Fast Algorithms at Low Temperatures via Markov Chains

Zongchen Chen
School of Computer Science, Georgia Institute of Technology, Atlanta, USA
chenzongchen@gatech.edu

Andreas Galanis
Department of Computer Science, University of Oxford, Oxford, UK
andreas.galanis@cs.ox.ac.uk

Leslie Ann Goldberg
Department of Computer Science, University of Oxford, Oxford, UK
leslie.goldberg@cs.ox.ac.uk

Will Perkins
Department of Mathematics, Statistics, and Computer Science,
University of Illinois at Chicago, Chicago, USA
william.perkins@gmail.com

James Stewart
Department of Computer Science, University of Oxford, Oxford, UK
james.stewart@cs.ox.ac.uk

Eric Vigoda
School of Computer Science, Georgia Institute of Technology, Atlanta, USA
ericvigoda@gmail.com

Abstract

For spin systems, such as the hard-core model on independent sets weighted by fugacity $\lambda > 0$, efficient algorithms for the associated approximate counting/sampling problems typically apply in the high-temperature region, corresponding to low fugacity. Recent work of Jenssen, Keevash and Perkins (2019) yields an FPTAS for approximating the partition function (and an efficient sampling algorithm) on bounded-degree (bipartite) expander graphs for the hard-core model at sufficiently high fugacity, and also the ferromagnetic Potts model at sufficiently low temperatures. Their method is based on using the cluster expansion to obtain a complex zero-free region for the partition function of a polymer model, and then approximating this partition function using the polynomial interpolation method of Barvinok. We present a simple discrete-time Markov chain for abstract polymer models, and present an elementary proof of rapid mixing of this new chain under sufficient decay of the polymer weights. Applying these general polymer results to the hard-core and ferromagnetic Potts models on bounded-degree (bipartite) expander graphs yields fast algorithms with running time $O(n \log n)$ for the Potts model and $O(n^2 \log n)$ for the hard-core model, in contrast to typical running times of $n^{O(\log \Delta)}$ for algorithms based on Barvinok's polynomial interpolation method on graphs of maximum degree Δ. In addition, our approach via our polymer model Markov chain is conceptually simpler as it circumvents the zero-free analysis and the generalization to complex parameters. Finally, we combine our results for the hard-core and ferromagnetic Potts models with standard Markov chain comparison tools to obtain polynomial mixing time for the usual spin system Glauber dynamics restricted to even and odd or "red" dominant portions of the respective state spaces.

2012 ACM Subject Classification Theory of computation → Random walks and Markov chains; Theory of computation → Design and analysis of algorithms

Keywords and phrases Markov chains, approximate counting, Potts model, hard-core model, expander graphs

Digital Object Identifier 10.4230/LIPIcs.APPROX-RANDOM.2019.41

Category RANDOM

Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2019).
Editors: Dimitris Achlioptas and László A. Végh; Article No. 41; pp. 41:1–41:14
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Related Version A full version of the paper is available at `https://arxiv.org/abs/1901.06653`, and the theorem numbering here matches that of the full version.

Funding *Zongchen Chen*: Research supported in part by NSF grants CCF-1617306 and CCF-1563838.
Andreas Galanis: The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013) ERC grant agreement no. 334828. The paper reflects only the authors' views and not the views of the ERC or the European Commission. The European Union is not liable for any use that may be made of the information contained therein.
Leslie Ann Goldberg: The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013) ERC grant agreement no. 334828. The paper reflects only the authors' views and not the views of the ERC or the European Commission. The European Union is not liable for any use that may be made of the information contained therein.
Will Perkins: Part of this work was done while WP was visiting the Simons Institute for the Theory of Computing.
James Stewart: The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013) ERC grant agreement no. 334828. The paper reflects only the authors' views and not the views of the ERC or the European Commission. The European Union is not liable for any use that may be made of the information contained therein.
Eric Vigoda: Research supported in part by NSF grants CCF-1617306 and CCF-1563838.

1 Introduction

The hard-core model from statistical physics is defined on the set of independent sets of a graph G, where the independent sets are weighted by a fugacity $\lambda > 0$. The associated Gibbs distribution $\mu_{G,\lambda}$ is defined as follows, for an independent set I:

$$\mu_{G,\lambda}(I) = \frac{\lambda^{|I|}}{Z_{G,\lambda}} \tag{1}$$

where $Z_{G,\lambda} = \sum_{I \in \mathcal{I}(G)} \lambda^{|I|}$ is the hard-core partition function (also called the independence polynomial), $\mathcal{I}(G)$ is the set of independent sets of G, and $\lambda > 0$ is the *fugacity*.

In applications, there are two important computational tasks associated to a spin system such as the hard-core model. Given an error parameter $\varepsilon \in (0,1)$, an ε-approximate counting algorithm outputs a number $\hat{Z}$ so that $e^{-\varepsilon} Z_{G,\lambda} \leq \hat{Z} \leq e^{\varepsilon} Z_{G,\lambda}$, and an ε-approximate sampling algorithm outputs a random sample I with distribution $\hat{\mu}$ so that the total variation distance satisfies $\|\mu_\lambda - \hat{\mu}\|_{TV} < \varepsilon$.

While classical statistical physics is most interested in studying the hard-core model on the integer lattice $\mathbb{Z}^d$, the perspective of computer science is to consider wider families of graphs, such as the set of all graphs, all graphs of maximum degree Δ, or all bipartite graphs of maximum degree Δ.

Almost all proven efficient algorithms for approximate counting and sampling from the hard-core model work for low fugacities (*high temperatures* in the language of statistical physics). In the high temperature regime there are at least three distinct algorithmic approaches to approximate counting and sampling: Markov chains, correlation decay, and polynomial interpolation. One striking advantage of the Markov chain approach is that the algorithms are much faster and simpler than the algorithms from the other approaches. In particular, it is common for a Markov chain sampling algorithm to run in time $O(n \log n)$, e.g., see [8, 10], while typical running times for algorithms based on correlation decay [26, 21] and polynomial interpolation [1] are $n^{O(\log \Delta)}$ where Δ is the maximum degree of the graph.

In general there are no known efficient algorithms at low temperatures (high fugacities), but recently efficient algorithms have been developed for some special classes of graphs including subsets of $\mathbb{Z}^d$ [14], random regular bipartite graphs, and bipartite expander graphs in general [16, 20]. What these bipartite graphs have in common is that for large enough λ, typical independent sets drawn from $\mu_{G,\lambda}$ align closely with one side or the other of the bipartition (the two ground states). This phenomenon is related to the phase transition phenomenon in infinite graphs, and implies the exponentially slow mixing time of local Markov chains [4, 12, 22]. The algorithms introduced in [14] exploit this phenomenon by expressing the partition function $Z_{G,\lambda}$ in terms of deviations from the two ground states, and then using a truncation of a convergent series expansion (the Taylor series or the cluster expansion) to approximate the log partition function. In statistical physics this is called a *perturbative* approach, and while in general it does not work in the largest possible range of parameter space, when it does work it gives a very detailed probabilistic understanding of the model [24, 6, 7].

To apply the perturbative approach at low temperatures, one rewrites the original spin model as a new model in which single spin interactions are replaced by the interaction of connected components representing deviations from a chosen ground state. Such models are called *abstract polymer models*, as detailed below, and have long been used in statistical physics to understand phase transitions. In this paper, we show that once a low temperature spin model has been transformed into an abstract polymer model, Markov chains once again become an effective algorithmic tool. Using this approach we obtain nearly linear and quadratic time sampling algorithms for low temperature models on expander graphs in cases where only $n^{O(\log \Delta)}$-time algorithms were previously known.

1.1 Abstract polymer models

Abstract polymer models, as defined by Gruber and Kunz in 1971 [13], (or "animal models" in Dobruishin's terminology [7]) are an important tool in studying the equilibrium phases of statistical physics models on lattices (e.g. [19, 6] among many others; see [3] for a brief history of their use in statistical physics and combinatorics). Recently they have been used to develop efficient algorithms for sampling and approximating the partition functions of statistical physics models on lattices [14] and expander graphs [16, 20] at low temperatures, the regime in which Markov chains like the Glauber dynamics are known to mix slowly.

We will study the following polymer models. We start with a host graph G and a set $[q] = \{0, \dots, q-1\}$ of spins. For each vertex v, there is a ground-state spin g_v. A polymer γ consists of a connected set of vertices together with an assignment σ_γ of spins from $\{0, \dots, q-1\} \setminus g_v$ to each vertex $v \in \gamma$ (we abuse notation and use γ to denote both the polymer and the associated set of vertices). The size of a polymer, $|\gamma|$, is the number of vertices in γ. The set of all polymers is $\mathcal{P}(G)$.

A polymer model on G consists of a set $\mathcal{C}(G) \subseteq \mathcal{P}(G)$ of "allowed" polymers, and a non-negative weight w_γ for each polymer $\gamma \in \mathcal{C}(G)$. We denote this model by $(\mathcal{C}(G), w)$. Two polymers γ and γ' are "compatible" (written $\gamma \sim \gamma'$) if their distance in the host graph is at least 2; otherwise they are incompatible (written $\gamma \nsim \gamma'$). The state space of allowable configurations is $\Omega = \{\Gamma \subseteq \mathcal{C}(G) \mid \forall \gamma, \gamma' \in \Gamma, \gamma \sim \gamma'\}$.

The partition function of the polymer model is $Z(G) = \sum_{\Gamma \in \Omega} \prod_{\gamma \in \Gamma} w_\gamma$, where the empty set of polymers contributes 1 to the partition function. The Gibbs measure μ_G is the probability distribution on Ω given by $\mu_G(\Gamma) = \frac{\prod_{\gamma \in \Gamma} w_\gamma}{Z(G)}$.

The polymer model is in fact a hard-core model on the "incompatibility graph" of $\mathcal{C}(G)$ (two polymers joined by an edge if they are incompatible), with non-uniform fugacities given by the weights w_γ. However, the geometry inherited from the host graph G and the sizes of the polymers adds additional structure to the model.

▶ **Example 1.** One instance of a polymer model is the hard-core model itself: polymers are single vertices of the graph G, labeled with "1" (for occupied) against a ground state "0" (for unoccupied). Each polymer (vertex) v comes with the weight function $w_v = \lambda$. Then the set of allowable polymer configurations is exactly the set of independent sets of G, and so the polymer model partition function is exactly the partition function of the hard-core model on G.

▶ **Example 2.** A second instance of a polymer model is related to the ferromagnetic q-color Potts model on a graph G (see Definition 8 below). Fix a color $g \in [q]$ to be the ground state color, and define polymers to be connected subgraphs of G of size at most M, with vertices labeled by the remaining colors $[q] \setminus \{g\}$. A polymer γ has weight function $w_\gamma = e^{-\beta B(\gamma)}$ where $B(\gamma)$ is the number of bichromatic edges in γ plus the size of the edge boundary of γ in G. A configuration of compatible polymers maps to a Potts configuration σ in which all connected components of non-g-colored vertices have size at most M, and the weight of σ in the Potts model is exactly the product of the weight functions of the polymers. The polymer model partition function $Z(G)$, with an appropriate choice of M, represents the contribution to the Potts model partition function of colorings with dominant color g.

As with the hard-core model, there are two main computational problems associated to a polymer model: approximate sampling from μ_G and approximate counting of $Z(G)$. We will approach them both via Markov chain algorithms. In general we will be interested in families of polymer models defined on classes of graphs. We denote such a family $(\mathcal{C}(\cdot), w, \mathcal{G})$, where for each graph $G \in \mathcal{G}$, $(\mathcal{C}(G), w)$ is a polymer model. We will always use n to denote the number of vertices of a graph G.

We consider two conditions on the weight functions w_γ and give their algorithmic consequences.

▶ **Definition 1.** *A polymer model $(\mathcal{C}(\cdot), w, \mathcal{G})$ satisfies the polymer mixing condition if there exists $\theta \in (0,1)$ such that*

$$\sum_{\gamma' \nsim \gamma} |\gamma'| w_{\gamma'} \le \theta |\gamma| \tag{2}$$

for all $G \in \mathcal{G}$ and all $\gamma \in \mathcal{C}(G)$.

We postpone the formal definition of mixing time to Section 2 and state our first main result here.

▶ **Theorem 2.** *Suppose that a polymer model $(\mathcal{C}(\cdot), w, \mathcal{G})$ satisfies the polymer mixing condition (2). Then for each $G \in \mathcal{G}$ there is a Markov chain making single polymer updates with stationary distribution μ_G and mixing time $T_{\mathrm{mix}}(\varepsilon) = O(n \log(n/\varepsilon))$.*

Theorem 2 on its own does not guarantee an efficient algorithm for sampling from μ_G because the Markov chain only yields an efficient sampling algorithm if we can implement each step efficiently. We will show that under a stronger condition we can do this.

▶ **Definition 3.** *A polymer model $(\mathcal{C}(\cdot), w, \mathcal{G})$ is said to be* computationally feasible *if, for each $G \in \mathcal{G}$ and each $\gamma \in \mathcal{P}(G)$, we can determine, in time polynomial in $|\gamma|$, whether $\gamma \in \mathcal{C}(G)$, and compute w_γ if it is.*

▶ **Definition 4.** *A computationally feasible polymer model $(\mathcal{C}(\cdot), w, \mathcal{G})$ with q spins on a class $\mathcal{G}$ of graphs of maximum degree Δ satisfies the polymer sampling condition with constant $\tau \geq 5 + 3\log((q-1)\Delta)$ if*

$$w_\gamma \leq e^{-\tau|\gamma|} \tag{3}$$

for all $G \in \mathcal{G}$ and all $\gamma \in \mathcal{C}(G)$.

We have the following theorem.

▶ **Theorem 5.** *If a computationally feasible polymer model $(\mathcal{C}(\cdot), w, \mathcal{G})$ satisfies the polymer sampling condition (3) then for all $G \in \mathcal{G}$ there is an ε-approximate sampling algorithm for μ_G with running time $O(n\log(n/\varepsilon))$.*

Finally, we can use the sampling algorithm and simulated annealing to give a fast approximate counting algorithm.

▶ **Theorem 6.** *If a computationally feasible polymer model $(\mathcal{C}(\cdot), w, \mathcal{G})$ satisfies the polymer sampling condition (3) then for all $G \in \mathcal{G}$ there is a randomized ε-approximate counting algorithm for $Z(G)$ with running time $O((n/\varepsilon)^2\log^2(n/\varepsilon))$ and success probability at least $3/4$.*

Fernández, Ferrari, and Garcia [11] introduced a condition very similar to the polymer mixing condition in the setting of polymer models on $\mathbb{Z}^d$. Their objective was to derive probabilistic properties of polymer models directly, without going through the combinatorics and complex analysis inherent in the cluster expansion for the log partition function. They introduced a continuous time stochastic process whose stationary distribution was the infinite volume Gibbs measure of their polymer model and their version of condition (2) implied an exponentially fast rate of convergence of this process. They remarked that such an approach had the potential to be an efficient computational tool.

Here we take an algorithmic point of view, and use the polymer mixing and sampling conditions to show that a simple discrete time Markov chain mixes rapidly and can be used to design efficient sampling and approximation algorithms. Our approach differs from that of [11] in that while they are interested primarily in the probabilistic properties of spin models on $\mathbb{Z}^d$, we are interested in algorithmic problems involving spin models on general families of graphs. Our setting of discrete time processes on finite graphs is also more suitable to studying algorithmic questions. Our work confirms the central point of [11]: that complex analysis and absolute convergence of the cluster expansion is not necessary to derive many important properties of a polymer model.

1.2 Applications

We apply our results for abstract polymer models to two specific examples: the ferromagnetic Potts model and the hard-core model on expander graphs. To state these results we need some definitions.

▶ **Definition 7.** *Let $\alpha > 0$. A graph G is an α-expander graph if for all $S \subset V(G)$ with $|S| \leq |V(G)|/2$, we have $e(S, S^c) \geq \alpha|S|$, where $S^c = V(G) \setminus S$ and $e(S, S^c)$ is the number of edges exiting the set S.*

▶ **Definition 8.** *The q-color ferromagnetic Potts model with parameter $\beta > 0$ is a random assignment of q colors to the vertices of a graph defined by*

$$\mu_{G,\beta}(\sigma) = \frac{e^{-\beta m(G,\sigma)}}{Z_{G,\beta}}$$

where $m(G, \sigma)$ is the number of bichromatic edges of G under the coloring σ and $Z_{G,\beta} = \sum_\sigma e^{-\beta m(G,\sigma)}$ is the Potts model partition function. The parameter β is known as the inverse temperature.

Jenssen, Keevash, and Perkins [16] gave an FPTAS and polynomial-time sampling algorithm for the Potts model on expander graphs, with an algorithm based on the cluster expansion and Barvinok's method of polynomial interpolation. Under essentially the same conditions on the parameters we give a Markov chain based sampling algorithm with near linear running time.

▶ **Theorem 9.** *Suppose $q \geq 2$, $\Delta \geq 3$ are integers and $\alpha > 0$ is a real. Then for $\beta \geq \frac{5+3\log((q-1)\Delta)}{\alpha}$ and any $qe^{-n} \leq \varepsilon < 1$, there is an ε-approximate sampling algorithm for the q-state ferromagnetic Potts model with parameter β on all n-vertex α-expander graphs of maximum degree Δ with running time $O(n \log(n/\varepsilon))$. There is also an ε-approximate counting algorithm with running time $O((n/\varepsilon)^2 \log^2(n/\varepsilon))$ and success probability at least 3/4.*

▶ **Definition 10.** *Let $\alpha \in (0, 1)$. A bipartite graph $G = (V, E)$ with bipartition $V = V^0 \cup V^1$ is a bipartite α-expander if, for $i \in \{0, 1\}$ and all $S \subseteq V^i$ where $|S| \leq |V^i|/2$, we have $N_G(S) \geq (1 + \alpha)|S|$ where $N_G(S)$ denotes the set of vertices that are adjacent to some vertex in S.*

Again we give a fast Markov chain based algorithm for sampling from the hard-core model for essentially the same range of parameters for which an FPTAS is given in [16].

▶ **Theorem 11.** *Suppose $\Delta \geq 3$ is an integer and $\alpha \in (0, 1)$ is a real. Then for any $\lambda \geq (3\Delta)^{6/\alpha}$ and $4e^{-n} \leq \varepsilon < 1$, there is an ε-approximate sampling algorithm for the hard-core model with parameter λ on all n-vertex bipartite α-expander graphs of maximum degree Δ. There is also an ε-approximate counting algorithm for the hard-core model with success probability at least $1 - \varepsilon$. Both algorithms run in time $O((n/\varepsilon)^2 \log^3(n/\varepsilon))$.*

The extra factor of n in the running time of the sampling algorithm for the hard-core model as compared to the Potts model is due to the fact that the hard-core model on a bipartite graph does not in general exhibit exact symmetry between the ground states, and so we must approximate the partition functions of the even and odd dominant independent sets to sample.

We can extend these algorithms to obtain fast sampling algorithms in most situations in which a counting problem can be put in the framework of abstract polymer models. For instance, we can use Theorems 5 and 6 to improve the running times of the algorithms given by [17, 20] for sampling and counting proper q-colorings in Δ-regular bipartite graphs (for large Δ). Section 5 of [17] gives a polymer model for proper q-colorings on Δ-regular bipartite graphs. The polymer model is computationally feasible. They prove in Section 5.1 that it satisfies the Kotecký-Preiss condition – in fact, their proof establishes the polymer sampling condition (3). Thus, we get the following corollary of Theorem 5 and 6.

▶ **Corollary 12.** *There is an absolute constant $C > 0$ so that for all even $q \geq 3$, all $\Delta \geq Cq^2 \log^2 q$ and all $\varepsilon > e^{-n/(8q)}$, there is an ε-approximate sampling algorithm to sample a uniformly random proper q-coloring from a random Δ-regular bipartite graph running in time $O(n \log(n/\varepsilon))$. Furthermore, there is a randomized ε-approximation algorithm for the number of proper q-colorings with running time $O((n/\varepsilon)^2 \log^2(n/\varepsilon))$ and success probability at least $3/4$. For odd q, there are ε-approximate counting and sampling algorithms that both run in time $O((n/\varepsilon)^2 \log^3(n/\varepsilon))$.*

As with independent sets, the extra factor of n in the running time for odd q comes from the fact that the ground states (colorings in which one side of the bipartition is assigned $\lceil q/2 \rceil$ colors and the other side $\lfloor q/2 \rfloor$ colors) are exactly symmetric only if q is even.

Finally, we remark that the approximate counting algorithms for these applications based on truncating the cluster expansion can run faster than $n^{O(\log \Delta)}$ if the parameters (expansion, fugacity, inverse temperature) are high enough (see [17, Theorem 8]), but the sampling algorithms derived from this approach will not match the $\tilde{O}(n)$ or $\tilde{O}(n^2)$ sampling algorithms we obtain here.

1.3 Comparison to spin Glauber dynamics

A very natural idea to sample at low temperatures (large β for the Potts model, large λ for the hard-core model) is to use a single-spin update Markov chain like the Glauber dynamics, but to start in one of the ground states of the model chosen at random. For example, pick one of the q-colors with equal probability then start the Potts model Glauber dynamics in the monochromatic configuration with that color. The intuition is that the Glauber dynamics will mix well within the portion of the state space close to the chosen ground state, and the randomness in the choice of ground state will ensure that an accurate sample from the full measure is obtained. Analyzing this algorithm was suggested in [14] and [16].

While we are not yet able to show that this algorithm succeeds, we make partial progress. We show that Glauber dynamics, restricted to remain in a portion of the state space, mixes rapidly (in polynomial time). It is easiest to state our result for the ferromagnetic Potts model.

For a ground state color $g \in [q]$ and an integer M, let $\Omega_M^g(G)$ be the set of q-colorings of the vertices of G so that every connected component of G colored with the palette of colors $[q] \setminus g$ is of size at most M. The set $\Omega_M^g(G)$ consists of colorings that come from the valid polymer configurations from Example 2 above. In [16] it is shown that for an appropriate choice of M, the set $\{\Omega_M^g(G)\}_{g \in [q]}$ forms an "almost partition" of the set of all colorings, in that the weight of both the overlap of the almost partition and the set of colorings uncovered by the almost partition is at most ε under the conditions of Theorem 9. In particular, an ε-approximate sample from the Potts model restricted to $\Omega_M^g(G)$ for $M = O(\log(n/\varepsilon))$ is enough (by symmetry) to obtain a $(q\varepsilon)$-approximate sample from the Potts distribution $\mu_{G,\beta}$ (cf. Lemma 28 of the full version). Using Markov chain comparison, we show in Section 5.3.1 of the full version that an efficient sampler can be obtained using the usual spin Glauber dynamics restricted to remain in $\Omega_M^g(G)$.

▶ **Theorem 13.** *Under the conditions of Theorem 9, and with $M = O(\log(n/\varepsilon))$, the Glauber dynamics restricted to $\Omega_M^g(G)$ has mixing time $T_{\text{mix}}(\varepsilon)$ polynomial in n and $1/\varepsilon$.*

Theorem 13 shows that, despite the exponentially slow mixing of the Glauber dynamics on the full state space, it can still be used by restricting the state space to obtain a polynomial-time approximate sampling algorithm.

In Section 5 of the full version, we give a result (Theorem 23) which is similar to Theorem 13 but applies much more generally – to polymer models which satisfy the polymer mixing condition and other mild conditions. We also obtain a similar theorem (Theorem 29) specifically for the hard-core model.

We leave for future work two important extensions that would complete the picture: 1) showing that *unrestricted* Glauber dynamics starting from a well chosen configuration works, and 2) reducing the running time to $O(n \log n)$ from the large polynomial that we obtain in the theorem.

2 Polymer models and Markov chains

In the full version, we show that the polymer sampling condition (3) implies the well-known Kotecký–Preiss [18] condition $\sum_{\gamma' \nsim \gamma} e^{|\gamma'|} w_{\gamma'} \leq |\gamma|$. The Kotecký–Preiss condition, in turn, implies the polymer mixing condition (2), which is weaker than the Kotecký–Preiss [18] condition.

We next introduce the polymer Markov chain. For each $v \in V(G)$, let $\mathcal{A}(v) = \{\gamma \in \mathcal{C}(G) : v \in \gamma\}$ denote the collection of all polymers containing v and let $a(v) = \sum_{\gamma \in \mathcal{A}(v)} w_\gamma$. By applying (2) to the smallest γ' containing v we have $a(v) \leq \theta < 1$ for all $v \in V(G)$. Define the probability distribution ν_v on $\mathcal{A}(v) \cup \{\emptyset\}$ by $\nu_v(\gamma) = w_\gamma$ for $\gamma \in \mathcal{A}(v)$ and $\nu_v(\emptyset) = 1 - a(v)$.

The polymer dynamics on Ω are defined by the following transition rule from a configuration Γ_t to a configuration Γ_{t+1}:

Polymer Dynamics

1. Choose $v \in V(G)$ uniformly at random. Let $\gamma_v \in \Gamma_t \cap \mathcal{A}(v)$ if $\Gamma_t \cap \mathcal{A}(v) \neq \emptyset$ and let $\gamma_v = \emptyset$ otherwise. Note that γ_v is well defined since Γ_t can have at most one polymer containing v.
2. Mutually exclusively do the following:
   - With probability $\frac{1}{2}$, let $\Gamma_{t+1} = \Gamma_t \setminus \gamma_v$.
   - With probability $\frac{1}{2}$, sample $\boldsymbol{\gamma}$ from ν_v, set $\Gamma_{t+1} = \Gamma_t \cup \boldsymbol{\gamma}$ if this is in Ω and set $\Gamma_{t+1} = \Gamma_t$ otherwise.

In the full version, we verify that the stationary distribution of the polymer dynamics is μ_G by checking detailed balance. Recall that if $\mathcal{M}$ is an ergodic Markov chain with transition matrix P and stationary distribution ν then the mixing time of $\mathcal{M}$ from a state x is given by

$$T_x(\varepsilon) = \min\{t > 0 \mid \text{ for all } t' \geq t, \|P^{t'}(x, \cdot) - \nu(\cdot)\|_{TV} \leq \varepsilon\},$$

where $\|\nu' - \nu\|_{TV}$ denotes the total variation distance between distributions ν and ν'. The mixing time of $\mathcal{M}$ is given by $T_{\text{mix}}(\varepsilon) = \max_x T_x(\varepsilon)$.

Proof of Theorem 2. We will show that under condition 2 the mixing time of the polymer dynamics is $O(n \log(n/\varepsilon))$ by applying the path coupling technique. We define a metric $D(\cdot, \cdot)$ on Ω by setting $D(\Gamma, \Gamma') = 1$ if $\Gamma' = \Gamma \cup \{\gamma\}$ or $\Gamma = \Gamma' \cup \{\gamma\}$ for a polymer γ and extending this as a shortest path metric; i.e., $D(\Gamma, \Gamma') = |\Gamma \triangle \Gamma'|$ for any $\Gamma, \Gamma' \in \Omega$ where $\triangle$ denotes the symmetric difference of two sets. The diameter W of Ω under $D(\cdot, \cdot)$ is no more than $2n$.

Now suppose we couple two chains X_t and Y_t by attempting the same updates in both chains at each step. Suppose that $X_t = Y_t \cup \{\gamma\}$ for some polymer γ. With probability $\frac{|\gamma|}{n} \cdot \frac{1}{2}$ we pick $v \in \gamma$ and remove γ_v which yields $X_{t+1} = Y_{t+1} = X_t$. On the other hand, we may attempt to add a polymer $\gamma' \nsim \gamma$ so that $Y_t \cup \{\gamma'\} \in \Omega$. That is, $X_{t+1} = X_t = Y_t \cup \{\gamma\}$ and $Y_{t+1} = Y_t \cup \{\gamma'\}$. This occurs with probability $\frac{|\gamma'|}{n} \cdot \frac{1}{2} \cdot w_{\gamma'}$ and in this case $D(X_{t+1}, Y_{t+1}) \le 2$. Putting these together we can bound

$$\mathbb{E}[D(X_{t+1}, Y_{t+1})] \le 1 + \frac{1}{2n}\left[-|\gamma| + \sum_{\gamma' \nsim \gamma} |\gamma'| w_{\gamma'}\right].$$

Using (2) we have $\sum_{\gamma' \nsim \gamma} |\gamma'| w_{\gamma'} \le \theta|\gamma|$, and so $\mathbb{E}[D(X_{t+1}, Y_{t+1})] \le 1 - |\gamma|\frac{1-\theta}{2n} \le 1 - \frac{1-\theta}{2n}$. By the path coupling lemma (see [9, Section 6]), the mixing time is at most $\log(W/\varepsilon)2n/(1-\theta) = O(n\log(n/\varepsilon))$. ◀

To prove Theorem 5 we will show that a single update of the polymer dynamics can be computed in constant expected time.

Assume the polymer sampling condition (3) holds with constant $\tau \ge 5 + 3\log((q-1)\Delta)$. We will use the following algorithm. Let $r = \tau - 2 - \log((q-1)\Delta) \ge 3 + 2\log((q-1)\Delta)$ and let $\mathcal{A}_k(v) = \{\gamma \in \mathcal{A}(v) : |\gamma| \le k\}$.

Single polymer sampler

1. Choose $\mathbf{k}$ according to the following geometric distribution: for k a non-negative integer, $\Pr[\mathbf{k} = k] = (1 - e^{-r})e^{-rk}$. This gives $\Pr[\mathbf{k} \ge k] = e^{-rk}$.
2. Enumerate all polymers in $\mathcal{A}_{\mathbf{k}}(v)$ and compute their weight functions.
3. Mutually exclusively output $\gamma \in \mathcal{A}_{\mathbf{k}}(v)$ with probability $w_\gamma e^{r|\gamma|}$, and with all remaining probability output $\emptyset$. In particular if $\mathbf{k} = 0$, then output $\emptyset$ with probability 1.

We now proceed to prove the following lemma.

▶ **Lemma 16.** *Under the polymer sampling condition* (3) *the output distribution of the* ***single polymer sampler*** *is ν_v and its expected running time is constant.*

Proof. We first show that the probabilities $w_\gamma e^{r|\gamma|}$ sum to less than 1, which shows the last step of the sampling algorithm is well defined. Since $\tau - r = 2 + \log((q-1)\Delta)$,

$$\sum_{\gamma \in A(v)} w_\gamma e^{r|\gamma|} \le \frac{1}{2}\sum_{k \ge 1} (e\Delta)^{k-1}(q-1)^k e^{-\tau k + rk} = \frac{1}{2e\Delta}\sum_{k\ge 1} e^{-k} < 1,$$

where the first inequality uses the fact that, given a degree Δ graph and a vertex v, there are at most $(e\Delta)^{k-1}/2$ connected size-k subgraphs containing v – a fact proved by Borgs, Chayes, Kahn, and Lovász [5, Lemma 2.1].

We next show that the output of the algorithm has distribution ν_v. Given $\gamma \in \mathcal{A}(v)$, to output γ we must choose $\mathbf{k} \ge |\gamma|$. This happens with probability $e^{-r|\gamma|}$ by the distribution of $\mathbf{k}$. Conditioned on choosing such a $\mathbf{k}$, the probability we output γ is $w_\gamma e^{r|\gamma|}$, and multiplying these probabilities together gives w_γ as desired. Since this is true for all $\gamma \in \mathcal{A}(v)$, the output distribution is exactly ν_v.

Finally we analyze the expected running time. To do this, we appeal to Lemma 3.7 of [23] which gives an algorithm with running time $O(k^5(e\Delta)^{2k})$ for listing all connected subgraphs containing a given vertex v of size at most k (given a graph of degree at most Δ). Consequently, conditioned on the event that $\mathbf{k} = k$, the enumeration step of our algorithm takes time $O(k^5(e\Delta)^{2k})$, and the time taken to determine which polymers are allowed and to compute their weights is $O(k^c(q-1)^k(e\Delta)^{k-1}/2)$ for some $c > 0$, since the polymer model is computationally feasible. In expectation therefore, the running time is

$$O\left(1+\sum_{k\geq 1}\Pr[\mathbf{k}=k]\left(k^5(e\Delta)^{2k}+k^c(e(q-1)\Delta)^k\right)\right)$$
$$=O\left(1+\sum_{k\geq 1}e^{-rk}\,k^c(e(q-1)\Delta)^{2k}\right)=O\left(1+\sum_{k\geq 1}k^c\,e^{-(\tau'+1)k}\right)=O(1)\,,$$

where $\tau'=\tau-5-3\log((q-1)\Delta)\geq 0$. ◀

Proof of Theorem 5. By Theorem 2, there is $T_\varepsilon=O(n\log(n/\varepsilon))$ so that if we start with the empty configuration $\Gamma_0=\emptyset$ and run the polymer dynamics, then Γ_{T_ε} has distribution within $\varepsilon/2$ total variation distance of μ_G. By Lemma 16, in expectation the running time will be $O(n\log(n/\varepsilon))$, but we want an upper bound on the worst case running time as well. To do this, we will simply stop the algorithm and output the empty configuration if the total running time exceeds L for some $L=O(n\log(n/\varepsilon))$ with a sufficiently large leading constant. We next show that the probability that the algorithm terminates in L steps is at most $\varepsilon/2$, which therefore yields that the output distribution has total variation distance at most ε from μ_G.

The randomness in the running time comes from the choice of the geometric random variable $\mathbf{k}$ at each step and the time taken to enumerate polymers in $\mathcal{A}_{\mathbf{k}}(v)$. By the choice of r, the random variable that takes the value $k^5(e\Delta)^{2k}+k^c(e(q-1)\Delta)^k$ with probability $(1-e^{-r})e^{-rk}$ has exponential tails, and so a Chernoff bound shows that the probability that the sum of $\Theta(n\log(n/\varepsilon))$ independent copies of such a random variable is at least twice its expectation is bounded by $e^{-\Theta(n\log(n/\varepsilon))}$ which is at most $\varepsilon/2$ (for large enough choice of constants), finishing the proof. ◀

3 Approximate counting algorithm

In this section we show how to use a sampling oracle to approximately compute the partition function of the polymer model. One standard way is by self-reducibility. In [14] an efficient sampling algorithm for polymer models is derived from an efficient approximate counting algorithm by applying self-reducibility on the level of polymers. While we could apply polymer self-reducibility in the other direction to obtain counting algorithms from our sampling algorithm, here we use the simulated annealing method instead (see [2, 15, 25]) to obtain a faster implementation of counting from sampling.

Suppose that $(\mathcal{C}(G),w)$ is a computationally feasible polymer model. Let ρ be a parameter and define a weight function $w_\gamma(\rho)=w_\gamma e^{-\rho|\gamma|}$ for all $\gamma\in\mathcal{C}(G)$. Then for each $\rho\geq 0$ this defines a computationally feasible polymer model $(\mathcal{C}(G),w(\rho))$ on G, where setting $\rho=0$ recovers the original model $(\mathcal{C}(G),w)$. If the original model $(\mathcal{C}(G),w)$ satisfies the polymer sampling condition (3), then so does $(\mathcal{C}(G),w(\rho))$ for every $\rho\geq 0$ as the weight function $w_\gamma(\rho)$ is monotone decreasing in ρ. Given the graph G, we write the partition function of the polymer model $(\mathcal{C}(G),w(\rho))$ as a function of ρ:

$$Z(\rho)=Z(G;\rho)=\sum_{\Gamma\in\Omega}\prod_{\gamma\in\Gamma}w_\gamma(\rho)=\sum_{\Gamma\in\Omega}\prod_{\gamma\in\Gamma}w_\gamma e^{-\rho|\gamma|}.$$

The associated Gibbs distribution is denoted by $\mu_\rho=\mu_{G;\rho}$. Since $\lim_{\rho\to\infty}w_\gamma(\rho)=0$, we have $\lim_{\rho\to\infty}Z(\rho)=1$ (only the empty configuration Γ contributes to this limit), and so we will use simulated annealing to interpolate between $Z(\infty)=1$ and our goal $Z(0)$, assuming access to a sampling oracle for $(\mathcal{C}(G), w(\rho))$ for all $\rho \geq 0$. To apply the simulated annealing method, roughly speaking, we find a sequence of parameters $0 = \rho_0 < \rho_1 < \cdots < \rho_\ell < \infty$ called a *cooling schedule* where $\ell \in \mathbb{N}^+$, and then estimate $Z(0)$ using the telescoping product

$$\frac{1}{Z(0)} = \frac{1}{Z(\rho_0)} = \frac{Z(\rho_1)}{Z(\rho_0)}\frac{Z(\rho_2)}{Z(\rho_1)} \cdots \frac{Z(\rho_\ell)}{Z(\rho_{\ell-1})}\frac{1}{Z(\rho_\ell)}.$$

To estimate each term $Z(\rho_{i+1})/Z(\rho_i)$, we define independent random variables $W_i = \prod_{\gamma\in\Gamma_i} \frac{w_\gamma(\rho_{i+1})}{w_\gamma(\rho_i)}$, where $\Gamma_i \sim \mu_{\rho_i}$. It is straightforward to see that $\mathbb{E}[W_i] = Z(\rho_{i+1})/Z(\rho_i)$ (see Lemma 17 of the full version, where we also require the variance). Using the sampling oracle for μ_{ρ_i}, we can sample W_i for all i, and by taking the product we get an estimate for $1/Z(0)$.

The key ingredient of simulated annealing is finding a good cooling schedule. There are nonadaptive schedules [2] that depend only on n, and adaptive schedules [15, 25] that also depend on the structure of $Z(\cdot)$. Usually the latter leads to faster algorithms than the former. In this paper we use a simple nonadaptive schedule: $\rho_i = i/n$ for $i = 1, \dots, \ell$ where $\ell = O(n\log(n/\varepsilon))$. We show that this cooling schedule already gives us a fast algorithm for the polymer model. The reason behind it is that the weight function $w_\gamma(\rho)$ decays exponentially fast, and so (see Lemma 18 of the full version) the partition function $Z(\rho_\ell)$ is bounded by a constant when $\rho_\ell = O(\log n)$, leading to a short cooling schedule. Our algorithm is as follows.

Polymer approximate counting algorithm

```
1. Let $\rho_i = i/n$ for $i = 0, 1, \dots, \ell$ where $\ell = \lceil n\log(4e(q-1)\Delta n/\varepsilon)\rceil$;
2. For $j = 1, \dots, m$ where $m = \lceil 64\varepsilon^{-2}\rceil$:
   a. For $0 \leq i \leq \ell - 1$:
      (i) Sample $\Gamma_i^{(j)}$ from $\mu_{\rho_i}$;
      (ii) Let $W_i^{(j)} = \prod_{\gamma\in\Gamma_i^{(j)}} e^{-|\gamma|/n}$;
   b. Let $W^{(j)} = \prod_{i=0}^{\ell-1} W_i^{(j)}$;
3. Let $\widehat{W} = \frac{1}{m}\sum_{j=1}^{m} W^{(j)}$ and output $\widehat{Z} = 1/\widehat{W}$.
```

For $0 \leq i \leq \ell - 1$ we define Γ_i to be an independent random sample from μ_{ρ_i} and $W_i = \prod_{\gamma\in\Gamma_i} e^{-|\gamma|/n}$. Finally, we let $W = \prod_{i=0}^{\ell-1} W_i$.

Proof of Theorem 6. In this version, we assume that we have access to an exact sampler $\mathcal{S}_{\text{exact}}$ that samples from μ_ρ for all $\rho \geq 0$ (in the full version we show how to adapt the argument to the situation where we only have an approximate sampler). Using this sampler in the Polymer approximate counting algorithm, we find that, for each j and each i, $\Gamma_i^{(j)}$ is an exact sample from the distribution μ_{ρ_i} and hence $W_i^{(j)}$ is an exact sample of W_i, independently for every j and i. Thus, $W^{(j)}$ is a sample of W independently for every j, and $\widehat{W}$ is the sample mean of $W^{(j)}$'s. We deduce from Lemmas 17 and 18 of the full version that

$$(1+\varepsilon/2)\mathbb{E}[W] \leq \frac{e^{\varepsilon/2}Z(\rho_\ell)}{Z(0)} \leq \frac{e^{\varepsilon}}{Z(0)} \text{ and } (1-\varepsilon/2)\mathbb{E}[W] \geq \frac{e^{-\varepsilon}Z(\rho_\ell)}{Z(0)} \geq \frac{e^{-\varepsilon}}{Z(0)}$$

where we use $1+\varepsilon/2 \leq e^{\varepsilon/2}$ and $e^{-\varepsilon} \leq 1-\varepsilon/2$ for all $0 < \varepsilon < 1$. Then

$$\Pr\left(\frac{e^{-\varepsilon}}{Z(0)} \leq \widehat{W} \leq \frac{e^{\varepsilon}}{Z(0)}\right) \geq \Pr\left(\left|\widehat{W} - \mathbb{E}[W]\right| \leq (\varepsilon/2)\mathbb{E}[W]\right).$$

By Chebyshev's inequality we have

$$\Pr\left(\left|\widehat{W}-\mathbb{E}[W]\right| \geq (\varepsilon/2)\mathbb{E}[W]\right) \leq \frac{4\operatorname{Var}(W)}{\varepsilon^2 m\left(\mathbb{E}[W]\right)^2} \leq \frac{4(e-1)}{\varepsilon^2 m} \leq \frac{1}{8}$$

where the second to last inequality follows from Lemmas 17 and 19 of the full version which enable us to show that

$$\frac{\operatorname{Var}(W)}{(\mathbb{E}[W])^2} = \frac{\mathbb{E}[W^2]}{(\mathbb{E}[W])^2} - 1 = \frac{Z(0)}{Z(\rho_1)}\frac{Z(\rho_{\ell+1})}{Z(\rho_\ell)} - 1 \leq e-1.$$

Thus, we deduce that

$$\Pr\left(e^{-\varepsilon}Z(0) \leq \widehat{Z} \leq e^{\varepsilon}Z(0)\right) = \Pr\left(\frac{e^{-\varepsilon}}{Z(0)} \leq \hat{W} \leq \frac{e^{\varepsilon}}{Z(0)}\right) \geq \frac{7}{8}$$

(so the error probability is at most 1/8). Note that the number of samples that we used is ℓm. Finally, we consider the running time of our algorithm. By Theorem 5, the running time of step 2(a)(i) is $O(n\log(8\ell mn)) = O(n\log(n/\varepsilon))$, and for step 2(a)(ii) the running time is $O(n)$. Thus, the running time of the algorithm is upper bounded by $\ell m \cdot O(n\log(n/\varepsilon)) = O((n/\varepsilon)^2\log^2(n/\varepsilon))$. ◀

4 Applications

In this section, we prove Theorem 9 for the Potts model. The proof of Theorem 11 (for the hard-core model) can be found in Section 4.2 of the full version. Throughout this section, we will work under the assumptions/conditions of Theorem 9. That is, we fix a real number $\alpha > 0$, integers $q \geq 3$ and $\Delta \geq 3$ and a real number $\beta \geq \frac{5+3\log((q-1)\Delta)}{\alpha}$. We let $\mathcal{G}$ be the class of α-expander graphs G with maximum degree at most Δ.

Consider the polymer model defined in Example 2 on an n-vertex graph $G \in \mathcal{G}$ with $M = n/2$ and ground state color $g \in [q]$. We will use $\mathcal{C}^g = \mathcal{C}^g(G)$ to denote the polymers and w^g_γ to denote the weight of a polymer $\gamma \in \mathcal{C}^g$; recall that $w^g_\gamma = e^{-\beta B(\gamma)}$, where $B(\gamma)$ counts the number of external edges of γ plus the number of bichromatic internal edges. Let $Z^g(G)$ be the partition function of the polymer model $(\mathcal{C}^g(G), w^g)$.

▶ **Lemma 20.** *Under the conditions of Theorem 9, the polymer model $(\mathcal{C}^g(\cdot), w^g, \mathcal{G})$ satisfies the polymer sampling condition* (3) *with $\tau = \alpha\beta$.*

Proof. Since every $G \in \mathcal{G}$ is an α-expander, for $\gamma \in \mathcal{C}^g$ we have $B(\gamma) \geq \alpha|\gamma|$ and hence $w^g_\gamma \leq e^{-\tau|\gamma|}$. ◀

▶ **Lemma 21** ([17, Lemma 12])**.** *For any n-vertex α-expander graph G and $\beta \geq 2\log(eq)/\alpha$, $qZ^g(G)$ is an e^{-n}-approximation of the Potts partition function $Z_{G,\beta}$.*

Proof of Theorem 9. Let $\mathcal{G}$ be the class of α-expander graphs of maximum degree at most Δ. Clearly, the polymer models $(\mathcal{C}^g(\cdot), w^g, \mathcal{G})$ are computationally feasible. By Lemma 20, the models also satisfy the polymer sampling condition and therefore Theorems 5 and 6 apply. Consider any n-vertex graph $G \in \mathcal{G}$. Since $\beta \geq \frac{5+3\log((q-1)\Delta)}{\alpha} > \frac{2\log(eq)}{\alpha}$, Lemma 21 applies to G.

For the sampling algorithm, we pick a color $g \in [q]$ uniformly at random and generate an (ε/q)-approximate sample from the Gibbs measure associated to $Z^g(G)$ using the algorithm of Theorem 5, in time $O(n\log(n/\varepsilon))$. By Lemma 21, we conclude that the resulting output is an ε-approximate sample for the Potts model.

For the counting algorithm, we pick an arbitrary $g \in [q]$ and produce using the algorithm of Theorem 6 a number $\hat{Z}$ in time $O((n/\varepsilon)^2 \log^2(n/\varepsilon))$, which is an $\varepsilon/(2q)$-approximation to $Z^g(G)$ with probability $\geq 3/4$. By Lemma 21, we conclude that $q\hat{Z}$ is an ε-approximation for the partition function of the Potts model (with the same probability). ◀

References

1 A. Barvinok. *Combinatorics and Complexity of Partition Functions.* Algorithms and Combinatorics. Springer International Publishing, 2017.

2 I. Bezáková, D. Štefankovič, V. V. Vazirani, and E. Vigoda. Accelerating simulated annealing for the permanent and combinatorial counting problems. *SIAM Journal on Computing*, 37(5):1429–1454, 2008.

3 C. Borgs. Absence of zeros for the chromatic polynomial on bounded degree graphs. *Combinatorics, Probability and Computing*, 15(1-2):63–74, 2006.

4 C. Borgs, J. T. Chayes, A. Frieze, J. H. Kim, P. Tetali, E. Vigoda, and V. H. Vu. Torpid mixing of some Monte Carlo Markov chain algorithms in statistical physics. In *Proceedings of the 40th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 218–229, 1999.

5 C. Borgs, J. T. Chayes, J. Kahn, and L. Lovász. Left and right convergence of graphs with bounded degree. *Random Structures & Algorithms*, 42(1):1–28, 2013.

6 C. Borgs and J. Z. Imbrie. A unified approach to phase diagrams in field theory and statistical mechanics. *Communications in mathematical physics*, 123(2):305–328, 1989.

7 R. L. Dobrushin. Estimates of semi-invariants for the Ising model at low temperatures. *Translations of the American Mathematical Society-Series 2*, 177:59–82, 1996.

8 M. E. Dyer and C. S. Greenhill. On Markov Chains for Independent Sets. *J. Algorithms*, 35(1):17–49, 2000. doi:10.1006/jagm.1999.1071.

9 Martin Dyer and Catherine Greenhill. Random walks on combinatorial objects. *London Mathematical Society Lecture Note Series*, pages 101–136, 1999.

10 C. Efthymiou, T. P. Hayes, D. Štefankovič, E. Vigoda, and Y. Yin. Convergence of MCMC and Loopy BP in the Tree Uniqueness Region for the Hard-Core Model. In *Proceedings of the 57th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 704–713, 2016.

11 R. Fernández, P. A. Ferrari, and N. L. Garcia. Loss network representation of Peierls contours. *Annals of Probability*, 29(2):902–937, 2001.

12 D. Galvin and P. Tetali. Slow mixing of Glauber dynamics for the hard-core model on regular bipartite graphs. *Random Structures & Algorithms*, 28(4):427–443, 2006.

13 C. Gruber and H. Kunz. General properties of polymer systems. *Communications in Mathematical Physics*, 22(2):133–161, 1971.

14 T. Helmuth, W. Perkins, and G. Regts. Algorithmic Pirogov-Sinai theory. *arXiv preprint*, arXiv:1806.11548, 2018. arXiv:1806.11548.

15 M. Huber. Approximation algorithms for the normalizing constant of Gibbs distributions. *The Annals of Applied Probability*, 25(2):974–985, 2015.

16 M. Jenssen, P. Keevash, and W. Perkins. Algorithms for #BIS-hard problems on expander graphs. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2235–2247. SIAM, 2019.

17 M. Jenssen, P. Keevash, and W. Perkins. Algorithms for #BIS-hard problems on expander graphs. *arXiv preprint*, 2019. arXiv:1807.04804v2.

18 R. Kotecký and D. Preiss. Cluster expansion for abstract polymer models. *Communications in Mathematical Physics*, 103(3):491–498, 1986. URL: http://projecteuclid.org/euclid.cmp/1104114796.

19 L. Laanait, A. Messager, S. Miracle-Solé, J. Ruiz, and S. Shlosman. Interfaces in the Potts model I: Pirogov-Sinai theory of the Fortuin-Kasteleyn representation. *Communications in Mathematical Physics*, 140(1):81–91, 1991.

20 C. Liao, J. Lin, P. Lu, and Z. Mao. Counting independent sets and colorings on random regular bipartite graphs. *arXiv preprint*, 2019. `arXiv:1903.07531`.

21 J. Liu and P. Lu. FPTAS for #BIS with degree bounds on one side. In *Proceedings of the 47th Annual ACM on Symposium on Theory of Computing (STOC)*, pages 549–556, 2015.

22 E. Mossel, D. Weitz, and N. Wormald. On the hardness of sampling independent sets beyond the tree threshold. *Probability Theory and Related Fields*, 143(3-4):401–439, 2009.

23 V. Patel and G. Regts. Deterministic polynomial-time approximation algorithms for functions and graph polynomials. *SIAM Journal on Computing*, 46(6):1893–1919, 2017.

24 S. A. Pirogov and Ya. G. Sinai. Phase diagrams of classical lattice systems. *Teoret. Mat. Fiz.*, 25(3):358–369, 1975.

25 D. Štefankovič, S. Vempala, and E. Vigoda. Adaptive simulated annealing: A near-optimal connection between sampling and counting. *Journal of the ACM*, 56(3):18, 2009.

26 D. Weitz. Counting independent sets up to the tree threshold. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing (STOC)*, pages 140–149, 2006.