Machine Learning: Science and Technology

BENCHMARK • OPEN ACCESS

Machine-learning-assisted Monte Carlo fails at sampling computationally hard problems

MACHINE
LEARNING
Science and Technology

OPEN ACCESS

RECEIVED
8 November 2022

REVISED
30 January 2023

ACCEPTED FOR PUBLICATION
23 February 2023

PUBLISHED
14 March 2023

BENCHMARK

Machine-learning-assisted Monte Carlo fails at sampling computationally hard problems

Simone Ciarella[1,3,*], **Jeanne Trinquier**[1,2,3,*], **Martin Weigt**[2] **and Francesco Zamponi**[1]

[1] Laboratoire de Physique de l'Ecole Normale Supérieure, ENS, Université PSL, CNRS, Sorbonne Université, Université de Paris, F-75005 Paris, France
[2] Sorbonne Université, CNRS, Institut de Biologie Paris Seine, Biologie Computationnelle et Quantitative LCQB, F-75005 Paris, France
[3] These authors contributed equally.
[*] Authors to whom any correspondence should be addressed.

E-mail: simone.ciarella@ens.fr and jeanne.trinquier@ens.fr

Keywords: statistical physics, Monte Carlo Markov Chains, sampling

Supplementary material for this article is available online

Abstract

Several strategies have been recently proposed in order to improve Monte Carlo sampling efficiency using machine learning tools. Here, we challenge these methods by considering a class of problems that are known to be exponentially hard to sample using conventional local Monte Carlo at low enough temperatures. In particular, we study the antiferromagnetic Potts model on a random graph, which reduces to the coloring of random graphs at zero temperature. We test several machine-learning-assisted Monte Carlo approaches, and we find that they all fail. Our work thus provides good benchmarks for future proposals for smart sampling algorithms.

1. Introduction

1.1. Motivations

Sampling from a given target probability distribution $P_t(\sigma_1,\ldots,\sigma_N)$ over N degrees of freedom can become extremely hard when N is large. A universal (i.e. system-independent) strategy for sampling consists in starting from a random configuration of $\sigma = \{\sigma_i\}_{i=1,\ldots,N}$, and generating a local Monte Carlo Markov Chain (MCMC), by sequentially proposing an update of one of the σ_i, and accepting or rejecting it with a proper probability (e.g. Metropolis–Hastings), until convergence [1]. However, for large N, the convergence time of the MCMC can grow exponentially in N, because of non-trivial long-range correlations that make local decorrelation extremely hard [2].

A solution to this problem consists in identifying the proper set of correlated variables, and proposing *global* updates of such variables together, in such a way to speed up convergence [3]. However, this process is not universal, because it relies on the proper identification of system-dependent correlations, which is not always possible. For instance, in disordered systems such as spin glasses, the nature of correlated domains is extremely elusive and proper global moves are not easy to identify [4, 5]. Another approach, which has been particularly successful in atomistic models of glasses, consists in unconstraining some degrees of freedom, evolve them and constrain them back [6–10], but again it is model-specific. Alternative proposals based on a renormalization group approach [11, 12] also rely on the identification of system-dependent collective variables.

A recently developed line of research, see e.g. [13–20], proposed to solve the problem in an elegant and universal way, by machine learning proper MCMC moves. In a nutshell, the idea is to learn an auxiliary probability distribution $P_a(\sigma)$, which (i) can be sampled efficiently (e.g. linearly in N) and (ii) provides a good approximation of the target probability. Then, the hope is to use the auxiliary distribution to propose smart MCMC moves. Using this strategy with autoregressive (AR) architectures that ensure efficient sampling, some authors found convergence speedup [13, 14, 16], but others found less promising results [20].

In order to make these studies more systematic, and really assess the performance of the method, it is important to have good benchmarks, i.e. problems that are guaranteed to be really hard to sample by local MCMC. In the early 90s, the very same problem had to be faced to assess the performance of local search algorithms that looked for solution of optimization or satisfiability problems [21]. In that case, the problem of generating good benchmarks was solved by introducing an ensemble of random instances of the problem under study [21–24]. It was later shown, both numerically and analytically, that these random optimization/satisfiability problems require a time scaling exponentially in N for proper sampling at low enough temperatures in certain regions of parameter space [2]. Hence, they provide very good benchmarks for sampling algorithms. Yet, the recent attempts to apply machine learning methods to speed-up sampling have not considered these benchmarks.

In this paper, we consider a prototypical hard-to-sample random problem, namely the coloring of random graphs, and we show that all the proposed methods fail to solve it. Our results confirm that this class of problems are a real challenge for sampling methods, even assisted by smart machine-learned moves. The model investigated in [20] possibly belong to this class. In addition, we discuss some practical issues such as mode-collapse in learning the auxiliary model, which happens when the target probability distribution has multiple peaks and the auxiliary model only learns one (or a subset) of them.

1.2. State of the art

Before proceeding, we provide a short review of the papers that motivated our study. Because the field is evolving rapidly, this does not aim at being an exhaustive review, and despite our best efforts, it is possible that we missed some relevant references.

Merchan and Nemenman [25] considered the general problem of whether a target probability distribution P_t can be approximated by a simpler one P_a, in particular by considering the Kullback–Leibler (KL) divergence:

$$D_{\mathrm{KL}}(P_t||P_a) = \left\langle \log \frac{P_t(\sigma)}{P_a(\sigma)} \right\rangle_{P_t} . \tag{1}$$

If this quantity is proportional to N for $N \to \infty$, then $P_t(\sigma)/P_a(\sigma)$ is typically exponential in N, and as a result samples proposed from P_a are very unlikely to be accepted in P_t. A small $D_{\mathrm{KL}}(P_t||P_a)/N$ (ideally vanishing for $N \to \infty$) seems therefore to be a necessary condition for a good auxiliary probability, which provides a quantitative measure of condition (ii) above. Merchan and Nemenma [25] suggested, by using small disordered systems ($N \sim 20$), that there might be a phase transition, for $N \to \infty$, separating a phase where $D_{\mathrm{KL}}(P_t||P_a)/N$ vanishes identically and a phase where it is positive.

Wu *et al* [13] proposed, more specifically, to use AR models as tractable architectures for P_a. In these architectures, P_a is represented using Bayes' rule,

$$P_a(\sigma) = P_a^1(\sigma_1) P_a^2(\sigma_2|\sigma_1) \dots P_a^N(\sigma_N|\sigma_{N-1}, \dots, \sigma_1). \tag{2}$$

Each term P_a^i is then approximated by a neural network, which takes as input $\{\sigma_1, \dots, \sigma_{i-1}\}$ and gives as output P_a^i, i.e. the probability of σ_i conditioned to the input. Such a representation of P_a, also called Masked Autoencoder for Distribution Estimator (MADE) [26], allows for very efficient sampling, because one can first sample σ_1, then σ_2 given σ_1, and so on, in a time scaling as the sum of the computational complexity of evaluating each of the P_a^i, which is typically polynomial in N for reasonable architectures. Hence, this scheme satisfies condition (i) above. The simplest choice for such a neural network is a linear layer followed by a softmax activation function. Wu *et al* [13] showed that using such an architecture, several statistical models could be well approximated, and the Boltzmann distribution of a Sherrington–Kirkpatrick (SK) spin glass model (with $N = 20$) could be efficiently sampled. Note that the model in [13] was trained by a variational procedure, which minimizes $D_{\mathrm{KL}}(P_a||P_t)$ instead of $D_{\mathrm{KL}}(P_t||P_a)$. This method is computationally very efficient as it only requires an average over P_a, which can be sampled easily, instead of P_t, but it is prone to mode-collapse (see section 2 for details). Moreover, this work was limited to quite small N.

Following up on [13], McNaughton *et al* [14] considered as target probability the Boltzmann distribution of a two-dimensional (2d) Edwards-Anderson (EA) spin glass model at various temperatures T, and used a Neural AR Distribution Estimator (NADE) [27], which is a variation of the MADE meant to reduce the number of parameters. Furthermore, the model was trained using a different scheme from Wu *et al* [13], called sequential tempering, which tries to minimize $D_{\mathrm{KL}}(P_t||P_a)$, thus preventing mode-collapse. To this aim, at first, a sample from P_t is generated at high temperature, which is easy, and used to learn P_a. Then, temperature is slightly reduced and smart MCMC sampling is performed using the P_a learned at the previous step, to generate a new sample from P_t, which is then used in the next step. If P_a remains a good approximation to P_t and MCMC sampling is efficient, this strategy ensures a correct minimization of

$D_{\mathrm{KL}}(P_t||P_a)$. This was shown to be the case in [14], down to low temperatures for a 2d EA model of up to $N = 225$ spins.

Gabrié *et al* [15] introduced a different scheme for learning P_a. This adaptive scheme combines local MCMC moves with smart P_a-assisted MCMC moves, together with an online training of P_a. It was successfully tested using a different architecture for P_a (called normalizing flows), on problems with two stable states separated by a high free energy barrier. Note that normalized flows can be equivalently interpreted as AR models [28–30]. Wu *et al* [16] also proved the effectiveness of smart assisted MCMC moves in a 2d Ising model and an Ising-like frustrated plaquette model.

Several other groups [17–20] investigated a problem related to sampling, namely that of simulated annealing [31] for finding ground states of optimization problems. This is an *a priori* slightly easier problem, because simulated annealing does not need to equilibrate at all temperatures to find a solution [32, 33]. In these works, simulated annealing moves were once again assisted by machine learning. Hibat-Allah *et al* [17] tested their procedure on the 2d EA and SK models, and Fan *et al* [18] considered a 2d, 3d, and 4d EA model. However, while finding the exact ground state of the SK and EA (for $d \geqslant 3$) models is hard, in practice for not too large random instances the problem can be solved by a proper implementation of standard simulated annealing [34], and the scaling of these methods with system size remains poorly investigated. Schuetz *et al* [19] considered the graph coloring problem, which is the zero-temperature version of the benchmark problem we propose to use in this work, and found that a graph neural network (GNN) can propose moves that allow one to efficiently find a proper coloring with comparable performances to (but not outperforming) state-of-the-art local search algorithm. Additionally, GNN have shown to be successful at solving discrete combinatorial problems [35], but they do not provide much advantage over classical greedy algorithms, and sometimes they can even show worse performance [36, 37]. Finally, Inack *et al* [20] showed that the machine-learning-assisted simulated annealing scheme does not work on a glassy problem with a rough energy landscape.

These works provided a series of inspiring ideas to improve sampling in disordered systems via machine learning smart MCMC moves. Yet, the question of whether machine learning can really speed up sampling in problems that are exponentially hard to sample via local MCMC remains open. This wide class of systems include many problems of interest, such as optimization problems (e.g. random satisfaction or random graph coloring) [38] and mean-field glass-forming materials [39–42].

1.3. Summary

In this work, we test machine-learning-assisted MCMC in what is considered to be a prototypical hard-to-sample model, namely the coloring of random graphs [21, 43, 44]. Before doing that, we also tested and reproduced previous results in simpler cases.

The models we consider are:

(1) The mean-field ferromagnetic problem, usually called the Curie-Weiss (CW) model, to gain some analytical insight into the different ways of training the auxiliary model.
(2) A 2d EA spin glass model. We consider this as an ‘easy’ problem (because, for instance, its ground state can be found in polynomial time), and we use it to reproduce previous results, compare different architectures, and gain insight on the role of some hyperparameters.
(3) The coloring (COL) of a random graph, which at finite temperature becomes an antiferromagnetic Potts model. In the proper range of parameters, this problem is proven to be exponentially hard to sample via local MCMC [2, 44], and we use it as a benchmark to understand whether smart MCMC can improve the sampling efficiency.

Any machine learning model that satisfies the AR property can be trained and used as an auxiliary distribution to propose smart moves. However, on the one hand, for complex problems, shallow or simple models might not be expressive enough to accurately learn the target distribution. On the other hand, if a problem can be easily solved by a simpler model, there is no need to employ complex deep architectures. In this paper we used several standard architectures illustrated in figure 1 and detailed in the SI:

- The MADE [13], which is an AR deep neural network; when its depth is equal to zero, this corresponds to a ‘shallow’ or single-layer AR model.
- The NADE [14, 27], which corresponds to a MADE with additional constraints on the parameters, with different depths and number of hidden units. This architecture has proven to be effective in image detection [45], filtering [46] and quantum systems [47].
- For the coloring, because neither the MADE nor the NADE perform well, we also tested an AR GNN that we called Graph AR Distribution Estimator (GADE), and a non-symmetric MADE (called ColoredMADE).

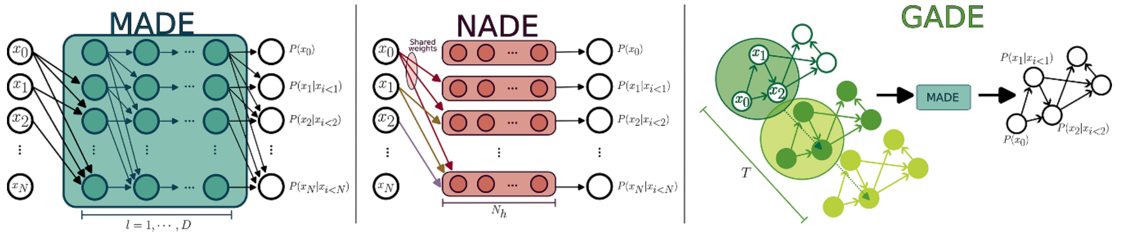

Figure 1. Sketches of the AR architectures used in this work.

Finally, we tried several strategies to learn the auxiliary model:

(I) *Maximum likelihood*: we generate a sample from the target distribution, and we use it to train the auxiliary model by maximum likelihood. While this is not a technique to generate samples from P_t, because one needs the samples to begin with, it is the best way to test if a given architecture for P_a is expressive enough.

(II) *Variational*: we minimize the KL divergence $D_{\mathrm{KL}}(P_a||P_t)$, which also corresponds to the variational free energy of the auxiliary model when considered as an approximation of the true one [13].

(III) *Sequential tempering*: we train the auxiliary model at higher temperature T, then use it to generate samples at lower T, and use the new samples to re-train the auxiliary model, and so on [14, 15].

The core of our work is the application of methods (II) and (III) to attempt a sampling at low temperatures, for which local MCMC does not decorrelate fast enough. For the 2d EA model, we find that basically all the techniques and architectures perform well down to very low temperatures, although (II) is more prone to mode-collapse. We confirm that machine learning MCMC moves can provide a speedup in this case [14]. For the COL problem, we find that none of these methods work well, even at moderately high temperatures located within the paramagnetic phase of the model.

2. Methods

We consider a specific application of the general scheme discussed in section 1, in which we want to approximate the Boltzmann distribution associated to the 'true' Hamiltonian $H(\sigma)$ at a fixed inverse temperature $\beta = 1/T$ (the target),

$$P_t(\sigma) = P_B(\sigma) = \frac{\mathrm{e}^{-\beta H(\sigma)}}{Z}, \tag{3}$$

with an AR network, i.e. $P_a(\sigma) = P_{\mathrm{AR}}(\sigma)$. In most cases, sampling is easy at small β and becomes harder and harder as β is increased. We now discuss different possible strategies to learn a proper $P_{\mathrm{AR}}(\sigma)$.

2.1. Maximum likelihood

If a sufficiently large sample of configurations $\{\sigma^m\}_{m=1,\dots,M}$ is available, independently and identically sampled from $P_B(\sigma)$, it is possible to train the model by maximizing the probability of the sample according to the AR model itself, i.e. to use the maximum likelihood method.

Assuming that the AR model is specified by a set of parameters θ, we maximize the likelihood of the observed data, defined as:

$$\mathcal{L}(\theta) = \prod_{m=1}^{M} P_{AR}(\sigma^m|\theta). \tag{4}$$

Equivalently, if $P_{\mathrm{emp}}(\sigma) = \frac{1}{M}\sum_{m=1}^{M}\delta_{\sigma,\sigma^m}$ is the empirical distribution of the sample, we minimize:

$$\begin{aligned}\mathcal{D}(\theta) &= D_{\mathrm{KL}}(P_{\mathrm{emp}}||P_{\mathrm{AR}}) \\ &= -\log M - \frac{1}{M}\sum_{m=1}^{M}\log P_{\mathrm{AR}}(\sigma^m|\theta).\end{aligned} \tag{5}$$

The optimal parameters $\hat{\theta}$ are given by:

$$\hat{\theta} = \mathrm{argmax}_\theta \mathcal{L}(\theta) = \mathrm{argmin}_\theta \mathcal{D}(\theta). \tag{6}$$

The gradient of $\mathcal{D}(\theta)$ can be computed analytically in terms of the σ^m, because $\log P_{\mathrm{AR}}(\sigma|\theta)$ is given explicitly (or by back-propagation) as a function of θ at fixed σ, and the training can thus be performed efficiently.

Note that if $P_B(\sigma)$ has multiple peaks, for sufficiently large M all these peaks are represented in the empirical distribution with the correct weights. Hence, when performing maximum likelihood to learn $P_{\mathrm{AR}}(\sigma)$, the learned model should be able to represent all the peaks of $P_B(\sigma)$, provided the AR model has enough free parameters, i.e. is expressive enough. Then, mode-collapse will not occur.

Obviously, the maximum-likelihood approach relies on the quality of the initial sample, which has to be representative of the true distribution. Such a sample is by definition difficult to obtain for the really hard sampling problems that we want to solve. Moreover, if we were able to obtain such a sample by conventional means, there would be no need for any smart MCMC scheme. Nevertheless, the maximum-likelihood approach constitutes a reliable and effective way to test if a specific AR architecture is capable of learning the complexity of a specific problem. We will thus use this scheme, in cases where standard sampling from $P_B(\sigma)$ is possible, to test the expressive quality of our AR architectures.

2.2. Variational approach

Wu *et al* [13] proposes to bypass the need for sampling from $P_B(\sigma)$ by using a variational approach. Instead of minimizing $D_{\mathrm{KL}}(P_B||P_{\mathrm{AR}})$ or its approximation $D_{\mathrm{KL}}(P_{\mathrm{emp}}||P_{\mathrm{AR}})$, as in section 2.1, we want here to minimize $D_{\mathrm{KL}}(P_{\mathrm{AR}}||P_B) = \sum_\sigma P_{\mathrm{AR}}(\sigma)\log\frac{P_{\mathrm{AR}}(\sigma)}{P_B(\sigma)}$, or equivalently [13] the variational free energy:

$$\begin{aligned}\beta F[P_{\mathrm{AR}}] &= \sum_\sigma P_{\mathrm{AR}}(\sigma)(\beta H(\sigma) + \log P_{\mathrm{AR}}(\sigma)) \\ &= \beta F[P_B] + D_{\mathrm{KL}}(P_{\mathrm{AR}}||P_B).\end{aligned} \tag{7}$$

As it is well known in statistical mechanics, because the KL divergence is positive, $F[P_{\mathrm{AR}}]$ is minimized when $P_{\mathrm{AR}} = P_B$ and $D_{\mathrm{KL}}(P_{\mathrm{AR}}||P_B) = 0$, and otherwise it provides an upper bound to the true free energy.

The gradient with respect to the parameters θ that define the AR model can be written as an expectation value over the AR model itself [13],

$$\begin{aligned}\beta\nabla_\theta F[P_{\mathrm{AR}}] &= \langle Q(\sigma)\nabla_\theta \log P_{\mathrm{AR}}(\sigma)\rangle_{P_{\mathrm{AR}}}, \\ Q(\sigma) &= \beta H(\sigma) + \log P_{\mathrm{AR}}(\sigma).\end{aligned} \tag{8}$$

The learning can then be done by gradient descent on $F[P_{\mathrm{AR}}]$, sampling from the AR model to estimate the gradient via equation (8). We used, as a condition to stop the gradient descent, that the variance of $Q(\sigma)$ over batches of generated data is smaller than a given threshold. Indeed, if the AR distribution is exactly the Boltzmann one, then $Q(\sigma)$ is a constant. Reciprocally, if the variance of $Q(\sigma)$ is zero, then the AR distribution is proportional to the Boltzmann distribution whenever $P_{\mathrm{AR}}(\sigma) > 0$, but not necessarily over all possible σ, due to mode-collapse. More specifically, if we have mode-collapse, then $P_{\mathrm{AR}}(\sigma) = P_B(\sigma)/\mathcal{Z}$ in some regions of the space of σ (typically around some of the peaks of P_B) and $P_{\mathrm{AR}}(\sigma) = 0$ elsewhere, where the proportionality constant $\mathcal{Z} < 1$ represents the total probability covered by $P_{\mathrm{AR}}(\sigma)$ in $P_B(\sigma)$. We then obtain $D_{\mathrm{KL}}(P_{\mathrm{AR}}||P_B) = -\log\mathcal{Z} > 0$, because the regions where $P_{\mathrm{AR}}(\sigma) = 0$ do not contribute to the sum. While this solution has larger KL divergence with respect to the optimal one ($P_{\mathrm{AR}} = P_B$, $D_{\mathrm{KL}}(P_{\mathrm{AR}}||P_B) = 0$), it can be a local minimum of $D_{\mathrm{KL}}(P_{\mathrm{AR}}||P_B)$ in which the variationally-trained AR model can be trapped, thus learning only a part of the landscape.

2.3. Local versus global MCMC

Standard local MCMC usually consists in selecting a variable at random and then proposing a random (or semi-random) change. If the MC moves respect microscopic detailed balance, the MCMC is guaranteed to converge to the correct Boltzmann distribution. This can be achieved, e.g. by accepting/rejecting the proposed MC moves following the Metropolis rule, where the acceptance probability of a move from configuration σ_{old} to σ_{new} is defined as:

$$\mathrm{Acc}\left[\sigma_{\mathrm{old}} \to \sigma_{\mathrm{new}}\right] = \min\left[1, \frac{P_B(\sigma_{\mathrm{new}})}{P_B(\sigma_{\mathrm{old}})}\right]. \tag{9}$$

We call this scheme local MCMC because each move consists in a change of a single degree of freedom.

In contrast, we can sample from our AR model $P_{\mathrm{AR}}(\sigma)$ to generate a new proposed configuration σ_{new}. It is still useful to respect microscopic detailed balance in order to ensure convergence to equilibrium, and for this reason the replacement $\sigma_{\mathrm{old}} \to \sigma_{\mathrm{new}}$ is accepted with probability:

$$\mathrm{Acc}\left[\sigma_{\mathrm{old}} \to \sigma_{\mathrm{new}}\right] = \min\left[1, \frac{P_B(\sigma_{\mathrm{new}}) \times P_{\mathrm{AR}}(\sigma_{\mathrm{old}})}{P_B(\sigma_{\mathrm{old}}) \times P_{\mathrm{AR}}(\sigma_{\mathrm{new}})}\right]. \tag{10}$$

Note that, because σ_{new} is generated from scratch by the AR model, it is in most cases completely different from σ_{old}, hence the resulting move is non-local and this is why this scheme is called global MCMC.

We also note that this global MCMC scheme is very similar to importance sampling, in which M i.i.d. samples σ^m are generated from $P_{AR}(\sigma)$, and then reweighted by $W(\sigma^m) = P_B(\sigma^m)/P_{\text{AR}}(\sigma^m)$ to compute averages. However, the formulation in terms of a MCMC is convenient to compare with local MCMC, to monitor efficiency via the acceptance rate (which is morally equivalent to a participation ratio in importance sampling), and to perform smart protocols (e.g. sequential tempering) during the MCMC dynamics [14, 15]. This is why we stick to this formulation in this paper.

The reweighting factor $W(\sigma) = P_B(\sigma)/P_{\text{AR}}(\sigma)$ that appears both in importance sampling and in equation (10) is the crucial quantity for the efficiency of the global MCMC scheme. If $W(\sigma)$ is typically exponential in N, then its fluctuations are too wild and moves are almost never accepted. The KL divergence is precisely the average of $\log W(\sigma)$, either on the Boltzmann or on the AR distribution, and if it is too large (in particular, growing linearly in N) the method is doomed to failure.

2.4. Sequential tempering

Sequential tempering is a technique used in [14] to learn the AR probability at a larger β using data from lower β, which can be convenient because collecting data becomes harder upon increasing β. The first step consists in collecting a sample via local MCMC at low β, where sampling is easy, and then training an AR model to reproduce this sample by maximum likelihood (section 2.1). Next, in order to create a new sample at $\beta + \delta\beta$, we use global MCMC by proposing moves with the previous AR model, at the new temperature. The acceptance rule then becomes:

$$\text{Acc}\left[\sigma_{\text{old}} \to \sigma_{\text{new}}\right] = \min\left[1, \frac{\mathrm{e}^{-(\beta+\delta\beta)H(\sigma_{\text{new}})} P_{\text{AR}}(\sigma_{\text{old}})}{\mathrm{e}^{-(\beta+\delta\beta)H(\sigma_{\text{old}})} P_{\text{AR}}(\sigma_{\text{new}})}\right]. \tag{11}$$

We then learn a new AR model from the new sample by maximum likelihood, and iterate until either we reach the temperature of interest, or the convergence time of the global MCMC exceeds some fixed threshold, indicating a failure of the training procedure. A related adaptive global MCMC scheme has been introduced in [15] and is detailed in the SI.

2.5. Evaluation of the AR model

Once the learning is completed, one can use several observables to evaluate the quality of the AR model. By completion of the learning we mean convergence of the gradient ascent in maximum likelihood (section 2.1), convergence of the gradient descent in the variational free energy (section 2.2), or reaching the target temperature with high enough acceptance rate in the sequential tempering (section 2.4).

As a first check, we can use the AR model to estimate thermodynamic observables (energy, entropy, correlations) of the true Hamiltonian. If the AR model has a lower entropy than the true one, the AR model is probably suffering mode-collapse. Another interesting observable is the KL divergence, either $D_{\text{KL}}(P_{\text{AR}}||P_B)$ or $D_{\text{KL}}(P_B||P_{\text{AR}})$, which measures how well the AR model approximates the target one, and more quantitatively provides the average of the reweighting factor, as discussed in section 2.3. A more easily accessible quality measure consists in generating samples with the AR model, then evolving them with local MCMC and checking if the energy remains constant and the correlation functions remain time-translationally invariant [6, 41], as expected if the initial configuration is a good equilibrium one.

A very important measure of the quality of the AR model is the acceptance rate of the global MCMC, as a high acceptance rate indicates that the AR model describes well the true model, at least in the region explored by the global MCMC. Yet, the AR model could be mode-collapsed and still keep a high acceptance rate, because the global MCMC would then only explore the region on which the AR model has collapsed [15]. In particular, this could happen if:

- The true model has a few (or a polynomial number of) probability peaks. In that case, the AR model could have learned the correct energy and correct free energy, and still the entropy could be lower than the true one, but by a sub-extensive (hence difficult to detect) amount. This is the standard scenario of mode-collapse, e.g. in a ferromagnetic model with two states, when the AR model is only learning one of them.
- The true model has a number of peaks growing exponentially in N, hence associated to an extensive contribution to the entropy (usually called configurational entropy or complexity in the glass literature [38]). In this case, if the AR model has learned the true energy (or a higher one), and its entropy is much lower than the true one, then its free energy should be appreciably higher than the true one. If instead the AR model with mode-collapse learns a good approximation to the true free energy, this happens by lowering both the entropy and the energy. In this case, mode-collapse is easily detected by a drift of the energy, that increases with the number of iterations when MCMC is started from AR-generated samples.

In practice, if (i) the global MCMC has good acceptance rate, (ii) the global MCMC dynamics initialized in a configuration generated by the AR model is close to stationary (i.e. one-time quantities such as the energy are constant in time, and two-time quantities such as correlations only depend on the time difference), and (iii) the time-dependent correlation of global MCMC vanish for large time differences, then the model should be of high enough quality. Yet, ultimately, it is hard to assess the quality of the AR model in a general way. We will give more concrete examples in the rest of the paper.

3. CW model

As a first exercise, we discuss here a mean-field ferromagnetic model, namely the CW model, which provides some insight on the expressive properties of AR models.

3.1. The model and its Boltzmann distribution

The model is defined on a set of binary Ising spins, for convenience $\sigma_i \in \{-1, 1\}$, by the Hamiltonian:

$$H(\sigma) = -\frac{N}{2}m^2, \quad m = \frac{1}{N}\sum_i \sigma_i. \tag{12}$$

In the thermodynamic limit, the value of m that dominates the Boltzmann probability distribution is the solution of $m = \tanh(\beta m)$ (see the SI for details), which gives $m = 0$, i.e. a paramagnetic phase, for $T > 1$, and $m = \pm\hat{m}$, i.e. a ferromagnetic phase, for $T < 1$. Note that the two solutions with positive or negative m have the same free energy, hence the probability distribution of the model has two distinct peaks corresponding to these two ferromagnetic states. The model has a spin-flip symmetry $\sigma_i \to -\sigma_i$, which is spontaneously broken for $T < 1$.

For $N \to \infty$ and $T > 1$, the Boltzmann probability distribution is well approximated by a product distribution,

$$P_B(\sigma) = \frac{\mathrm{e}^{-\beta H(\sigma)}}{Z} \sim \prod_i p(\sigma_i), \quad p(\sigma_i) = \frac{1}{2}, \tag{13}$$

in the sense that the KL divergence (per spin) between the two sides of equation (13) vanishes for $N \to \infty$. When $T < 1$, this holds separately for each of the two ferromagnetic states, resulting in a mixture of two product distributions:

$$\begin{aligned} P_B(\sigma) &\sim \frac{1}{2}\prod_i p_+(\sigma_i) + \frac{1}{2}\prod_i p_-(\sigma_i), \\ p_\pm(\sigma_i) &= \frac{1 \pm \hat{m}\sigma_i}{2}, \end{aligned} \tag{14}$$

each peak being approximated by the product form with $m = \hat{m}$ and $m = -\hat{m}$.

3.2. Expressivity of the AR model

The CW model is well approximated by a mixture of two product distributions for large N. We now want to ask whether this structure can also be well approximated by a single AR model. Because training an AR model on a sample generated by the CW model is only possible numerically, we consider instead a sample generated by the mixture model, which coincides with the CW one for $N \to \infty$. For this model, we can provide an analytic solution.

We then assume that an infinite sample of spin configurations is generated from the mixture model in equation (14), i.e. with probability 1/2 we choose $m = \pm\hat{m}$ and we then choose each spin independently with probability $p_\pm(\sigma_i)$. We use this infinite sample to train the AR model via maximum likelihood (section 2.1). We consider here a shallow AR model (see the SI for details). Because of the spin-flip symmetry, which is realized in an infinite sample from the true model, the local fields $h_i = 0$ have to vanish. Furthermore, because the Hamiltonian is invariant under any relabeling of the $\{\sigma_i\}$, in each conditional probability $P^i_{\mathrm{AR}}(\sigma_i|\sigma_{i-1}, \ldots, \sigma_1)$ the spins $\sigma_{i-1}, \ldots, \sigma_1$ are seen as equivalent by spin σ_i, hence the couplings J_{ij} should be independent of the index $j < i$, leading to:

$$P^i_{\mathrm{AR}}(\sigma_i|\sigma_{<i}) = \frac{\exp\left(\sum_{j(<i)} J_i \sigma_i \sigma_j\right)}{2\cosh(\sum_{j(<i)} J_i \sigma_j)} = \frac{\exp\left(J_i \sigma_i m_{<i}\right)}{2\cosh(J_i m_{<i})}, \tag{15}$$

where $m_{<i} = \sum_{j(<i)} \sigma_j$ is the magnetization of the first $(i-1)$ spins. Hence, each term P^i_{AR} is parametrized by a single coupling J_i.

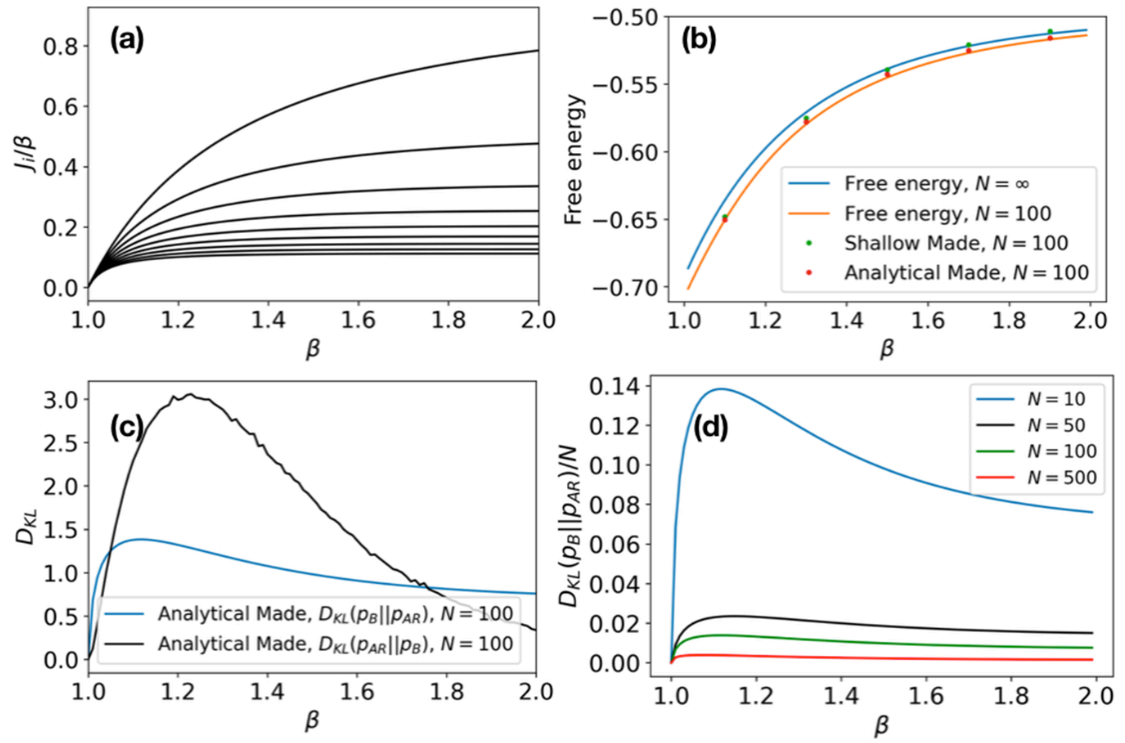

Figure 2. CW model - (a) Analytical solution of equation (16) for the first nine couplings $J_2, J_3, \ldots, J_{10}$ (from top to bottom, rescaled by inverse temperature β) of the AR model as a function of β, in the thermodynamic limit $N \to \infty$. (b) Comparison of the exact free energy per spin, obtained from the exact solution for $N \to \infty$ (full blue line) or for $N = 100$ (full orange line), with its estimate obtained from two AR models at $N = 100$, either a shallow MADE trained via maximum likelihood on equilibrium samples generated via local MCMC on the true Boltzmann distribution of the CW model (green circles), or the shallow MADE obtained from the solution of equation (16) (orange circles). (c) KL divergences between the shallow MADE obtained from equation (16) and the target model, computed for $N = 100$ as a function of β. (d) Scaling of the KL divergence with N.

Minimization with respect to J_i of the KL divergence between the mixture model in equation (14) and the AR model in equation (15) leads to a set of coupled equations:

$$\begin{aligned} p(k) &= \binom{i-1}{k}\left(\frac{1+\hat{m}}{2}\right)^{k}\left(\frac{1-\hat{m}}{2}\right)^{i-1-k}, \\ \hat{m}^2 &= \sum_{k=0}^{i-1} p(k)\frac{2k-i+1}{i-1}\tanh[J_i(2k-i+1)]. \end{aligned} \tag{16}$$

Once $\hat{m}$ is computed by solving $m = \tanh(\beta m)$, the J_is can be computed by solving equation (16) numerically. We can also compute the resulting minimum KL divergence (SI).

Because $\hat{m} = 0$ for $T > 1$, all the $J_i = 0$ and the AR model trivially reduces to the product distribution in equation (13). Results for $T < 1$ are shown in figure 2(a) for the first few couplings. While we find a non-trivial relation between J_i and β in the vicinity of the critical temperature $\beta = 1$ where all the J_i vanish, figure 2(a) suggests that J_i roughly scales as $\beta/(i-1)$ at large β. A comparison of the free energy per spin, obtained from the exact solution and from the AR model, is given in figure 2(b). The KL divergences between P_{AR} and P_B (in both directions) are given in figure 2(c); note that these are not divided by N, hence their value per spin is very small, and shows a mild peak in the vicinity of the phase transition. Finally, figure 2(d) shows that $D_{\mathrm{KL}}(P_B||P_{\mathrm{AR}})/N$ vanishes at all temperatures when $N \to \infty$, which confirms that the AR model can perfectly approximate the target one in the thermodynamic limit, even in the ferromagnetic phase.

This calculation gives some useful insight on the expressivity and training of AR models:

- If training is done via maximum likelihood with a perfect sample of the target model, no mode-collapse is observed, as expected.
- A single shallow AR model is capable of expressing a two-peaked distribution, at the price of having finite couplings that are much stronger than the CW ones (which where of order β/N, taking into account the factor of β in the Boltzmann measure). This happens as follows. The first spin σ_1 is chosen at random, $P^1_{\mathrm{AR}}(\sigma_1) = 1/2$. The second spin has a strong coupling with the first, $J_2 \sim \beta$. The third spin also has a strong

coupling with the first two, $J_3 \sim \beta/2$, and so on. More generally, this suggests that for q-state variables, a single AR model can express a distribution with q peaks, each peak being selected by the choice of the first spin. Whether an AR model can express a distribution with more than q peaks remains open at this stage, but we will see in section 5 that this is not possible when there are too many peaks.

- The maximum-likelihood training converges to a solution at all temperatures, and does not suffer from any slowdown in the vicinity of the critical point.

We thus conclude that AR models can represent well models with second-order phase transitions, without suffering from slowing down, at least when training is done via maximum likelihood.

3.3. Mode-collapse

We also consider a shallow AR model trained in a variational way (section 2.2) to represent the CW model. In this case, we do not impose any restriction on the couplings J_{ij} and the fields h_i (see the SI), and the variational free energy cannot be written explicitly. For $T < 1$, we found numerically that the AR model can end up in three different fixed points. The first one corresponds to the 'correct' form in equation (15), i.e. with $h_i = 0$ and $J_{ij} = J_i$. The other two correspond to mode-collapse in a solution with $J_{ij} = 0$ and $h_i = h = \pm T\,\mathrm{atanh}(\hat{m})$: in each of these two solutions, the AR model learns only one of the two peaks of the Boltzmann distribution, resulting in a loss of entropy per spin of $(\log 2)/N$, which is however too small to be detected for large N. Moreover, the basin of attraction of the correct solution seems to be vanishingly small for large N, and the two mode-collapsed solutions dominate the training. In fact, to obtain the solution with $h_i = 0$, we have to force the fields to vanish, thus imposing the spin-flip symmetry. We conclude that, as expected, one has to be careful about mode-collapse when training the AR model variationally.

4. EA model

We now consider a two-dimensional (2d) EA spin glass model, chosen because it is not exponentially hard to sample (finding its ground state requires polynomial time in N [48–50]), there is no phase transition at finite temperature (hence no *a priori* risk of mode-collapse), and yet the relaxation time of local MCMC grows quite fast upon lowering temperature. Moreover, previous work [14] has explored this model, finding an important speedup using a NADE architecture.

4.1. Model and architectures

We consider a 2d EA model defined on a 10×10 square lattice with periodic boundary conditions, i.e. with $N = 100$ Ising spins, $\sigma_i \in \{-1, 1\}$, as:

$$H(\sigma) = -\sum_{\langle ij \rangle} J_{ij}\sigma_i\sigma_j. \tag{17}$$

Here, the sum runs over neighboring pairs of sites in the square lattice, and the couplings J_{ij} are i.i.d. from a Gaussian distribution with zero mean and unit variance as in [14].

For the AR decomposition in equation (2), we order the variables moving along e.g. horizontal lines from the top left to the bottom right of the lattice. We consider two different AR architectures: a shallow MADE without fields ($h_i = 0$) and with $N(N-1)/2$ independent coupling parameters J_{ij}, and a NADE model with N_h hidden nodes (see the SI). In the NADE, in order to impose the inversion symmetry $\sigma_i \to -\sigma_i$, we set all the biases to zero, and we use sigmoid functions in the hidden layers. The NADE then has $2 \times N_h \times N$ parameters.

In order to train the two AR models, we use sequential tempering (section 2.4) as in [14]. The initial sample used to learn the model by maximum likelihood is generated at $\beta = 1$, where local MCMC still decorrelates fast enough. It contains $M = 10^5$ configurations, and is obtained by a single local MCMC, adding a new configuration every 10 sweeps, resulting in a total of 10^6 sweeps. The models are learned with gradient descent using mini-batches of 256 configurations, during 50 training epochs and with a learning rate of 10^{-3}. During the sequential tempering, each sequence of the sample is evolved for 10 global MCMC steps, at a new inverse temperature $\beta \leftarrow \beta + 0.1$. The acceptance rate is also computed during this global MCMC. The model is then trained again, with the same hyperparameters, using the new sample. The times needed on a standard laptop to perform local (one local MC sweep corresponds to N single spin-flip attempts) and global MCMC, and to train the model, are given in table 1.

4.2. Results

We begin by reproducing the results of McNaughton *et al* [14] with the NADE, choosing $N_h = 64$ which is the minimal number needed to obtain accurate results according to McNaughton *et al* [14]. Having reproduced

Table 1. Computational time needed to perform local MCMC, global MCMC, and model learning, on a standard laptop.

Local MCMC time for one sweep	10^{-5} s
Global MCMC time for one move	$3\cdot 10^{-4}$ s
Total learning time at given β	5 mn

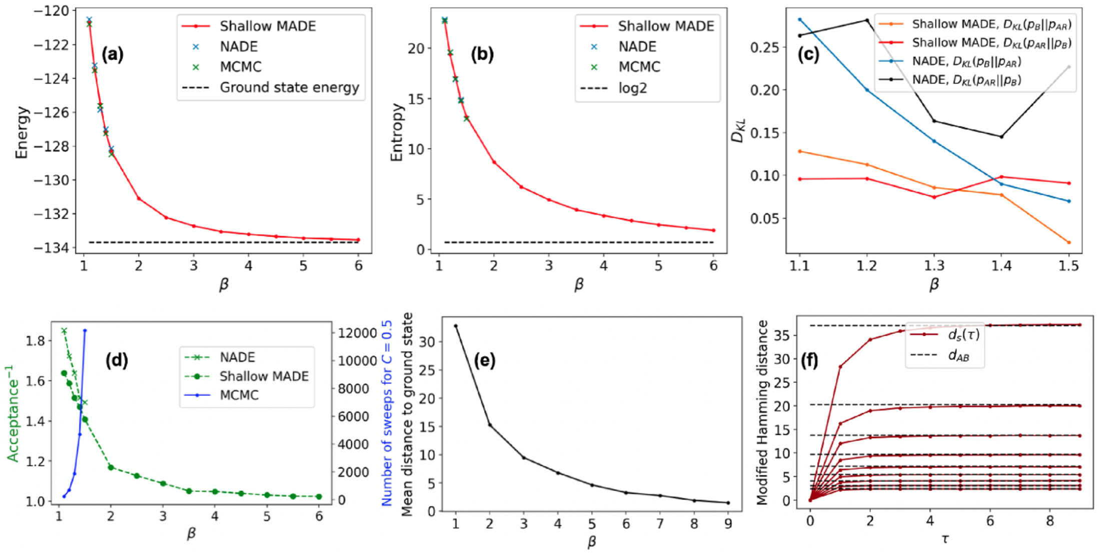

Figure 3. 2d EA model with $N = 100$ spins (10×10 periodic square lattice) and a training sample of $M = 10^5$ configurations. (a) Total energy and (b) total entropy as a function of β, as estimated by the $N_h = 64$ NADE and by the shallow MADE, compared with the exact result obtained by local MCMC. The entropy is obtained by thermodynamic integration, and the exact ground state energy has been obtained via the McGroundstate server [50]. (c) KL divergences between the AR models and the Boltzmann distribution as a function of β. (d) Inverse of the acceptance rate of global MCMC as a function of inverse temperature β, for the $N_h = 64$ NADE and the shallow MADE, both trained via sequential tempering. For comparison, we also show the decorrelation time of local MCMC. (e) Average Hamming distance (i.e. number of flipped spins) between the ground state and equilibrium configurations generated by global MCMC with the shallow MADE, as a function of β. (f) Average Hamming distance d_{AB} between two independent global MCMC runs with the shallow MADE (dashed horizontal lines), compared with the average Hamming distance $d_s(\tau)$ between a single global MC chain at two distinct times separated by τ global MC steps (red dots and lines). From top to bottom, $\beta = 1, 2, 3, \dots, 9$.

the NADE results of McNaughton *et al* [14], we also considered a shallow MADE, which has a slightly smaller number of parameters, i.e. $N(N-1)/2 = 4950$ versus $2NN_h = 12800$ for $N = 100$ and $N_h = 64$. The shallow MADE performs identically to the NADE (figure 3), and we find the shallow MADE preferable because it has less parameters and it is more interpretable [51]. It might be interesting, in future work, to check whether or not the shallow MADE has a better scaling upon increasing N with respect to the NADE.

Figure 3(d) shows the number of local MCMC sweeps τ needed to reach a time-averaged spin-spin correlation:

$$C(\tau) = \frac{1}{T}\sum_{t=1}^{T}\frac{1}{N}\sum_{i=1}^{N}\sigma_i(t)\sigma_i(t+\tau) = 0.5. \tag{18}$$

Note that already for $\beta = 1.5$, the time to decorrelate is of the order of magnitude of 10^4 sweeps, and it grows quickly upon decreasing temperature. We stop investigating the local MCMC at $\beta = 1.5$ because below this temperature measuring its relaxation time becomes too computationally costly.

Next, we train the AR models using the sequential tempering procedure with $M = 10^5$ training configurations, as detailed in section 4.1. We observe in figure 3(d) that the acceptance rate of global MCMC moves remains very close to one at all temperatures, and it actually *increases* upon decreasing temperature, reaching one for $\beta \to \infty$. For this paramagnetic model, new configurations proposed by the global MCMC are completely uncorrelated from previous ones, and the decorrelation time for the global MCMC is of the order of magnitude of the inverse acceptance rate, hence between one and two global MCMC moves. We checked explicitly in figure 3(f), where we show that the average Hamming distance (i.e. number of distinct spins) between two configurations obtained by independent global MCMC, and between a single MCMC and itself a few global MCMC later, coincide.

Using the numbers reported in table 1, we find that at $\beta = 1.5$, decorrelation on a standard laptop takes about 0.12 s using local MCMC (i.e. about 10^{-5} s per MC sweep times 12000 time sweeps to decorrelate), and about 10^{-3} s for global MCMC (that requires a few moves to decorrelate, see figure 3(f), each taking about $3 \cdot 10^{-4}$ s), hence with a speedup of two orders of magnitude. Furthermore, the decorrelation time of global MCMC slightly decreases upon lowering temperature, while that of local MCMC strongly increases, leading to an even stronger speedup, divergent for $\beta \to \infty$. Note that the learning time is negligible at large enough β.

To monitor the quality of the AR models upon decreasing temperature, we compare the exact energy (figure 3(a)) and entropy (figure 3(b)) computed with local MCMC with those estimated by sampling the AR models, finding extremely good agreement. For the local MCMC computation, we perform an annealing from high temperature measuring the energy and the specific heat, and we obtain the entropy by integrating the latter in temperature. For the AR models, we estimate the energy and the entropy by sampling from the model at each T, and computing the average of $H(\sigma)$ and of $-\log P_{\mathrm{AR}}(\sigma)$ over this sample, respectively. We note, in particular, that the AR entropy coincides with that of the true model, which shows that the AR model is not mode-collapsed. In figure 3(c), we also show the KL divergences $D(P_{\mathrm{AR}}||P_B)$ and $D(P_B||P_{\mathrm{AR}})$ as a function of temperature. Both are extremely small, which implies that $\log W(\sigma) = \log[P_B(\sigma)/P_{\mathrm{AR}}(\sigma)]$ is extremely small for configurations σ that are typical both of the true equilibrium and of the AR model, consistently with the high acceptance rate shown in figure 3(d) (see the discussion in section 2.3).

For future reference, we also checked the dependence of the acceptance rate of global MCMC on the training set size M. We found that a minimal training set size of about $M = 10^4$ is needed to obtain good performance, while below this size the AR model cannot be trained properly and global MCMC fails (see the SI).

To summarize, we found that:

- A 2d EA model with $N = 100$ spins can be approximated very accurately by an AR model, as shown by the very small KL divergence.
- Correspondingly, global MCMC is very efficient and leads to an important speedup with respect to local MCMC, as shown in [14]. We have shown that the speedup diverges for $\beta \to \infty$.
- A simple shallow MADE performs as well as the NADE used in [14], even with a slightly smaller number of parameters.
- A sufficiently large training set ($M \gtrsim 10^4$) must be used to achieve a good efficiency of the global MCMC.

We thus conclude that, at least for $N = 100$ (a 10×10 periodic lattice), the 2d EA model is easy to approximate with an AR model, leading to efficient global MCMC. Yet, the fact that the AR model concentrates at low temperatures on the two ground states (related by the inversion symmetry), which can be found in short time by algorithms such as McGroundstate [50], leads us to believe that the instance we studied is too small to be really hard to sample.

Furthermore, we repeated the study on a 3d EA model on a $5 \times 5 \times 5$ cubic periodic lattice ($N = 125$ spins). Also in this case, because of the small size, the ground state is found easily by McGroundstate [50], despite the fact that the problem is in principle NP-hard [48]. We found that the AR model approximates well the 3d EA model, leading to efficient global MCMC at all temperatures, despite the presence of a phase transition at finite temperature [52], and converging to the ground state in the zero-temperature limit (see the SI).

We believe that a more complete study should investigate systematically the dependence on system size for much larger N, as well as the dependence on the space dimension, because the EA model is known to become harder and harder to sample upon increasing spatial dimensions. Because this is not the main focus of the present work, we leave this for future investigation. The main goal of the present section was to show that our implementation of the global MCMC is able to reproduce and extend previous positive results [14], and it can thus be considered reliable and efficient.

5. Coloring

We now consider our main benchmark, namely the hard-to-sample coloring problem. We will show that all the global MCMC implementations discussed above fail for this model, and we could not find an implementation that achieves a better efficiency. In particular, we will show that both the maximum-likelihood and the variational training fail, but for quite different reasons.

5.1. Model

The model is formulated in terms of N variables $\sigma_i \in \{0, \dots, q-1\}$, each taking q possible colors. The variables are the nodes of an Erdös-Rényi random graph $\mathcal{G}$, i.e. a graph in which each link $\langle i,j \rangle$ is present at

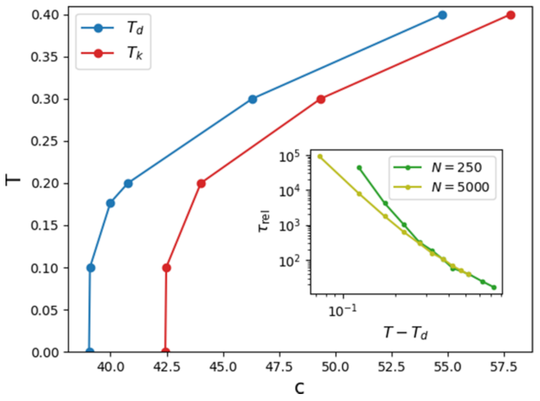

Figure 4. Coloring—Phase diagram of the coloring of Erdös-Rényi random graphs with $q = 10$ colors, as a function of the average node connectivity c and of temperature T [53]. The upper line is the dynamical glass transition $T_d(c)$, below which the decorrelation time increases exponentially with N; the lower line is the condensation transition $T_K(c)$ below which quiet planting is not possible. Inset: decorrelation time of the equilibrium local MCMC for $c = 40$, for two different system sizes, as a function of $T - T_d(c = 40)$.

random with the same probability, chosen such that the average connectivity of a node is c. The model Hamiltonian can then be written as:

$$H(\sigma) = \sum_{\langle i,j\rangle \in \mathcal{G}} \delta_{\sigma_i,\sigma_j}, \tag{19}$$

i.e. a q-state Potts model where antiferromagnetic couplings (whose value is fixed to one without loss of generality) are only present on the edges of $\mathcal{G}$. The Hamiltonian counts the monochromatic edges by assigning a zero energy to all links that connect sites with distinct colors, and energy one to links that connect sites with the same color. Hence, $H(\sigma) \geqslant 0$ and $H(\sigma) = 0$ if and only if σ is a proper coloring of $\mathcal{G}$. At zero temperature, the problem of finding the ground state of $H(\sigma)$ is therefore related to the random graph coloring problem [43, 44].

At finite temperature and for large enough q, the model belongs to the random first order transition (RFOT) universality class [54], together with many other optimization problems [38] and with mean-field structural glasses [39–42]. These models are characterized by a finite-temperature dynamical glass transition, below which the time needed for a proper sampling of the Boltzmann probability diverges exponentially with N [2], due to the presence of an exponential number of peaks (i.e. metastable states) [24, 38, 41, 42].

The coloring problem is specified by the number of sites N, the choice of the random graph $\mathcal{G}$ which is specified by its average connectivity c, the number of colors q, and temperature T. For the model to be a good representative of the RFOT class, we need q to be large enough, otherwise the transition is too close to a second order one, especially at finite N [54]. We thus chose $q = 10$, for which a finite RFOT dynamical transition $T_d(c)$ is present for all $c > 39.02$, and the corresponding phase diagram is reported in figure 4 (data were privately communicated by the authors of Cavaliere *et al* [53]).

Systems in the RFOT class also display a second phase transition, called condensation or Kauzmann transition $T_K(c)$. Below this transition, the number of metastable glassy states become sub-extensive [38]. For our purposes, this phase transition is relevant because for $T > T_K(c)$ one can use the quiet planting technique [55], which consists in exchanging the sampling of σ and the generation of $\mathcal{G}$. While one should first generate a graph $\mathcal{G}$ and then sample σ from the Boltzmann distribution for that $\mathcal{G}$ (which is exponentially hard in N at low T), in quiet planting one first takes a random configuration σ, and then constructs at a given T a random graph $\mathcal{G}$ such that σ is an equilibrium configuration for the model defined on that graph, at temperature T (which can be done in polynomial time). It can be shown [55] that for $T > T_K(c)$ in models such as the coloring, this exchange of order does not affect the statistical properties of $\mathcal{G}$. This allows one to run local MCMC from the configuration σ, which is guaranteed to be an equilibrium one, thus bypassing the need for equilibration, which takes an exponentially large time for $T < T_d(c)$. Our choice of $q = 10$ ensures the existence of a large enough region $T_K < T < T_d$ where model-generated samples may be compared to the planted configurations.

For our study, we thus chose $c = 40$, for which $T_d = 0.1768$ and $T_K = 0$ (figure 4). In this way, we can perform quiet planting at any temperature to study the equilibrium local MCMC dynamics, and we also have a wide range of temperature $T < T_d$ where this dynamics is exponentially slow in N. Notice that such graphs are full of short loops, so some finite size effects are measurable, like an energy drift in the planted solution.

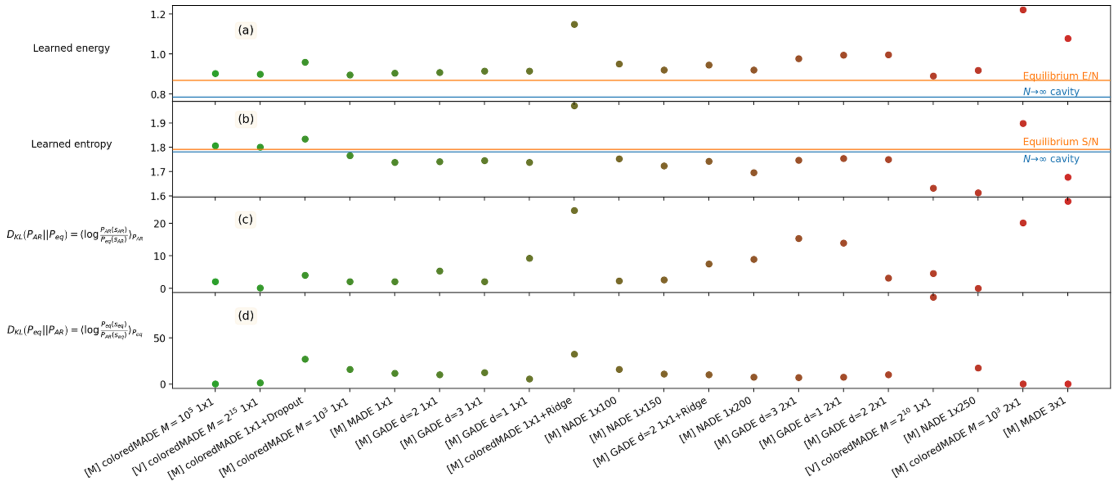

Figure 5. Coloring with $q = 10$, $c = 40$, $T = 1$ and $N = 250$. The four panels of the figure report the value of energy (a), entropy (b) and KL divergences (c), (d) for different AR architectures, trained with the techniques of section 2. The name of the models (i.e. its training technique, AR network type, etc) is reported on the x-axis and corresponds to a specific color. Overall, the models are ranked, from left to right, by free energy, which at $T = 1$ is the difference of energy minus entropy. The horizontal lines correspond to the equilibrium values, measured using local MCMC for $N = 250$ (orange, the entropy is obtained via thermodynamic integration) or calculated from the cavity method for $N \to \infty$ (blue).

We have confirmed that the scenario we describe in the next sections is equally reproduced for lower connectivity, $\{q = 5, c = 13\}$ and $\{q = 10, c = 30\}$, so that it is mainly determined by the complexity of the problem. In figure 4 we show the decorrelation time of the overlap correlation function,

$$C(t,\tau) = \frac{1}{N}\sum_{i=1}^{N} \delta_{\sigma_i(t),\sigma_i(t+\tau)}, \tag{20}$$

here measured using local MCMC and averaging over time t. Note that with this definition, complete decorrelation corresponds to $C(t,\tau) \sim 1/q$. The decorrelation time is defined by $\langle C(t,\tau)\rangle_t = 0.5$. When plotted as a function of $T - T_d$, it displays the characteristic power-law divergence predicted by RFOT theory. In the vicinity and below T_d, the dynamics becomes so slow that we are unable to observe decorrelation of local MCMC in reasonable time. We consider a fixed size $N = 250$, which is a good compromise between avoiding too large finite size effects, and having a small enough size to allow for an extensive testing of AR model architectures.

It is also important to mention that the thermodynamics of the model can be solved, for $T > T_K(c)$ and for $N \to \infty$, by a simple cavity computation [53–55]. In particular, the energy and entropy per spin are given by:

$$\begin{aligned} e(T) &= \frac{c}{2}\frac{\mathrm{e}^{-\beta}}{q-1+e^{-\beta}}, \\ s(T) &= \log q + \frac{c}{2}\log\left[\frac{q-1+\mathrm{e}^{-\beta}}{q}\right] + \beta e(T). \end{aligned} \tag{21}$$

Although for $N = 250$ we observe significant deviations from the thermodynamic values (figure 5), these expressions are still useful as a reference.

5.2. Model selection

Our goal is to use AR models to generate good trial configurations and speed up the dynamics of the hard-to-sample coloring problem. Before exploring the hard region below T_d, we test how different AR architectures perform in the paramagnetic phase at $T = 1$, where the local MCMC dynamics is fast and we can easily generate an equilibrium sample from the Boltzmann distribution. Our intermediate goal is to select the best architectures and training methods (see the SI for details).

For the AR decomposition in equation (2), variables are ordered naively, which means that we first fix an arbitrary ordering $\sigma_1, \ldots, \sigma_N$, and then construct a graph $\mathcal{G}$ as described in section 5.1. We also considered ordering the variables from higher to lower connectivity, and found no qualitative difference. Results are reported here for this second choice of ordering. The models are trained either via maximum-likelihood on a sample of M equilibrium configurations generated via local MCMC, or via the variational method using M

model-generated samples to compute the gradient, see section 2.2. The training is done using the Adam optimizer [56] and mini-batches of 1024 configurations. In order to enforce color symmetry, each configuration in the sample is subjected to a random permutation of colors before being used, in each epoch of training. We start from a learning rate of 0.01 that we reduce by a factor 2 after every 250 epochs of no improvement [57]. We set an early stop criterion that triggers if the likelihood of a validation set made of independent equilibrium samples starts decreasing.

At the end of the training, we compute the model energy (figure 5(a)) and entropy (figure 5(b)) by sampling from the AR model and computing the average of $H(\sigma)$ and $-\log P_{\mathrm{AR}}(\sigma)$, respectively. We compare these results with the exact ones obtained via local MCMC. We observe that the AR energy and entropy are generally quite close to the equilibrium values. In particular, the samples generated by the models often have an energy that is closer to the average equilibrium energy than that of the planted configuration, hence the difference falls within the typical equilibrium fluctuations. As a general trend, we observe that increasing the model expressivity does not improve the energy, probably because of overfitting, as shown by the decrease in entropy. On the other hand, adding a regularization (either dropout or ridge) degrades the performance significantly, by increasing both the energy and the entropy. We thus conclude that the best model is once again a shallow MADE (ColoredMADE, see the SI), which can achieve a very low energy and a very high entropy, close to the target values, both with variational and maximum-likelihood training.

In figures 5(c) and (d) we report the KL divergences between the Boltzmann distribution and that of the AR models, which confirms that the two shallow ColoredMADE have the smallest KL divergence. Having selected two good architectures at $T = 1$, we now focus on these two models and check how they perform upon lowering the temperature. We consider, for each T, a ColoredMADE trained by maximum likelihood on an independent equilibrium sample generated from the Boltzmann distribution by long enough local MCMC at that T, and another ColoredMADE trained variationally at fixed T (which, we recall, does not require to generate equilibrium samples).

We will show next that both models fail at lower T. Of course, there might be other architectures that we did not consider here, which might provide better results. However, because both adding regularization or increasing the expressivity of the model (also by using GNNs) did not seem to improve the efficiency, we do not see a clear path for improvement.

5.3. Failure of sequential tempering

We first consider the ColoredMADE trained by maximum likelihood, and compare the average energy of the equilibrium configurations with those generated by the AR model (figure 6(d)). We observe that the model generates configurations with higher energy as soon as temperature goes below $T \sim 1$, while its entropy (figure 6(e)) remains close to the equilibrium value. Yet, once the entropy is plotted parametrically as a function of energy (figure 6(f)), the difference from the equilibrium result becomes apparent.

We also compare the correlation functions of local MCMC dynamics starting from the equilibrium samples and the AR-generated samples. (To avoid confusion, we stress that here we are not yet considering AR-assisted global MCMC.) In figure 6(a) we report the overlap correlation function defined by equation (20), here for fixed $t = 0$ as a function of τ measured in MC sweeps. The dashed lines are obtained starting at $t = 0$ from equilibrium samples, while the full lines are obtained starting from samples generated by the AR model. We observe that AR-generated samples have the correct equilibrium local MCMC dynamics only at very high temperatures. In figure 6(c) we compare the relaxation times, defined by $C(0,\tau_{\mathrm{rel}}) = 1/e$, and show that AR-samples have systematically faster dynamics upon lowering T. Overall, these results suggest that the maximum-likelihood training is effective at high temperatures only, because upon lowering temperature the model is unable to propose configurations of low enough energy. This study is restricted to $T \geqslant 0.3$ because below that temperature we are unable to generate the equilibrium samples needed to train the model by local MCMC.

Next, we consider the global MCMC dynamics, and we measure its acceptance rate at $T = 1$. As in section 4.2, global MCMC can be initialized (i) on AR-generated configurations, (ii) on the training set and (iii) on an independent test set made by additional equilibrium configurations. In figure 7(a), we consider a ColoredMADE trained on $M = 10^3$ configurations, and we report the average acceptance rate together with the geometric average of the Boltzmann ratio $P_B(\sigma_{\mathrm{new}})/P_B(\sigma_{\mathrm{old}})$ and the model ratio $P_{AR}(\sigma_{\mathrm{old}})/P_{\mathrm{AR}}(\sigma_{\mathrm{new}})$. Similarly to the 2d EA model (see the SI), we observe that for such a small M, the acceptance rate on the test set is vanishingly small, while using the other initializations, it decreases with time. The situation improves strongly upon increasing the size of the training set to $M = 10^5$ (figure 7(b)), leading to a good acceptance rate. Global MCMC can then decorrelate reasonably fast in this case.

We explore more systematically the dependence on T and M in figure 7(c), where we report the acceptance as a function of β for different architectures and different M. We observe that for a given architecture and M, the acceptance drops exponentially fast with β, in some cases becoming so small that we

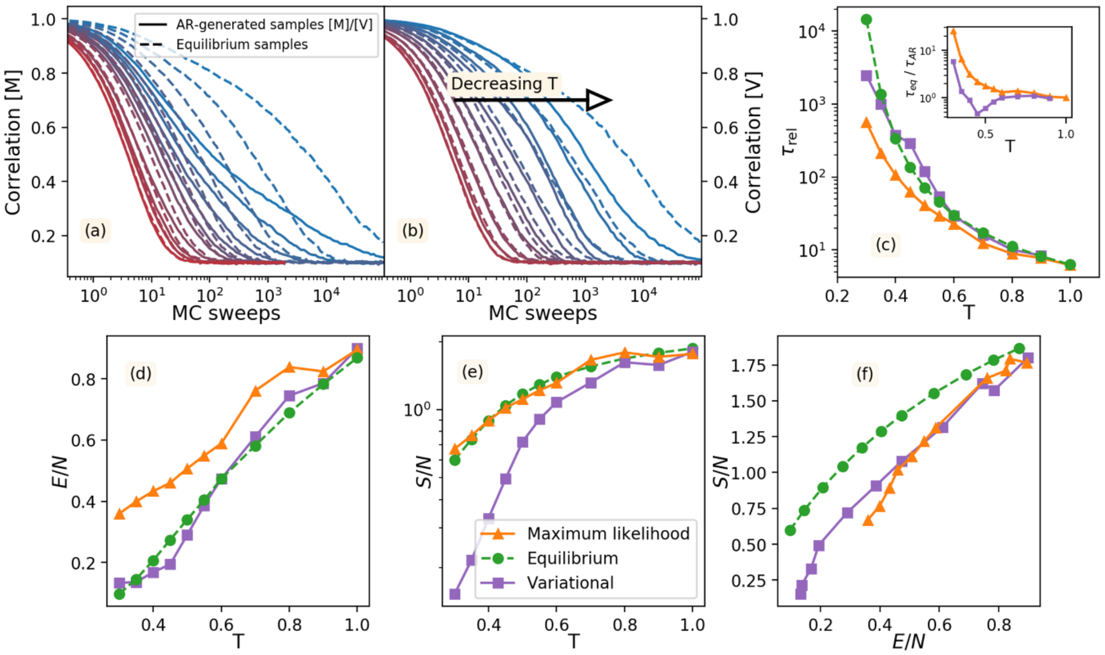

Figure 6. Coloring with $q = 10$, $c = 40$ and $N = 250$. (a), (b) Overlap correlation function for local MCMC dynamics starting from samples generated either in equilibrium (dashed lines, via long enough local MCMC) or by a ColoredMADE AR model trained both by maximum likelihood (a, continuous lines) and variationally (b, continuous lines). The temperature decreases from $T = 1$ (red) to $T = 0.3$ (blue) following the color gradient. (c) Relaxation time τ_{rel} defined from $C(0, \tau_{\text{rel}}) = 1/e$ for the samples generated by the AR model trained using maximum likelihood (orange) or variationally (purple) compared to equilibrium (green). Relaxation is measured averaging over 100 local MCMC runs. In the inset we report the ratio of the equilibrium time to that of AR models. The difference becomes larger upon decreasing the temperature, but the variational model (purple) remains accurate down to $T \sim 0.35$. (d) Average energy and (e) entropy of samples generated by the two AR models and in equilibrium, with the same color code as in (c). In panel (e), the green data correspond to the exact entropy in the thermodynamic limit, equation (21). Using the same data of (d), (e), we plot in (f) the entropy of the generated samples as a function of their energy.

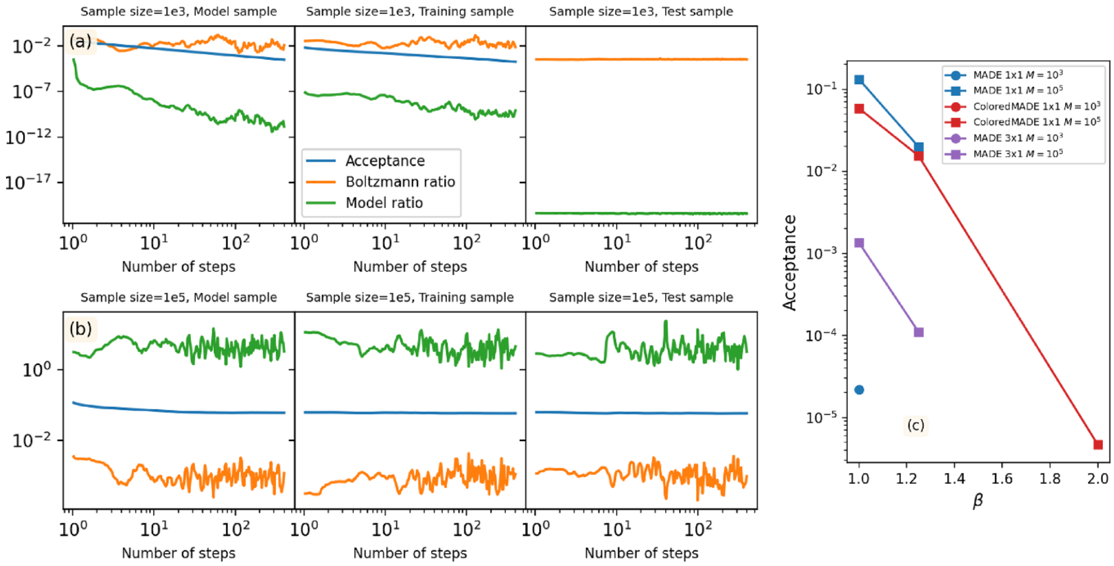

Figure 7. Coloring with $q = 10$, $c = 40$ and $N = 250$. (a) Acceptance rate of global MCMC for the ColoredMADE trained via maximum likelihood at $T = 1$ using a sample of $M = 10^3$ equilibrium configurations. Here, $K = 100$ independent global MCMC are initialized in configurations generated by the AR model itself (model sample), taken from the training sample, or from additional independent equilibrium configurations (test sample). We report the average acceptance rate, and the geometric average of the Boltzmann and model ratio, over the K chains. (b) Same with $M = 10^5$. (c) Acceptance rate (on the test set) of global MCMC for several AR models trained with the maximum likelihood technique on equilibrium configurations generated via local MCMC at several temperatures. Missing data indicate that the acceptance rate is so small that it is impossible to measure it.

cannot measure it (no move is accepted during the longest global MCMC we can run). We conclude that the maximum likelihood approach fails because, upon lowering temperature, the AR models are not expressive enough to properly fit the target energy, and as a consequence they propose moves that cannot be accepted

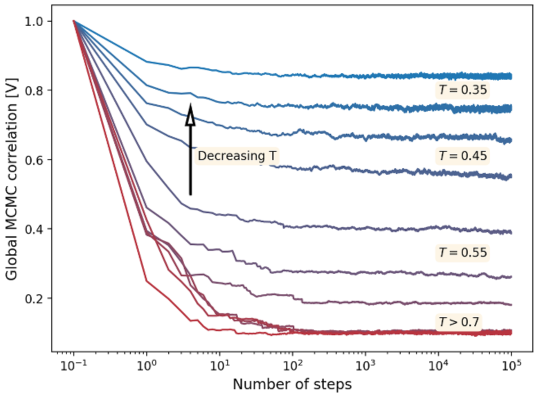

Figure 8. Coloring with $q = 10$, $c = 40$ and $N = 250$. Overlap correlation function of the global MCMC using the variational ColoredMADE as a function of the number of MC steps.

by the global MCMC. Increasing M does not help at low temperature (figure 7(c)), and we have already seen that increasing the model expressivity only leads to overfitting (figure 5).

It is possible that by improving model expressivity and at the same time considering larger M, the acceptance rate could improve. However, the AR architectures we considered are already quite hard to train, and $M \sim 10^5$ corresponds to a quite large dataset, and still the method fails at $T = 0.5$, which is a quite high temperature in the paramagnetic phase. These results thus highlight the lack of computational efficiency of this approach.

Note that the global MCMC fails even when the AR model is trained using maximum likelihood on a perfect equilibrium sample, i.e. under the best possible conditions. Hence, sequential tempering cannot work, because upon decreasing temperature the global MCMC remains stuck and does not lead to a proper new sample on which the AR model can be trained. In fact, performing sequential tempering runs (not shown) we observe that the global MCMC does not equilibrate, hence the AR model has even worse energy and leads to an even slower global MCMC.

5.4. Failure of the variational training

We now consider the variationally-trained ColoredMADE, and note that its energy remains accurate down to $T = 0.3$ (figure 6(d)), while its entropy decreases quickly upon lowering T (figure 6(e)), suggesting mode collapse, i.e. the model only learns a limited portion of the support of the Boltzmann distribution. Surprisingly, the parametric curve of entropy versus energy (figure 6(f)) is very similar to that obtained using maximum-likelihood training. Consistently, when we run global MCMC using the variational model, we observe that the acceptance rate remains good, but the proposed moves do not lead to any decorrelation (figure 8), and as a result the sampling of the Boltzmann distribution is not efficient. Global MCMC indeed remain confined in the limited portion of the phase space that has been learned by the variational model. This is consistent with the discussion of section 3.3 for the CW model.

In figure 6(b), we report the correlation functions of the local MCMC dynamics initialized both in equilibrium and in configurations generated by the variational AR model, and note that they are essentially indistinguishable down to $T \sim 0.35$. Because this method does not require the generation of equilibrium samples for training, one might wonder whether, despite being mode-collapsed and thus useless for global MCMC, the variational model could still generate configurations that are close enough to some of the equilibrium ones, such that the *local* MCMC dynamics coincides with the equilibrium one. If true, this would be interesting, because one could use the variational model to generate *some* typical equilibrium states, although only a small subset of them. However, we show in figure 9 that this is not the case. Below $T \sim 0.3$, upon approaching the glass transition, the energy of the variational model remains higher than the equilibrium one, and the generated configurations display an initially faster local MCMC dynamics, whose relaxation time increases during the evolution (the so-called aging behavior typical of glasses). Hence, the variational model also fails in providing good equilibrium configurations.

5.5. Role of the color symmetry

One might wonder whether the failure of the AR models is somehow related to the presence of the color symmetry (see the SI for details), which might lead to a proliferation of states, thus requiring a lot of training samples to be taken into account correctly. Or, maybe, breaking the color symmetry would allow for a better

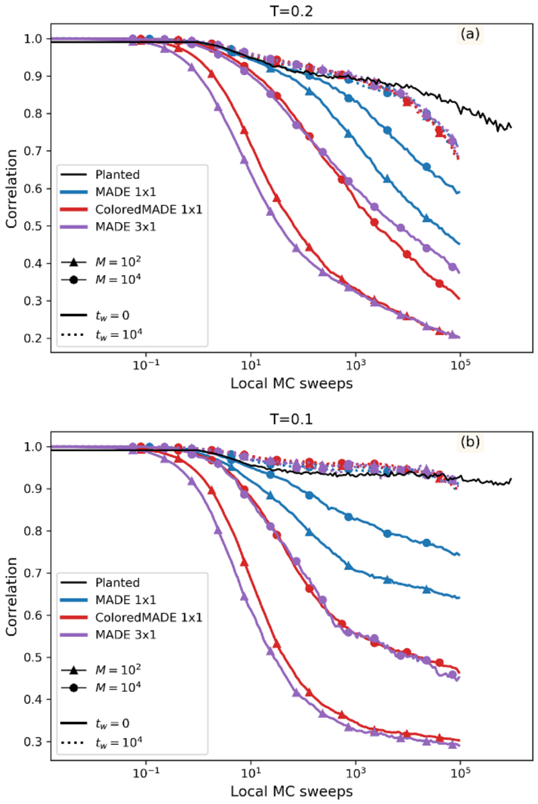

Figure 9. Coloring with $q = 10$, $c = 40$ and $N = 250$. Correlation $C(t_w, \tau)$ of local MCMC initialized at $t = 0$ in AR-generated samples at low temperatures $T = 0.2$ (a) and $T = 0.1$ (b). We used the following model architectures: shallow MADE (blue), shallow ColoredMADE (red) and deep MADE ($D = 3$, purple). The models have been trained via the variational approach with batches of $M = 10^2$ (triangles) or $M = 10^4$ (circles) configurations. The full line is the correlation measured from $t_w = 0$, while the dashed line is measured after waiting $t_w = 10^4$ sweeps. In black we report, for comparison, the exact equilibrium dynamics, obtained initializing the local MCMC in the planted configuration.

learning of the AR model, that could use local fields to obtain more information on the Boltzmann distribution, thus reducing the role of couplings.

To address this question, we tried to break the color symmetry by adding small random local fields on each spin, thus lifting the degeneracy of the $q!$ equivalent configurations. Recall that $q = 10$, hence $q!$ is very large, although the contribution to the entropy per spin, $\log(q!)/N$, is negligible. Yet, we repeated the same analysis and we obtained very similar results. We thus conclude that the addition of small external fields is not enough to make the approach work, because its failure is due to the intrinsic difficulty of representing the multiplicity of states typical of a glassy system.

6. Conclusions

In this paper, we addressed the general problem of whether machine learning can assist Monte Carlo methods to provide a computational speed up in hard-to-sample glassy problems.

We confirmed previous findings [13–19] that identified a class of problems, such as simple ferromagnetic models and the 2d EA spin glass model, which can be well approximated by simple AR models. These models can then be used to sample the Boltzmann distribution by machine-learning-assisted global MCMC, providing a computational speedup of several orders of magnitude. For these type of problems, several different architectures and training schemes have been proposed, but we found that all these methods are roughly equivalent. In particular, we tend to prefer a shallow AR architecture, which provides a simple and interpretable model. We also highlighted the need for a large enough training sample for the efficiency of the approach.

We found that the situation is totally different when one considers really hard-to-sample problems belonging to the RFOT universality class, such as the coloring of random graphs and, possibly, the model considered in [20]. For these type of problems, the landscape is very rough and complex, displaying a multitude of locally stable glass states that are known to trap standard local MCMC [2]. We showed that machine-learning-assisted global MCMC fails, either because of the mode-collapse of the variationally learned model into a restricted portion of the phase space (that is not even representative of equilibrium in the vicinity of the glass transition), or because the sequential tempering scheme fails to achieve a high enough acceptance rate of global MCMC moves.

We tried several architectures in order to overcome the problem, but we did not find any promising sign of improvement. In particular, we found that adding regularization leads to underfitting (high entropy), while increasing the expressivity of the model leads to overfitting (low entropy), and both are detrimental to the global MCMC efficiency.

We are thus led to believe that there is an intrinsic difficulty of these problems that prevents machine learning methods to solve them efficiently. Of course, we might have missed something and future work could find a solution using smarter architectures. Yet, we believe that our work is useful because it establishes a benchmark against which, in our opinion, such machine learning methods should be tested to assess their performance as universal methods for sampling speedup.

Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: 10.5281/zenodo.7567683.

Acknowledgments

We thank Giulio Biroli, Giuseppe Carleo, Angelo Cavaliere, Marylou Gabrié, Ilya Nemenman and Federico Ricci-Tersenghi for many useful suggestions related to this work.

This project has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (Grant Agreement No. 723 955 - GlassUniversality). J T is supported by a PhD Fellowship of the i-Bio Initiative from the Idex Sorbonne University Alliance.

ORCID iDs

Simone Ciarella https://orcid.org/0000-0002-9247-139X
Martin Weigt https://orcid.org/0000-0002-0492-3684
Francesco Zamponi https://orcid.org/0000-0001-9260-1951

References

[1] Krauth W 2006 *Statistical Mechanics: Algorithms and Computations* (New York, NY: Oxford University Press)
[2] Montanari A and Semerjian G 2006 Rigorous inequalities between length and time scales in glassy systems *J. Stat. Phys.* **125** 23
[3] Swendsen R H, Wang J-S and Ferrenberg A M 1992 New Monte Carlo methods for improved efficiency of computer simulations in statistical mechanics *The Monte Carlo Method in Condensed Matter Physics* ed K Binder (Berlin: Springer) pp 75–91
[4] Jörg T 2005 Cluster Monte Carlo algorithms for diluted spin glasses *Prog. Theor. Phys. Suppl.* **157** 349
[5] Zhu Z, Ochoa A J and Katzgraber H G 2015 Efficient cluster algorithm for spin glasses in any space dimension *Phys. Rev. Lett.* **115** 077201
[6] Ninarello A, Berthier L and Coslovich D 2017 Models and algorithms for the next generation of glass transition studies *Phys. Rev.* X **7** 021039
[7] Kapteijns G, Ji W, Brito C, Wyart M and Lerner E 2019 Fast generation of ultrastable computer glasses by minimization of an augmented potential energy *Phys. Rev.* E **99** 012106
[8] Ciarella S *et al* 2021 Soft particles at liquid interfaces: from molecular particle architecture to collective phase behavior *Langmuir* **37** 5364
[9] Hagh V F, Nagel S R, Liu A J, Manning M L and Corwin E I 2022 Transient learning degrees of freedom for introducing function in materials *Proc. Natl Acad. Sci.* **119** e2117622119
[10] Ozawa M, Iwashita Y, Kob W and Zamponi F 2022 Creating bulk ultrastable glasses by random particle bonding (arXiv:2203.14604)
[11] Li S-H and Wang L 2018 Neural network renormalization group *Phys. Rev. Lett.* **121** 260601
[12] Marchand T, Ozawa M, Biroli G and Mallat S 2022 Wavelet conditional renormalization group (arXiv:2207.04941)
[13] Wu D, Wang L and Zhang P 2019 Solving statistical mechanics using variational autoregressive networks *Phys. Rev. Lett.* **122** 80602
[14] McNaughton B, Milosevic M V, Perali A and Pilati S 2020 Boosting monte carlo simulations of spin glasses using autoregressive neural networks *Phys. Rev.* E **101** 053312
[15] Gabrié M, Rotskoff G M and Vanden-Eijnden E 2021 Adaptive monte carlo augmented with normalizing flows *Proc. Natl Acad. Sci.* **119** e2109420119
[16] Wu D, Rossi R and Carleo G 2021 Unbiased monte carlo cluster updates with autoregressive neural networks *Phys. Rev. Res.* **3** L042024
[17] Hibat-Allah M, Inack E M, Wiersema R, Melko R G and Carrasquilla J 2021 Variational neural annealing *Nat. Mach. Intell.* **3** 952
[18] Fan C, Shen M, Nussinov Z, Liu Z, Sun Y and Liu Y-Y 2021 Finding spin glass ground states through deep reinforcement learning (arXiv:2109.14411)
[19] Schuetz M J A, Brubaker J K, Zhu Z and Katzgraber H G 2022 Graph coloring with physics-inspired graph neural networks (arXiv:2202.01606)
[20] Inack E M, Morawetz S and Melko R G 2022 Neural annealing and visualization of autoregressive neural networks in the Newman-Moore model *Condens. Matter* **7** 38

[21] Cheeseman P C *et al* 1991 Where the really hard problems are *Proc. 12th Int. Joint Conf. on Artificial intelligence* (*IJCAI*) **91** 331–7

[22] Kirkpatrick S and Selman B 1994 Critical behavior in the satisfiability of random boolean expressions *Science* **264** 1297

[23] Selman B, Mitchell D G and Levesque H J 1996 Generating hard satisfiability problems *Artif. Intell.* **81** 17

[24] Monasson R, Zecchina R, Kirkpatrick S, Selman B and Troyansky L 1999 Determining computational complexity from characteristic 'phase transitions *Nature* **400** 133

[25] Merchan L and Nemenman I 2016 On the sufficiency of pairwise interactions in maximum entropy models of networks *J. Stat. Phys.* **162** 1294

[26] Germain M, Gregor K, Murray I and Larochelle H 2015 Made: Masked autoencoder for distribution estimation *Proc. 32 nd Int. Conf. on Machine Learning* (*Lille, France*) vol 37, ed F Bach and D Blei (PMLR) pp 881–9

[27] Uria B, Coté M-A, Gregor K, Murray I and Larochelle H 2016 Neural autoregressive distribution estimation *J. Mach. Learn. Res.* **17** 1 (available at: www.jmlr.org/papers/v17/16-272.html)

[28] Papamakarios G, Pavlakou T and Murray I 2017 Masked autoregressive flow for density estimation *Advances in Neural Information Processing Systems* vol 30, ed I Guyon, U V Luxburg, S Bengio, H Wallach, R Fergus, S Vishwanathan and R Garnett (Curran Associates, Inc.) (available at: https://proceedings.neurips.cc/paper/2017/hash/6c1da886822c67822bcf3679d04369fa-Abstract.html)

[29] Kingma D P, Salimans T, Jozefowicz R, Chen X, Sutskever I and Welling M 2016 Improved variational inference with inverse autoregressive flow *Advances in Neural Information Processing Systems* vol 29, ed D Lee, M Sugiyama, U Luxburg, I Guyon and R Garnett (Curran Associates, Inc.) (available at: https://proceedings.neurips.cc/paper/2016/hash/ddeebdeefdb7e7e7a697e1c3e3d8ef54-Abstract.html)

[30] Hartnett G S and Mohseni M 2020 Self-supervised learning of generative spin-glasses with normalizing flows (arXiv:2001.00585)

[31] Kirkpatrick S, Gelatt C D and Vecchi M P 1983 Optimization by simulated annealing *Science* **220** 671

[32] Krzakala F and Kurchan J 2007 Landscape analysis of constraint satisfaction problems *Phys. Rev.* E **76** 021122

[33] Krzakala F and Zdeborová L 2013 Performance of simulated annealing in *p*-spin glasses *J. Phys.: Conf. Ser.* **473** 12022

[34] Wang W, Machta J and Katzgraber H G 2015 Comparing Monte Carlo methods for finding ground states of ising spin glasses: population annealing, simulated annealing and parallel tempering *Phys. Rev.* E **92** 013303

[35] Schuetz M J A, Brubaker J K and Katzgraber H G 2022 Combinatorial optimization with physics-inspired graph neural networks *Nat. Mach. Intell.* **4** 367

[36] Angelini M C and Ricci-Tersenghi F 2022 Modern graph neural networks do worse than classical greedy algorithms in solving combinatorial optimization problems like maximum independent set *Nat. Mach. Intell.* **5** 29–31

[37] Boettcher S 2022 Inability of a graph neural network heuristic to outperform greedy algorithms in solving combinatorial optimization problems *Nat. Mach. Intell.* **5** 24–25

[38] Krzakala F, Montanari A, Ricci-Tersenghi F, Semerjian G and Zdeborová L 2007 Gibbs states and the set of solutions of random constraint satisfaction problems *Proc. Natl Acad. Sci.* **104** 10318

[39] Kirkpatrick T R and Wolynes P G 1987 Connections between some kinetic and equilibrium theories of the glass transition *Phys. Rev.* A **35** 3072

[40] Kirkpatrick T R and Thirumalai D 1988 Comparison between dynamical theories and metastable states in regular and glassy mean-field spin models with underlying first-order-like phase transitions *Phys. Rev.* A **37** 4439

[41] Cugliandolo L F and Kurchan J 1993 Analytical solution of the off-equilibrium dynamics of a long-range spin-glass model *Phys. Rev. Lett.* **71** 173

[42] Parisi G, Urbani P and Zamponi F 2020 *Theory of Simple Glasses: Exact Solutions in Infinite Dimensions* (Cambridge: Cambridge University Press)

[43] Mulet R, Pagnani A, Weigt M and Zecchina R 2002 Coloring random graphs *Phys. Rev. Lett.* **89** 268701

[44] Zdeborová L and Krzakala F 2007 Phase transitions in the coloring of random graphs *Phys. Rev.* E **76** 31131

[45] Hashemzehi R, Mahdavi S J S, Kheirabadi M and Kamel S R 2020 Detection of brain tumors from mri images base on deep learning using hybrid model CNN and NADE *Biocybernetics Biomed. Eng.* **40** 1225

[46] Zheng Y, Tang B, Ding W and Zhou H 2016 A neural autoregressive approach to collaborative filtering *Proc. 33rd Int. Conf. on Machine Learning* vol 48, ed M F Balcan and K Q Weinberger (PMLR) pp 764–73 (available at: https://proceedings.mlr.press/v48/zheng16.html)

[47] Sharir O, Levine Y, Wies N, Carleo G and Shashua A 2020 Deep autoregressive models for the efficient variational simulation of many-body quantum systems *Phys. Rev. Lett.* **124** 20503

[48] Barahona F 1982 On the computational complexity of ising spin glass models *J. Phys. A: Math. Gen.* **15** 3241

[49] Hartmann A K 2011 Ground states of two-dimensional ising spin glasses: fast algorithms, recent developments and a ferromagnet-spin glass mixture *J. Stat. Phys.* **144** 519

[50] Charfreitag J, Jünger M, Mallach S and Mutzel P 2022 McSparse: exact solutions of sparse maximum cut and sparse unconstrained binary quadratic optimization problems *2022 Proc. Symp. on Algorithm Engineering and Experiments* (*ALENEX*), ed C A Phillips and B Speckmann pp 54–66 (available at: https://epubs.siam.org/doi/abs/10.1137/1.9781611977042.5)

[51] Trinquier J, Uguzzoni G, Pagnani A, Zamponi F and Weigt M 2021 Efficient generative modeling of protein sequences using simple autoregressive models *Nat. Commun.* **12** 1

[52] Katzgraber H G, Körner M and Young A P 2006 Universality in three-dimensional ising spin glasses: a monte carlo study *Phys. Rev.* B **73** 224432

[53] Cavaliere A G, Lesieur T and Ricci-Tersenghi F 2021 Optimization of the dynamic transition in the continuous coloring problem *J. Stat. Mech.* **2021** 113302

[54] Krzakala F and Zdeborová L 2008 Potts glass on random graphs *Europhys. Lett.* **81** 57005

[55] Zdeborová L and Krzakala F 2016 Statistical physics of inference: thresholds and algorithms *Adv. Phys.* **65** 453

[56] Kingma D P and Ba J 2014 Adam: a method for stochastic optimization (arXiv:1412.6980)

[57] Bengio Y 2012 Practical recommendations for gradient-based training of deep architectures *Neural Networks: Tricks of the Trade* 2nd edn, ed G Montavon, G B Orr and K-R Müller (Berlin: Springer) pp 437–78