Perfect Sampling for Hard Spheres from Strong Spatial Mixing

Konrad Anand
Queen Mary, University of London, UK

Andreas Göbel
Hasso Plattner Institute, University of Potsdam, Germany

Marcus Pappik
Hasso Plattner Institute, University of Potsdam, Germany

Will Perkins
School of Computer Science, Georgia Institute of Technology, Atlanta, GA, USA

Abstract

We provide a perfect sampling algorithm for the hard-sphere model on subsets of $\mathbb{R}^d$ with expected running time linear in the volume under the assumption of strong spatial mixing. A large number of perfect and approximate sampling algorithms have been devised to sample from the hard-sphere model, and our perfect sampling algorithm is efficient for a range of parameters for which only efficient approximate samplers were previously known and is faster than these known approximate approaches. Our methods also extend to the more general setting of Gibbs point processes interacting via finite-range, repulsive potentials.

2012 ACM Subject Classification Theory of computation → Randomness, geometry and discrete structures

Keywords and phrases perfect sampling, hard-sphere model, Gibbs point processes

Digital Object Identifier 10.4230/LIPIcs.APPROX/RANDOM.2023.38

Category RANDOM

Related Version *Full Version*: `https://arxiv.org/abs/2305.02450` [3]

Funding *Konrad Anand*: funded by a studentship from Queen Mary, University of London.
Andreas Göbel: funded by the project PAGES (project No. 467516565) of the German Research Foundation (DFG).
Marcus Pappik: funded by the HPI Research School on Data Science and Engineering.
Will Perkins: supported in part by NSF grant DMS-2309958.

Acknowledgements We thank Mark Jerrum for very helpful discussions on this topic.

1 Introduction

Gibbs point processes, or classical gases, are mathematical models of interacting particles. In statistical physics they are used to model gases, fluids, and crystals, while in other fields they are used to model spatial phenomena such as the growth of trees in a forest, the distribution of stars in the universe, or the location of cities on a map (see e.g. [68, 59, 73, 12]).

Perhaps the longest and most intensively studied Gibbs point process is the hard-sphere model: a model of a gas in which the only interaction between particles is a hard-core exclusion in a given radius around each particle. That is, it is a model of a random packing of equal-sized spheres. Despite the simplicity of its definition, the hard-sphere model is expected to exhibit the qualitative behavior of a real gas [2], and in particular exhibits gas, liquid, and solid phases, thus giving evidence for the hypothesis, dating back to at

least Boltzmann, that the macroscopic properties of a gas or fluid are determined by its microscopic interactions. This rich behavior exhibited by the hard-sphere model is very difficult to analyze rigorously, and the most fundamental questions about phase transitions in this model are open mathematical problems [68, 50].

In studying the hard-sphere model (or Gibbs point processes more generally), a fundamental task is to sample from the model. Sampling is used to estimate statistics, observe evidence of phase transitions, and perform statistical tests on data. A wide variety of methods have been proposed to sample from these distributions; for instance, the Markov chain Monte Carlo (MCMC) method was first proposed by Metropolis, Rosenbluth, Rosenbluth, Teller, and Teller [53] to sample from the two-dimensional hard-sphere model. Understanding sampling methods for point processes in theory and in practice is a major area of study [58, 59, 16, 39, 47], and advances in sampling techniques have led to advances in the understanding of the physics of these models [53, 2, 50, 7, 6, 16].

In this paper we will be concerned with provably efficient sampling from the hard-sphere model. Rigorous guarantees for sampling algorithms come in several different varieties. One question is what notion of "efficient" to use; another is what guarantee we insist on for the output. In this paper we will provide an efficient sampling algorithm under the strictest possible terms with respect to both running time and accuracy of the output: a linear-time, *perfect* sampling algorithm.

For simplicity we focus on sampling from the hard-sphere model defined on finite boxes in $\mathbb{R}^d$. For fixed parameter values of the model, the typical number of points appearing in such a region is linear in the volume, and so any sampling algorithm will require at least this much time.

As for guarantees on the output, there are two main types of guarantees. The first type is an *approximate sampler*: the output of such an algorithm must be distributed within ε total variation distance of the desired target distribution. Perhaps the main approach to efficient sampling from distributions normalized by intractable normalizing constants is the MCMC method. In this approach, one devises a Markov chain with the target distribution as the stationary distribution and runs a given number steps of the chain from a chosen starting configuration; if the number of steps is at least the ε-mixing time, then the final state has distribution within ε total variation distance of the target [42, 65, 13]. In general, however, computing or bounding the mixing time can be a very challenging problem.

The second type of guarantee is that of a *perfect sampler* [63]. Such an algorithm has a running time that is random, but the distribution of the output is guaranteed to be *exactly* that of the target distribution. The main advantage of perfect sampling algorithms – and the primary reason they are studied and used in practice – is that one need not prove a theorem or understand the mixing time of a Markov chain to run the algorithm and get an accurate sample; one can simply run the algorithm and know that the output has the correct distribution. The drawback is that the running time may be very large, depending on the specific algorithm and on the parameter regime. Some naive sampling methods such as rejection sampling return perfect samples but are inefficient on large instances (exponential expected running time in the volume). The breakthrough of Propp and Wilson in introducing "coupling from the past" [63, 64] was to devise a procedure for using a Markov chain transition matrix to design perfect sampling algorithms which, under some conditions, could run in time polylogarithmic in the size of a discrete state space (polynomial-time in the size of the graph of a spin system), matching the efficiency of fast mixing Markov chains which only return approximate samples (see also [5, 49] for precedents in perfect sampling). The work of Propp and Wilson led to numerous constructions of perfect sampling algorithms for problems with

both discrete and continuous state spaces including [17, 27, 45, 60, 28, 21, 46, 58, 23]. Notably, many of the first applications of Propp and Wilson's technique were in designing perfect sampling algorithms for Gibbs point processes (though often without rigorous guarantees on the efficiency of the algorithms).

Perfect sampling continues to be a very active area of research today, with a special focus on improving the range of parameters for which perfect sampling algorithms can (provably) run in expected linear or polynomial time [9, 40, 30]

In this paper we design a perfect sampling algorithm for the hard-sphere model (and Gibbs point processes interacting with a finite-range, repulsive pair potential more generally) that is guaranteed to run in linear expected time for activity parameters up to the best known bound for efficient approximate sampling via MCMC.

What is this bound and how do we design the algorithm? One central theme in the analysis of discrete spin systems is the relationship between spatial mixing (correlation decay properties) and temporal mixing (mixing times of Markov chains) [35, 1, 72, 52, 15]. At a high level, these works show that for discrete lattice systems a strong correlation decay property (*strong spatial mixing*) implies a near-optimal convergence rate for local-update Markov chains like the Glauber dynamics. Recently it has been showed that strong spatial mixing in a discrete lattice model also implies the existence of efficient *perfect* sampling algorithms [18, 4]. In parallel, there has been work establishing the connection between strong spatial mixing and optimal temporal mixing for Markov chains in the setting of the hard-sphere model and Gibbs point processes [33, 55, 56]. At a high level, our aim is to combine these threads to show that strong spatial mixing for Gibbs point processes implies the existence of an efficient perfect sampler. One challenge is that the approaches of [18, 4] are inherently discrete in that key steps of the algorithms involve enumerating over all possible configurations in a subregion, something that is not possible in the continuum. To overcome this we make essential use of Bernoulli factories – a method for perfect simulation of a coin flip with a bias $f(p)$ given access to coin flips of bias p. Bernoulli factories have recently been used in perfect sampling algorithms for solutions to constraint satisfaction problems in [31, 32].

1.1 The hard-sphere model, strong spatial mixing, and perfect sampling

The hard-sphere model is defined on a bounded, measurable subset Λ of $\mathbb{R}^d$ with an activity parameter $\lambda \geq 0$ that governs the density of the model and a parameter $r > 0$ that governs the range of interaction (though by re-scaling there is really only one meaningful parameter, and we could take $r = 1$ without loss of generality). In words, the hard-sphere model is the distribution of finite point sets in Λ obtained by taking a Poisson point process of activity λ on Λ and conditioning on the event that all pairs of points are at distance at least r from each other; in other words, on the event that spheres of radius $r/2$ centered at the given points form a sphere packing.

We can equivalently define the model more explicitly, and in doing so, introduce objects and notation we work with throughout the paper. To begin, let $\mathcal{N}_\Lambda$ be the set of all finite point sets in Λ. Each point set $\eta \in \mathcal{N}_\Lambda$ represents a particle configuration in Λ. Write $\mathfrak{R}_\Lambda \subseteq 2^{\mathcal{N}_\Lambda}$ for the σ-field generated by the maps $\{\mathcal{N}_\Lambda \to \mathbb{N}_0, \eta \mapsto |\eta \cap B| \;\big|\; B \subseteq \Lambda \text{ Borel-measurable}\}$. The hard-sphere model (or in fact any Gibbs point process) is a probability measure μ_λ on the space $(\mathcal{N}_\Lambda, \mathfrak{R}_\Lambda)$.

Define for every $x_1, \ldots, x_k \in \mathbb{R}^d$ the indicator that the points are centers of non-overlapping spheres of radius $r/2$; that is,

$$D(x_1,\dots,x_k) = \prod_{\{i,j\}\in\binom{[k]}{2}} \mathbb{1}_{\mathrm{dist}(x_i,x_j)\geq r}\,.$$

Then define the partition function

$$Z_\Lambda(\lambda) = \sum_{k\geq 0}\frac{\lambda^k}{k!}\int_{\Lambda^k} D(x_1,\dots,x_k)\,\mathrm{d}x_1\dots\,\mathrm{d}x_k\,.$$

For an event $A\in\mathfrak{R}_\Lambda$, the hard-sphere model assigns the probability

$$\mu_\lambda(A) = \frac{1}{Z_\Lambda(\lambda)}\sum_{k\geq 0}\frac{\lambda^k}{k!}\int_{\Lambda^k}\mathbb{1}_{\{x_1,\dots,x_k\}\in A}D(x_1,\dots,x_k)\,\mathrm{d}x_1\dots\,\mathrm{d}x_k\,. \tag{1}$$

A very useful generalization of this model is to allow for a non-constant (but measurable) activity function $\boldsymbol{\lambda}:\Lambda\to[0,\infty)$. Here the model is a Poisson process with inhomogenous activity $\boldsymbol{\lambda}$ conditioned on the points forming the centers of a sphere packing; the partition function is now

$$Z_\Lambda(\boldsymbol{\lambda}) = \sum_{k\geq 0}\frac{1}{k!}\int_{\Lambda^k}\prod_{i=1}^{k}\boldsymbol{\lambda}(x_i)D(x_1,\dots,x_k)\,\mathrm{d}x_1\dots\,\mathrm{d}x_k$$

and the measure $\mu_{\boldsymbol{\lambda}}$ is defined analogously to (1). This generalization allows modeling of non-homogenous spaces and generalizes the concept of imposing boundary conditions on the model. To see this, suppose we fix a particle configuration $\eta\in\mathcal{N}_\Lambda$ as boundary conditions. Additional points are forbidden within the balls of radius r around each point $x\in\eta$; we can implement the distribution of additional points by considering the measure $\mu_{\boldsymbol{\lambda}}$ with $\boldsymbol{\lambda}(y)=0$ if $\mathrm{dist}(y,x)<r$ for some $x\in\eta$; and $\boldsymbol{\lambda}(y)=\lambda$ otherwise. We denote the resulting activity function by $\boldsymbol{\lambda}$ by λ_η. Further, we can use this generalization to restrict a point process to only place points in a subregion $\Lambda'\subseteq\Lambda$ by considering the measure $\mu_{\lambda\mathbb{1}_{\Lambda'}}$ with activity function $\lambda\mathbb{1}_{\Lambda'}: x\mapsto\lambda\mathbb{1}_{x\in\Lambda'}$. Of course the generalization to measurable activity functions is much more general than this, and activity functions $\boldsymbol{\lambda}$ need not be realizable by boundary conditions or restriction to a subregion, nor take only two values.

This generalization to activity functions is crucial for defining *strong spatial mixing*, the condition under which we can guarantee the efficiency of our perfect sampling algorithm.

To define the concept of strong spatial mixing we consider projections of the measure $\mu_{\boldsymbol{\lambda}}$ to subregions $\Lambda'\subseteq\Lambda$. We write $\mu_{\boldsymbol{\lambda}}[\Lambda']$ for the probability measure on $(\mathcal{N}_{\Lambda'},\mathfrak{R}_{\Lambda'})$ induced by $\mu_{\boldsymbol{\lambda}}$ (we make this definition formal in Section 3). We can impose two distinct boundary conditions on Λ' by choosing two different activity functions $\boldsymbol{\lambda},\boldsymbol{\lambda}'$. Strong spatial mixing asserts that the distributions $\mu_{\boldsymbol{\lambda}}[\Lambda']$, $\mu_{\boldsymbol{\lambda}'}[\Lambda']$ are close in total variation when $\boldsymbol{\lambda},\boldsymbol{\lambda}'$ differ only on points far from Λ'; i.e., when $\mathrm{dist}(\Lambda',\mathrm{supp}(\boldsymbol{\lambda}-\boldsymbol{\lambda}'))$ is large (as $\mathrm{supp}(\boldsymbol{\lambda}-\boldsymbol{\lambda}')$ is the set of points at which the two activity functions disagree).

Writing $|\Lambda'|$ for the volume of Λ', strong spatial mixing with exponential decay is defined as follows.

▶ **Definition 1.1.** *Given $a,b\in\mathbb{R}_{>0}$, the hard-sphere model on $\mathbb{R}^d$ exhibits (a,b)-**strong spatial mixing** up to $\lambda\in\mathbb{R}_{>0}$ if for all bounded measurable $\Lambda\subset\mathbb{R}^d$ the following holds: For all measurable $\Lambda'\subseteq\Lambda$ and all activity functions $\boldsymbol{\lambda},\boldsymbol{\lambda}'\leq\lambda$ it holds that*

$$d_{TV}(\mu_{\boldsymbol{\lambda}}[\Lambda'],\mu_{\boldsymbol{\lambda}'}[\Lambda']) \leq a|\Lambda'|\mathrm{e}^{-b\cdot\mathrm{dist}\left(\Lambda',\mathrm{supp}\left(\boldsymbol{\lambda}-\boldsymbol{\lambda}'\right)\right)},$$

where $d_{TV}(\cdot,\cdot)$ denotes total variation distance.

This definition of strong spatial mixing comes from [56], which in turn adapted similar notions from discrete spin systems [15, 74]. Strong spatial mixing has proved to be an essential definition in the analysis, both probabilistic and algorithmic, of spin systems on graphs, and many recent works are focused on either proving strong spatial mixing for a particular model, range of parameters, and class of graphs (e.g. [74, 22, 51, 69, 66, 10]) or deriving consequences of strong spatial mixing (e.g. [70, 19, 48, 18, 4]).

Our main result is a linear expected-time perfect sampling algorithm for the hard-sphere model under the assumption of strong spatial mixing.

▶ **Theorem 1.2.** *There is a perfect sampling algorithm for the hard-sphere model on finite boxes $\Lambda \subset \mathbb{R}^d$ with the property that if the hard-sphere model exhibits (a, b)-strong spatial mixing up to λ, then the expected running time of the algorithm at activity λ is $O(|\Lambda|)$, where the implied constant is a function of a, b, and λ.*

In particular, one can run the algorithm for any value of λ (without knowing whether or not strong spatial mixing holds) and the algorithm will terminate in finite time with an output distributed exactly as μ_λ; under the assumption of strong spatial mixing the expected running time is guaranteed to be linear in the volume.

Using bounds from [56] on strong spatial mixing in the hard-sphere model, we obtain the following explicit bounds on the activities for which the algorithm is efficient.

▶ **Corollary 1.3.** *The above perfect sampling algorithm runs in expected time $O(|\Lambda|)$ when $\lambda < \frac{\mathrm{e}}{v_d(r)}$, where $v_d(r)$ is the volume of the ball of radius r in $\mathbb{R}^d$.*

In comparison, near-linear time MCMC-based approximate samplers were given in [56] for the same range of parameters (following results for more restricted ranges in [43, 33]). For perfect sampling from the hard-sphere model, linear expected time algorithms were given in [36, 25] for more restrictive ranges of parameters.

1.2 Gibbs point processes with finite-range repulsive potentials

We now give a closely related result in the more general setting of Gibbs point processes interacting via finite-range, repulsive pair potentials.

Gibbs point processes are defined via a density against an underlying Poisson point process. In general, this density is the exponential of (the negative of) an energy function on point sets that captures the interactions between points. In many of the most studied cases, this energy function takes a special form: it is the sum of potentials over pairs of points in a configuration.

A *pair potential* is a measurable symmetric function $\phi : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R} \cup \{\infty\}$. For a bounded, measurable activity function $\boldsymbol{\lambda}$ on Λ the Gibbs point process with pair potential ϕ on Λ is defined via the partition function

$$Z_\Lambda(\boldsymbol{\lambda}) = \sum_{k \geq 0} \frac{1}{k!} \int_{\Lambda^k} \left(\prod_{i \in [k]} \boldsymbol{\lambda}(x_i) \right) \mathrm{e}^{-H(x_1,\dots,x_k)} \, \mathrm{d}x_1 \dots \, \mathrm{d}x_k$$

where

$$H(x_1, \dots, x_k) = \sum_{\{i,j\} \in \binom{[k]}{2}} \phi(x_i, x_j) \, .$$

Again the corresponding probability measure $\mu_{\boldsymbol{\lambda}}$ is obtained as in (1). A pair potential ϕ is *repulsive* if $\phi(x, y) \geq 0$ for all x, y. It is of *finite-range* if there exists $r \geq 0$ so that $\phi(x, y) = 0$ whenever $\mathrm{dist}(x, y) > r$. As with the hard-sphere model, we can use the

activity function to encode the influence of boundary conditions by defining the activity function $\lambda_\eta : y \mapsto \lambda e^{-\sum_{x\in\eta}\phi(x,y)}$ for any activity $\lambda \in [0,\infty)$ and particle configuration $\eta \in \mathcal{N}_\Lambda$. Moreover, strong spatial mixing for a Gibbs point process is defined exactly as in Definition 1.1.

The hard-sphere model is one example of a model interacting via a finite-range, repulsive pair potential; it is obtained by letting $\phi(x,y)$ take the value $+\infty$ if $\operatorname{dist}(x,y) \le r$ and 0 otherwise. The Strauss process [71, 44] is another such example.

Our next result is a near-linear expected time perfect sampling algorithm for Gibbs point processes interacting via finite-range, repulsive potentials under the assumption of strong spatial mixing.

▶ **Theorem 1.4.** *Suppose ϕ is a finite-range, repulsive potential on $\mathbb{R}^d$ and suppose ϕ exhibits (a,b)-strong spatial mixing up to λ for some constants $a,b>0$. Then there is a perfect sampling algorithm for the Gibbs point process defined by ϕ and activity functions bounded by λ on boxes Λ in $\mathbb{R}^d$ with expected running time $O\Big(|\Lambda|\log^{O(1)}|\Lambda|\Big)$.*

One difference between this algorithm and the hard-sphere algorithm of Theorem 1.2 is that this algorithm needs knowledge of the constants a,b in the assumption of strong spatial mixing, whereas the hard-sphere algorithm does not.

Using the results of [56], we can get explicit bounds for the existence of efficient perfect sampling algorithms in terms of the *temperedness constant* of the potential defined by

$$C_\phi := \sup_{x\in\mathbb{R}^d} \int_{\mathbb{R}^d} |1 - e^{-\phi(x,y)}|\, dy\,. \tag{2}$$

Under the assumption that ϕ is repulsive and of finite range r, we have $0 \le C_\phi \le v_d(r)$.

▶ **Corollary 1.5.** *The above perfect sampling algorithm runs in expected time $O\Big(|\Lambda|\log^{O(1)}|\Lambda|\Big)$ when $\lambda < \frac{e}{C_\phi}$.*

▶ Remark 1.6. In fact, using the results of Michelen and Perkins [54], one can push the bound for strong spatial mixing up to e/Δ_ϕ, where $\Delta_\phi \le C_\phi$ is the *potential-weighted connective constant* defined therein; our perfect sampling algorithm is efficient up to that point.

1.3 Related work and future directions

Related work

In recent years there has been a moderate flurry of activity around proving rigorous results for Gibbs point processes in both the setting of statistical physics and probability theory and in the setting of provably efficient sampling algorithms.

Work on provably efficient approximate sampling methods for the hard-sphere model begins with the seminal paper of Kannan, Mahoney, and Montenegro [43], who used techniques from the analysis of discrete spin systems to prove mixing time bounds for Markov chains for the hard-sphere model. Improvements to the range of parameters for which fast mixing holds came in [29, 33], before Michelen and Perkins proved the bound $e/v_d(r)$ in [56], which we match with a perfect sampling algorithm in Corollary 1.3.

Perfect sampling algorithms for the hard sphere model have been considered in [27, 46, 21, 25, 38]. In terms of rigorous guarantees of efficiency, Huber proved a bound of $2/v_d(r)$ for a near-linear expected time perfect sampler in [36]. The perfect sampling algorithm of Guo and Jerrum in [25] does not match this bound, but the algorithm, based on "partial rejection sampling" [26] is novel and particularly simple. Several of these approaches also apply for finite-range, repulsive potentials or can be extended to that setting (e.g. [57]).

In parallel, there has been much work on proving bounds on the range of activities for which no phase transition can occur in the hard-sphere model; and, in recent years in particular, the techniques used have close connections to algorithms and the study of Markov chains. The classic approach to proving absence of phase transition is by proving convergence of the cluster expansion; the original bound here is $1/(ev_d(r))$ due to Groeneveld [24]. In small dimensions (most significantly in dimension 2) improvements to the radius of convergence can be obtained [20]. On the other hand, this approach is inherently limited by the presence of non-physical singularities on the negative real axis. Alternative approaches avoiding this obstruction include using the equivalence of spatial and temporal mixing [33, 56]; or disagreement percolation [11, 34, 8]. The best current bound for absence of phase transition for the hard-sphere model and for repulsive pair potentials is the bound of e/C_ϕ (and e/Δ_ϕ) obtained by Michelen and Perkins [55, 56, 54]. Theorem 1.4 brings the bound for efficient perfect sampling up to this bound.

On a technical level, the most relevant past work is [18], in which the authors prove that for discrete spin systems, strong spatial mixing and subexponential volume growth of a sequence of graphs imply the existence of an efficient perfect sampling algorithm. We take their approach as a starting point but need new ideas to replace their exhaustive enumeration of configurations.

A key step in our algorithm is the use of a Bernoulli factory to implement a Bayes filter. Bernoulli factories are algorithms by which a Bernoulli random variable with success probability $f(p)$ can be simulated (perfectly) by an algorithm with access to independent Bernoulli p random variables, where the algorithm does not know the value p. Whether a Bernoulli factory exists (and how efficient it can be) depends on the function $f(\cdot)$ and a priori bounds on the possible values p. Bernoulli factories have been studied in [61, 37, 14] and recently used in the design of perfect sampling algorithms for CSP solutions in [31, 32].

Future directions

There are a number of extensions and improvements to these results one could pursue. Perhaps most straightforward would be to relax the notion of strong spatial mixing from exponential decay to decay faster than the volume growth of $\mathbb{R}^d$ and to extend the results to repulsive potentials of unbounded range but finite temperedness constant C_ϕ. Moreover, it would be nice to upgrade the guarantees of the algorithm in Theorem 1.4 to that of Theorem 1.2: that the algorithm does not need prior knowledge of the strong spatial mixing constants a, b to run correctly.

An ambitious and exciting direction would be to remove the assumption of a repulsive potential and find efficient perfect sampling algorithms for the class of *stable* potentials (see e.g. [62, 67, 68] for a definition). A stable potential is repulsive at short ranges but can include a weak attractive part; such potentials include the physically realistic Lenard-Jones potential among others [75]. This would require some very new ideas, as much of the recent probabilistic and algorithmic work on Gibbs point processes (e.g. [55, 56, 8, 54]) has used repulsiveness as an essential ingredient (for one, repulsiveness of the potential implies stochastic domination by the underlying Poisson point process). As a notable exception, a deterministic approximation algorithm for partition functions of finite-range stable potentials based on cluster expansion was recently proposed in [41].

1.4 Outline of the paper

In Section 2, we describe the high-level idea and intuition behind the algorithm. In Section 3 we introduce some notation and present some preliminary results that we will use throughout the paper. In Section 4 we present the algorithm that we will apply to both hard spheres

and more general processes, and we state the main theorems and lemmas that we use for proving Theorem 1.2. The more general setting of bounded-range repulsive potentials (i.e., Theorem 1.4) can be found in the full version of the paper [3]. Intermediate steps and proves are omitted and can be found in the full version as well.

2 Intuitive idea behind the algorithm

Our algorithm is an adaptation to continuum models of the work by Feng, Guo, and Yin [18] on perfect sampling from discrete spin systems. We mimic their setting of a spin system on a graph $G = (V, E)$ by putting a graphical structure on sub-regions of our continuous space.

Let $\Lambda = [0, L)^d \subset \mathbb{R}^d$ be the region considered, $\lambda > 0$ the activity, and let ϕ be a repulsive potential of range $r > 0$. We subdivide Λ into $(\Lambda_{\boldsymbol{v}})_{\boldsymbol{v}\in\mathcal{V}}$, a set of smaller boxes of side length r indexed by vertices of a graph: each box corresponds to a vertex and boxes are connected if they are within r of each other, i.e., particles in the boxes can interact directly through the potential ϕ. We fix the index set for the boxes to be $\mathcal{V} \subset \mathbb{N}_0^d$, where each $\boldsymbol{v} \in \mathcal{V}$ corresponds to the box $\Lambda_{\boldsymbol{v}} = [v_1 r, (v_1 + 1)r) \times \cdots \times [v_d r, (v_d + 1)r)$. We extend this notation to sets of indices $S \subseteq \mathcal{V}$ by setting $\Lambda_S = \bigcup_{\boldsymbol{v}\in S} \Lambda_{\boldsymbol{v}}$. Further, we denote by $\mathbb{B}_k(\boldsymbol{v})$ the set of indices $\boldsymbol{w} \in \mathcal{V}$ with $\|\boldsymbol{v} - \boldsymbol{w}\|_\infty \le k$. To shorten notation, we write $\partial S = \left(\bigcup_{\boldsymbol{v}\in S} \mathbb{B}_1(\boldsymbol{v})\right) \setminus S$ for the outer boundary of a set of boxes indexed by $S \subseteq \mathcal{V}$.

Our algorithm runs iteratively, keeping track of two random variables: a point configuration $X_t \in \mathcal{N}_\Lambda$ with $X_0 = \emptyset$, and a set of "incorrect" boxes $\mathcal{U}_t \subseteq \mathcal{V}$ with $\mathcal{U}_0 = \mathcal{V}$. With each iteration t we maintain the following *invariant*: the partial configuration $X_t \cap (\Lambda_{\mathcal{U}_t})^{\mathrm{c}}$ is distributed according to the projection of μ_λ to $(\Lambda_{\mathcal{U}_t})^{\mathrm{c}}$ under the boundary condition $X_t \cap \Lambda_{\mathcal{U}_t}$. It follows that X_t is distributed according to μ_λ once we reach the state $\mathcal{U}_t = \emptyset$.

We proceed by sketching an iteration of the algorithm. An example for the involved subregions is given in Figure 1. Each iteration runs as follows:

1. We choose $\boldsymbol{u}_t \in \mathcal{U}_t$ uniformly at random and attempt to "repair" it by updating X_t on a neighborhood of boxes $B = \{\boldsymbol{u}_t\} \cup (\mathbb{B}_\ell(\boldsymbol{u}_t) \setminus \mathcal{U}_t)$ for some *update radius* $\ell \in \mathbb{N}$.
2. We sample a Bayes filter F_t (i.e., a Bernoulli random variable) with probability depending on the potential ϕ, the activity λ, and the current point configuration X_t on $\Lambda_{\boldsymbol{u}_t}$ and $\Lambda_{\partial B}$.
3. **a)** If $F_t = 1$, we set $\mathcal{U}_{t+1} = \mathcal{U}_t \setminus \{\boldsymbol{u}_t\}$ and we get X_{t+1} by updating X_t on Λ_B according to a projection of μ_λ conditioned on the boundary configuration $X_t \cap (\Lambda_B)^{\mathrm{c}}$.
   b) If $F_t = 0$, the configuration is unchanged and we add the boundary boxes to our "incorrect" list, i.e., $X_{t+1} = X_t$ and $\mathcal{U}_{t+1} = \mathcal{U}_t \cup \partial B$.

We use the Bayes filter, as in [18], to remove bias from the resulting distribution. To give some intuition for its role, suppose we run a naive version of the algorithm where we always update X_t on Λ_B as in step 3.a) above. Assuming the desired invariant holds after t iterations, this naive algorithm gives a bias to the distribution of X_{t+1} proportional to $\frac{Z_{\Lambda_{B\setminus\{\boldsymbol{u}_t\}}}\left(\lambda_{X_t\cap\Lambda_{\partial B\cup\{\boldsymbol{u}_t\}}}\right)}{Z_{\Lambda_B}\left(\lambda_{X_t\cap\Lambda_{\partial B}}\right)}$. We choose the Bayes filter such that, conditioned on $F_t = 1$, the bias term gets canceled. This suggests the choice

$$\mathbb{P}[F_t = 1 \mid X_t, \mathcal{U}_t, \boldsymbol{u}_t] = C(\mathcal{U}_t, \boldsymbol{u}_t, X_t) \cdot \frac{Z_{\Lambda_B}(\lambda_{X_t\cap\Lambda_{\partial B}})}{Z_{\Lambda_{B\setminus\{\boldsymbol{u}_t\}}}\left(\lambda_{X_t\cap\Lambda_{\partial B\cup\{\boldsymbol{u}_t\}}}\right)}, \tag{3}$$

where the choice $C(\mathcal{U}_t, \boldsymbol{u}_t, X_t)$ serves three main purposes.

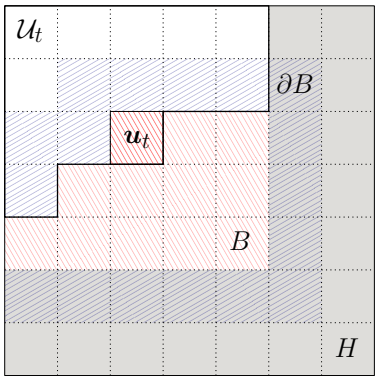

Figure 1 The box-shaped region $\Lambda \subset \mathbb{R}^2$ is divided into boxes of side length r (dotted lines). The boxes $\mathcal{U}_t$ are bordered by bold black lines. For $\boldsymbol{u}_t$ as given and update radius $\ell = 2$, the corresponding set B of boxes to be updated is indicated by the red hatched area (falling left to right). Its boundary boxes ∂B are shown as blue hatched area (rising left to right). The boxes in $H = (\mathcal{U}_t \cup B)^{\mathrm{c}}$ are shown with gray background.

First, it must guarantee that the right-hand side of (3) is a probability. To achieve this we need, for $H = (\mathcal{U}_t \cup B)^{\mathrm{c}}$ and almost all realizations of X_t, $\mathcal{U}_t$ and $\boldsymbol{u}_t$, that

$$C(\mathcal{U}_t, \boldsymbol{u}_t, X_t) \leq \inf_{\xi \in \mathcal{N}_{\Lambda_H}} \frac{Z_{\Lambda_{B \setminus \{\boldsymbol{u}_t\}}}\left(\lambda_{\xi \cup (X_t \cap \Lambda_{\mathcal{U}_t})}\right)}{Z_{\Lambda_B}\left(\lambda_{\xi \cup (X_t \cap \Lambda_{\mathcal{U}_t \setminus \{\boldsymbol{u}_t\}})}\right)}. \tag{4}$$

Second, $C(\mathcal{U}_t, \boldsymbol{u}_t, X_t)$ must introduce no new bias. Carrying out the calculations, it can be seen that this is guaranteed if $C(\mathcal{U}_t, \boldsymbol{u}_t, X_t)$ only depends on $X_t \cap \Lambda_{\mathcal{U}_t}$. Finally, it must ensure that the algorithm terminates almost surely. It suffices to ensure $C(\mathcal{U}_t, \boldsymbol{u}_t, X_t)$ is uniformly bounded away from 0 for almost all realizations of X_t, implying that the same holds for the right-hand side of (3). We refer to a function $C(\cdot)$ satisfying these requirements as a *Bayes filter correction*.

If we use a Bayes filter as given in (3), keeping X_t and $\mathcal{U}_t$ unchanged whenever $F_t = 0$ introduces new bias. To prevent this, we set $\mathcal{U}_{t+1} = \mathcal{U}_t \cup \partial B$ in step 3.b), effectively deleting the part of the configuration that was revealed by the filter. Since the algorithm only terminates once $\mathcal{U}_t = \emptyset$, we further require the Bayes filter correction to ensure that the probability of $F_t = 0$ is small to guarantee efficiency.

Constructing a Bayes filter correction that satisfies the requirements above and allows for efficient sampling of F_t is a non-trivial task. In the next subsections, we present two approaches for this, the first specialized to the hard-sphere model without requirements, and the second one for more general potentials with strong spatial mixing of the point process. Crucially, assuming strong spatial mixing, both constructions allow us to control the success probability of the Bayes filter via the update radius ℓ in the construction of the updated set of boxes B (see step 1).

2.1 Bayes filter for the hard-sphere model

To construct a Bayes filter for the hard-sphere model, we efficiently approximate the right-hand side of (4). To approximate the infimum over the uncountable set of configurations $\xi \in \mathcal{N}_{\Lambda_H}$ we take the minimum over a finite, but sufficiently rich set of configurations, balancing the quality of approximation with the computation required. In fact the number

of configurations needed will depend only on the volume of $\Lambda_{B\cup\partial B}$. We approximate the fraction of partition functions in (4) with running time only depending on the volume of $\Lambda_{B\cup\partial B}$. As a result, we efficiently compute a Bayes filter correction $C_\varepsilon(\cdot)$, with the parameter $\varepsilon > 0$ controlling how much $C_\varepsilon(\mathcal{U}_t, \boldsymbol{u}_t, X_t)$ deviates from the right-hand side of (4).

While our construction of $C_\varepsilon(\cdot)$ guarantees correctness of the sampling algorithm for any $\varepsilon > 0$, proving efficiency requires more. With strong spatial mixing, we choose ε so that the probability that $F_t = 0$ is uniformly bounded above, ensuring $\mathrm{O}(|\Lambda|)$ iterations in expectation.

It remains to argue that we can efficiently sample F_t, using the Bayes filter correction $C_\varepsilon(\cdot)$. Explicitly computing the success probability of F_t as in (3) would require computing the fraction of partition functions on the right-hand side exactly, while approximating these partition functions would require that the approximation error only depends on $X_t \cap \Lambda_{\mathcal{U}_t}$, to avoid new bias.

It is unclear how to implement these approaches, so instead we use Bernoulli factories to sample F_t without knowing the success probability. To do so, we observe that the fraction of partition functions can be written as a ratio of probabilities for drawing the empty set from a conditional hard-sphere model on Λ_B and $\Lambda_{B\setminus\{\boldsymbol{u}_t\}}$. Since both regions have constant volume, rejection sampling obtains Bernoulli random variables with these success probabilities in constant time. Hence, we obtain a Bernoulli factory for F_t with constant expected running time. Wald's identity yields a total expected running time $\mathrm{O}(|\Lambda|)$ for the algorithm.

2.2 Bayes filter for general potentials

We now consider the case of general bounded-range, repulsive potentials. Unlike the hard sphere model, it is not clear here how to approximate the infimum in (4) from a finite set of boundary configurations. However, given constants $a, b > 0$ such that ϕ satisfies (a, b)-strong spatial mixing, we can explicitly compute a function $\delta(a, b)$ so that

$$C_{a,b}(\mathcal{U}_t, \boldsymbol{u}_t, X_t) = \delta(a,b) \cdot \frac{Z_{\Lambda_{B\setminus\{\boldsymbol{u}_t\}}}\left(\lambda_{X_t\cap\Lambda_{\mathcal{U}_t}}\right)}{Z_{\Lambda_B}\left(\lambda_{X_t\cap\Lambda_{\mathcal{U}_t\setminus\{\boldsymbol{u}_t\}}}\right)}$$

is a Bayes filter correction. With strong spatial mixing, we use $C_{a,b}(\cdot)$ to construct a Bayes filter such that probability that $F_t = 0$ is bounded above, again implying a bound of $\mathrm{O}(|\Lambda|)$ on the expected number of iterations of the algorithm.

Note that in this setting, we require spatial mixing for both correctness and efficiency, while for the hard-sphere model we only need it for efficiency. Another crucial difference is that, while we can explicitly compute $\delta(a, b)$, the same does not hold for $C_{a,b}(\cdot)$ due to the fraction of partition functions involved. Again we circumvent this by rewriting the success probability of the Bayes filter in a suitable way and applying a Bernoulli factory for sampling F_t. Finally, we point out that we do not obtain a constant bound for the expected running time of each iteration, but instead the bound depends on the number of points in $X_t \cap \Lambda_{\partial B}$. Possible dependencies between the configuration X_t and the number of iterations prevent us from bounding the total expected running time using Wald's identity. Instead, we provide tail bounds on the number of iterations and the running time of each iteration, allowing us to derive an expected total running time that is linear in the volume of Λ up to polylogarithmic factors.

3 Preliminaries

Throughout the paper, we write $\mathbb{N}$ for the set of strictly positive integers, and we write $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$. For any $k \in \mathbb{N}$, we denote by $[k]$ the set $[1, k] \cap \mathbb{N}$.

For a bounded measurable region $\Lambda \subset \mathbb{R}^d$ and any finite point configuration $\eta \in \mathcal{N}_\Lambda$, we write $|\eta|$ for the number of points in η. Note that this notation is the same that as the one we use for the volume of a region. The particular meaning will be clear from the context. Moreover, for $k \in \mathbb{N}$, we write $\binom{\eta}{k}$ for the set $\{\eta' \subseteq \eta \mid |\eta'| = k\}$.

3.1 Gibbs point processes

We introduce some additional notation for Gibbs point processes, used in the rest of the paper. Firstly, when dealing with a tuple $(x_1, \dots, x_k) \in (\mathbb{R}^d)^k$ we frequently denote it by the corresponding bold letter $\boldsymbol{x}$. Based on this, we write $\mathrm{d}\boldsymbol{x}$ for $\mathrm{d}x_1 \dots \mathrm{d}x_k$ and $H(\boldsymbol{x})$ for $H(x_1, \dots, x_k)$. Moreover, for any $k \in \mathbb{N}_0$ and $\boldsymbol{x} = (x_1, \dots, x_k) \in (\mathbb{R}^d)^k$ we write $\eta_{\boldsymbol{x}}$ for the set $\{x_1, \dots, x_k\}$, where the case $k = 0$ results in $\eta_{\boldsymbol{x}} = \emptyset$. Finally, for $\boldsymbol{x} \in \Lambda^k$ we write $\boldsymbol{\lambda}^{\boldsymbol{x}}$ for $\prod_{i\in[k]} \boldsymbol{\lambda}(x_i)$. This simplifies the definition of $\mu_{\boldsymbol{\lambda}}$ given in the introduction to

$$\mu_{\boldsymbol{\lambda}}(A) = \frac{1}{Z_\Lambda(\boldsymbol{\lambda})} \sum_{k\geq 0} \frac{1}{k!} \int_{\Lambda^k} \mathbb{1}_{\eta_{\boldsymbol{x}} \in A} \boldsymbol{\lambda}^{\boldsymbol{x}} \mathrm{e}^{-H(\boldsymbol{x})} \, \mathrm{d}\boldsymbol{x}.$$

Next, we formalize two different notions of restricting a Gibbs point process on Λ to a subregion $\Lambda' \subseteq \Lambda$ that are relevant throughout the paper.

The first is based on restricting the support of $\boldsymbol{\lambda}$ by defining a new activity function $\boldsymbol{\lambda}\mathbb{1}_{\Lambda'} : y \mapsto \boldsymbol{\lambda}(y) \cdot \mathbb{1}_{y\in\Lambda'}$ (for constant activity λ, we write $\lambda\mathbb{1}_{\Lambda'} : y \mapsto \lambda\mathbb{1}_{y\in\Lambda'}$). The resulting Gibbs point process is a probability measure on $(\mathcal{N}_\Lambda, \mathfrak{R}_\Lambda)$ with

$$\begin{aligned}\mu_{\boldsymbol{\lambda}\mathbb{1}_{\Lambda'}}(A) &= \frac{1}{Z_\Lambda(\boldsymbol{\lambda}\mathbb{1}_{\Lambda'})} \sum_{k\geq 0} \frac{1}{k!} \int_{\Lambda^k} \mathbb{1}_{\eta_{\boldsymbol{x}} \in A} (\boldsymbol{\lambda}\mathbb{1}_{\Lambda'})^{\boldsymbol{x}} \mathrm{e}^{-H(\boldsymbol{x})} \, \mathrm{d}\boldsymbol{x} \\ &= \frac{1}{Z_{\Lambda'}(\boldsymbol{\lambda})} \sum_{k\geq 0} \frac{1}{k!} \int_{\Lambda'^k} \mathbb{1}_{\eta_{\boldsymbol{x}} \in A} \boldsymbol{\lambda}^{\boldsymbol{x}} \mathrm{e}^{-H(\boldsymbol{x})} \, \mathrm{d}\boldsymbol{x}\end{aligned}$$

for all $A \in \mathfrak{R}_\Lambda$. In particular, for $A = \{\eta \in \mathcal{N}_\Lambda \mid \eta \cap (\Lambda')^{\mathrm{c}} > 0\}$, it holds that $\mu_{\boldsymbol{\lambda}\mathbb{1}_{\Lambda'}}(A) = 0$.

The second way of restricting a Gibbs point process $\mu_{\boldsymbol{\lambda}}$ is by projecting it to a measurable subregion $\Lambda' \subseteq \Lambda$. To this end, we write $\mu_{\boldsymbol{\lambda}}[\Lambda']$ for the image measure of $\mu_{\boldsymbol{\lambda}}$ under the map $\mathcal{N}_\Lambda \to \mathcal{N}_{\Lambda'}$, $\eta \mapsto \eta \cap \Lambda'$. By construction, $\mu_{\boldsymbol{\lambda}}[\Lambda']$ is a probability distribution on $(\mathcal{N}_{\Lambda'}, \mathfrak{R}_{\Lambda'})$ that, for every $A \in \mathfrak{R}_{\Lambda'}$, assigns a probability

$$\mu_{\boldsymbol{\lambda}}[\Lambda'](A) = \frac{1}{Z_\Lambda(\boldsymbol{\lambda})} \sum_{k\geq 0} \frac{1}{k!} \int_{\Lambda^k} \mathbb{1}_{\eta_{\boldsymbol{x}} \cap \Lambda' \in A} \boldsymbol{\lambda}^{\boldsymbol{x}} \mathrm{e}^{-H(\boldsymbol{x})} \, \mathrm{d}\boldsymbol{x}.$$

As discussed in Section 1, we frequently modify the activity function to encode the effect of fixing a certain point set (boundary condition). More precisely, for a fixed potential ϕ, an activity function $\boldsymbol{\lambda}$ and a point set $\eta \in \mathcal{N}_\Lambda$ we write $\boldsymbol{\lambda}_\eta$ for the function $y \mapsto \boldsymbol{\lambda}(y)\mathrm{e}^{-\sum_{x\in\eta}\phi(x,y)}$. Similarly, for $k \in \mathbb{N}$ and $\boldsymbol{x} \in \Lambda^k$ we write $\boldsymbol{\lambda}_{\boldsymbol{x}}$ for the activity function $y \mapsto \boldsymbol{\lambda}(y)\mathrm{e}^{-\sum_{i\in[k]}\phi(x_i,y)}$. We extend this notation to constant activity $\lambda \in \mathbb{R}_{\geq 0}$, writing $\lambda_\eta : y \mapsto \lambda\mathrm{e}^{-\sum_{x\in\eta}\phi(x,y)}$ and $\lambda_{\boldsymbol{x}} : y \mapsto \lambda\mathrm{e}^{-\sum_{i\in[k]}\phi(x_i,y)}$. Using this notation, a useful alternative definition of $\mu_{\boldsymbol{\lambda}}[\Lambda']$ is given by

$$\begin{aligned}\mu_{\boldsymbol{\lambda}}[\Lambda'](A) &= \frac{1}{Z_\Lambda(\boldsymbol{\lambda})} \sum_{k\geq 0} \frac{1}{k!} \int_{\Lambda'^k} \mathbb{1}_{\eta_{\boldsymbol{x}} \in A} \boldsymbol{\lambda}^{\boldsymbol{x}} \mathrm{e}^{-H(\boldsymbol{x})} Z_\Lambda\big(\boldsymbol{\lambda}_{\boldsymbol{x}}\mathbb{1}_{(\Lambda')^{\mathrm{c}}}\big) \, \mathrm{d}\boldsymbol{x} \\ &= \frac{1}{Z_\Lambda(\boldsymbol{\lambda})} \sum_{k\geq 0} \frac{1}{k!} \int_{\Lambda'^k} \mathbb{1}_{\eta_{\boldsymbol{x}} \in A} \boldsymbol{\lambda}^{\boldsymbol{x}} \mathrm{e}^{-H(\boldsymbol{x})} Z_{(\Lambda')^{\mathrm{c}}}(\boldsymbol{\lambda}_{\boldsymbol{x}}) \, \mathrm{d}\boldsymbol{x}\end{aligned}$$

for $A \in \mathfrak{R}_{\Lambda'}$. In particular, note that

$$\mu_{\boldsymbol{\lambda}\mathbb{1}_{\Lambda'}}[\Lambda'](A) = \frac{1}{Z_{\Lambda'}(\boldsymbol{\lambda})} \sum_{k\geq 0} \frac{1}{k!} \int_{\Lambda'^k} \mathbb{1}_{\eta_{\boldsymbol{x}} \in A} \boldsymbol{\lambda}^{\boldsymbol{x}} \mathrm{e}^{-H(\boldsymbol{x})} \, \mathrm{d}\boldsymbol{x}.$$

While $\mu_{\boldsymbol{\lambda}\mathbb{1}_{\Lambda'}}[\Lambda']$ and $\mu_{\boldsymbol{\lambda}\mathbb{1}_{\Lambda'}}$ seem similar, the former is a distribution on $(\mathcal{N}_{\Lambda'}, \mathfrak{R}_{\Lambda'})$ whereas the latter is defined on $(\mathcal{N}_{\Lambda}, \mathfrak{R}_{\Lambda})$.

3.2 Bernoulli factories

In designing our sampling algorithm, it will be useful to consider the following Bernoulli factory problem. We are given access to a sampler for $\mathrm{Ber}(p)$ and for $\mathrm{Ber}(q)$, that is samplers of Bernoulli random variables with parameters p and q respectively, where we further assume $p < q$. We want to sample a random variable $Z \sim \mathrm{Ber}\left(\frac{p}{q}\right)$.

Most work on Bernoulli factories studies their running time in terms of the number of coin flips required. In our setting, the time needed to generate each of these coin flips is random variable. Fortunately, suitable independence assumptions hold in our setting allowing us to prove the following lemma.

▶ **Lemma 3.1.** *Fix some $p, q \in [0,1]$ such that $q - p \geq \epsilon$ for some $\epsilon > 0$. Further assume that we have oracle access to a sampler from $\mathrm{Ber}(p)$ and $\mathrm{Ber}(q)$ in the following sense:*

1. *every sample from $\mathrm{Ber}(p)$ (resp. $\mathrm{Ber}(q)$) is independent from all previous samples;*
2. *the expected running time for obtaining a sample from $\mathrm{Ber}(p)$ (resp. $\mathrm{Ber}(q)$), conditioned on previously obtained samples, is uniformly bounded by some $t \in \mathbb{R}_{\geq 0}$.*

Then we can sample from $\mathrm{Ber}\left(\frac{p}{q}\right)$ in $\mathrm{O}\left(t\epsilon^{-2}\right)$ expected time.

4 The algorithm

Let $\Lambda = [0, L)^d$ and consider a Gibbs point processes on Λ with uniform activity $\boldsymbol{\lambda}(x) \equiv \lambda$ for some $\lambda \in \mathbb{R}_{>0}$ and repulsive potential ϕ with finite range $r \in \mathbb{R}_{>0}$. Throughout the analysis of our algorithm, it will be useful to focus on configurations $\eta \in \mathcal{N}_\Lambda$ such that $\phi(x,y) < \infty$ for all $\{x, y\} \in \binom{\eta}{2}$, in which case we call η a *feasible configuration*.

Before stating our algorithm, we first formalize how we divide Λ into smaller boxes, following the description given in Section 2. For a r and L as above, let $N = \lceil L/r \rceil$. We set $\mathcal{V} = \{0, \ldots, N-1\}^d$ to be the set of box indices and associate each box index $\boldsymbol{v} = (v_1, \ldots, v_d) \in \mathcal{V}$ with the region $\Lambda_{\boldsymbol{v}} = \big([v_1 r, (v_1+1)r) \times \cdots \times [v_d r, (v_d+1)r)\big) \cap \Lambda$. As in Section 2, we extend this notation to sets of box indices $S \subseteq \mathcal{V}$ by setting $\Lambda_S = \bigcup_{\boldsymbol{v} \in S} \Lambda_{\boldsymbol{v}}$. Further, recall that, for $\boldsymbol{v} \in \mathcal{V}$, we write $\mathbb{B}_k(\boldsymbol{v})$ for the set of boxes $\boldsymbol{w} \in \mathcal{V}$ with $\|\boldsymbol{v} - \boldsymbol{w}\|_\infty < k$. As mentioned earlier, our algorithm tries to update in each step the point configuration on a subset of boxes $B \subseteq \mathcal{V}$. To this end, for $S \subseteq \mathcal{V}$, $\boldsymbol{v} \in S, r \in \mathbb{R}_{>0}$ and $\ell \in \mathbb{N}$, we define

$$B(S, \boldsymbol{v}, \ell) \coloneqq \{\boldsymbol{v}\} \cup (\mathbb{B}_{\boldsymbol{v}}(\ell) \setminus S).$$

We refer to the parameter ℓ as the *update radius*. Finally, recall that we write $\partial S = (\bigcup_{\boldsymbol{v} \in S} \mathbb{B}_1(\boldsymbol{v})) \setminus S$ for the outer boundary of $S \subseteq \mathcal{V}$.

Whether the algorithm updates the point configuration in iteration t depends on the outcome of a Bernoulli random variable F_t, called the *Bayes filter*. We introduce the following definition.

▶ **Definition 4.1.** *Fix a repulsive potential ϕ of range $r \in \mathbb{R}_{>0}$, an activity $\lambda \in \mathbb{R}_{>0}$ and some $\ell \in \mathbb{N}$. We call a function $C : 2^{\mathcal{V}} \times \mathcal{V} \times \mathcal{N}_\Lambda \to [0,1]$ a* Bayes filter correction *if, for all non-empty $S \subseteq \mathcal{V}$ and $\boldsymbol{v} \in S$, it holds that*

1. *$C(S, \boldsymbol{v}, \cdot)$ is $\mathfrak{R}_{\Lambda_S}$-measurable (in particular $C(S, \boldsymbol{v}, \eta) = C(S, \boldsymbol{v}, \eta \cap \Lambda_S)$ for all $\eta \in \mathcal{N}_\Lambda$),*
2. *there is some $\varepsilon > 0$ such that for $B = B(S, \boldsymbol{v}, \ell)$, $H = (S \cup B)^{\mathrm{c}}$ and all feasible $\eta \in \mathcal{N}_\Lambda$ it holds that*

$$\varepsilon \leq C(S, \boldsymbol{v}, \eta) \leq \inf_{\substack{\xi \in \mathcal{N}_{\Lambda_H} \\ \xi \cup (\eta \cap \Lambda_S) \text{ is feasible}}} \left\{ \frac{Z_{\Lambda_{B \setminus \{\boldsymbol{v}\}}}\left(\lambda_{\xi \cup (\eta \cap \Lambda_S)}\right)}{Z_{\Lambda_B}\left(\lambda_{\xi \cup (\eta \cap \Lambda_{S \setminus \{\boldsymbol{v}\}})}\right)} \right\}.$$

Our perfect sampling procedure is stated in Algorithm 1.

Algorithm 1 Perfect sampling algorithm for repulsive Gibbs point processes.

Data: region $\Lambda = [0, L)^d$, repulsive potential ϕ of range at most $r \in \mathbb{R}_{>0}$, activity $\lambda \in \mathbb{R}_{>0}$, update radius $\ell \in \mathbb{N}$

1 set $t = 0$, $\mathcal{U}_t = \mathcal{V}$, $X_t = \emptyset$
2 **while** $\mathcal{U}_t \neq \emptyset$ **do**
3 draw $\boldsymbol{u}_t \in \mathcal{U}_t$ uniformly at random
4 set $B = B(\mathcal{U}_t, \boldsymbol{u}_t, \ell)$
5 draw F_t from $\mathrm{Ber}\left(C(\mathcal{U}_t, \boldsymbol{u}_t, X_t) \cdot \frac{Z_{\Lambda_B}\left(\lambda_{X_t \cap \Lambda_{\partial B}}\right)}{Z_{\Lambda_{B \setminus \{\boldsymbol{u}_t\}}}\left(\lambda_{X_t \cap \Lambda_{\partial B \cup \{\boldsymbol{u}_t\}}}\right)} \right)$ where C is a Bayes filter correction as in Definition 4.1
6 **if** $F_t = 1$ **then**
7 draw Y from $\mu_{\lambda_{X_t \cap (\Lambda_B)^{\mathrm{c}}} \mathbb{1}_{\Lambda_B}}[\Lambda_B]$
8 set $X_{t+1} = (X_t \setminus \Lambda_B) \cup Y$
9 set $\mathcal{U}_{t+1} = \mathcal{U}_t \setminus \{\boldsymbol{u}_t\}$
10 **else**
11 set $\mathcal{U}_{t+1} = \mathcal{U}_t \cup \partial B$
12 increase t by 1
13 **return** X_t

Before we get to the question of how to sample an appropriate Bayes filter in step 5, the following statement ensures that the algorithm produces the correct output distribution.

▶ **Theorem 4.2.** *Let $T = \inf_{t \in \mathbb{N}} \{\mathcal{U}_t = \emptyset\}$. Then T is almost surely finite and for all $t \in \mathbb{N}_0$ with $\mathbb{P}[t \geq T] > 0$ and all $A \in \mathfrak{R}_\Lambda$, it holds that $\mathbb{P}[X_t \in A \mid t \geq T] = \mu_\lambda(A)$.*

We proceed to exemplify how we use Bernoulli factories to sample the Bayes filter. For brevity, we focus on the hard-sphere model here. The more general case of bounded-range repulsive potential can be found in the full version of the paper [3].

Bayes filter for the hard-sphere model

Recall that for the hard-sphere model, we have $\phi(x, y) = \infty$ if $\mathrm{dist}(x, y) < r$ and 0 otherwise. Given a non-empty set of boxes $S \subseteq \mathcal{V}$, $\boldsymbol{v} \in S$ and a feasible configuration $\eta \in \mathcal{N}_\Lambda$, we want to construct a Bayes filter correction $C(S, \boldsymbol{v}, \eta)$ that allows us to efficiently sample the filter.

To this end, set $B = B(S, \boldsymbol{v}, \ell)$ and $H = (S \cup B)^{\mathrm{c}}$. Our construction makes use of two ingredients. Firstly, we argue that, instead of minimization over the uncountable set of boundary conditions $\mathcal{N}_{\Lambda_H}$, it suffices to minimize over subsets of the finite set $(\delta_1\mathbb{Z})^d \cap \Lambda_{H\cap\partial B}$ for a sufficiently small $\delta_1 > 0$. Secondly, choosing a sufficiently small $\delta_2 > 0$, we show that we can approximate the involved partition functions using the function

$$\hat{Z}(S, \eta, \delta_2) = \sum_{\gamma \subseteq (\delta_2\mathbb{Z})^d \cap \Lambda_S} \lambda^{|\gamma|} \delta_2^{|\gamma|} \cdot D(\gamma) \cdot D(\gamma \mid \eta \cap \Lambda_{\partial S}), \tag{5}$$

where $D(\gamma) = \prod_{\{x,y\} \in \binom{\gamma}{2}} \mathbb{1}_{\mathrm{dist}(x,y) \geq r}$ and $D(\gamma \mid \eta) = \prod_{x\in\gamma} \prod_{y\in\eta} \mathbb{1}_{\mathrm{dist}(x,y)\geq r}$.

The following lemma then gives a way to construct a Bayes filter correction for the hard-sphere model.

▶ **Lemma 4.3.** *For non-empty $S \subseteq \mathcal{V}$, $\boldsymbol{v} \in S$, feasible $\eta \in \mathcal{N}_\Lambda$ and $\varepsilon, \delta_1, \delta_2 > 0$ define*

$$C_\varepsilon(S, \boldsymbol{v}, \eta) \coloneqq \mathrm{e}^{-\varepsilon} \cdot \min_{\gamma \subseteq (\delta_1\mathbb{Z})^d \cap \Lambda_{H\cap\partial B}} \frac{\hat{Z}(B \setminus \{\boldsymbol{v}\}, \gamma \cup (\eta \cap \Lambda_S), \delta_2)}{\hat{Z}\big(B, \gamma \cup (\eta \cap \Lambda_{S\setminus\boldsymbol{v}}), \delta_2\big)},$$

where $B = B(S, \boldsymbol{v}, \ell)$ and $H = (S \cup B)^{\mathrm{c}}$. For δ_1, δ_2 sufficiently small, depending only on d, r, ℓ and ε, it holds that $C_\varepsilon(S, \boldsymbol{v}, \eta)$ is a Bayes filter correction.

In fact, we will not use C_ε directly for our Bayes filter, but a slightly scaled version $0 < \mathrm{e}^{-\varepsilon} C_\varepsilon$, which is again a Bayes filter correction. The additional slack allows us to efficiently sample the Bayes filter by using a Bernoulli factory, as we argue in the next lemma.

▶ **Lemma 4.4.** *Let $S \subseteq \mathcal{V}$ be non-empty, $\boldsymbol{v} \in S$ and $\eta \in \mathcal{N}_\Lambda$ be feasible, and set $B = B(S, \boldsymbol{v}, \ell)$. For all $\varepsilon > 0$ we can sample a Bernoulli random variable with success probability*

$$\mathrm{e}^{-\varepsilon} C_\varepsilon(S, \boldsymbol{v}, \eta) \cdot \frac{Z_{\Lambda_B}(\lambda_{\eta\cap\Lambda_{\partial B}})}{Z_{\Lambda_{B\setminus\{\boldsymbol{v}\}}}\big(\lambda_{\eta\cap\Lambda_{\partial B\cup\{\boldsymbol{v}\}}}\big)}$$

with expected running time only depending on ε, ℓ, r, λ and d.

The core idea of the above lemma to express $\frac{Z_{\Lambda_B}(\lambda_{\eta\cap\Lambda_{\partial B}})}{Z_{\Lambda_{B\setminus\{\boldsymbol{v}\}}}\big(\lambda_{\eta\cap\Lambda_{\partial B\cup\{\boldsymbol{v}\}}}\big)}$ as a fraction of probabilities. Together with the fact that $\mathrm{e}^{-\varepsilon} C_\varepsilon(S, \boldsymbol{v}, \eta) < 1$, this allows us to write the success probability of the Bayes filter as a fraction of probabilities $\frac{p}{q}$. Arguing that $p < q$, and that we can sample $\mathrm{Ber}(p)$ and $\mathrm{Ber}(q)$ efficiently allows us to apply Lemma 3.1 to prove Lemma 4.4.

While the above suffices to perform each iteration of Algorithm 1 efficiently, we still need to bound the number of iterations. For this, we derive a lower bound on the success probability of the Bayes filter with correction $\mathrm{e}^{-\varepsilon} C_\varepsilon(\cdot)$ for a particular choice of ε, using the assumption of strong spatial mixing.

▶ **Lemma 4.5.** *Consider a hard-sphere model that exhibits (a, b)-strong spatial mixing up to λ. Then there are constants a', b', only depending on a, b, r, λ and d, such that for all non-empty $S \subseteq \mathcal{V}$, $\boldsymbol{v} \in S$ and feasible $\eta \in \mathcal{N}_\Lambda$ it holds that*

$$\mathrm{e}^{-\mathrm{e}^{-\ell}} C_{\mathrm{e}^{-\ell}}(S, \boldsymbol{v}, \eta) \cdot \frac{Z_{\Lambda_B}(\lambda_{\eta\cap\Lambda_{\partial B}})}{Z_{\Lambda_{B\setminus\{\boldsymbol{v}\}}}\big(\lambda_{\eta\cap\Lambda_{\partial B\cup\{\boldsymbol{v}\}}}\big)} \geq 1 - a'\mathrm{e}^{-b'\ell}.$$

Lemma 4.5 allows us to control the success probability of the Bayes filter in terms of ℓ. Combining the results above gives the following theorem.

▶ **Theorem 4.6.** *Consider Algorithm 1 on a hard-sphere model with* $C(\cdot) = \mathrm{e}^{-\mathrm{e}^{-\ell}} C_{\mathrm{e}^{-\ell}}(\cdot)$ *as Bayes filter correction in line 5. We can run the algorithm in almost-surely finite running time and, on termination, it outputs a sample from the hard-sphere Gibbs measure* μ_λ *on* Λ*. Moreover, if the hard-sphere model satisfies* (a, b)*-strong spatial mixing and if* ℓ *is chosen as a sufficiently large constant, depending on* a*,* b*,* r*,* λ *and* d*, then we can run the algorithm in expected time* $\mathrm{O}(|\Lambda|)$*.*

References

1 Michael Aizenman and Richard Holley. Rapid convergence to equilibrium of stochastic Ising models in the Dobrushin Shlosman regime. *Percolation theory and ergodic theory of infinite particle systems*, pages 1–11, 1987.

2 Berni Julian Alder and Thomas Everett Wainwright. Phase transition for a hard sphere system. *The Journal of Chemical Physics*, 27(5):1208–1209, 1957.

3 Konrad Anand, Andreas Göbel, Marcus Pappik, and Will Perkins. Perfect sampling for hard spheres from strong spatial mixing. *arXiv preprint*, 2023. `arXiv:2305.02450`.

4 Konrad Anand and Mark Jerrum. Perfect sampling in infinite spin systems via strong spatial mixing. *SIAM Journal on Computing*, 51(4):1280–1295, 2022.

5 Søren Asmussen, Peter W Glynn, and Hermann Thorisson. Stationarity detection in the initial transient problem. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 2(2):130–157, 1992.

6 Etienne P Bernard and Werner Krauth. Two-step melting in two dimensions: first-order liquid-hexatic transition. *Physical Review Letters*, 107(15):155704, 2011.

7 Etienne P Bernard, Werner Krauth, and David B Wilson. Event-chain Monte Carlo algorithms for hard-sphere systems. *Physical Review E*, 80(5):056704, 2009.

8 Steffen Betsch and Günter Last. On the uniqueness of Gibbs distributions with a non-negative and subcritical pair potential. In *Annales de l'Institut Henri Poincare (B) Probabilites et statistiques*, volume 59(2), pages 706–725. Institut Henri Poincaré, 2023.

9 Siddharth Bhandari and Sayantan Chakraborty. Improved bounds for perfect sampling of k-colorings in graphs. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pages 631–642, 2020.

10 Zongchen Chen, Kuikui Liu, Nitya Mani, and Ankur Moitra. Strong spatial mixing for colorings on trees and its algorithmic applications. *arXiv preprint*, 2023. `arXiv:2304.01954`.

11 Hofer Temmel Christoph. Disagreement percolation for the hard-sphere model. *Electronic Journal of Probability*, 24:1–22, 2019.

12 David Dereudre. Introduction to the theory of Gibbs point processes. In *Stochastic Geometry*, pages 181–229. Springer, 2019.

13 Persi Diaconis. The Markov Chain Monte Carlo revolution. *Bulletin of the American Mathematical Society*, 46(2):179–205, 2009.

14 Shaddin Dughmi, Jason Hartline, Robert D Kleinberg, and Rad Niazadeh. Bernoulli factories and black-box reductions in mechanism design. *Journal of the ACM (JACM)*, 68(2):1–30, 2021.

15 Martin Dyer, Alistair Sinclair, Eric Vigoda, and Dror Weitz. Mixing in time and space for lattice spin systems: A combinatorial view. *Random Structures & Algorithms*, 24(4):461–479, 2004.

16 Michael Engel, Joshua A Anderson, Sharon C Glotzer, Masaharu Isobe, Etienne P Bernard, and Werner Krauth. Hard-disk equation of state: First-order liquid-hexatic transition in two dimensions with three simulation methods. *Physical Review E*, 87(4):042134, 2013.

17 Stefan Felsner and Lorenz Wernisch. Markov chains for linear extensions, the two-dimensional case. In *SODA*, pages 239–247, 1997.

18 Weiming Feng, Heng Guo, and Yitong Yin. Perfect sampling from spatial mixing. *Random Structures & Algorithms*, 61(4):678–709, 2022.

19 Weiming Feng and Yitong Yin. On local distributed sampling and counting. In *Proceedings of the 2018 ACM Symposium on Principles of Distributed Computing*, pages 189–198, 2018.

20 Roberto Fernández, Aldo Procacci, and Benedetto Scoppola. The analyticity region of the hard sphere gas. Improved bounds. *J. Stat. Phys.*, 5:1139–1143, 2007.

21 Pablo A Ferrari, Roberto Fernández, and Nancy L Garcia. Perfect simulation for interacting point processes, loss networks and Ising models. *Stochastic Processes and their Applications*, 102(1):63–88, 2002.

22 David Gamarnik, Dmitriy Katz, and Sidhant Misra. Strong spatial mixing of list coloring of graphs. *Random Structures & Algorithms*, 46(4):599–613, 2015.

23 Nancy L Garcia. Perfect simulation of spatial processes. *Resenhas do Instituto de Matemática e Estatística da Universidade de São Paulo*, 4(3):283–325, 2000.

24 J Groeneveld. Two theorems on classical many-particle systems. *Phys. Letters*, 3, 1962.

25 Heng Guo and Mark Jerrum. Perfect simulation of the hard disks model by partial rejection sampling. *Annales de l'Institut Henri Poincaré D*, 8(2):159–177, 2021.

26 Heng Guo, Mark Jerrum, and Jingcheng Liu. Uniform sampling through the Lovász local lemma. *Journal of the ACM (JACM)*, 66(3):1–31, 2019.

27 Olle Häggström and Karin Nelander. Exact sampling from anti-monotone systems. *Statistica Neerlandica*, 52(3):360–380, 1998.

28 Olle Häggström, Marie-Colette N.M. van Lieshout, and Jesper Møller. Characterization results and Markov chain Monte Carlo algorithms including exact simulation for some spatial point processes. *Bernoulli*, 5(4):641–658, 1999.

29 Thomas P Hayes and Cristopher Moore. Lower bounds on the critical density in the hard disk model via optimized metrics. *arXiv preprint*, 2014. `arXiv:1407.1930`.

30 Kun He, Xiaoming Sun, and Kewen Wu. Perfect sampling for (atomic) Lovász Local Lemma. *arXiv preprint*, 2021. `arXiv:2107.03932`.

31 Kun He, Chunyang Wang, and Yitong Yin. Sampling Lovász Local Lemma for general constraint satisfaction solutions in near-linear time. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 147–158. IEEE, 2022.

32 Kun He, Kewen Wu, and Kuan Yang. Improved bounds for sampling solutions of random CNF formulas. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 3330–3361. SIAM, 2023.

33 Tyler Helmuth, Will Perkins, and Samantha Petti. Correlation decay for hard spheres via Markov chains. *The Annals of Applied Probability*, 32(3):2063–2082, 2022.

34 Christoph Hofer-Temmel and Pierre Houdebert. Disagreement percolation for Gibbs ball models. *Stochastic Processes and their Applications*, 129(10):3922–3940, 2019.

35 Richard Holley. Possible rates of convergence in finite range, attractive spin systems. *Part. Syst. Random Media Large Deviat.*, 41:215, 1985.

36 Mark Huber. Spatial birth–death swap chains. *Bernoulli*, 18(3):1031–1041, 2012.

37 Mark Huber. Nearly optimal Bernoulli factories for linear functions. *Combin. Probab. Comput.*, 25(4):577–591, 2016.

38 Mark Huber, Elise Villella, Daniel Rozenfeld, and Jason Xu. Bounds on the artificial phase transition for perfect simulation of hard core Gibbs processes. *Involve, a Journal of Mathematics*, 5(3):247–255, 2013.

39 Masaharu Isobe. Hard sphere simulation in statistical physics—methodologies and applications. *Molecular Simulation*, 42(16):1317–1329, 2016.

40 Vishesh Jain, Ashwin Sah, and Mehtaab Sawhney. Perfectly sampling $k \geq (8/3 + o(1))\Delta$-colorings in graphs. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 1589–1600, 2021.

41 Matthew Jenssen, Marcus Michelen, and Mohan Ravichandran. Quasipolynomial-time algorithms for repulsive Gibbs point processes. *arXiv preprint*, 2022. `arXiv:2209.10453`.

42 Mark Jerrum and Alistair Sinclair. The Markov chain Monte Carlo method: an approach to approximate counting and integration. *Approximation algorithms for NP-hard problems*, pages 482–520, 1996.

43 Ravi Kannan, Michael W. Mahoney, and Ravi Montenegro. Rapid mixing of several Markov chains for a hard-core model. In *Algorithms and computation*, volume 2906 of *Lecture Notes in Comput. Sci.*, pages 663–675. Springer, Berlin, 2003.

44 Frank P Kelly and Brian D Ripley. A note on Strauss's model for clustering. *Biometrika*, pages 357–360, 1976.

45 Wilfrid S Kendall. Perfect simulation for the area-interaction point process. In *Probability towards 2000*, pages 218–234. Springer, 1998.

46 Wilfrid S Kendall and Jesper Møller. Perfect simulation using dominating processes on ordered spaces, with application to locally stable point processes. *Advances in Applied Probability*, pages 844–865, 2000.

47 Botao Li, Yoshihiko Nishikawa, Philipp Höllmer, Louis Carillo, AC Maggs, and Werner Krauth. Hard-disk pressure computations – A historic perspective. *The Journal of Chemical Physics*, 157(23):234111, 2022.

48 Jingcheng Liu, Alistair Sinclair, and Piyush Srivastava. Correlation decay and partition function zeros: Algorithms and phase transitions. *SIAM Journal on Computing*, pages FOCS19–200, 2022.

49 Laszlo Lovasz and Peter Winkler. Exact mixing in an unknown Markov chain. *The Electronic Journal of Combinatorics*, pages R15–R15, 1995.

50 Hartmut Löwen. Fun with hard spheres. In *Statistical physics and spatial statistics*, volume 554, pages 295–331. Springer, 2000.

51 Pinyan Lu and Yitong Yin. Improved FPTAS for multi-spin systems. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques: 16th International Workshop, APPROX 2013, and 17th International Workshop, RANDOM 2013, Berkeley, CA, USA, August 21-23, 2013. Proceedings*, pages 639–654. Springer, 2013.

52 Fabio Martinelli. Lectures on Glauber dynamics for discrete spin models. In *Lectures on probability theory and statistics*, pages 93–191. Springer, 1999.

53 Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953.

54 Marcus Michelen and Will Perkins. Potential-weighted connective constants and uniqueness of Gibbs measures. *arXiv preprint*, 2021. `arXiv:2109.01094`.

55 Marcus Michelen and Will Perkins. Analyticity for classical gasses via recursion. *Communications in Mathematical Physics*, pages 1–22, 2022.

56 Marcus Michelen and Will Perkins. Strong spatial mixing for repulsive point processes. *Journal of Statistical Physics*, 189(1):9, 2022.

57 Sarat B Moka, Dirk P Kroese, et al. Perfect sampling for Gibbs point processes using partial rejection sampling. *Bernoulli*, 26(3):2082–2104, 2020.

58 Jesper Møller. A review of perfect simulation in stochastic geometry. *Lecture Notes-Monograph Series*, pages 333–355, 2001.

59 Jesper Møller and Rasmus Plenge Waagepetersen. *Statistical inference and simulation for spatial point processes*. CRC Press, 2003.

60 Duncan J Murdoch and Peter J Green. Exact sampling from a continuous state space. *Scandinavian Journal of Statistics*, 25(3):483–502, 1998.

61 Serban Nacu and Yuval Peres. Fast simulation of new coins from old. *The Annals of Applied Probability*, 15(1A):93–115, 2005.

62 Oliver Penrose. Convergence of fugacity expansions for fluids and lattice gases. *Journal of Mathematical Physics*, 4(10):1312–1320, 1963.

63 James Gary Propp and David Bruce Wilson. Exact sampling with coupled Markov chains and applications to statistical mechanics. *Random Structures & Algorithms*, 9(1-2):223–252, 1996.

64 James Gary Propp and David Bruce Wilson. How to get a perfectly random sample from a generic Markov chain and generate a random spanning tree of a directed graph. *Journal of Algorithms*, 27(2):170–217, 1998.

65 Dana Randall. Rapidly mixing Markov chains with applications in computer science and physics. *Computing in Science & Engineering*, 8(2):30–41, 2006.

66 Guus Regts. Absence of zeros implies strong spatial mixing. *Probability Theory and Related Fields*, pages 1–21, 2023.

67 David Ruelle. Correlation functions of classical gases. *Annals of Physics*, 25:109–120, 1963.

68 David Ruelle. *Statistical mechanics: Rigorous results*. World Scientific, 1999.

69 Alistair Sinclair, Piyush Srivastava, Daniel Štefankovič, and Yitong Yin. Spatial mixing and the connective constant: Optimal bounds. *Probability Theory and Related Fields*, 168(1-2):153–197, 2017.

70 Yinon Spinka. Finitary codings for spatial mixing Markov random fields. *Ann. Probab.*, 48(3):1557–1591, 2020.

71 David J Strauss. A model for clustering. *Biometrika*, 62(2):467–475, 1975.

72 Daniel W Stroock and Boguslaw Zegarlinski. The logarithmic Sobolev inequality for discrete spin systems on a lattice. *Communications in Mathematical Physics*, 149(1):175–193, 1992.

73 Marie-Colette N.M. van Lieshout. *Markov Point Processes and Their Applications*. Imperial College Press, 2000.

74 Dror Weitz. Counting independent sets up to the tree threshold. In *Proceedings of the Thirty-Eighth Annual ACM Symposium on Theory of Computing*, STOC 2006, pages 140–149. ACM, 2006.

75 William W. Wood, F. Raymond Parker, and Jack David Jacobson. Recent monte carlo calculations of the equation of state of lenard-jones and hard sphere molecules. *Il Nuovo Cimento (1955-1965)*, 9:133–143, 1958.