A new class of Markov random fields enabling lightweight sampling

Jean-Baptiste Courbot
IRIMAS UR 7499, Université de Haute-Alsace, Mulhouse, France
and
Hugo Gangloff
Université Paris-Saclay, AgroParisTech, INRAE UMR MIA Paris-Saclay, France
and
Bruno Colicchio
IRIMAS UR 7499, Université de Haute-Alsace, Mulhouse, France

October 6, 2025

Abstract

This work addresses the problem of efficient sampling of Markov random fields (MRF). The sampling of Potts or Ising MRF is most often based on Gibbs sampling, and is thus computationally expensive. We consider in this work how to circumvent this bottleneck through a link with Gaussian Markov Random fields. The latter can be sampled in several cost-effective ways, and we introduce a mapping from real-valued GMRF to discrete-valued MRF. The resulting new class of MRF benefits from a few theoretical properties that validate the new model. Numerical results show the drastic performance gain in terms of computational efficiency, as we sample at least 35x faster than Gibbs sampling using at least 37x less energy, all the while exhibiting empirical properties close to classical MRFs.

Keywords: Gaussian Markov Random field, computational efficiency, image processing

1 Introduction

In this paper, we consider two classes of random fields: Markov Random fields (MRFs) and Gaussian Markov Random Fields (GMRFs). This introduction summarizes their main properties as well as procedures to sample from these processes.

1.1 Markov random fields

1.1.1 Definition

In the context of image processing, Markov Random Fields (MRFs) are a class of statistical processes that are often used to represent latent (or hidden) processes of interest, typically in segmentation tasks. Let $\mathcal{S}$ be the set of n sites in an image, and let us denote $\mathbf{X} = \{X_s\}_{s\in\mathcal{S}}$. Denoting $\mathbf{X}$ a random vector and $\mathbf{x}$ its realization, we will also note, for brevity, the distributions $p(\mathbf{X} = \mathbf{x})$ as $p(\mathbf{x})$ when there is no ambiguity.

The Markov property for $\mathbf{X}$ is the following. $\mathbf{X}$ is a MRF if and only if there exists a neighboring system N such that $\forall s$:

$$p(x_s|\mathbf{x}_{\mathcal{S}\setminus s}) = p(x_s|\mathbf{x}_{N_s}). \tag{1}$$

The neighboring system can be chosen at discretion, but usually refers to 4- or 8-sets of pixels adjacent to the location s. Thanks to the Hammersley-Clifford theorem (Clifford & Hammersley 1971), and considering only pairwise site interactions, the density of the process can be written as:

$$p(\mathbf{x}) = \frac{1}{\gamma}\exp\left(-\sum_{s\in\mathcal{S}}\sum_{s'\in N_s}\psi(x_s, x_{s'})\right), \tag{2}$$

where ψ is a pairwise potential and γ is the partition function. Typically, one can choose for $\beta > 0$:

$$\begin{array}{rrcll} & \psi(x_s, x_{s'}) & = & \beta\delta_{\{x_s=x_{s'}\}} & \text{(Potts potential)}, \\ \text{or} & \psi(x_s, x_{s'}) & = & \beta(1-\delta_{\{x_s=x_{s'}\}}) & \text{(Ising potential)}, \end{array} \tag{3}$$

with δ the Kronecker function. Later, the corresponding fields will be referred to as Potts or Ising MRFs, respectively.

1.1.2 Sampling

Computing γ is in the general case intractable as it requires to sum over all possible configurations of the n-valued random vector $\mathbf{x}$. Thus, realizations $\mathbf{X} = \mathbf{x}$ are approximated through iterative sampling techniques. This is mainly performed from Gibbs sampling as proposed by (Geman & Geman 1984). Because this sampling has a high computational cost, various efforts have been made in the literature to approximate γ (Giovannelli 2007), to find other models (Laferté et al. 2000), or to improve updates within the Metropolis-Hastings sampler (Grathwohl et al. 2021). Gibbs sampling nevertheless remains overall the main technique for sampling MRFs.

As Gibbs sampling is a MCMC algorithm, its computational cost is mainly dependent on the expected time it takes to approximate the target distribution. This depends on the

mixing rate of the underlying Markov chain, which in turn depends on the eigenvalues of its transition matrix. For instance, Frigessi et al. (1997) discuss the convergence time t^*, that is, the number of individual site update the Gibbs sample requires to reach an arbitrary precision, depending on the number of sites n. In their Theorem 1, the authors show, in the Ising case that $t^* \leq \mathcal{O}(n\log(n))$ or $t^* < \mathcal{O}(n^2\log(n))$; when there is no phase transition, *i.e.* for small values of the Ising parameter β.

Such values however imply that the resulting realization has no large homogeneous area, and thus these results are no relevant in most of image-processing based situations. For large values of β and in dimension 2, a higher bound is found as $t^* \leq \mathcal{O}(\exp(\beta C\sqrt{n}))$, with C a constant. Overall, works dealing with the convergence of Potts MRF sampling are quite seldom, and the results are also obtained for values of the Potts parameter that also makes them irrelevant for image processing tasks (see, *e.g.*,(Gheissari & Lubetzky 2016, Mossel & Sly 2013)).

More recent work proposes using auxiliary Gaussian variables in an MCMC scheme in order to assist in sampling Ising (Martens & Sutskever 2010) and Potts (Margossian & Mukherjee 2021) distributions, enabling in the latter case a computational complexity of $\mathcal{O}(n^3 + mn^2K)$, K being the number of classes and m the number of Gibbs iterations. Other propositions involve the reformulation of Gibbs sampling over relevant sub-lattices, leading to *chromatic* (Gonzalez et al. 2011) or *conclique* (Kaplan et al. 2020) Gibbs sampling. The computational cost of a chromatic Gibbs sampler is $\mathcal{O}(\frac{n}{q} + c)$ with c colors (*i.e.*, c independent sub-lattices) and q processors over n sites. However, the works reported in this paragraph address the computational cost for a single Gibbs path over the grid, and thus the statistical convergence cost overhead must also be accounted for. To our knowledge, the chromatic Gibbs sampler is, to date, the most relevant sampler for Ising/Potts MRF when dealing with large images. It is reported in Appendix A.1 and will serve as a reference for comparison in the numerical section.

1.2 Gaussian Markov random fields

1.2.1 Definition

Gaussian random fields (GRFs) are another class of statistical processes that are commonly used to model spatial dependencies between real-valued variables. In the case of image processing, GRFs may also be indexed by a lattice. Let us define $\mathbf{Z}$ as a GRF on the lattice $\mathcal{S}$ iff $\mathbf{Z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\boldsymbol{\mu} \in \mathbb{R}^S$ and $\boldsymbol{\Sigma} \in \mathbb{R}^{S\times S}$ are the mean vector and the covariance matrix, respectively.

Upon known conditions on $\boldsymbol{\Sigma}$ (Rozanov 1982, p. 120), GRFs are also Markovian, and are then referred to as Gaussian Markov random fields (GMRFs). These conditions imply that the precision matrix $\mathbf{Q} = \boldsymbol{\Sigma}^{-1}$ is sparse, in other words, that conditionally to any given site $s \in \mathcal{S}$, the correlation spans a limited neighborhood, yielding a Markov property similar to (1).

1.2.2 Sampling

The algorithmic complexity of GMRF sampling relies on the assumptions one is willing to make on $\mathbf{\Sigma}$. Following (Rue & Held 2005), the simplest option is to rely on a Cholesky decomposition of $\mathbf{\Sigma}$. However, this is rather ineffective as n grows, as its computational complexity is $\mathcal{O}(n^3)$.

Fourier sampling. On the other hand, assuming that $\mathbf{\Sigma}$ is circulant enables computations based on discrete Fourier sampling. Thus, under this assumption, there is no need to compute or store neither $\mathbf{Q}$ nor $\mathbf{\Sigma}$ as their bases are sufficient to perform computations. However, sampling under this assumption implies that $\mathcal{S}$ is joined at the sides, making it a torus, and that $\mathbf{Z}$ is stationary on $\mathcal{S}$, *i.e.*, that it locally behaves similarly regardless of the site s.

Spectral sampling. More recent works (Allard et al. 2020) leveraged the torus constraint by introducing a *spectral* sampling method. The only requirement here is that the covariance function belongs to an extended Gneiting class of covariances, which includes in particular Matérn correlation functions.

Fourier and spectral sampling of GMRFs are both computationally efficient and will be considered later on. Details of their implementations are reported in Appendix A.2.

1.3 This paper

This paper introduces a novel class of Markov random fields, designed such that samples can be obtained through GMRF sampling. In Section 2 we describe the processes involved and their key properties, as well as the sampling procedure. Section 3 proposes then a numerical study, focusing on the sampling time and energy consumption, and on the empirical properties of sampled fields. Then, Section 4 wraps up the results and highlights some relations to other works. Finally, we gather in the Appendix the main algorithms mentioned in the paper, as well as additional numerical results. Note also that a preliminary version of this work was presented in (Courbot & Gangloff 2025).

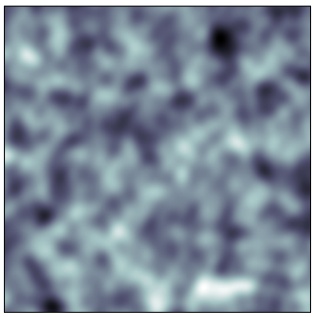
We also use a few hypotheses and notational conventions through this paper, that we summarize here. We assume all random processes are indexed by a finite grid $\mathcal{S}$. Discrete-valued MRFs will take their values ω in the K-class set $\Omega \overset{\text{def.}}{=} \{\omega_0, \omega_1, \dots, \omega_{K-1}\} \subseteq \mathbb{N}$. Bold notation will refer to vectors, matrices, or random processes, while non-bold notation will refer to scalar values or random variables. Besides, $\mathbf{0}_N$ will refer to a zero-valued vector of size N while $\mathbf{I}_N$ will refer to a square identity matrix of size $N \times N$, and Euclidean distances will be denoted $\|\cdot\|$.

2 Gaussian Unit simplex Markov random fields

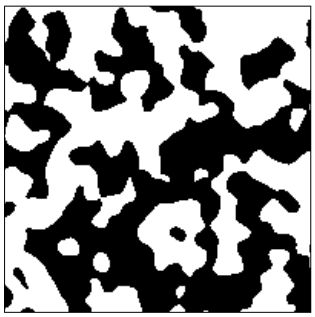
2.1 Definitions

The intuition underlying the link between continuous GMRF and discrete MRFs is the following: a thresholded version of a GMRF yields a two-class field that is visually similar to a binary MRF (see Fig. 1). In this section, we propose an expansion beyond the intuitive

$K = 2$ classes case and using a continuous function of a GMRF $\mathbf{Z}$ to perform the mapping towards the discrete case. The expansion in more than 2 dimensions requires handling more than one GMRF realization and splitting the space they take value in. To do so, we use the following definition of unit simplex.



(a) GMRF realization $\mathbf{z} \in \mathbb{R}^n$.



(b) Discretized field $\mathbf{f} \in \{0,1\}^n$ such that $\forall s \in \mathcal{S}, f_s = \mathbb{1}_{\{z_s > 0\}}$.

(c) Realization of a Potts MRF ($K = 1$).

Figure 1: Original intuition motivating this work: thresholding a GMRF realization (a) yields a discrete random field (b) that behaves similarly to a discrete MRF realization (c).

Definition 1 (Unit simplex)**.** *A unit P-simplex is a regular simplex belonging in $\mathbb{R}^P$, whose $P+1$ vertices lie on a unit sphere. From (Anderson & Thron 2022) we take its vertices $\mathbf{v} \in \mathbb{R}^P$ as:*

$$\mathbf{v}_j = \sqrt{\frac{P+1}{P}}\mathbf{e}_j - \frac{\sqrt{P+1}-1}{P\sqrt{P}}\mathbf{1} \; \forall 1 \leq j \leq P \text{ and } \mathbf{v}_{P+1} = \frac{-1}{\sqrt{P}}\mathbf{1} \tag{4}$$

with $\mathbf{1} \in \mathbb{R}^P$ a vector of ones and $\mathbf{e}_j \in \mathbb{R}^P$ the $j-$th basis vector.

Let us denote $\mathbf{u}_P = \{\mathbf{v}_1, \ldots, \mathbf{v}_{P+1}\}$ the set of vertices of the unit P-simplex. Note that these are coordinate-invariant, so any permutation also lies on the unit sphere.

The second ingredient towards a generalized thresholding lies in the definition of a multivariate GMRF, which is defined as a GMRF with vector values on each $s \in \mathcal{S}$. Throughout this section, we will handle the multivariate GMRF $\mathbf{Z}$ such that for a given number of classes K:

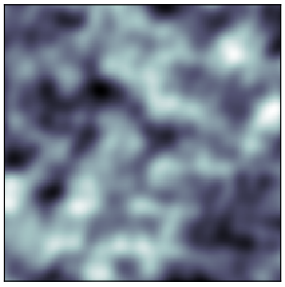
$$\mathbf{Z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \text{ with mean } \boldsymbol{\mu} \in \mathbb{R}^{n(K-1)} \text{ and covariance matrix } \boldsymbol{\Sigma} \in \mathbb{R}^{n(K-1)\times n(K-1)} \tag{5}$$

Each random variable within $\mathbf{Z}$ is indexed both in terms of location on the grid $\mathcal{S}$, and in terms of *component* among the $K-1$ values given at each s. Later in the paper, we will also refer to components of $\mathbf{Z}$, that is, the $K-1$ fields $\mathbf{Z}_k$ taking values in $\mathbb{R}^n$. We are now equipped to define a mapping of $\mathbf{Z} = \{\mathbf{Z}_k\}_{k=0}^{K-2}$ with respect to vertices of a unit simplex.

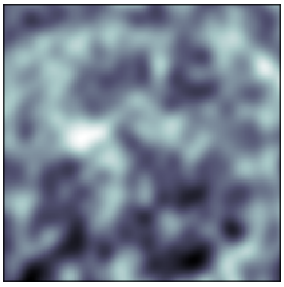
Definition 2 (Mapping with respect to unit simplex)**.** *Let $c > 0$. We design $\pi_i^c : \mathbb{R}^{K-1} \mapsto [0,1]$ to be a measure of the distance between $\mathbf{Z}$ and the $i-$th vertice $\mathbf{v}_i$ of $\mathbf{u}_{K-1}$, such that $\forall s \in \mathcal{S}$:*

$$\pi_i^c(\mathbf{Z}_s) = \frac{\exp(-c^{-2}\|\mathbf{Z}_s - \mathbf{v}_i\|^2)}{\sum_{k=0}^{K-1}\exp(-c^{-2}\|\mathbf{Z}_s - \mathbf{v}_k\|^2)} \tag{6}$$

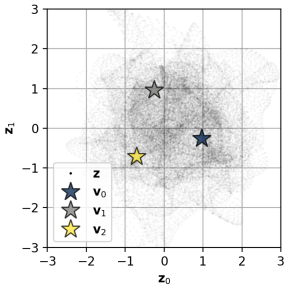
with $c > 0$ and $\mathbf{v}_k, \mathbf{v}_i \in \mathbf{u}_{K-1}$ unit simplex vertices.



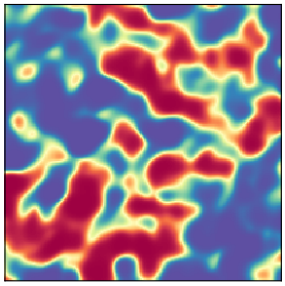
(a) $\mathbf{Z} = \mathbf{z}$: first component $\mathbf{z}_0$



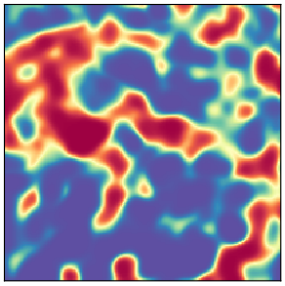
(b) $\mathbf{Z} = \mathbf{z}$: second component $\mathbf{z}_1$.



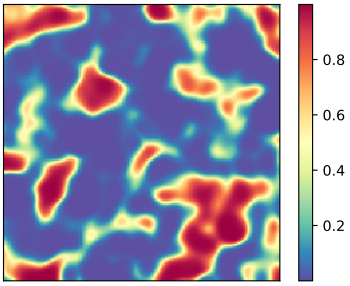
(c) $\mathbf{Z} = \mathbf{z}$ projected in $\mathbb{R}^2$, and locations of the unit simplex vertices.



(d) $\pi_0^c(\mathbf{z})$.



(e) $\pi_1^c(\mathbf{z})$.



(f) $\pi_2^c(\mathbf{z})$.

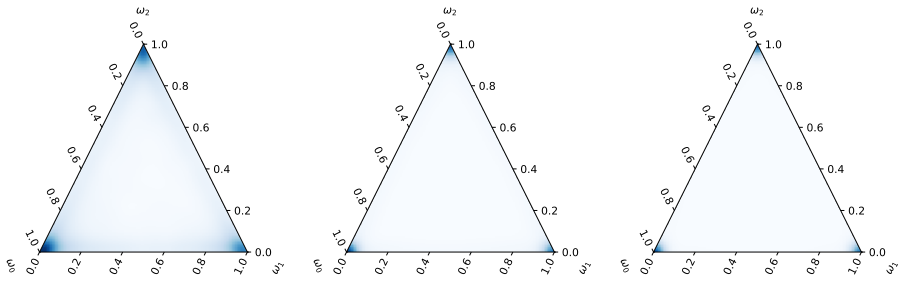
Figure 2: Depiction of a realization of $\mathbf{Z}$ in the $K = 3$ case, together with the mapping π_k^c, with $c = 1$.

An illustration of the application of π_i^c to a realization $\mathbf{Z} = \mathbf{z}$ is given in Fig. 2.

Remark 1. *Besides, as* $\sum_{i=0}^{K-1} \pi_i^c(\mathbf{Z}_s) = 1$, *the set* $\{\pi_i^c(\mathbf{Z}_s)\}_{0 \leq i \leq K-1}$ *can be seen as a set of probabilities and represented on the* probability *simplex. Figure 3 depicts this mapping for several values of* c *in the* $K = 3$ *case. This mapping highlights the similarity of the mapping* (6) *with other distributions, namely the logit-normal multivariate distribution, and the Concrete or Gumbel softmax distributions (Maddison et al. 2017, Jang et al. 2017). While its behavior at the limit of* $c \to \infty$ *is similar to those distributions, the distribution induced by* (6) *is however different:*

- *it can be seen as a softmax of* distances *of Gaussian variables of the vertices of the simplex.*
- *it yields spatial correlations as* $\mathbf{Z}$ *is a GMRF.*
- *our purpose here is sampling discrete, spatially correlated variables.*

Definition 3 (Gaussian Unit-simplex Markov random field (GUM))**.** *Let* $\mathbf{u}_{K-1}$ *be a unit* $(K-1)$*-simplex (Definition 1),* $\mathbf{Z}$ *a multivariate GMRF (Eq. (5)), the set of classes* $\Omega = \{\omega_0, \ldots, \omega_{K-1}\} \subseteq \mathbb{N}$, *and* $c > 0$.



(a) $c = 1$ (b) $c = 0.5$ (c) $c = 0.25$

Figure 3: Kernel density estimation obtained from the distribution of the triplet $\left(\pi_0^c(z_s), \pi_1^c(z_s), \pi_2^c(z_s)\right)$ in the probability simplex whose vertices locates classes in $\Omega = \{\omega_0, \omega_1, \omega_2\}$, over all $s \in \mathcal{S}$ for a given realization $\mathbf{Z} = \mathbf{z}$. As $c \to \infty$, the distribution tends towards Dirac masses located at the vertices of the probability simplex.

We define the mapping $\phi_{K,c} \colon \mathbb{R}^{n(K-1)} \mapsto \mathbb{R}^n$ *such that:*

$$\phi_{K,c}(\mathbf{Z}) = \sum_{i=0}^{K-1} \omega_i \pi_i^c(\mathbf{Z}) \tag{7}$$

$\phi_{K,c}(\mathbf{Z})$ *is named a* GUM random field.

This transformation can be seen as a class-weighted measure of distance (seen from a Gaussian distribution of standard deviation c) between the points in $\mathbf{Z} = \mathbf{z}$ and the vertices of the unit simplex.

Remark 2. *Note that for this mapping to be well-defined, we need to represent classes with a set* $\Omega \subseteq \mathbb{R}^+$*. We set* $\Omega \subseteq \mathbb{N}$ *for this purpose. We consider this is not limiting, as any discrete set can be indexed in* $\mathbb{N}$ *and thus mapped to positive integers.*

We finally introduce a limiting process that will yield a discretization of $\phi_{K,c}(\mathbf{Z})$.

Definition 4 (Discrete GUM)**.** *Let* $\mathbf{Z}$ *be a GMRF. From Definition 3, we have:*

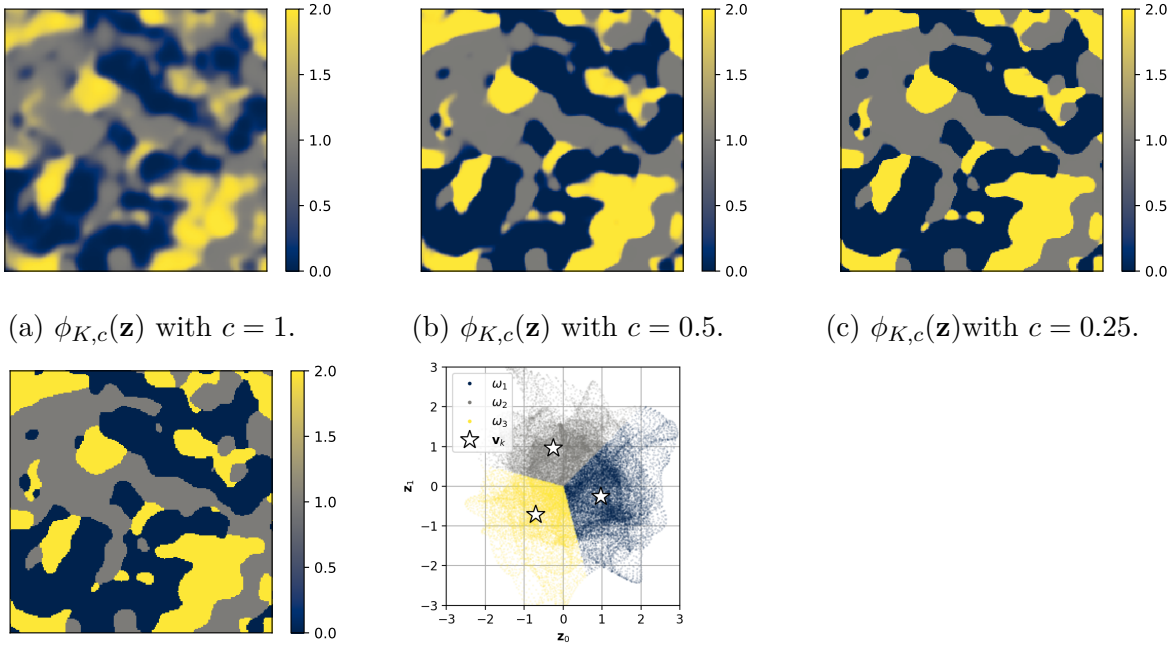
$$\phi_{K,c}(\mathbf{Z}) \underset{c \to 0}{\longrightarrow} \sum_{i=0}^{K-1} \omega_i \delta_{\left[\|\mathbf{Z}-\mathbf{v}_i\| \leq \|\mathbf{Z}-\mathbf{v}_k\|,\ \forall \mathbf{v}_k \in \mathbf{U}_{K-1}\right]} \tag{8}$$

Denoting $\lim_{c \to 0} \phi_{K,c}(\mathbf{Z}) = \phi_K(\mathbf{Z}) = \mathbf{X} = \{X_s\}_{s \in \mathcal{S}}$*, we have* $\forall s \in \mathcal{S}$*:*

$$X_s = \omega_{k^*} \text{ with } k^* \text{ chosen such that } \mathbf{v}_{k^*} = \underset{\mathbf{v} \in \mathbf{U}_{K-1}}{\arg\min} \|\mathbf{Z}_s - \mathbf{v}\| \tag{9}$$

This discrete limit process will be referred to as a Discretized GUM *or* DGUM*.*

In other words, the mapping $\mathbf{Z} \mapsto \lim_{c \to 0} \phi_{K,c}(\mathbf{Z})$ indicates pointwise from which vertices among the unit $(K-1)$-simplex vertices each $\mathbf{z}_s$ is the closest. The process linking a realization of a GMRF $\mathbf{Z} = \mathbf{z}$ to its DGUM is reported in Fig. 4.



(a) $\phi_{K,c}(\mathbf{z})$ with $c = 1$.

(b) $\phi_{K,c}(\mathbf{z})$ with $c = 0.5$.

(c) $\phi_{K,c}(\mathbf{z})$with $c = 0.25$.

(d) DGUM of $\mathbf{z}$.

(e) $\mathbf{Z} = \mathbf{z}$ colored according to the classes in $\mathbf{x}$.

Figure 4: Illustration of the DGUM sampling for $K = 3$ classes. The sampling is performed from $\mathbf{Z} = \mathbf{z}$ depicted in Fig. 2. (a)-(c) depict the GUM realization (7), and (d) depicts its limit DGUM realization $\mathbf{x}$ (9). (e) depicts the classes in $\mathbf{x}$ for the values of $\mathbf{z}$ in $\mathbb{R}^2$.

2.2 Properties

In this section, we state and prove a few properties of the GUM and DGUM processes defined in the previous section in order to better understand their theoretical behavior.

Proposition 1 (Markovianity of $\phi_{K,c}(\mathbf{Z})$)**.** *Let* $\mathbf{Z}$ *be a GMRF. Then,* $\phi_{K,c}(\mathbf{Z})$ *is a Markov random field.*

Proof. Let us denote $\mathcal{F}(A)$ the $\sigma-$algebra induced by a random variable A. Rephrasing Equation (1), $\mathbf{Z}$ is Markovian if and only if there exists a neighborhood system N such that for any $s \in \mathcal{S}$, $\mathcal{F}(\mathbf{Z}_{N_s})$ is splitting $\mathcal{F}(\mathbf{Z}_s)$ and $\mathcal{F}(\mathbf{Z}_{\mathcal{S}\setminus s,N_s})$ (see (Rozanov 1982, Chap. 2)).

Besides, $\phi_{K,c}$ being continuous, it is a Borel function. Then Doob-Dynkin lemma (Kallenberg 1997) states that its application on random variables contracts the sigma-algebra, that is: $\mathcal{F}(\phi_{K,c}(\mathbf{Z}_s)) \subseteq \mathcal{F}(\mathbf{Z}_s)$, $\mathcal{F}(\phi_{K,c}(\mathbf{Z}_{N_s})) \subseteq \mathcal{F}(\mathbf{Z}_{N_s})$ and $\mathcal{F}(\phi_{K,c}(\mathcal{S} \setminus s, N_s)) \subseteq \mathcal{F}(\mathcal{S} \setminus s, N_s)$. Hence, $\mathcal{F}(\phi_{K,c}(\mathbf{Z}_{N_s}))$ is also splitting $\mathcal{F}(\phi_{K,c}(\mathbf{Z}_s))$ and $\mathcal{F}(\phi_{K,c}(\mathbf{Z}_{\mathcal{S}\setminus s,N_s}))$. Thus, $\phi_{K,c}(\mathbf{Z})$ is also Markovian.

□

Proposition 2 (Markovianity of $\mathbf{X}$)**.** *Let* $\mathbf{Z}$ *be a GMRF. Then, the DGUM* $\phi_K(\mathbf{Z})$ *from Definition 4 is also Markovian.*

Proof. In the proof of Proposition 1, we saw that $\phi_{K,c}$ being a Borel function, it preserves the Markov property assumed in $\mathbf{Z}$. Let us index c by $u \in \mathbb{N}$ such that $\{c_u\}_{u\in\mathbb{N}}$ decreases and $\lim_{u\to\infty} c_u = 0$. Then, $\mathbf{X} = \lim_{c\to 0} \phi_{K,c}(\mathbf{Z}) = \lim_{u\to\infty} \phi_{K,c_u}(\mathbf{Z})$, so that ϕ_K is the pointwise limit of ϕ_{K,c_u}. As the latter is a Borel function, then ϕ_K is also a Borel function. Thus, $\phi_K(\mathbf{Z})$ is also Markovian. □

Remark 3. *The Markovianity of* $\mathbf{X}$*, provided that* $\mathbf{Z}$ *is Markovian, implies that* $\mathbf{X}$ *and* $\mathbf{Z}$ *share the same neighborhood structure. In practice, the covariance matrix of* $\mathbf{Z}$ *might dampen the correlation; thus the observed* $\mathbf{X}$ *may have a smaller apparent neighborhood dependency than the one described from* $\mathbf{Z}$*.*

We now specify further the conditions to reach class balance in DGUM realization, for which $\mathbf{Z}$ is required to be centered and isotropic.

Definition 5 (Balanced and unbalanced GUMs)**.** *Let us denote* $\boldsymbol{\mu} \in \mathbb{R}^{K-1}$ *the vector of scalar means of the* $(K-1)$ *random fields composing* $\mathbf{Z}$*. When* $\boldsymbol{\mu} = \mathbf{0}_{K-1}$*, we denote* $\mathbf{Z}$ *as a* balanced *GUM. Otherwise, it will be referred to as* unbalanced*. We similarly denote balanced and unbalanced DGUM from Definition 4.*

Definition 6 (Isotropic and anisotropic GUMs)**.** *Let* $\boldsymbol{\Sigma}$ *be the covariance matrix of* $\mathbf{Z}$*. We define the GUM to be* isotropic *when there exists a covariance matrix* $\boldsymbol{\Xi} \in \mathbb{R}^{n\times n}$ *such that* $\boldsymbol{\Sigma} = \mathbf{I}_{K-1} \otimes \boldsymbol{\Xi}$*, with* $\otimes$ *the Kronecker product. In other words, all* $(K-1)$ *components of* $\mathbf{Z}$ *share the same covariance matrix, and are independent from each other. Otherwise, the GUM is said to be* anisotropic*. We hold the same definition regarding the DGUM.*

Figure 5 depicts the distributions of $\mathbf{Z}$ in the unbalanced and anisotropic cases. In the remainder of the paper, we will focus on balanced isotropic GUMs and DGUMs.

Proposition 3 (Class balance)**.** *The balanced isotropic DGUM (definitions 5 and 6) reaches class balance:*

$$\forall k,\ p(X_s = \omega_k) = \frac{1}{K} \tag{10}$$

Proof. Definition 5 implies that $\mathbf{Z}$ is centered, and definition 6 that all components within $\mathbf{Z}$ are independent.

Then, the marginal is written, $\forall s$:

$$p(\mathbf{Z}_s) \sim \mathcal{N}(\mathbf{0}_{K-1}, \mathbf{I}_{K-1}\Xi_{s,s}). \tag{11}$$

In other words, when looking at a specific $s \in \mathcal{S}$, $\mathbf{Z}_s$ follows an isotropic, centered normal distribution of variance $\Xi_{s,s}$.

Besides, the unit simplex partitions $\mathbb{R}^{(K-1)}$ into K regions $B_k = \{\mathbf{z} \in \mathbb{R}^{K-1} : \|\mathbf{z} - \mathbf{v}_k\| \leq \|\mathbf{z} - \mathbf{v}_i\|, \forall i\}$ that correspond to Voronoï cells of the simplex vertices. The Voronoï cells are defined symmetrically, as the v_k are equidistant and centered at the origin. Thus, for any i, j there exists a rotation $\mathcal{R}_{ij}$ such that $\mathcal{R}_{ij}(v_i) = v_j$ and $\mathcal{R}_{ij}(B_i) = B_j$.

As the normal distribution is invariant under orthogonal transformation, $\mathcal{R}_{ij}(\mathbf{Z}_s)$ also follows an isotropic, centered normal distribution. Besides, the B_k are rotationally symmetric under

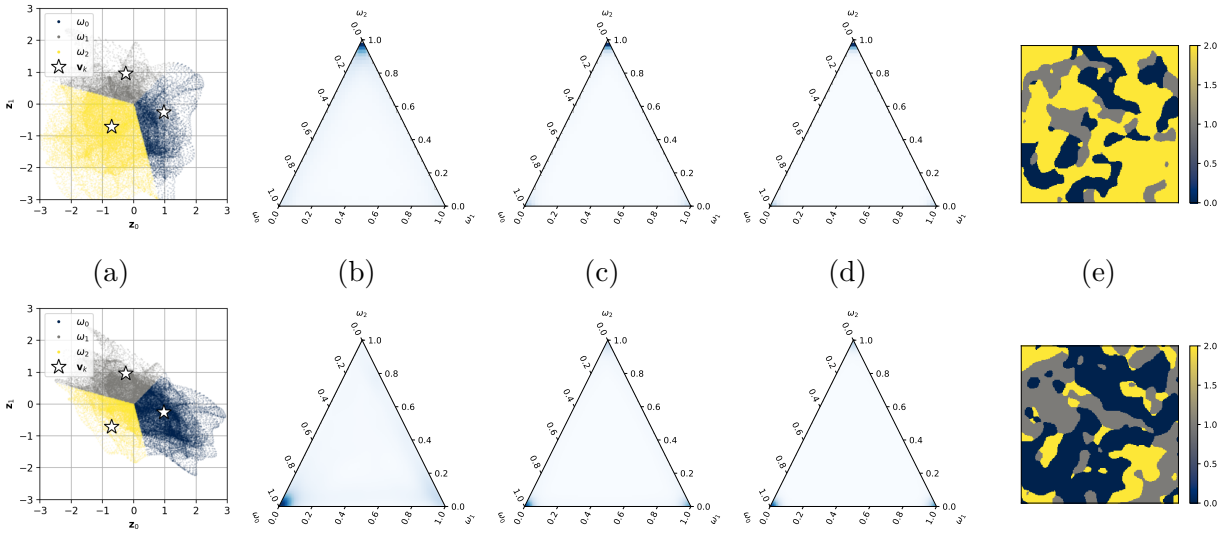


Figure 5: Depiction of unbalanced (first line) and anisotropic (second line) GUM and DGUM distribution and realization, in the $K = 3$ case. (a) and (f) depict the marginal locations in $\mathbb{R}^2$ of a realization $\mathbf{z}$, colored according to the class ω, as in Fig. 4e. (b-d) and (g-i) depict the ternary distribution of π (as in Fig. 3), showing in particular how these two situations enable class unbalance. (e) and (j) depict the corresponding DGUM realization.

permutation of the vertices. Thus, $\forall s$:

$$p(\mathbf{Z}_s \in B_i) = p(\mathcal{R}_{ij}(\mathbf{Z}_s) \in B_j) = p(\mathbf{Z}_s \in B_j) \tag{12}$$

As this is true for any i, j, then $\forall s$ and $\forall k$:

$$p(\mathbf{Z}_s \in B_k) = p(X_s = \omega_k) = \frac{1}{K} \tag{13}$$

□

2.3 Sampling

In this paper, we propose to use DGUMs as surrogates of classical MRFs. To sample a K class balanced isotropic DGUM, the procedure is the following:

- Sample $K-1$ independent realizations of centered GMRFs sharing the same covariance matrix $\mathbf{\Xi}$, yielding $\mathbf{Z} = \mathbf{z}$.
- Compute $\mathbf{X} = \mathbf{x}$ through (9).

Thus, the computational complexity of DGUM sampling is directly that of the GMRF sampling. Because the latter is performed directly (*i.e.*, no iterative sampling), there is no statistical convergence issue, unlike the Gibbs sampling approach. As mentioned in Section 1.2.2, we consider two approaches for sampling, that are appealing because of their tractability.

Fourier sampling. Fourier sampling relies on two 2D DFTs over n sites and the sampling of n independent Gaussian random variables (see Appendix A.2). This yields a computational complexity of

$$\mathcal{O}\Big(n + 2(K-1)n\log(n)\Big) = \mathcal{O}\Big((K-1)n\log(n)\Big).$$

Spectral sampling. This approach (Allard et al. 2020) is based on the summation of p cosine waves, each depending on n random variables sampled from a Gaussian variable of inverse-gamma-sampled variance (see Appendix A.2). The value of p should not be neglected, as it should be large enough to ensure an approximation based on the central limit theorem ($p = 5.10^3$ in (Allard et al. 2020)). As the sampling along a Gamma distribution is only slightly slower than sampling a normal distribution (Marsaglia & Tsang 2000), we consider its complexity as equivalent. Thus, spectral-based sampling of DGUM has a complexity of

$$\mathcal{O}((K-1)np).$$

3 Numerical results

This section presents numerical experiments and results regarding the sampling of DGUM fields, with comparison to Potts MRF (3) sampling.

Experimental setup. We evaluate the DGUM sampling using a Matérn covariance function C, defined between any sites $s, s' \in \mathcal{S}$ as:

$$C(s-s') = \frac{\sigma^2}{2^{\nu-1}\Gamma(\nu)}(\kappa\|s-s'\|)^\nu K_\nu(\kappa\|s-s'\|), \tag{14}$$

with K_ν the modified Bessel function of the second kind. We set $\sigma = 1$, $\kappa = 0.1$, and $\nu = 1$ in the numerical experiments.

As there is no formal equivalence between a Potts MRF and a DGUM, it is not possible to determine an equivalent value for β in the Potts MRF (3). Nevertheless, we found that $\beta = 0.5$ yields similar images appearance. Potts MRF were sampled using the *chromatic* Gibbs sampler (Gonzalez et al. 2011), which parallelizes sampling on mutually independent sets of sites in $\mathcal{S}$. The chromatic Gibbs sampler convergence is assessed numerically, and the sampler is stopped when less than 5% of pixel classes change between pixel classes of the last image sample and the most frequent pixel class from the previous 10 iterations.

Implementation. We implemented the DGUM and Potts MRF models and sampling techniques in JAX (Bradbury et al. 2024), which is one of the most efficient scientific computing libraries available nowadays. For improved performance, our code makes use of Just-In-Time (JIT) compilation, vectorization and is easily executable on either a GPU or a CPU. The code to reproduce the experiments will be made available at https://github.com/HGangloff/mrfx.

We expect timings to be highly favorable to the GPU in the spectral sampling and the Gibbs chromatic cases because, respectively, the gamma sampling and the site-wise updates on graph colorings are very well parallelized on a GPU.

In general, we also expect better JAX performances on a GPU because JIT compilation is optimized for such devices. However, those internal optimizations are in a black-box from the user viewpoint, preventing a fine-grained interpretation of the improvements.

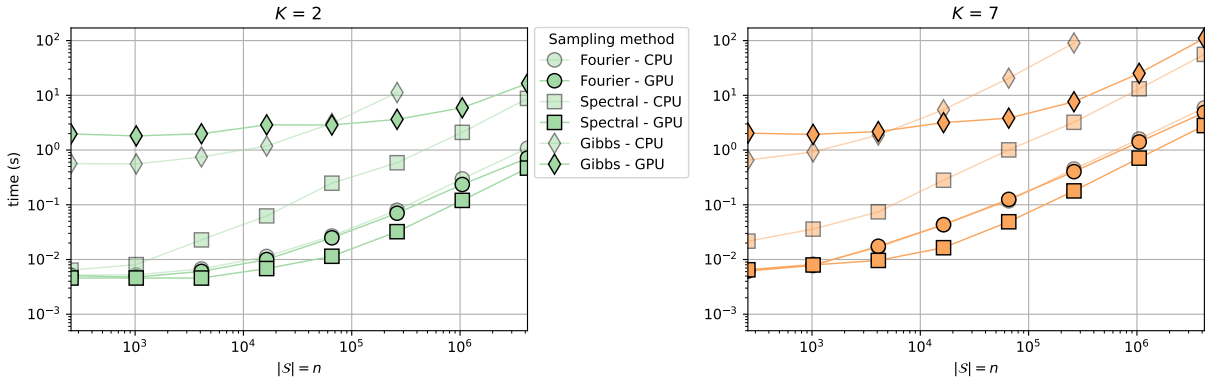


Figure 6: Computation times for $K = 2$ and $K = 7$ classes MRF sampling, as function of the number of sites n. The times reported here use a NVIDIA T600 Laptop GPU and are averaged over 100 repetitions. Additional results, for other values of K, are reported in Appendix B.1. Due to their time cost, the results regarding Gibbs sampling on CPU are not reported for the largest images ($n = 1024^2$ and 2048^2).

3.1 Computational cost

Time complexity. At first, we evaluate the sampling time complexity as a function of the number of sites n and classes K. Results are reported in Fig. 6 and suggest the following observations:

- DGUM sampling is by far faster than Gibbs sampling, even in the case of the chromatic version.
- Moving from CPU to GPU mainly benefits to the spectral sampling method, as Fourier sampling benefits from already well-implemented routines on CPU.
- Fourier sampling yields the best performances in the CPU implementation, as it benefits from already-implemented parallelized routines. On the other hand, the spectral DGUM sampling benefits the most from the GPU version.

Thus, the fastest MRF sampling technique is, for most cases, the spectral sampling computed on GPU.

Energy consumption. We also report the energy consumption for the sampling of an image, with the same varying parameters as in the previous section. We perform the measurement using the Python library Zeus (You et al. 2023) which provides easy-to-use energy measurement of CPUs and GPUs in PyTorch and JAX by wrapping the low-level Nvidia Management Library. The results are reported in Fig. 7. Overall, the observed energy consumption follows expectations, as it increases with both n and K. Besides, GPU-based methods are for almost every n and K more energy-efficient than their CPU counterpart.

Overall, the DGUM enables a drastic gain both in computation time and consumption, as shown in Fig. 8a: for any image size n, DGUM are sampled at least 35x faster and using at least 37x less energy than the best Gibbs alternative, which is the key finding of this study. For large images, a tradeoff remains to be made between sampling speed (spectral sampling, on GPU) and energy efficiency (Fourier sampling, on GPU), as is depicted in Fig. 8b for two fixed image sizes.

Remark 4. *While analyzing the results, the reader should bear in mind that the Zeus energy measurement might not be totally accurate (in absolute value) and that computing power is not the only source of energy consumption, resource utilization, waste, etc. (Kaack et al. 2022). Another critical aspect should be noted in the comparison between GPUs and CPUs. While a GPU implementation of CPU code can indeed induce both faster executions and reduced energy consumption at runtime, this should not hide the environmental footprint of manufacturing GPUs (Morand et al. 2025). The authors then want to argue that investing in GPUs must remain a careful choice.*

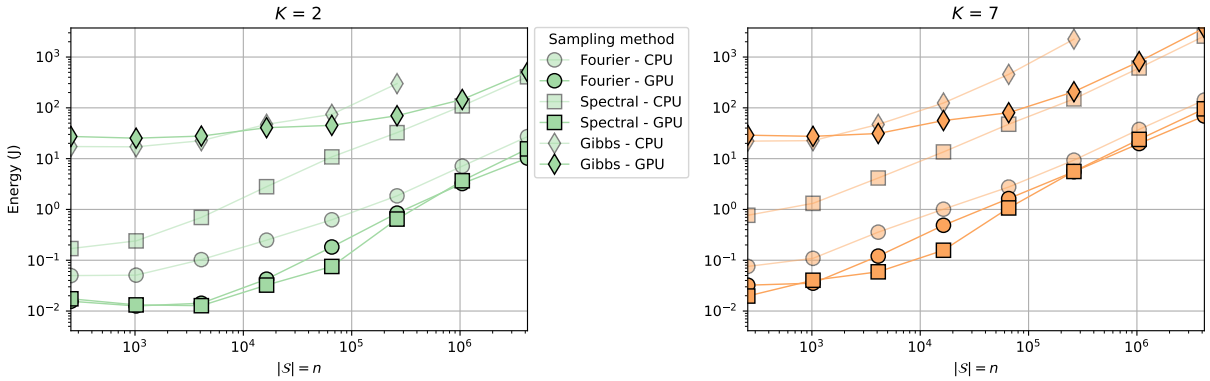


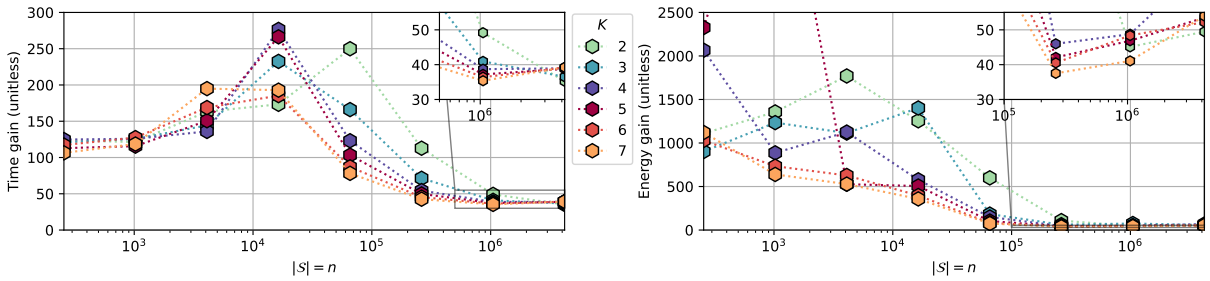
Figure 7: Estimated energy consumption of the considered sampling methods, for $K = 2$ and $K = 7$, as a function of the number of sites n. Additional results, for other values of K, are reported in Appendix B.1.

3.2 Statistical properties

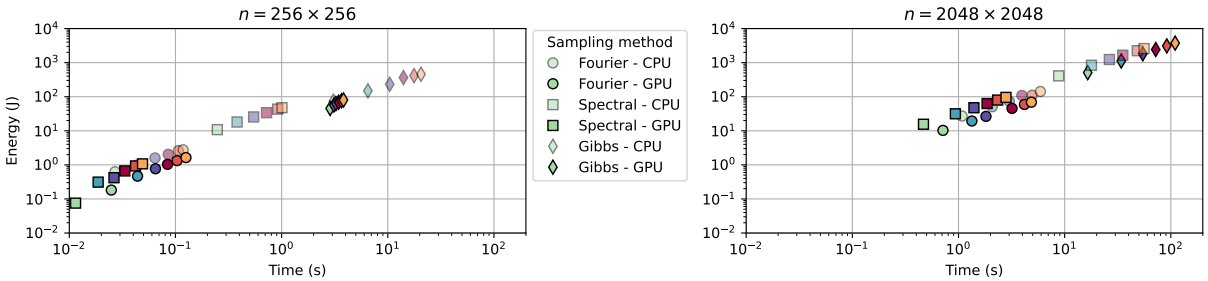
This section reports statistical measures obtained from sampled DGUM fields in comparison with Potts MRF sampled through chromatic Gibbs sampling. We focus on estimations of class proportion, pairwise similarity function, and additional observations regarding a phase transition behavior are included in Appendix B.2.

First order statistic. Focusing on class proportion, we have shown in Proposition 3 that sampling isotropic, balanced GUMS yields class balance. Table 1 reports the statistics of the class proportion estimator, showing the adequation of numerical implementation to the expected $1/K$ ratio. Besides, the standard deviation of the estimator (unknown theoretical value) is also reported. This table shows thus that, in terms of class balance estimator, the DGUM sampling behaves similarly to the chromatic Gibbs sampler of Potts MRFs.

Second order statistics. Figure 9 provides an insight into the pairwise similarity between sites of the realization as a function of their distance d on $\mathcal{S}$, namely $p\Big(x_i = x_j \big| \|i-j\|_2 = d\Big)$.



(a) Improvements of time (left) and energy consumption (right), depicted as a ratio between the measures from the best Gibbs sampler vs. the best DGUM sampler.



(b) Tradeoff between energy consumption and computational speed for the sampling of medium-sized (256^2 pixels, left) and large (2048^2 pixels, right) images. The colors depict the same classes as in (a).

Figure 8: Synthesis of the numerical results in terms of improvements (a) and time-energy tradeoff (b).

This function is expected to decrease towards $1/K$: indeed, large distances will yield mutually independent realizations $\mathbf{z}_s$ and x_s. The figure exhibits that the overall behavior of a Potts MRF is similar to that of a DGUM, regardless of the sampling method. Remaining differences can be attributed to different constructions of neighborhoods, which spans only 8 sites in Potts MRFs but is larger (depending on the covariance function) with DGUMs.

Wrapping up the results of this section, we can summarize our main numerical results as:

1. the DGUM sampling outperforms traditional Gibbs sampling in terms of both speed and energy consumption by several orders of magnitude, as highlighted in Fig. 8a.
2. the resulting fields still share the main numerical properties expected from an Ising / Potts MRF, namely, class balance and pairwise similarity between sites.

4 Discussion and conclusion

This paper allowed us to introduce the DGUM random fields, that makes use of the sampling techniques from GMRFs to sample discrete MRFs. We exhibited some of the DGUM properties, in particular the fact that DGUMs are also Markov random fields. The numerical results confirmed the computational gain of doing so, while the overall structure of a classical MRF is preserved in DGUM samples. Several perspectives stem from this

	K	2	3	4	5	6	7
$\lvert\hat{f}_0 - f_0\rvert$	DGUM / Fourier	0.0076	0.0055	0.0131	0.0095	0.0043	0.0041
	DGUM / Spectral	0.0036	0.0037	0.0034	0.0005	0.0067	0.0029
	Gibbs sampling	0.0020	0.0137	0.0020	0.0031	0.0053	0.0010
$\mathrm{std}(\hat{f}_0)$	DGUM / Fourier	0.0650	0.0497	0.0533	0.0375	0.0454	0.0490
	DGUM / Spectral	0.0935	0.0780	0.0711	0.0481	0.0541	0.0541
	Gibbs sampling	0.0399	0.0581	0.0505	0.0504	0.0482	0.0490

Table 1: Class balance numerical study. Focusing on the estimator $\hat{f}_0 = \hat{p}(x_s = \omega_0)$, the table depicts the bias $|\hat{f}_0 - f_0| = |\hat{f}_0 - \frac{1}{K}|$ and standard deviation of the frequency estimator.). We use a Matérn covariance with $\kappa = 0.1$ for the DGUM sampling, and $\beta = 1.0$ for Gibbs sampling. Expectations are estimated over 50 repetitions on images of size 150×150.

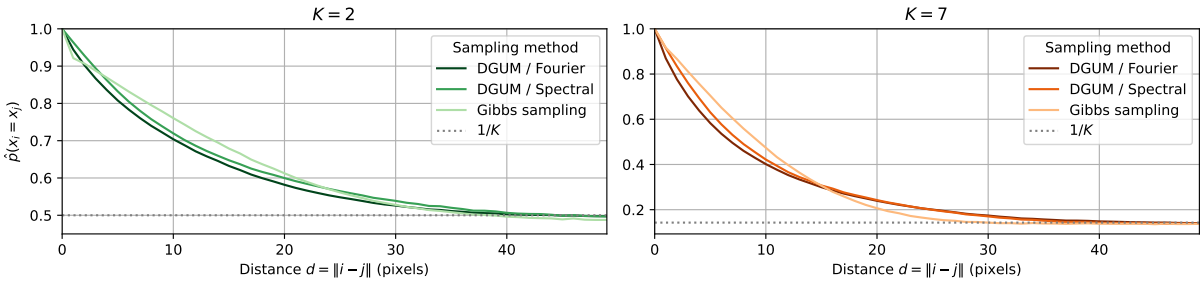


Figure 9: Empirical MRF / DGUM comparison setting $\kappa = 0.1$. The Potts MRF was sampled using $\beta = 0.5$ for the $K = 2$ case (left) and $\beta = 1$ for $K = 7$ (right).

work, and we discuss two avenues for future work.

Inference. Gibbs sampling is also a bottleneck in inverse problems with latent MRF processes. However, the mapping $\phi_{K,c}$ and its limit when $c \to 0$, noted ϕ_K, are not surjective. Thus, going back from $\mathbf{X} = \mathbf{x}$ to $\mathbf{Z} = \mathbf{z}$ is not directly feasible, which hinders the use of $\mathbf{x} = \phi_K(\mathbf{z})$ as a latent variable in an inverse problem. Indeed, the posterior in such a model (*e.g.* with an observation $\mathbf{Y} = \mathbf{y}$ resulting from a noisy measurement of $\mathbf{x}$) cannot be computed directly for a given $\mathbf{x}$ but only for a given $\mathbf{z}$; and as $\mathbf{z} \in \mathbb{R}^{N(K-1)}$ this makes the problem difficult to invert. Possible solutions involve the use of approximate sampling techniques, such as variational inference, whose computational cost might counteract the benefit of the DGUM sampling.

Beyond the grid. The processes presented here were assumed to lie on a regularly indexed 2D grid $\mathcal{S}$, but this is not a necessary condition for DGUM sampling: indeed, one could easily extend this work to handle 3D lattices or graphs. We could also consider the process to lie beyond the grid, as GMRFs are commonly defined outside grids to depict real-valued spatial data. Thus, DGUMs could be sampled at an arbitrary precision, yielding leads for potential use in super-resolution problems.

A Sampling algorithms

For the sake of completeness, this appendix gathers the sampling algorithms considered in this paper, using our notations. All these algorithms are implemented and available at https://github.com/HGangloff/mrfx.

A.1 Sampling Markov Random Fields

The main sampler for Markov random fields is the Gibbs sampler (Geman & Geman 1984) introduced by Geman and Geman in 1984. It is reported in Alg. 1.

An important improvement was provided in the *chromatic* Gibbs sampler proposed in (Gonzalez et al. 2011). The main idea is that mutually independent sites that can be sampled simultaneously, enabling in practice array-wise sampling and even parallelize part of the loop. Note the similarity with the *conclique* based Gibbs sampling (Kaplan et al. 2020), which has a more general perspective regarding the support of the random field on a graph. As their principle is essentially identical, we report only the chromatic Gibbs sampler in Alg. 2.

Algorithm 1 Gibbs Sampler (Geman & Geman 1984)

Require: Distribution $p(x_s|\mathbf{x}_{N_s})$ and its parameters, set of sites $\mathcal{S}$
Ensure: Sequence $\{\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \dots, \mathbf{x}^{(P)}\}$
  Choose an initial value $\mathbf{x}^{(0)}$.
  while a convergence criterion is not reached – iteration p **do**
    Set $\mathbf{x}^{(p)}$ to $\mathbf{x}^{(p-1)}$.
    for each site $s \in \mathcal{S}$ **do** ▷ this loop can not be parallelized
      Sample $x_s^{(p)}$ from $p(x_s|\mathbf{x}_{N_s}^{(p)})$.
    end for
  end while

Algorithm 2 Chromatic Gibbs Sampler (Gonzalez et al. 2011)

Require: Distribution $p(x_s|\mathbf{x}_{N_s})$ and its parameters, and subdivisions of $\mathcal{S}$ into mutually independent $\mathcal{S}_0, \mathcal{S}_1, ..., \mathcal{S}_j$ with respect to the chosen neighborhood.
Ensure: Sequence $\{\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \dots, \mathbf{x}^{(P)}\}$
  Choose an initial value $\mathbf{x}^{(0)}$.
  while a convergence criterion is not reached – iteration p **do**
    Set $\mathbf{x}^{(p)}$ to $\mathbf{x}^{(p-1)}$.
    for each set $\mathcal{S}_i$ **do** ▷ this loop can not be parallelized.
      for each site $s \in \mathcal{S}_i$ **do** ▷ this loop can be parallelized
        Sample $x_s^{(p)}$ from $p(x_s|\mathbf{x}_{N_s}^{(p)})$.
      end for
    end for
  end while

A.2 Sampling Gaussian Markov Random Fields

We detail here two samplers for GMRFs: Fourier sampling (Rue & Held 2005) and spectral sampling (Allard et al. 2020). Fourier sampling relies on the hypothesis that the covariance matrix is circulant, thus assuming that the grid $\mathcal{S}$ is a torus, *i.e.*, that the top/bottom and left/right borders are joint. Denoting $\mathbf{b}_{\mathbf{\Sigma}}$ the basis of a covariance matrix $\mathbf{\Sigma}$ and $\mathbf{b}_{\mathbf{Q}}$ the basis of the related precision matrix $\mathbf{Q} = \mathbf{\Sigma}^{-1}$, we have the following properties:

- $\mathbf{b}_{\mathbf{Q}} = \mathrm{IDFT}(1 \oslash \mathrm{DFT}(\mathbf{b}_{\mathbf{\Sigma}}))$, with DFT and IDFT the discrete Fourier transform and its inverse, and $\oslash$ denoting element-wise division.
- $\forall \mathbf{z} \in \mathbb{R}^n$, $\mathbf{Q}\mathbf{z} = \mathrm{IDFT}(\mathrm{DFT}(\mathbf{b}_{\mathbf{Q}}) \odot \mathrm{DFT}(\mathbf{z})) = \mathbf{z} * \mathbf{b}_{\mathbf{Q}}$, with $\odot$ the element-wise product and $*$ the convolution operator.

Thus, computations involving $\mathbf{\Sigma}$ or $\mathbf{Q}$ do not need to involve these matrices fully, but only through their bases $\mathbf{b}_{\mathbf{\Sigma}}$ and $\mathbf{b}_{\mathbf{Q}}$. Then, Fourier sampling builds upon these properties, using a sampling of a complex i.i.d. Gaussian vector and the computation of the eigenvalues of $\mathbf{b}_{\mathbf{Q}}$. The procedure is reported in Alg. 3.

Algorithm 3 Fourier sampling of GMRF (Rue & Held 2005).

Require: Circulant precision matrix $\mathbf{Q}$ and its basis $\mathbf{b}_{\mathbf{Q}}$.
Ensure: GMRF sample with zero-mean and precision $\mathbf{Q}$.
Sample $\mathbf{u} \in \mathbb{C}^n$ such that $\mathrm{Re}(\mathbf{u}) \sim \mathcal{N}(\mathbf{0}_n, \mathbf{I}_n)$ and $\mathrm{Im}(\mathbf{u}) \sim \mathcal{N}(\mathbf{0}_n, \mathbf{I}_n)$
Compute $\mathbf{L} = \sqrt{n}\,\mathrm{DFT}(\mathbf{b}_{\mathbf{Q}}))$
Return $\mathbf{z} = \mathrm{Re}\Big((\mathbf{L} \wedge (-\frac{1}{2})) \odot \mathbf{u}\Big)$, where $\wedge$ is a pointwise power operator.

We also report the more recent spectral sampling procedure (Allard et al. 2020). We use it together with a covariance matrix $\mathbf{\Sigma}$ designed from a Matérn correlation function (14). It relies on the idea that summing over a large enough number of well-chosen cosine waves, denoted *bands*, one reaches the conditions of the central limit theorem and thus approximates a Gaussian distribution with the desired covariance. The algorithm is reported, using the notations of this paper, in Alg. 4.

Algorithm 4 Spectral sampling using a Matérn correlation function (Allard et al. 2020)

Require: number of bands p, Matérn parameters ν and κ
Ensure: Sample $\mathbf{z}$, a zero-mean GMRF with covariance function $\mathbf{\Sigma}$ following a Matérn correlation function (14).
for $i = 1$ to p **do**
  Sample $g_i \sim \mathcal{G}(\nu, \kappa^2/2)$;
  Set $\xi_i = 1/(2g_i)$;
  Sample $\eta_i \sim \sqrt{2\xi_i}\mathcal{N}(0, \mathbf{I}_n)$;
  Sample $u_i \sim \mathcal{U}(0, 2\pi)$;
end for
for each $s \in \mathcal{S}$ **do**
  compute $z_s = \sqrt{\frac{2}{p}} \sum_{i=1}^{p} \cos(\eta_i^\top s + u_i)$
end for

B Additional numerical results

We report here additional results complimenting those presented in Section 3, regarding time and energy consumption, as well as phase transition observations.

B.1 Time and energy consumption

For the sake of brevity, we depicted the results on sampling time and energy consumption for $K = 2$ and $K = 7$ classes in section 3.1. We provide here additional results for $2 < K < 7$, depicted in Fig. 10. Overall, the trend is similar along the different values of K and image size n: the best results for time and energy consumption are obtained using Spectral sampling on GPU, then Fourier sampling on GPU, then Fourier sampling on CPU.

B.2 Observations of a phase transition

Depending on their β parameter, Ising and Potts MRF exhibit a well known phase transition effect that separates low values of β, yielding a noisy field, and higher values, exhibiting a constant behavior, with patch-like patterns in between. A relevant metric indicating phase transition is the proportion of sites identical to their neighbors, and we measure this quantity for two kinds of random fields.

1. By construction, the GUM random vectors are built upon the quantity $\pi_k^c(\mathbf{z})$, which measures the distance between any point in $\mathbb{R}^{K-1}$ to the $k-$th vertex of the unit simplex. As $c \to 0$, the $\pi_k^c(\mathbf{z})$ will take values close to either 0 or 1; thus, a field $\tilde{\mathbf{x}}$ sampled from $\pi_k^c(\mathbf{z})$ is expected to follow a phase transition depending on c.
2. More directly, the DGUMs are driven by the covariance structure of the GMRF $\mathbf{Z}$ that is sampled beforehand. Thus, we also study the behavior of DGUMs obtained through $\mathbf{Z} = \mathbf{z}$ realizations using a Matérn correlation function (14) within the covariance matrix, while varying κ values.

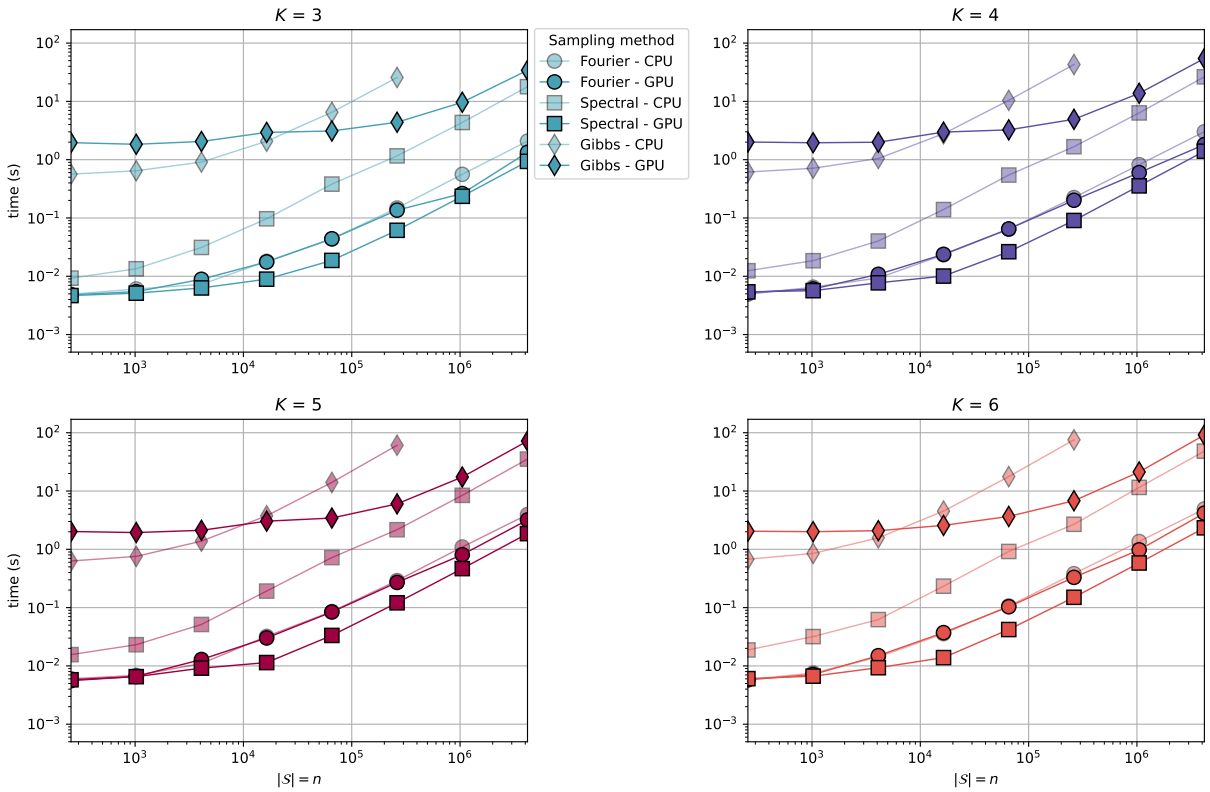
Figure 11 depicts the results of the phase transition study. We can observe, on the π_k^c, that c increases the randomness of the resulting sampled field, while a stationary behavior happens for $c < 0.5$. Regarding the phase transition of the DGUM, we note that a large κ yields an i.i.d. random fields (with $1/K$ probability of reaching each class), while $\kappa < 0.1$ yields a patch-like behavior.

References

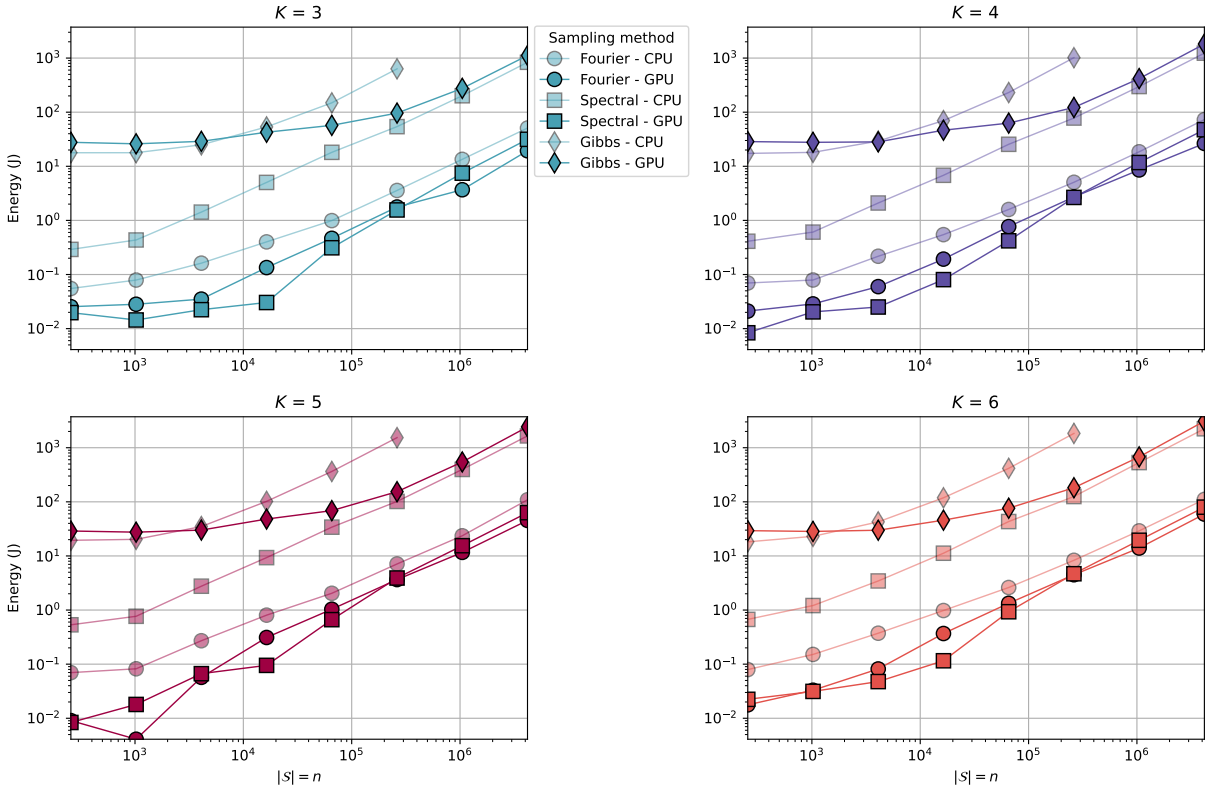
Allard, D., Emery, X., Lacaux, C. & Lantuéjoul, C. (2020), 'Simulating space-time random fields with nonseparable Gneiting-type covariance functions', *Statistics and Computing* **30**(5), 1479–1495.

Anderson, G. & Thron, C. (2022), 'Coordinate permutation-invariant unit N-simplexes in N dimensions', *International Journal of Pure and Applied Mathematics Research* (2), 1–14.

Bradbury, J., Frostig, R., Hawkins, P., Johnson, M. J., Leary, C., Maclaurin, D., Necula, G., Paszke, A., VanderPlas, J., Wanderman-Milne, S. & Zhang, Q. (2024), 'JAX: composable

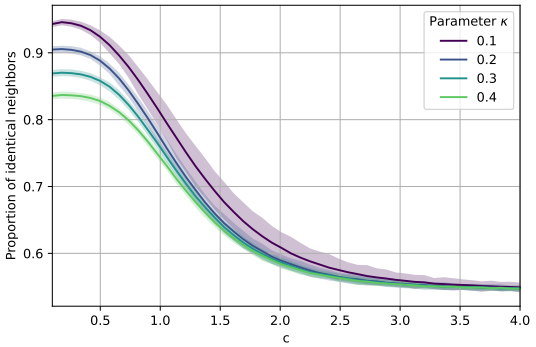


(a) Computation times.

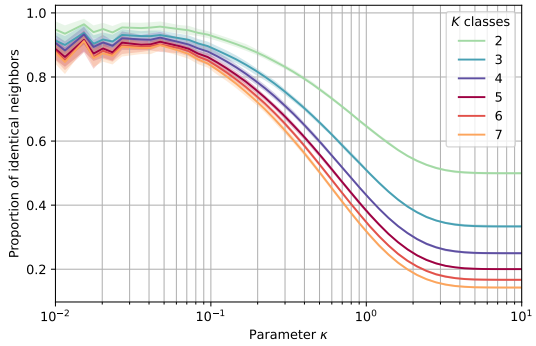


(b) Estimated energy consumption.

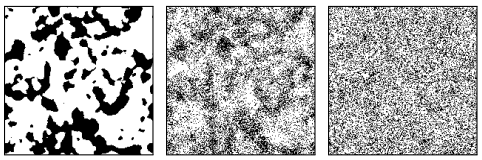
Figure 10: Computational burden (in time, and energy) extending the results presented in Figs. 6 and 7 to $3 \leq K \leq 6$ classes.



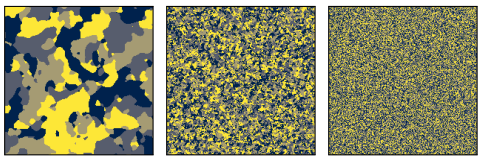
(a) Phase transition when sampling a binary field from π^c_k.



(b) Phase transition when sampling a DGUM, depending on the parameter κ and the number of classes K.



(c) $c = 0.1$ (d) $c = 2$ (e) $c = 4$



(f) $\kappa = 0.1$ (g) $\kappa = 1$ (h) $\kappa = 10$

Figure 11: Phase transition phenomena observed within the DGUM sampling process. (a) and (b) depict the overall behavior, with plain lines averaging over 50 samples of size 150×150, and shaded regions corresponding to the 1^{st} and 9^{th} decile. (c)-(e) and (f)-(h) provide examples of realization corresponding to (a) and (b) respectively for a few parameters.

transformations of Python+NumPy programs'.
URL: *http://github.com/google/jax*

Clifford, P. & Hammersley, J. (1971), 'Markov fields on finite graphs and lattices'.

Courbot, J.-B. & Gangloff, H. (2025), Gaussian unit-simplex Markov random fields as a fast proxy for MRF sampling, *in* '2025 IEEE Statistical Signal Processing Workshop (SSP)', pp. 166–170.

Frigessi, A., Martinelli, F. & Stander, J. (1997), 'Computational complexity of Markov chain Monte Carlo methods for finite Markov random fields', *Biometrika* **84**(1), 1–18.

Geman, S. & Geman, D. (1984), 'Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images', *IEEE Transactions on pattern analysis and machine intelligence* (6), 721–741.

Gheissari, R. & Lubetzky, E. (2016), 'Mixing times of critical 2D Potts models', *arXiv preprint arXiv:1607.02182* .

Giovannelli, J.-F. (2007), 'Unsupervised Bayesian convex deconvolution based on a field with an explicit partition function', *IEEE Transactions on Image Processing* **17**(1), 16–26.

Gonzalez, J., Low, Y., Gretton, A. & Guestrin, C. (2011), Parallel Gibbs sampling: From colored fields to thin junction trees, *in* 'Proceedings of the Fourteenth International

Conference on Artificial Intelligence and Statistics', JMLR Workshop and Conference Proceedings, pp. 324–332.

Grathwohl, W., Swersky, K., Hashemi, M., Duvenaud, D. & Maddison, C. (2021), Oops i took a gradient: Scalable sampling for discrete distributions, *in* 'International Conference on Machine Learning', PMLR, pp. 3831–3841.

Jang, E., Gu, S. & Poole, B. (2017), Categorical reparametrization with gumble-softmax, *in* 'International Conference on Learning Representations (ICLR 2017)', OpenReview. net.

Kaack, L. H., Donti, P. L., Strubell, E., Kamiya, G., Creutzig, F. & Rolnick, D. (2022), 'Aligning artificial intelligence with climate change mitigation', *Nature Climate Change* **12**(6), 518–527.

Kallenberg, O. (1997), *Foundations of modern probability*, Vol. 2, Springer.

Kaplan, A., Kaiser, M. S., Lahiri, S. N. & Nordman, D. J. (2020), 'Simulating Markov random fields with a conclique-based Gibbs sampler', *Journal of Computational and Graphical Statistics* **29**(2), 286–296.

Laferté, J.-M., Pérez, P. & Heitz, F. (2000), 'Discrete Markov image modeling and inference on the quadtree', *IEEE Transactions on image processing* **9**(3), 390–404.

Maddison, C. J., Mnih, A. & Teh, Y. W. (2017), The concrete distribution: A continuous relaxation of discrete random variables, *in* 'International Conference on Learning Representations'.

Margossian, C. C. & Mukherjee, S. (2021), 'Simulating Ising and Potts models at critical and cold temperatures using auxiliary gaussian variables', *arXiv preprint arXiv:2110.10801* .

Marsaglia, G. & Tsang, W. W. (2000), 'A simple method for generating gamma variables', *ACM Transactions on Mathematical Software (TOMS)* **26**(3), 363–372.

Martens, J. & Sutskever, I. (2010), Parallelizable sampling of Markov random fields, *in* 'Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics', JMLR Workshop and Conference Proceedings, pp. 517–524.

Morand, C., Ligozat, A.-L. & Névéol, A. (2025), Does Efficiency Lead to Green Machine Learning Model Training? Analyzing Historical Trends in Impacts from Hardware, Algorithmic and Carbon Optimizations. working paper or preprint.
URL: *https://hal.science/hal-04839926*

Mossel, E. & Sly, A. (2013), 'Exact thresholds for Ising–Gibbs samplers on general graphs', *The Annals of Probability* **41**(1), 294 – 328.
URL: *https://doi.org/10.1214/11-AOP737*

Rozanov, Y. A. (1982), *Markov random fields*, Springer.

Rue, H. & Held, L. (2005), *Gaussian Markov random fields: theory and applications*, Chapman and Hall/CRC.

You, J., Chung, J.-W. & Chowdhury, M. (2023), Zeus: Understanding and optimizing GPU energy consumption of DNN training, *in* 'USENIX NSDI'.