 0\), when \(b \leq \frac{1}{2}\).

Now we are ready to bound \(E[\Delta \Phi]\), which will bound the mixing time. We define \(\beta_{\sigma, C_j}\) such that

\[
E[\Phi(\sigma', \sigma'C_j)] = \beta_{\sigma, C_j} \Phi(\sigma, \sigma'C_j).
\]
Thus we have
\[ E[\Delta \Phi] = E[\Phi(\sigma', \sigma'_C)] - \Phi(\sigma, \sigma_C) = (\beta_{\sigma, \sigma_C} - 1) \Phi(\sigma, \sigma_C), \]
and
\[ \beta_{\sigma, \sigma_C} = 1 + \frac{E[\Delta \Phi]}{\Phi(\sigma, \sigma_C)}. \]

According to Lemma 4, we set \( \beta = \max_{\sigma, \sigma_C} \beta_{\sigma, \sigma_C}. \) From Equation (5), i.e., the definition of \( \Phi, \) it is easy to know that \( \Phi(\sigma, \sigma_C) \leq \delta C_i. \) By the bound of \( E[\Delta \Phi] \) in Equation (16), we have
\[ \beta \leq 1 + \frac{1}{n(1 + b^\delta)} (b^\delta - 1 - b^\delta). \]

For any states \( \sigma \) and \( \eta, \) we have \( \Phi(\sigma, \eta) \leq n\delta. \) Thus, we have
\[ E[\Phi(X_t, Y_t)] \leq \beta^t \cdot E[\Phi(X_0, Y_0)] \leq \left( 1 + \frac{1}{n(1 + b^\delta)} (b^\delta - 1 - b^\delta) \right)^t \cdot n\delta \tag{17} \]

Equation (17) and Lemma 4 produce a bound of the mixing time
\[ \tau(\varepsilon) \leq \frac{n}{\delta^2 + 1 - \delta^3} \log \left( \frac{n\delta}{\varepsilon} \right). \]

\begin{itemize}
  \item Theorem 5. (Mixing time of Markov chain). For the six-vertex model with the constraint function \( f^*, \) whose underlying graph is two-by-two-intersection free, the Markov Chain \( \{Z_t\} \) mixes in time \( O\left( \frac{n^{(1+b^\delta)}}{\delta^2 + 1 - \delta^3} \log \left( \frac{n\delta}{\varepsilon} \right) \right) \) for \( \delta \geq 2, b \leq \frac{1}{\delta}. \)
\end{itemize}

References

1. Miriam Backens. A complete dichotomy for complex-valued holant ‘c. In Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Donal Sannella, editors, 45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, July 9-13, 2018, Prague, Czech Republic, volume 107 of LIPIcs, pages 12:1–12:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018. doi:10.4230/LIPIcs.ICALP.2018.12
2. Russ Bubley and Martin E. Dyer. Path coupling: A technique for proving rapid mixing in markov chains. In 38th Annual Symposium on Foundations of Computer Science, FOCS ’97, Miami Beach, Florida, USA, October 19-22, 1997, pages 223–231. IEEE Computer Society, 1997. doi:10.1109/SFCS.1997.646111
3. Andrei A. Bulatov. The complexity of the counting constraint satisfaction problem. J. ACM, 60(5):34:1–34:41, 2013. doi:10.1145/2528400
4. Andrei A. Bulatov, Martin E. Dyer, Leslie Ann Goldberg, Markus Jalsenius, Mark Jerrum, and David Richerby. The complexity of weighted and unweighted #csp. J. Comput. Syst. Sci., 78(2):681–688, 2012. doi:10.1016/j.jcss.2011.12.002
5. Jin-Yi Cai and Xi Chen. Complexity of counting CSP with complex weights. J. ACM, 64(3):19:1–19:39, 2017. doi:10.1145/2822891
Beyond Windability: An FPRAS for The Six-Vertex Model

6 Jin-Yi Cai, Xi Chen, and Pinyan Lu. Graph homomorphisms with complex values: A dichotomy theorem. *SIAM J. Comput.*, 42(3):924–1029, 2013. doi:10.1137/1008040194

7 Jin-Yi Cai, Zhuoqiu Fu, and Shuai Shao. A complexity trichotomy for the six-vertex model. *CoRR*, abs/1704.01657, 2017. URL: http://arxiv.org/abs/1704.01657

8 Jin-Yi Cai, Zhuoqiu Fu, and Mingji Xia. Complexity classification of the six-vertex model. *Inf. Comput.*, 259(Part):130–141, 2018. doi:10.1016/j.ic.2018.01.003

9 Jin-Yi Cai, Heng Guo, and Tyson Williams. A complete dichotomy rises from the capture of vanishing signatures. *SIAM J. Comput.*, 45(5):1671–1728, 2016. doi:10.1137/15M1049798

10 Jin-Yi Cai, Tianyu Liu, and Pinyan Lu. Approximability of the six-vertex model. In Timothy M. Chan, editor, *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019*, San Diego, California, USA, January 6-9, 2019, pages 2248–2261. SIAM, 2019. doi:10.1137/1.9781611975482.136

11 Jin-Yi Cai, Tianyu Liu, Pinyan Lu, and Jing Yu. Approximability of the eight-vertex model. In Shubhangi Saraf, editor, *35th Computational Complexity Conference, CCC 2020*, July 28-31, 2020, Saarbrücken, Germany (Virtual Conference), volume 169 of *LIPIcs*, pages 4:1–4:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPIcs.CCC.2020.4

12 Martin E. Dyer and Catherine S. Greenhill. On markov chains for independent sets. *J. Algorithms*, 35(1):17–49, 2000. doi:10.1006/jagm.1999.1071

13 Martin E. Dyer and David Richerby. An effective dichotomy for the counting constraint satisfaction problem. *SIAM J. Comput.*, 42(3):1245–1274, 2013. doi:10.1137/100811258

14 Joanna A. Ellis-Monaghan and Criel Merino. Graph polynomials and their applications I: the tutte polynomial. In Matthias Dehmer, editor, *Structural Analysis of Complex Networks*, pages 219–255. Birkhäuser / Springer, 2011. doi:10.1007/978-0-8176-4789-6_9

15 Leslie Ann Goldberg, Martin Grohe, Mark Jerrum, and Marc Thurley. A complexity dichotomy for partition functions with mixed signs. *SIAM J. Comput.*, 39(7):3336–3402, 2010. doi:10.1137/090757496

16 Leslie Ann Goldberg and Mark Jerrum. A dichotomy for real weighted holant problems. *SIAM J. Comput.*, 39(7):3336–3402, 2010. doi:10.1137/070697870

17 Leslie Ann Goldberg and Mark Jerrum. Approximating the partition function of the ferromagnetic potts model. *Journal of the ACM (JACM)*, 59(5):1–31, 2012.

18 Leslie Ann Goldberg, Russell A. Martin, and Mike Paterson. Random sampling of 3-colorings in $\mathbb{Z}^2$. *Random Struct. Algorithms*, 24(3):279–302, 2004. doi:10.1002/rsa.20002

19 Heng Guo and Tyson Williams. The complexity of planar boolean #csp with complex weights. In Fedor V. Fomin, Rusins Freivalds, Marta Z. Kwiatkowska, and David Peleg, editors, *Automata, Languages, and Programming - 40th International Colloquium, ICALP 2013*, Riga, Latvia, July 8-12, 2013, Proceedings, Part I, volume 7965 of Lecture Notes in Computer Science, pages 516–527. Springer, 2013. doi:10.1007/978-3-642-39206-1_44

20 Lingxiao Huang, Pinyan Lu, and Chihao Zhang. Canonical paths for MCMC: from art to science. In Robert Krauthgamer, editor, *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016*, Arlington, VA, USA, January 10-12, 2016, pages 514–527. SIAM, 2016. doi:10.1137/1.9781611974331.ch38

21 Sangxia Huang and Pinyan Lu. A dichotomy for real weighted holant problems. *Comput. Complex.*, 25(1):255–304, 2016. doi:10.1007/s00037-015-0118-3

22 Mark Jerrum and Alistair Sinclair. Approximating the permanent. *SIAM J. Comput.*, 18(6):1149–1178, 1989. doi:10.1137/0218077

23 Mark Jerrum, Alistair Sinclair, and Eric Vigoda. A polynomial-time approximation algorithm for the permanent of a matrix with nonnegative entries. *J. ACM*, 51(4):671–697, 2004. doi:10.1145/1008731.1008738
24 Mark Jerrum, Leslie G. Valiant, and Vijay V. Vazirani. Random generation of combinatorial structures from a uniform distribution. *Theor. Comput. Sci.*, 43:169–188, 1986. doi:10.1016/0304-3975(86)90174-X
25 Greg Kuperberg. Another proof of the alternative-sign matrix conjecture. *International Mathematics Research Notices*, 1996(3):139–150, 1996.
26 Jiabao Lin and Hanpin Wang. The complexity of boolean holant problems with nonnegative weights. *SIAM J. Comput.*, 47(3):798–828, 2018. doi:10.1137/17M113304X
27 Michael Luby, Dana Randall, and Alistair Sinclair. Markov chain algorithms for planar lattice structures. *SIAM J. Comput.*, 31(1):167–192, 2001. doi:10.1137/S0097539799360355
28 Michael Luby and Eric Vigoda. Fast convergence of the glauber dynamics for sampling independent sets. *Random Structures & Algorithms*, 15(3-4):229–241, 1999.
29 Colin McQuillan. Approximating holant problems by winding. *CoRR*, abs/1301.2880, 2013. URL: http://arxiv.org/abs/1301.2880
30 Milena Mihail and Peter Winkler. On the number of eulerian orientations of a graph. *Algorithmica*, 16(4/5):402–414, 1996. doi:10.1007/BF01940872
31 Linus Pauling. The structure and entropy of ice and of other crystals with some randomness of atomic arrangement. *Journal of the American Chemical Society*, 57(12):2680–2684, 1935.
32 Dana Randall and Prasad Tetali. Analyzing glauber dynamics by comparison of markov chains. In Claudio L. Lucchesi and Arnaldo V. Moura, editors, *LATIN ’98: Theoretical Informatics, Third Latin American Symposium, Campinas, Brazil, April, 20-24, 1998, Proceedings*, volume 1380 of *Lecture Notes in Computer Science*, pages 292–304. Springer, 1998. doi:10.1007/BFb0054330
33 Alistair Sinclair. Improved bounds for mixing rates of markov chains and multicommodity flow. *Combinatorics, probability and Computing*, 1(4):351–370, 1992.
34 Alistair Sinclair. *Algorithms for random generation and counting - a Markov chain approach*. Progress in theoretical computer science. Birkhäuser, 1993.
35 William Thomas Tutte. A contribution to the theory of chromatic polynomials. *Canadian journal of mathematics*, 6:80–91, 1954.
36 Leslie G. Valiant. Holographic algorithms. *SIAM J. Comput.*, 37(5):1565–1594, 2008. doi:10.1137/070682575
37 Michel Las Vergnas. On the evaluation at (3, 3) of the tutte polynomial of a graph. *J. Comb. Theory, Ser. B*, 45(3):367–372, 1988. doi:10.1016/0095-8956(88)90079-2
38 Doron Zeilberger. Proof of the alternating sign matrix conjecture. *Electron. J. Comb.*, 3(2), 1996. URL: http://www.combinatorics.org/Volumes/VOL3/Abstracts/v3i2r13.html