Limits and Performances of Algorithms Based on Simulated Annealing in Solving Sparse Hard Inference Problems

Maria Chiara Angelini[1,2] and Federico Ricci-Tersenghi[1,2,3]

[1]*Dipartimento di Fisica, Sapienza Università di Roma, 00185 Italy*
[2]*INFN-Sezione di Roma 1, 00185 Italy*
[3]*CNR-Nanotec, unità di Roma, 00185 Italy*

The planted-coloring problem is a prototypical inference problem for which thresholds for Bayes-optimal algorithms, like belief propagation (BP), can be computed analytically. In this paper, we analyze the limits and performances of simulated annealing (SA), a Monte-Carlo-based algorithm that is more general and robust than BP, and thus of broader applicability. We show that SA is suboptimal in the recovery of the planted solution because it gets attracted by glassy states that, instead, do not influence the BP algorithm. At variance with previous conjectures, we propose an analytic estimation for the SA algorithmic threshold by comparing the spinodal point of the paramagnetic phase and the dynamical critical temperature. This is a fundamental connection between thermodynamical phase transitions and out-of-equilibrium behavior of Glauber dynamics. We also study an improved version of SA, called replicated SA (RSA), where several weakly coupled replicas are cooled down together. We show numerical evidence that the algorithmic threshold for the RSA coincides with the Bayes-optimal one. Finally, we develop an approximated analytical theory explaining the optimal performances of RSA and predicting the location of the transition toward the planted solution in the limit of a very large number of replicas. Our results for RSA support the idea that mismatching the parameters in the prior with respect to those of the generative model may produce an algorithm that is optimal and very robust.

I. INTRODUCTION

Inference problems are widespread in all scientific disciplines and real-world applications based on data analysis. It is also well known that not all inference problems have the same difficulty, and the solving algorithms need possibly to be tested on the hardest problems. For this reason, focusing on classes of inference problems that are provably very hard is mandatory at the time of testing algorithms. Here we choose to work with inference problems defined on sparse random graphs that present the so-called hard phase that is a range of the signal-to-noise ratio (SNR) where the solution is in principle achievable (by unbounded computational power), but any known algorithm running in a time growing polynomially with the problem size is not able to reach such a solution. Below we define more in detail this class of hard inference

problems; for the moment, it is enough to stress that the presence of the hard phase is related to the so-called information-to-computational gap conjecture. This conjecture, partially proved for some inference problems and some classes of algorithms, assumes that the minimal SNR needed by any polynomial-time algorithm to solve one of these hard inference problems is larger than the information-theoretical lower bound [1–10].

In Fig. 1, we show a schematic phase diagram for a generic inference problem with a hard phase. The derivation of such a phase diagram has been possible working in the very commonly used setting of Bayesian inference known as the teacher-student scenario first introduced in Ref. [11]. In this setting, the teacher generates a random signal $\mathbf{x}$ with the prior probability distribution $P_p(\mathbf{x})$. The signal generated by the teacher is often called the planted configuration. Later, the teacher generates the data from the signal using a probabilistic model P_m. We assume that the data come into the form of a sparse graph G and that the SNR is directly related to the graph mean connectivity c [12]. Applying a probabilistic model to a random signal, we get a random graph G generated with the likelihood $P_m(G|\mathbf{x})$. The teacher provides the student with the prior

FIG. 1. The schematic phase diagram for hard inference problems showing an information-to-computational gap. The SNR here is given by the graph mean degree c. While for $c < c_{IT}$ it is impossible to detect the planted signal, for $c > c_{IT}$ it is in principle possible, but any known algorithm requires a much larger SNR $c > c_{alg}$ to achieve signal detection. The value of c_{alg} depends on the algorithm, and the present work aims to estimate it for simulated annealing algorithms.

P_p, the model P_m, and the data G, asking the student to infer the planted signal **x**. Using Bayes's theorem, the student can easily write down the posterior probability distribution

$$P(\mathbf{x}|G) \propto P_m(G|\mathbf{x})P_p(\mathbf{x}). \qquad (1)$$

The Bayes-optimal estimator $\hat{\mathbf{x}}_{opt}$ of the signal can be obtained by sampling the posterior $P(\mathbf{x}|G)$. A weak recovery of the signal can be achieved if the Bayes-optimal estimator $\hat{\mathbf{x}}_{opt}$ is better than a random guess: This happens for $c > c_{IT}$ that is above the information-theoretical threshold c_{IT}.

In hard inference problems, it is not enough to have a SNR larger than the c_{IT} threshold to recover the signal in a time growing polynomially with the problem size. A larger algorithmic threshold exists c_{alg}, such that only for $c > c_{alg}$ polynomial-time algorithms can actually detect the signal. The threshold c_{alg} clearly depends on the algorithm, and we show how to estimate it for stochastic samplers based on the simulated annealing (SA) algorithm.

When the graph G representing the data is a random graph [14], the posterior can be efficiently sampled by using the Bethe approximation and the belief propagation (BP) algorithm [15]. The algorithmic threshold for the BP algorithm c_{BP} is larger than c_{IT} for inference problems with a hard phase and is nonetheless considered the optimal algorithmic threshold for algorithms running in linear time in the problem size N in the vast majority of the problems [13,16]. Other algorithms which are not Bayes optimal, i.e., not sampling exactly the Bayesian posterior, may have even larger algorithmic thresholds.

Unfortunately, the BP algorithm is known to be not very robust and of limited applicability when the graph G is not locally treelike, as the presence of short loops may avoid convergence to a fixed point and, even at convergence, the estimated marginal probabilities may be far from the exact ones [21]. We are not saying BP can never be used for inference away from random graphs: Indeed, many works are showing the applicability of BP or inference methods strongly related to it on real-world data (see Ref. [22] just as an example). However, having at hand more robust algorithms that can work efficiently both on random graphs

and on more general structured data is of primary importance.

To this aim, it is natural to consider samplers of the posterior probability distribution based on a Monte Carlo Markov chain (MCMC). By construction, a MCMC will eventually sample exactly from the posterior, but its limits and performances are mostly unknown. In some recent works, it has been proved how the algorithmic threshold of some MCMC algorithms is higher than the algorithmic thresholds of other Bayes-optimal algorithms in specific planted problems [23–25]. However, apart from the straightforward statement that MCMC methods are slowed down in a phase where ergodicity is broken in the large-N limit [26], more precise general statements on when and why MCMC-based inference algorithms can find the solution are missing to the best of our knowledge.

MCMC-based inference algorithms could really be ideal if, on top of being robust, they would also be optimal in those cases where the optimal threshold can be analytically computed. To understand this last point, we believe that the study of inference problems defined on random sparse graphs is the perfect framework: Thresholds can be computed analytically thanks to the Bethe approximation for the posterior distribution, and MCMC simulations can be run efficiently on very large problem sizes since a single MC step is linear in N for sparse models.

We study one of the simplest and most common inference problems, namely, the planted-coloring problem (to be defined in more detail below), using simple, but effective, MCMC-based algorithms: SA and replicated simulated annealing (RSA). The main results that we achieve within this framework are the following.

(i) Algorithmic threshold for simulated annealing: We make a conjecture based on firm numerical evidence and strong analytical arguments for the determination of the algorithmic threshold of SA. Our conjecture is based on the comparison of the spinodal point and the dynamical phase transition. Suboptimal with respect to the Bayes-optimal threshold, we find that the SA algorithmic threshold is worse than what has been reported in previous studies.

(ii) Replicated simulated annealing is Bayes optimal: We provide strong numerical evidence that the algorithmic threshold for RSA is extremely close to the Bayes-optimal one of BP. RSA works very well even for a small number of coupled replicas, so the complexity overhead with respect to SA is minimal.

(iii) Nature of the phase transition: We provide both numerical and analytical evidence that coupling replicas changes the nature of the glass transition in the random problem from a discontinuous one to a continuous one. This observation may have important algorithmic consequences, as hard phases are known to exist only in models undergoing discontinuous phase transitions.

(iv) Analytical estimate for the RSA critical temperature: We provide an analytical estimate of the transition temperature in the limit of a large number of coupled replicas for the RSA algorithm. Convergence to this limiting value is fast, so it provides a good estimate of the critical temperature also for a reasonably small number of replicas, those used in practical simulations.

(v) Optimal temperature for a MCMC is not Bayes optimal: For the sake of simplicity, here we study an inference problem without additional noise, i.e., generated from a Gibbs distribution at $T = 0$. We provide analytical evidence that the optimal temperature for recovering the signal is strictly positive and not small at all, so far from the Bayes-optimal one (this is in agreement with previous numerical results).

(vi) Analytical control of finite-size effects: Analytical predictions in the large-N limit are of scarce practical utility if they are not complemented with the study of finite-size effect, which may be very large. We provide an analytical explanation of finite-size effects in the behavior of the SA algorithm, via the study of BP on finite graphs, where similar finite-size effects appear.

Taken all together, the above results provide a strong and coherent theory explaining the behavior of important algorithms based on simulated annealing in solving hard inference problems defined on sparse graphs. We believe this theory has broad applicability, and preliminary results in other planted models support it (however, we leave the study of other problems for future works, the planted coloring being already very interesting and of broad applicability *per se*).

The rest of the paper is organized as follows. In Sec. II, we define the planted-coloring problem and the associated BP algorithm. In Sec. III, we study the performances of SA and put forward a conjecture for its algorithmic threshold. In Sec. IV, we describe the RSA algorithm, and we show that it can reach the threshold of Bayes-optimal algorithms. In Sec. V, we describe the finite-size effects in SA and explain them by studying the BP behavior on single instances of finite size. Finally, in Sec. VI we summarize the results found, the general picture emerging from this work, and possible future implications. We leave to Appendices for more technical computations.

II. THE PLANTED-COLORING PROBLEM

In the present work, we focus our attention on the planted-coloring problem, which was the first planted problem studied in detail [27]. Let us start recalling briefly the coloring problem (both the random and the planted versions) and their phase transitions.

In the random q-coloring problem, a random graph of N vertices and mean degree c is given, and we have to assign one among the q available colors to each vertex in a way to avoid monochromatic edges, i.e., adjacent vertices with the same color. In the thermodynamical limit $N \to \infty$, a typical random q-coloring problem undergoes several phase transitions [28,29]. Before the so-called dynamical transition c_d, there are exponentially many (in N) solutions that all belong to a large single cluster. For $c_d < c < c_c$, with c_c being the condensation transition, the space of solutions splits in an exponential number of different clusters, whose number becomes subexponential for $c > c_c$. A typical random graph is q colorable up to the satisfiability threshold c_s, and for $c > c_s$ there is no solution to a typical random q-coloring problem. The actual values of the different transitions connectivities $c_d \leq c_c \leq c_s$ depend on q and so does the nature of the phase transition: For $q < 4$, the dynamical and condensation transition coincide, leading to a continuous phase transition, while for $q \geq 4$, the transition is discontinuous. This has consequences also on the corresponding planted model.

The planted-coloring problem has been introduced in Ref. [27] and consists in first assigning a random q coloring to the N vertices (this is the planted solution) and then building the random graphs of mean degree c by adding $M = cN/2$ edges connecting randomly chosen pairs of vertices of different colors. The inference problem corresponds to the recovery of the signal (the planted solution) given the random graph G built as explained above. So the mean degree c of the random graph can be seen as a sort of signal-to-noise ratio: For small c, the graph G has exponentially many solutions (colorings), and it is impossible to identify the planted one, while for very large c, the planted coloring is only compatible with the graph G and can potentially be identified.

To achieve a more detailed description of the phase transitions taking place in the planted-coloring problem, we need to discuss the solutions to the BP equations associated with the problem. Calling $\psi_s^{i \to j}$ the probability that the vertex i takes the color s in the absence of the link between i and j, the BP equations read [29]

$$\psi_s^{i \to j} = \frac{1}{Z^{i \to j}} \prod_{k \in \partial i \setminus j} (1 - \psi_s^{k \to i}), \tag{2}$$

where $Z^{i \to j}$ is a normalization imposing $\sum_s \psi_s^{i \to j} = 1$, and $\partial i \setminus j$ indicates all the neighbors of i with the exclusion of j. The BP equations are usually solved iteratively and at convergence provide the marginal probability that variable i takes color s via

$$\psi_s^i = \frac{1}{Z^i} \prod_{k \in \partial i} (1 - \psi_s^{k \to i}), \tag{3}$$

where Z^i is again a normalization constant.

The fixed-point marginals are mainly of two kinds [30]: (i) uniform over the q colors ($\psi_s^i = (1/q), \ \forall i, s$), identifying the paramagnetic (PM) phase and (ii) highly biased

toward the planted solution, identifying the signal recovery or ferromagnetic (FM) phase. The stability analysis of the paramagnetic fixed point toward perturbations in the direction of the planted state reveals that the paramagnetic phase is locally stable for $c < c_{KS} = (q-1)^2$ [27] (in that work, the threshold was called c_l, but we prefer a name making evident the connection with the Kesten-Stigum bound in tree reconstruction). So for $c < c_{KS}$, we expect that BP, if initialized with no information about the signal, will converge to the uninformative PM fixed point, and recovery via BP is impossible. Vice versa, for $c > c_{KS}$ any initial perturbation in the direction of the signal is amplified, and BP will spontaneously converge to the informative FM fixed point, highly correlated with the signal. Thus, c_{KS} is the algorithmic threshold for partial signal recovery via BP.

The procedure described above is called "quiet planting" as it adds the planted solution (and a cluster surrounding it) without changing the structure of solutions to the random problem for $c < c_c$. Since in this region the planted cluster is one among exponentially many others, finding it is impossible. Instead, for $c > c_c$, the planted cluster of solutions dominates the total number of solutions, and the planted solution can be detected in principle. For this reason, $c_c = c_{IT}$ is called the information-theoretical threshold.

Whether the planted solution can be actually found above c_{IT} depends on the nature of the phase transition [13]. In continuous phase transitions, we have $c_{IT} = c_{KS}$ and so, as soon as the planted cluster dominates the total number of solutions, BP can actually detect it. This happens in the planted q-coloring problem for $q < 4$. On the contrary, if the transition is discontinuous, we have a hard phase for $c_{IT} < c < c_{KS}$ where the planted solution is in principle detectable, but signal recovery via BP is impossible due to the stability (and the strong attraction) of the PM fixed point. A hard phase exists for $q > 4$ in the planted q-coloring problem. The case $q = 4$ has remained elusive for a while and was finally solved in Ref. [31].

In the present work, we focus on the case $q = 5$ that is an inference problem with a well-defined hard phase. The threshold values for $q = 5$ are $c_d = 12.837(3)$, $c_c = c_{IT} = 13.23(1)$, and $c_{KS} = 16$ [27,29]. Despite a great effort to break this hard phase, no polynomial algorithm is known that can detect the planted solution for $c < c_{KS}$.

Let us finally discuss a delicate point. By construction, the BP algorithm relies on the replica-symmetric (RS) approximation, assuming that most of the solutions form a single and well-connected cluster [15]. Consequently, the BP fixed points can describe only RS phases, like the PM and FM phases discussed above. However, from the solution of the random model, we know that for $c > c_d$ the space of solutions splits into many different clusters: This is the so-called glassy phase, as it induces a very strong slowing-down in many algorithms, and its description requires us to assume replica symmetry breaking

(RSB) [32]. In the planted graph, the planted cluster also starts to be locally stable at $c = c_d$, exactly as the glassy states. For $c_d < c < c_c$, the planted cluster shares the same properties as the random glassy states, while for $c > c_c$, its internal entropy becomes larger than the entropy of the glassy states, and it starts to be detectable. These glassy states, which are so important in the description of the random model, play no role in the BP behavior when it is applied to the planted case, and the performance of the BP algorithm is not improved by taking into account the glassy nature of the hard phase [26].

We have already mentioned that BP is Bayes optimal: It can find optimal solutions for $c > c_{KS}$ because it assumes the perfect knowledge of the model. Indeed, Eq. (3) assumes that we know that there is a hard constraint imposing the absence of links between nodes of the same color in the construction of the graph around the planted solution. It can be shown that, imposing the perfect knowledge of the model, there cannot exist a RSB phase, and the only possible transition is a usual ferromagnetic first-order transition between the PM and FM phases. In statistical mechanics language, assuming the perfect knowledge of the model corresponds to be on the so-called Nishimori line [33], where the RS assumption holds. So, in the planted-coloring problem, thanks to the perfect knowledge of the generative model, only the PM and FM phases can dominate the measure over the solutions. The first-order transition takes place at c_{IT}, where the free energies of the two phases are equal, with the FM free-energy being lower for $c > c_{IT}$ (thus, making the signal detectable in principle). The spinodal points are c_d and c_{KS}, as the FM planted state appears at c_d and the PM state becomes unstable at c_{KS}.

Leaving BP aside and considering broader classes of algorithms, it is very likely that one has to consider situations where algorithms do not satisfy the Nishimori condition: either because they have no perfect knowledge about the model or because even knowing the model, they go out of equilibrium, thus violating the Nishimori condition. In all of these cases, the glassy states—so important in the random model—should be taken again into account (even if they are irrelevant in the thermodynamic RS description of the planted model). We see in the next sections that they are particularly important to determine the behavior of MCMC-based solvers that do not assume the perfect knowledge of the model. This is an ideal example showing that by moving away from equilibrium statistical mechanics to out-of-equilibrium processes, one can find a much richer physics that requires a more complex description.

III. SIMULATED ANNEALING FOR THE PLANTED-COLORING PROBLEM

In this section, we introduce SA as a solver for the planted-coloring problem, and we analyze its performance.

SA is a MC-based algorithm introduced in Ref. [34], and it is widely used in many optimization problems, including the random coloring problem [35,36]. In the context of the planted-coloring problem, it was considered in Ref. [27], where the authors stated it can find the planted solution in the whole region $c > c_{KS}$. We show that this is not in agreement with our numerical results and propose a better conjecture.

Given a configuration s of colors of the nodes, in a statistical mechanics approach, we associate with it an energy counting the number of monochromatic edges

$$H(s) = \sum_{(i,j) \in E} \delta_{s_i, s_j}, \qquad (4)$$

where E is the edge set of the graph. We then introduce a temperature $T \equiv 1/\beta$ in the model and a Gibbs-Boltzmann-like probability measure on the configurations

$$P_{GB}(s) \propto e^{-\beta H(s)}. \qquad (5)$$

In the $T = 0$ limit, this probability measure becomes the uniform distribution over the set of solutions (proper colorings), which is always nonempty thanks to the presence of the planted solution s^* with $H(s^*) = 0$. Moreover, the $T = 0$ distribution coincides with the posterior $P(s|G)$ if the prior is uniform over all configurations. This indicates that the Bayes-optimal temperature is $T_{Bayes} = 0$. One could then naively conclude that the best strategy for any inference algorithm should be the sampling of the $T = 0$ distribution to detect s^*. We see that the optimal temperature for MC-based algorithms is different from the Bayes-optimal one $T_{Bayes} = 0$.

The meaning of the distribution $P_{GB}(s)$ with $T > 0$ is to relax the hard constraints into soft ones: Configurations with monochromatic edges are admitted, but with a probability that is more suppressed the lower the temperature is. We start our SA from a random configuration of colors and at sufficiently high T. At each step of the SA algorithm, we decrease the temperature by dT and we do a Monte Carlo sweep (MCS) that corresponds to the attempt to update the color of each of the N variables following the usual Metropolis rule: In practice, we change the color s_i to s_i', chosen at random between the q possible colors, with probability 1 if in this way the number of monochromatic edges decreases, while we update it with probability $e^{-\beta n}$ if the number of monochromatic edges increases by n after the update.

In Fig. 2, we show the intensive energy $e = H/N$ of the configurations visited by SA with parameter $dT = 10^{-7}$ in ten samples of size $N = 10^5$ of the planted-coloring problem with $q = 5$. The differences between samples are negligible (but at $c = 18$), while the behavior of SA changes drastically with the mean degree c.
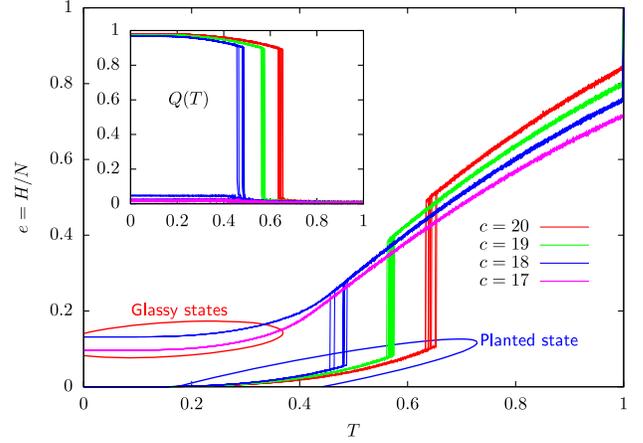
FIG. 2. The energy density as a function of the temperature in simulated annealing for planted coloring with $q = 5$, $N = 10^5$, and $dT = 10^{-7}$. For each value of the mean degree c, we show the results for ten different samples. For $c \lesssim 18$, SA does not find the planted solution and gets trapped in glassy states. Inset: overlap with the planted configuration as a function of the temperature (same parameters as in the main figure).

For $c > 18$, SA finds a configuration highly correlated with the planted solution s^* on each sample. This is confirmed also by the data in the inset showing the overlap between the configuration s reached by SA and the planted solution s^* defined as

$$Q = \frac{\max_{\pi \in S_q} \sum_i \delta_{s_i^*, \pi(s_i)}/N - 1/q}{1 - 1/q}, \qquad (6)$$

where S_q is the group of permutations of q elements. The overlap defined in this way is null for a random guess, while $Q = 1$ if there is a perfect recovery of the planted solution.

On the contrary, for $c < 18$, SA is unable to reach the planted solution in any sample, and it gets trapped in glassy states, which are metastable states (local minima of higher energy) orthogonal to the planted state ($Q \simeq 0$). So the numerical data strongly suggest that $c_{SA} \simeq 18$ is the algorithmic threshold for SA. Indeed, exactly at $c = 18$, we see in Fig. 2 that the behavior of SA is sample dependent: A typical behavior that is often observed at a spinodal point.

The strict inequality $c_{KS} < c_{SA}$ implies SA is not Bayes optimal, and there exists a region $c \in (c_{KS}, c_{SA})$ where BP performs better than SA.

Let us comment briefly on the choice of N and dT. We study several values of N and dT, all providing data consistent with the claims above, as long as dT is small enough. In Appendix B, we present the data for different N and dT values, showing that for $c > c_{KS}$ the SA algorithm can find solutions with high probability if $dT = O(N^{-a})$ with $a \simeq 1$. This means that SA is a solving algorithm running in a time scaling roughly quadratically with the

problem size N. We decide to show only data for $N = 10^5$ since this value is large enough that fluctuations are strongly suppressed.

Data in Fig. 2 clearly show that a dynamical first-order transition is taking place at the temperature where SA makes a jump to reach the planted cluster of solutions. Moreover, when such a dynamical first-order transition does not take place (i.e., for $c < c_{\text{SA}}$) the overlap remains null, and SA stays very far from (actually orthogonal to) the planted solution.

To explain why SA is not able to find the planted solution for $c < c_{\text{SA}}$, we need to look to the extension to the finite temperature of the phase transitions at $T = 0$. How the thresholds c_d and c_{KS} change in temperature was already computed in Ref. [27] (more details in the Appendix A): We report them with the names $T_d(c)$ and $T_{\text{KS}}(c)$ in Fig. 3. For $T < T_d$ glassy states appear, while for $T < T_{\text{KS}}$ the paramagnetic solution is no more stable with respect to perturbations toward the planted solution.

The SA protocol starts from very high temperatures and keeps cooling very slowly (ideally, in an adiabatic way). Which critical line (either T_d or T_{KS}) it will find first depends on the c value. Defining the SA algorithmic threshold c_{SA} as the crossing point of these two critical lines

$$T_d(c_{\text{SA}}) = T_{\text{KS}}(c_{\text{SA}}), \tag{7}$$
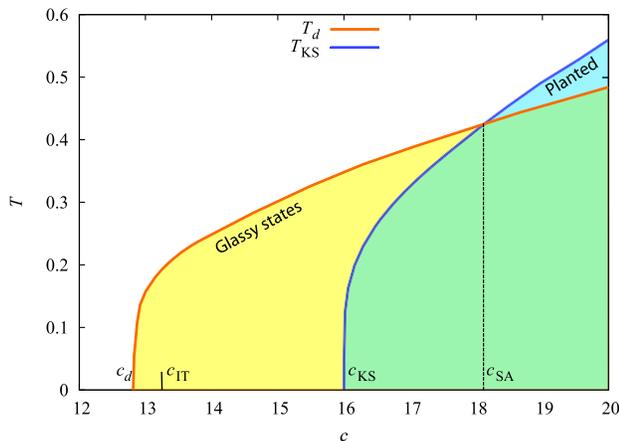
FIG. 3.   Phase diagram in the c-T plane for the planted-coloring problem with $q = 5$. $T_d(c)$ is the critical line below which glassy states appear, while below $T_{\text{KS}}(c)$ the paramagnetic solution becomes unstable toward the planted solution. The information-theoretical threshold c_{IT} corresponds to c_c, while we claim that SA is not able to find the planted solution for $c < c_{\text{SA}}$. We can extract c_{SA} as the connectivity at which $T_d(c) = T_{\text{KS}}(c)$. Both in the yellow and the green region, a MCMC algorithm gets trapped in glassy states, and the only region in which it can nucleate the planted solution is the blue one. On the other hand, the BP algorithm finds the paramagnetic solution in the yellow region, while it succeeds to find the planted solution both in the blue and in the green one.

we conjecture the following scenario. If $c > c_{\text{SA}}$, decreasing T the paramagnetic phase becomes unstable with respect to the planted state, and SA jumps into the planted cluster, thus achieving perfect recovery in the $T = 0$ limit (the planted state is very stable at low temperatures, and SA cannot leave it when glassy states appear below T_d). Conversely, if $c < c_{\text{SA}}$, decreasing T the paramagnet first becomes unstable toward the glassy states at T_d. We remind that for $T > 0$ the Nishimori condition is not satisfied, and so glassy states are not only possible but likely to play a central role. And indeed, we observe SA falling inside glassy states for $c < c_{\text{SA}}$. The barriers to escaping these glassy states are extensive, and SA cannot jump anymore to the planted solution, even if the temperature is decreased below T_{KS}: Once the dynamics is trapped by a glassy state, it will stay there for a time that is exponential in the size of the system.

Based on the arguments above, the conjecture that c_{SA} is the algorithmic threshold for SA is very plausible, on top of being fully supported by numerical observations.

IV. REPLICATED SIMULATED ANNEALING

Having understood that the SA algorithmic threshold c_{SA} is larger than the BP one c_{KS} because of the existence of glassy states than can trap the SA algorithm, a natural question is whether another MC-based algorithm could do better than SA. In this section, we study the RSA introduced in Ref. [37] and provide strong numerical evidence that its algorithmic threshold c_{RSA} matches c_{KS}, thus being as good as the Bayes-optimal BP.

The idea beyond RSA is to simulate y replicas (copies) of the system coupled together with ferromagnetic coupling (i.e., replicas prefer to be in similar configurations). The corresponding energy function is the following:

$$H(s) = \sum_{a=1}^{y} \sum_{(i,j) \in E} \delta_{s_i^a, s_j^a} - \frac{\gamma}{2y} \sum_{a \neq b} \sum_{i=1}^{N} \delta_{s_i^a, s_i^b}, \tag{8}$$

where γ controls the intensity of the ferromagnetic coupling between replicas that we decide to scale by a factor y in order to get a well-defined limit when $y \to \infty$. The idea of simulating different replicas is not new in the use of SA for inference problems [38], and the correlation among replicas was also used to modify the underlying graph [39]. The novelty in the RSA is that the different replicas do not evolve independently but are coupled together. This is a practical way to probe states that dominate not the original free energy, but a modified free energy where the contribution of local entropy is enhanced.

From a practical point of view, RSA works exactly as SA: We start at a high enough temperature from random independent configurations for each replica. Each MCS consists in the attempt to change the color of each variable in each replica, according to the energy in Eq. (8). After

each MCS, the temperature is decreased by dT. In all our numerical experiments, we set $\gamma = 1$ (a complete study of the optimal value of γ could be performed, but it is beyond the scope of the present work).

The lowering of the temperature forces the system to look for lower-energy configurations, both reducing the number of monochromatic edges in each replica and increasing the similarity between replicas. In this way, RSA should converge toward low-energy states of large entropy, as larger clusters of solutions can more easily accommodate $y \gg 1$ replicas. The reason why this should favor the planted cluster of solutions against the glassy states, is that beyond c_c the planted cluster has the largest entropy: In fact, glassy states are many, but the entropy of each glassy state is relatively small, such that their total entropy is not larger than the one of the states dominating the thermodynamics. This picture is not limited to the present problem and generalizes to other inference problems where the signal competes with many more random "glassy" configurations.

In Fig. 4, we show the behavior of RSA for $N = 10^5$ and $c = 17$. We report data for only one value $dT = 10^{-7}$, but in Appendix B we show that it is enough to have $dT = O(N^{-a})$ with $a \simeq 1$ to recover the signal. The curves for $y = 1$ are the same shown in Fig. 2 reporting the inability of SA in finding the planted solution. Instead, RSA can find the planted solution as long as $y > 1$. Surprising enough, already the smallest value $y = 2$ seems very effective in the task of detecting the planted signal, with nine of the ten samples shown in the figure reaching a perfect recovery. The temperature where detection takes place that corresponds to a jump in the energy fluctuates a

little bit among the samples, while for larger values of y, the behavior of RSA is practically sample independent and seems to follow an almost deterministic law. A possible explanation is that the PM state in the replicated model is becoming unstable at the temperature where the jump takes place: If this happens, the RSA dynamics has no other option than to flow toward the planted solution, thus eventually detecting the signal in the $T = 0$ limit.

To test the above hypothesis, we compute $T_{KS}(c)$ in the replicated planted model. Moreover, to test the conjecture of the previous section locating the algorithmic threshold c_{RSA} for RSA, we compute also $T_d(c)$ in the replicated random model. The details of these computations are presented in Appendices D and E, and the main results are summarized in Fig. 5 for $y = 2$, 4, 8, 16, 32. It is immediately evident that the critical lines for $y > 1$ are located at much higher temperatures than those for the original model ($y = 1$) shown in Fig. 3, and these critical temperatures are indeed very close to the temperature where the jump takes place. In particular, the vertical dashed line in Fig. 4 marks the value of T_{KS} in the $y \gg 1$ limit, which turns out to be a very accurate analytical prediction for the jump location (i.e., the signal detection) in the RSA run with $y \gg 1$ and a very good approximation also for finite values of y. We see that the sample-to-sample fluctuations in T_{KS} are larger in the case of small y and become almost null in the large-y limit, again in perfect agreement with the observation of the temperature of the jumps for RSA. Moreover, the critical lines $T_{KS}(c)$ and $T_d(c)$ shown in Fig. 5 for $y > 1$ are extremely close.

FIG. 4. The energy density as a function of the temperature in the replicated simulated annealing for planted coloring with $q = 5$, $N = 10^5$, $dT = 10^{-7}$, and $c = 17$, changing the number y of replicas. For each value of y, ten different samples are simulated. For $y = 1$ SA does not find the planted solution, while RSA finds it as long as $y > 1$ RSA finds it. The vertical line represents T_{KS} as found by the linearization of BP equations around the PM solution in the limit $y \to \infty$.

FIG. 5. T_{KS} (full lines) and T_d (dashes lines) as a function of c in the replicated model with $y = 2$, 4, 8, 16, 32. Data are averaged over ten samples of size $N = 10^6$ (which is large enough to avoid any significant finite-size effects). Inset: The difference $T_{KS} - T_d$ is plotted to help locate the crossing point that defines the algorithmic threshold c_{RSA}, perfectly compatible with the BP threshold $c_{KS} = 16$. Below this threshold, the difference $T_{KS} - T_d$ is compatible with zero, thus making plausible the scenario where the temperature at which the paramagnetic state loses its local stability is the same in the planted and the random ensemble.

Indeed, we report in the inset of Fig. 5 their difference to help the reader in locating their crossing point that defines the algorithmic threshold c_{RSA} (according to the conjecture put forward in the previous section). Within the statistical uncertainties, the difference $T_{KS} - T_d$ does not depend on y and becomes null at $c_{RSA} \simeq 16$, perfectly compatible with the algorithmic threshold $c_{KS} = 16$. Thus, RSA and the Bayes-optimal BP algorithm are observed to share the same (conjecturally optimal) algorithmic threshold. Please note that below c_{RSA} the difference $T_{KS} - T_d$ is compatible with zero: For these connectivities the temperature at which the paramagnetic state loses its local stability is the same in the planted and the random ensemble. As long as $y > 1$, changing the number of replicas y does not seem to lead to improvements, as we also see directly from the RSA simulation (see Fig. 4). The best choice is thus to use a small-y value, even $y = 2$, because the time for running RSA scales as $O(yN)$.

Let us finally discuss an important difference between the original model and the replicated one. For $y > 1$, in the random model, the transition toward the glassy states becomes continuous. So, while in the original model ($y = 1$) glassy states at T_d are already well-formed states and become dominant with a discontinuous transition, in the replicated model ($y > 1$) the glassy states arise exactly at T_d with a continuous transition (a plot showing how the order parameter changes introducing replicas is presented in Appendix E). This observation allows us to compute T_d very easily for $y > 1$ via the linearization of the BP equations in the random model, similar to the computation of T_{KS} in the planted model (more details in Appendices C, D, and E). The consequences of the changing of the order of the transition on the existence of hard phases are discussed in the concluding paragraph.

V. FINITE-SIZE EFFECTS

Finite-size effects are sizeable only close to the critical point. As a typical example, we show in Fig. 6 the behavior of SA at connectivity $c = 17 < c_{SA}$ for sizes $N = 10^3$, $10^4, 10^5$. At first sight, the behavior of SA seems strange enough: It can find the planted solution for $N \leq 10^4$, even if the PM state should be stable given that we are running SA with $c < c_{SA}$. As long as the PM is locally stable, the barrier one has to cross to leave it is $O(N)$, and thus the corresponding timescale should be $O(\exp(N))$, definitely a huge number for the sizes $N = 10^3, 10^4$ shown in Fig. 6.

How can we explain this observation? Can we achieve an analytical prediction, at least qualitatively, of the above finite-size effects? A clear and simple answer comes from the study of the critical values T_{KS} and T_d on samples of a given finite size. Indeed, both critical values can be computed by running BP on a given sample: T_{KS} is found by analyzing the stability of the PM state in the planted model via the linearization of BP equations around the
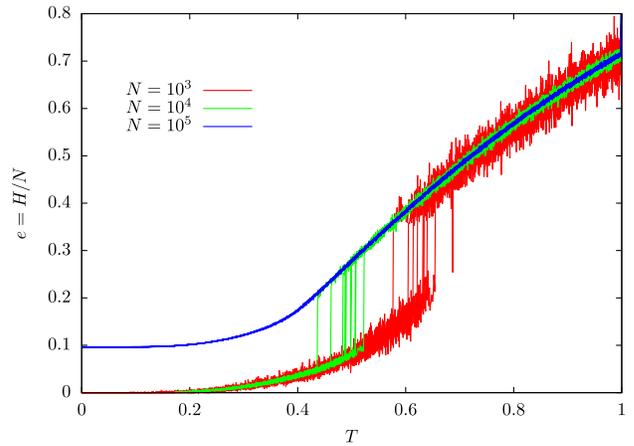
FIG. 6. The energy density as a function of the temperature during simulated annealing for planted coloring with $q = 5$, $c = 17$, $dT = 10^{-7}$, and different problem sizes. For each value of N, ten different samples are simulated. For $N = 10^5$, SA does not find the planted solution, while it always does for smaller sizes.

paramagnetic fixed point; T_d is found by running BP on the random model initialized in a planted glassy state at a given temperature and looking at the temperature at which the glassy state becomes unstable toward the paramagnet. More details are given in Appendix E.

We cannot compare directly T_{KS} and T_d for a given sample, because, as explained exhaustively in Appendices D and E, we are looking at the ensemble of graphs planted at $T = 0$ to compute T_{KS} and to the ensemble of graphs planted at $T > 0$ to find T_d. However, to compare the two, we can reasonably assume that the finite size is effectively changing the mean degree c. T_d and T_{KS} being both monotonic functions of c, we can assume that the sample that has the highest T_d will also have the highest T_{KS}. Following the same reasoning, we order values of T_{KS} and T_d and pair them following their order. For each pair, we can now look to the $\max(T_{KS}, T_d)$, because we conjecture that the highest of the two is the one determining the fate of SA in the search for the planted signal. In Fig. 7, we show the histograms of T_{KS} (filled curves) and T_d (empty curves) for different sizes at $c = 17$. The data are thus in perfect agreement with our claim that SA can find the planted state only if $\max(T_{KS}, T_d) = T_{KS}$, and this happens with a high probability for $N = 10^3$ and 10^4, while we never observe it for $N = 10^5$.

The reason beyond this unexpected behavior is due to the very different finite-size effects in T_{KS} and T_d: While the latter deviates slightly from the value it takes in the thermodynamic limit, we observe in T_{KS} substantial deviations, very biased toward larger temperatures. As a consequence, the PM state gets destabilized at higher temperatures in samples of finite size allowing for signal detection (a very favorable effect of finite-size effects).
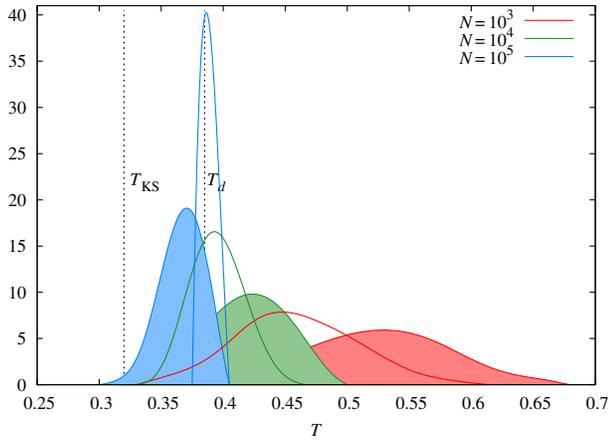
FIG. 7. Histograms of T_{KS} (filled curves) and T_d (empty curves) obtained by running linearized BP equations on given instances of different sizes at $c = 17$ (more details on the computation in the text). Ordering values of T_{KS} and T_d at a given N and pairing them following their order, for $N = 10^3, 10^4$, $\max(T_{\mathrm{KS}}, T_d) = T_{\mathrm{KS}}$ for almost each pair of temperatures, while for $N = 10^5$ it always happens that $\max(T_{\mathrm{KS}}, T_d) = T_d$ (and this is the reason why SA is not able to reach the planted state for $N = 10^5$). Vertical dashed lines represent T_{KS} and T_d in the thermodynamic limit.

VI. DISCUSSION AND PERSPECTIVES

In this work, we analyze the limits and the performances of simple, but efficient, algorithms based on a MCMC (i.e., SA and RSA) in solving the planted-coloring problem on sparse random graphs. This problem is a well-understood inference problem (in the teacher-student scenario) showing an information-to-computational gap: The planted solution can in principle be detected for $c > c_{\mathrm{IT}}$, but any known polynomial algorithm can find it only for $c > c_{\mathrm{alg}} \geq c_{\mathrm{KS}} > c_{\mathrm{IT}}$.

For planted problems defined on random graphs and for which the generative model is known—two conditions not always easy to meet—the BP algorithm is Bayes optimal and can find the planted solution as soon as $c > c_{\mathrm{KS}}$. The BP algorithm works at the same temperature used in the planting process ($T = 0$ in the present model), and for this reason, it satisfies the Nishimori condition that keeps the algorithm away from glassy states: Indeed, in glassy states, the replica symmetry is spontaneously broken, while the Nishimori condition implies the replica symmetry must be preserved. As a consequence, BP is the optimal one among a large class of message-passing algorithms: As shown in Ref. [26], generalizing BP to include RSB effects leads to the worst algorithmic performances (probably due to the presence of the uninformative glassy states that we have seen trapping the MCMC algorithms).

Willing to understand the more general situation where algorithms are run with a temperature T different from the planting temperature used in the problem generation ($T > T_p = 0$ in our case), we need to abandon the

Nishimori condition. In this situation, the glassy states, which are uninformative from the point of view of the inference problem, can play a key role, influencing negatively the performances of the algorithms. We show in the present work that algorithms based on a MCMC, like SA, can indeed be trapped by glassy states and fail to detect the signal, even in the region where the FM state has lower energy with respect to the PM state.

Decreasing the temperature in a SA (or RSA) simulation, the system faces a critical choice, which is schematically depicted in Fig. 8: When it leaves the PM state, it can either flow to the FM state (and thus detect the planted signal) or get trapped in glassy states (which are uncorrelated to the planted signal). In the latter case, the FM state becomes unreachable, even if it has lower energy because the dynamics in the glassy states is extremely slow, and the barriers (both energetic and entropic [40]) make paths leaving the glassy states extremely improbable.

The above observations allow us to make a conjecture on the location of the algorithmic threshold for MCMC-based algorithms. In the literature, there is not a clear indication on which is the correct threshold for these algorithms: While some works claim that MCMC solvers can reach the BP threshold [27], some other references suggest that indeed glassy states negatively influence the performances of the non-Bayes-optimal algorithms [26,41–44] but do not provide any estimate of the actual algorithmic threshold.

In this work, we conjecture that the fate of a MCMC-based annealing algorithm depends on which critical line it meets before. If the PM spinodal line $T_{\mathrm{KS}}(c)$ is met first, the MCMC evolution spontaneously leaves the PM state heading toward the FM states (the only existing before the appearance of glassy states) and detecting the planted signal. On the contrary, if the dynamical critical line
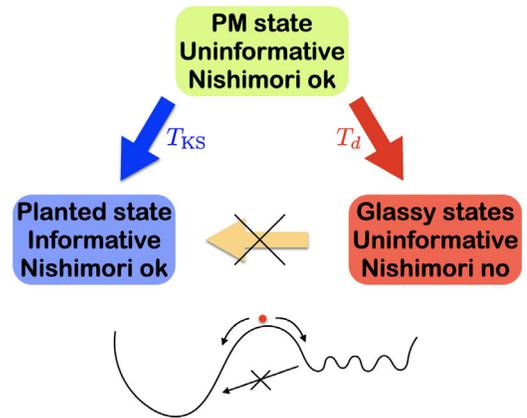
FIG. 8. A schematic picture of the general scenario faced by an annealing process looking for the planted signal. Decreasing the temperature, the fate of the annealing algorithm is determined by which critical temperature is met first between T_{KS} and T_d (as explained in the text). The lower diagram should remind us that once the system gets trapped in glassy states it cannot reach the planted signal anymore due to the large barriers.

$T_d(c)$ is met first, glassy states become the most attractive for the MCMC evolution, and the system gets trapped in glassy states forever. We see that the above conjecture predicts with high accuracy the algorithmic thresholds measured in numerical simulations for both SA and RSA. For SA, we find that the algorithmic threshold is suboptimal ($c_{SA} > c_{KS}$), thus correcting previous claims of optimality [27]. For RSA, instead, we find an optimal algorithmic threshold ($c_{RSA} \simeq c_{KS}$).

The optimality of RSA is a very good news: Being a MCMC-based algorithm, it is extremely robust and can be used with confidence also in more structured problems, like those coming from realistic applications, where message-passing algorithms are likely to fail. The RSA algorithm has been introduced in Ref. [37] as a smart way to sample configurations with large local entropy. To the best of our knowledge, it was never used before in inference problems. We adopt it with the following rationale: According to the scenario depicted above, the inference process can be seen as a competition between the single FM planted state and the exponentially many glassy states, the two competitors having similar chances of attracting the dynamics close to the algorithmic threshold. Given that the glassy states are many, we expect the basin of attraction of a single glassy state to be much smaller than the one of the FM planted state. For this reason, in the replicated system, the y copies are more likely to be attracted by the FM state having a larger basin than by any single glassy state (while being distributed over several glassy states is not convenient because of the coupling among replicas).

In other words, the improvement of RSA over SA is due to the planted state being different from glassy states (and this is often the case in inference problems), but it cannot be extended straightforwardly to other types of problems. For example, in a recent paper we compared the performances of SA and RSA for an optimization problem, the largest independent set problem on sparse random graphs [45], and in that case, we did not find a great improvement in the replicated version. However, in hard optimization problems, one is often trying to find a solution that lies inside a glassy state (e.g., in the clustered phase of constraint satisfaction problems [29,46,47]). Thus, if the competition is taking place between glassy states of similar internal entropy, the replicated version of the algorithms does not lead to any great improvement (as we saw in the largest independent set problem [45]). So, it is worth stressing that the very good performances of RSA in inference problems [48] are mainly due to the internal entropy of the planted state being larger than the one of each glassy state, and this difference is general, being the planted state the one with the lowest energy and being the basin of attraction larger for lower-energy states (see, e.g., the discussion in Appendix A of Ref. [40]).

We study annealing algorithms as they represent the safer way to test a broad range of temperatures: Any good practitioner would start running SA if they know anything about the model to be optimized, and the output of SA would already provide a rough but useful indication of the critical temperature region to be better explored. However, once one gets a more detailed picture of the energy landscape (as we do here for the planted-coloring problem), the natural question is "Is the optimal temperature to run a MCMC eventually the replicated one?" The answer is clear from Figs. 2 and 4: The optimal temperature for the MCMC is the one where the jump to the planted state takes place (let us call it T_{jump}). Indeed for $T > T_{jump}$, the MCMC is stacked in the PM phase, while decreasing the temperature the dynamics becomes slower, and the time for doing the jump increases (and even eventually diverges if glassy states with extensive barriers appear).

Noticing that in all cases where the signal can be detected by MCMC-based algorithms we have $T_{jump} > 0$, we can straightforwardly conclude that the optimal MCMC temperature is not the Bayes-optimal one. A similar conclusion has been achieved recently for MCMC-based algorithms solving the planted clique problem [24,49] and in the rank-one matrix-estimation problem [50]. So, it seems a general statement that, once we consider out-of-equilibrium processes solving an inference problem, the Bayes-optimal temperature plays a marginal or null role, and the optimal temperature is typically higher (larger thermal fluctuations enhance the exploration of the configuration space). This is reminiscent of what has been found for Langevin dynamics in another inference problem, the spiked matrix-tensor model: In that case, the Langevin algorithm is suboptimal with respect to message-passing algorithms, due to the presence of glassy states, and it is computationally advantageous for it to mismatch the parameters to find the planted solution of the problem [42–44].

The mismatch between the optimal MCMC temperature and the Bayes-optimal temperature is a good news for all those situations when the generative model is not known and matching the planting temperature would be simply impossible by ignorance. So, having robust algorithms that can work even without knowing exactly the properties of the generative model is extremely useful, and the only requirement is to have an idea of the optimal temperature at which the algorithm should be run (the jump temperature in the present case).

We attempt to get an analytical estimation of the jump temperature T_{jump} by checking the stability of the PM state in the replicated model (the details of the computation are in Appendix D). However, the replicated model has many short loops, and the Bethe approximation is no longer exact (one should consider supervariables involving the y replicas and taking q^y values, but we leave this computation for future work). Luckily enough in the large-y limit, the interaction among replicas becomes weak (we scale it by $1/y$ to get a well-defined limit for $y \to \infty$), and the Bethe approximation is again valid on a single graph. In the $y \gg 1$ limit, we can thus obtain an estimate of T_{jump} (it is reported in Fig. 4 with a vertical dashed line) that compares

extremely well with numerics even for smaller values of y. This is, in our opinion, a very relevant achievement as predicting the dynamical behavior of out-of-equilibrium algorithms is typically very challenging, and the cases where this can be put in connection with the thermodynamical phase diagrams (as those shown in Figs. 3 and 5) are very few, especially for sparse models.

A final comment is about the nature of the transitions in the replicated system. We already pointed out that the transition from the PM state toward the glassy states changes its nature adding replicas, passing from a discontinuous transition for $y = 1$ to a continuous transition for $y > 1$. This could sound strange and unexpected; however, this phenomenon is reminiscent of what happens in other disordered systems: Let us consider the formulation of the problem where the y-coupled variables on a single site form a supervariable taking q^y values. The supervariable is subject to a local field that favors configurations where the y replicas are similar. It is well known that disordered models undergoing a discontinuous replica symmetry-breaking transition (e.g., the famous p-spin model [51]) can change their nature to a continuous transition in presence of an external field. In the case of the y replicas, the change in the nature of the transition is caused by a self-induced internal field. This change in the type of transition should lead us to rethink the whole concept of the hard phase in inference problems, which is fundamentally grounded on the discontinuous nature of the phase transition. Indeed until now, any known model whose random version shows a discontinuous transition, has a hard phase in its planted version, while any model that shows a continuous transition in its random version does not have a hard phase in its planted version. The phase diagram of a model with a discontinuous transition has the same form as the one we show in Fig. 3 for the coloring problem. In the upper panel in Fig. 9, we show the typical phase diagram for a model with a continuous transition: The PM state loses stability as long as the glassy states or the planted state become locally stable, respectively, in the random graph or in the planted graph. The temperature at which the PM state becomes unstable toward the planted state is always higher than the temperature of the instability toward a random glassy state in a planted graph: As a result, there is no hard phase, because it is possible to recover the planted state as long as it becomes locally stable, both for a BP or for a MCMC algorithm. The phase diagram for the replicated model is instead different as shown in the lower panel of Fig. 9: Below a certain connectivity (that in the case of the coloring problem, we find numerically to coincide with c_{KS}) the temperature at which the PM state becomes unstable toward the planted state coincides with the one at which it becomes unstable toward the glassy states (in other words, the two critical lines merge). Moreover, as long as the glassy states become locally stable, both a MCMC algorithm and the BP algorithm get
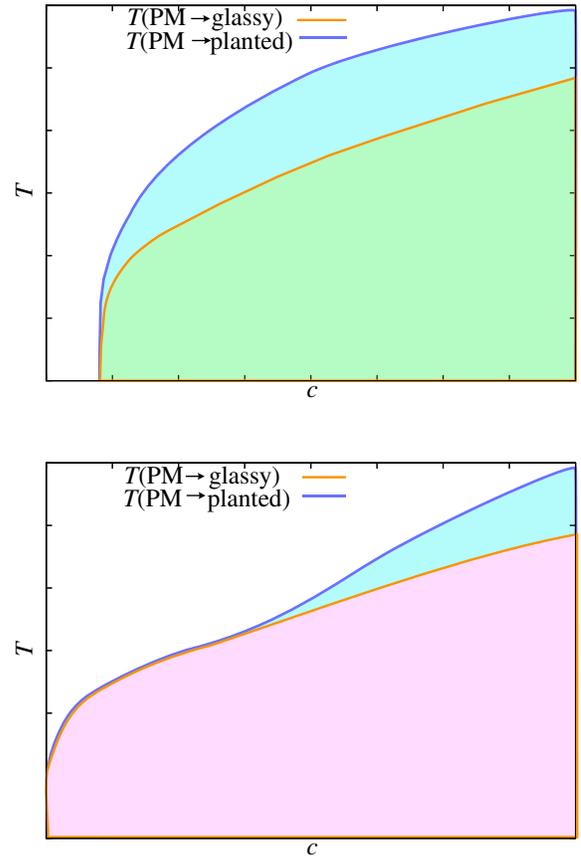
FIG. 9. Qualitative phase diagram for a "standard" model with a continuous transition (upper panel) and for the replicated coloring model (lower panel). While for a standard continuous model, the temperature at which the PM state becomes unstable toward the planted state is always higher than the temperature of the instability toward a random glassy state, for the replicated model below a certain connectivity, the two temperatures coincide. As in Fig. 9, in the blue regions both a MCMC algorithm and BP can nucleate the planted solution, while in the green one, MCMC algorithms get trapped in glassy states while BP is able to find the planted one. In the diagram of the replicated model, a new previously unseen region, colored pink, appears where both a MCMC algorithm and BP are trapped by a glassy state.

trapped by them and do not find the planted state. In this situation, the hard phase is present but its nature is different from the one arising in discontinuous models: While for discontinuous models, the hard phase is caused by the local stability of the PM state and the planted state is retrievable as soon as the PM state becomes locally unstable, in the replicated model, the continuous transition destabilizing the PM state may lead to the concurrent formation of glassy states (for $c < c_{KS}$ in our model) that trap the algorithms and make the planted state not retrievable.

At this point, one could ask why glassy states are not a problem for BP in the nonreplicated models, both discontinuous and continuous (in the green region of both Figs. 3 and 9 glassy states are stable but BP does not "see" them) while they trap the BP algorithm in the replicated case. In

the nonreplicated models, the number of glassy states goes to infinity in the thermodynamic limit, and their basin of attraction is very narrow: BP randomly initialized is not able to enter any of them. The introduction of replicas changes the free energy of the model, performing a sort of weighted average over the states, reducing the number of glassy states and enlarging their basin of attraction (for a pictorial representation of this phenomenon see Fig. 1 of Ref. [37]): In this way, they become accessible by the BP algorithm. This phenomenon should depend on the parameters y and γ and we leave its complete characterization for future works [52]. Please note that the PM state, which is trapping BP in the hard phase in the nonreplicated model, can be viewed as the white average over all the underlying glassy states [26]. When replicas are added, the role of PM trapping BP is taken by some new glassy states that can be viewed as a weighted average over the glassy states of the nonreplicated original model.

In conclusion, what is the most robust, eventually optimal, and easy-to-use algorithm to solve hard inference problems? We see that for the coloring problem, RSA with few replicas, say, $y = O(10)$, is the best option. It is more robust than BP, and it can be applied essentially to any kind of graph. Its running time is $O(yN)$, so only a factor y slower than the simpler SA, which is the first option for any practitioner. Maybe parallel tempering (PT) could be better than RSA, but it is more difficult to use and requires a preprocessing operation where the temperature schedule has to be optimized. Moreover, RSA can be restricted to a small temperature range around T_{KS}, thus reducing the running times, while PT requires many more temperatures to be run efficiently.

Let us also comment that RSA can be viewed as a standard annealing process that is a dynamical process trying to minimize the energy function, but using an effective Hamiltonian or cost function where configurations are reweighted. In the original work introducing RSA, the motivation was to sample configurations of large local entropy [37]. However, the idea can be generalized, and we believe it is really worth exploring further the possibility of increasing the chances of reaching the planted signal by modifying the energy landscape. This has been shown useful in random constraint satisfaction problems, where the reweighting of solutions can modify the critical lines and enhance the chance of finding a solution [53,54]. Also, in the tensor PCA problem, the modification of the energy landscape achieved via the use of many replicas can lead to sensible improvement in signal detection [55].

Having found a MCMC-based algorithm (RSA) that has the same algorithmic threshold c_{KS} as the Bayes-optimal BP, some questions naturally arise (that we leave for future works).

Does the RSA optimality hold in general? A simple way to paraphrase this question is, "Can one demonstrate that the equivalence

$$T_{KS}(c_{KS}, y) = T_d(c_{KS}, y)$$

is valid for all the planted inference problems whenever $y > 1$ or at least for y large enough?" And if yes, why? Can we get an improved RSA by optimizing the coupling among replicas? We believe the answer to these questions will come from studying numerically more inference problems and analytically the $y \to \infty$ limit.

Does there exist an even better MCMC-based algorithm that can work below c_{KS}, i.e., breaking the hard phase? The most immediate answer would be negative, given that BP (a Bayes-optimal algorithm) stops working at c_{KS}. Moreover, recent works connect the performance of a broad class of algorithms called stable algorithms with the structure of the configurational space, suggesting that the clustering of solutions in the hard phase is an insurmountable obstacle for stable algorithms [56]. However, we notice that MCMC-based algorithms do not belong to the class of stable algorithms (at least if run for a long enough time), so there is a hope they can work efficiently even in a clustered hard phase, where message-passing algorithms are deemed to fail. Moreover, smarter MCMC-based algorithms, like RSA or PT, actually work in an extended space (in replicas, the former algorithm, and in temperature, the latter), and in this extended space, the clustering of the measure may be less important. In support of this conjecture, we show in the present work how the nature of the phase transition changes in the replicated problem.

A last comment is on the scaling of the running times to find solutions in the regime where the algorithms we study do work, i.e., above c_{KS}. Given that we focus on sparse problems, each iteration step of an algorithm is linear in N. However, the number of iterations can scale with the system size N. For message-passing algorithms, the number of iterations scales as a small power of N or a large power of $\log(N)$, while for MCMC-based algorithms the number of iterations scales roughly linear in N (see Appendix B). This means we are already exploring the region of timescales growing with the system size, where analytical tools are at present unable to provide any detailed insight. Without reaching the timescales growing exponentially in N, which are practically useless for any real-world problem, we believe the possibility of developing polynomial in N solving algorithms based on the MCMC working below c_{KS} in the hard phase is to be seriously taken into account.

APPENDIX A: SUMMARY OF CRITICAL TEMPERATURES, THEIR PHYSICAL MEANING, AND THEIR CONNECTION WITH PREVIOUS DEFINITIONS

We define $T_{KS}(c)$ as the spinodal temperature at which the paramagnetic state develops an instability toward the planted solution. This temperature was already computed in Ref. [27] under the name T_2 looking at the divergence of the ferromagnetic susceptibility. Its exact value is

$$T_{KS}(c) = -\frac{1}{\log[\frac{c-(q-1)^2}{q-1+c}]}.$$

Such a spinodal line ends at zero temperature at $c_{KS} = (q-1)^2$ (previously called c_l in Ref. [27]). As usual in planted problems, c_{KS} corresponds also to the connectivity at which the paramagnetic state becomes unstable in a random graph, without the planted state: This is signaled by the divergence of the spin-glass susceptibility. However, one should be careful because this coincidence is not true anymore when temperature is added: The temperature at which the paramagnetic state becomes unstable in the random coloring problem has been computed in Ref. [29] and it reads

$$T_c(c) = -\frac{1}{\log\left[1 - \frac{q}{\sqrt{c+1}}\right]},$$

and it is always smaller than $T_{KS}(c)$ for $c > c_{KS}$.

The other temperature that is fundamental in our analysis is $T_d(c)$, which is defined as the spinodal temperature above which a glassy state in the random coloring problem is not stable anymore. Its behavior is already shown in Ref. [27]. At $T = 0$, this line ends in $c_d = 12.837(3)$ for $q = 5$ [29]. Analogous to c_{KS}, c_d turns out to be also the spinodal connectivity below which a planted configuration is not stable anymore. Again, this coincidence is not true anymore when temperature is added: The spinodal temperature for the planted configuration is the one called T_1 in Ref. [27], and $T_1(c) > T_d(c)$ holds for $c > c_d$.

One can generalize the above definitions for $T_d(c)$ and $T_1(c)$ by introducing $T_{sp}(T_{pl}, c)$ as the spinodal temperature at which a state planted at temperature T_{pl} becomes unstable. With this definition, we have $T_1(c) = T_{sp}(0, c)$ and $T_d(c) = T_{sp}(T, c)$.

APPENDIX B: SCALING OF RUNNING TIMES WITH SYSTEM SIZE

In the main text, we present data only for $N = 10^5$ and $dT = 10^{-7}$, but it is important to study how small one has

to set dT in order to ensure that the annealing algorithm will converge to the solution with high probability. For any well-normalized cost function, we expect the starting temperature of the annealing process to be $O(1)$, i.e., not to scale with the system size N. As a consequence, the running time of the annealing process is $O(dT^{-1})$, and given that a single iteration in sparse models requires a time $O(N)$, the total running time is $O(N/dT)$. Let us discuss now how dT should be scaled with N in order to ensure convergence to the planted state.

We run RSA for $c = 17$ and SA for $c = 20$ (in both cases, we expect the algorithm to work if dT is small enough). We consider sizes $N = 10^3, 10^4, 10^5$ and several dT values. From the data of the overlap presented in the inset of Fig. 2, it is clear that the condition $Q > 0.5$ selects the successful runs. For each pair (N, dT), we measure the probability of a successful run.

In Fig. 10, we report data obtained by running RSA for $c = 17$ with $y = 2, 4, 8$, and a good scaling of the success probability can be achieved by plotting the data as a function of NdT^a with a very close to 1. Using this renormalized variable, times for different sizes collapse and are well fitted by a function $t = \text{erfc}[(NdT^a - x_0)/b]$ [57]. The best scaling exponents are $a \simeq 1.27$ for $y = 2$, $a \simeq 0.9$ for $y = 4$, and $a \simeq 0.95$ for $y = 8$, but estimating the corresponding uncertainty is not easy as it depends on the finite-size corrections to scaling which are not under control. Nonetheless, all the estimated exponents are very close to 1, and this implies the RSA algorithm finds the planted solution in a time scaling roughly quadratically with the system size.

In Fig. 11, we present the same scaling analysis for the performances of SA at $c = 20$. Given the larger number of runs, the errors on the success probabilities are much smaller and consequently make evident the need to use finite-size corrections to scaling. For this reason, we compute the optimal scaling exponent $a \simeq 1.03$ by collapsing data for $N = 10^4, 10^5$ assuming the data for $N = 10^3$ are affected by some small corrections to scaling.

A simple numerical argument for the scaling $dT \approx N^{-1}$ in SA and RSA follows. Assume that the Monte Carlo (Glauber) dynamics is able to detect the planted state only in the region $T_d < T < T_{KS}$ (i.e., the one colored blue in Figs. 3 and 9, right) that we call the retrieval region. The time spent in the retrieval region by annealing at a constant rate is $(T_{KS} - T_d)/dT$, and we need this time to scale like the time needed by a MCMC simulation run at any fixed temperature in the retrieval region to find the planted state. We measure the number of MCSs needed to nucleate the planted solution running a standard MCMC simulation at fixed temperature T for $c = 20$ and $y = 1$. The results are presented in Fig. 12. Straight data correspond to a power-law growth with the problem size N, and the black line corresponds to a linear growth in N. Several observations are in order. The optimal time for small values of N seems
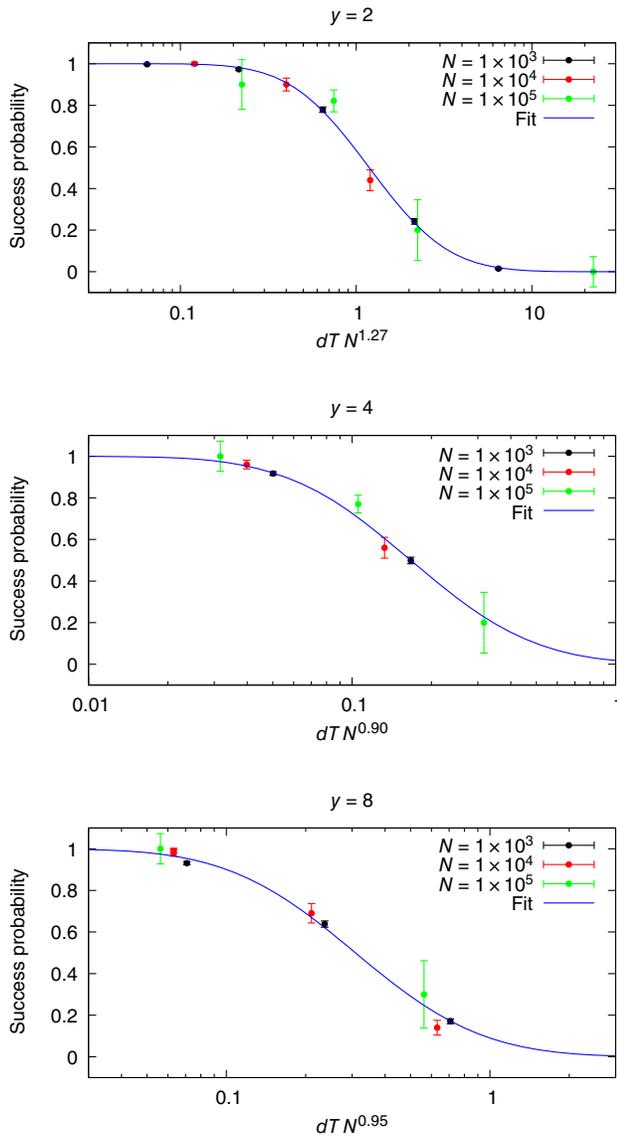
FIG. 10. The success probability of finding the planted state by running RSA on $c = 17$ planted-coloring problems of size N with a constant annealing rate dT is a function of $dT N^a$ with a scaling exponent a close to 1 (at least for $R = 4, 8$). This implies the running times of RSA are roughly quadratic in N. The fit function is $f(x) = \text{erfc}[(x - x_0)/b]$.

to be achieved for T slightly above $T_{KS} \simeq 0.56$, and this is expected as setting the temperature to the largest possible value speeds up the MC dynamics and finite-size corrections make T_{KS} larger (see Sec. V on the finite-size effects). If T is too large, eventually it becomes larger than T_{KS} (that decreases toward its large-N limit) and leaves the retrieval region: When this happens, the MC dynamics is no longer able to reach the planted state in a polynomial time, and the corresponding time grows enormously with N (see data for $T = 0.7$). If T is below T_d (see data for $T = 0.4$), the time to find the planted state grows immediately very fast (likely superpolynomially). Last, but not least, choosing the temperature in the retrieval region (better toward its upper
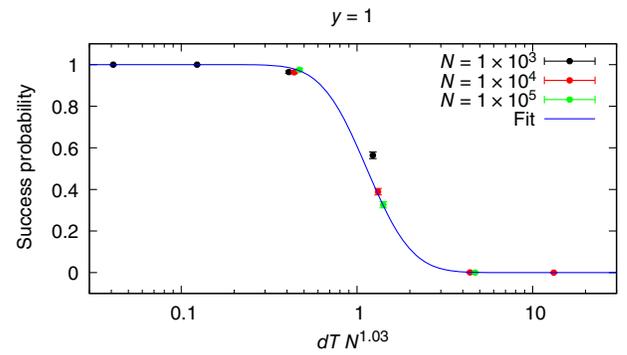
FIG. 11. As in Fig. 10, but for $c = 20$ and running SA. Again the scaling $dT N^a$ with $a \approx 1$ is very good, although the smaller errors suggest the presence of finite-size corrections to scaling.
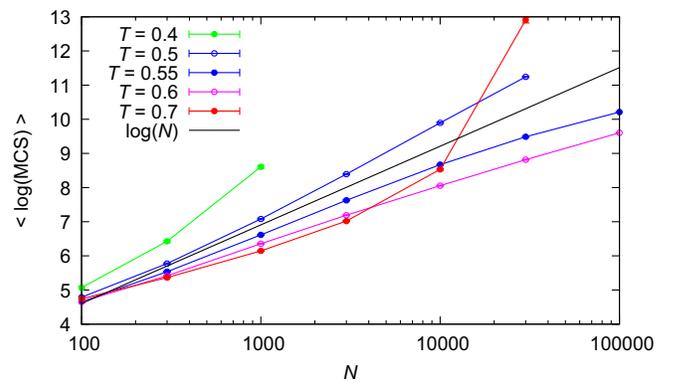
FIG. 12. The mean of the logarithm of the number of MCSs needed by a Glauber dynamics at fixed temperature T to find the planted state in $c = 20$ coloring problems.

end) the running times to find the planted state scale roughly linearly with the system size (the black line is a linear growth law for the ease of reading). This last observation brings us to the conclusion that, although running an annealing algorithm is very comfortable because one does not need to spend time selecting the optimal running temperature (the annealing will visit the retrieval region wherever it is), it is not optimal, because all the time spent outside the retrieval region is actually useless. The optimal and robust retrieval algorithm is running MC dynamics in the retrieval region. Since a single MCS takes a time of order $O(N)$, the time needed by MC dynamics to find the planted state in the retrieval region is again roughly quadratic.

As a comparison, in Fig. 13 we show the number of iterations that the BP algorithm needs to converge to the fixed point corresponding to the planted state. Each single BP iteration takes a time of order $O(cN)$. As shown, the growth of the number of iterations with the size of the system can be fitted both by a power law with a very small exponent and with a logarithmic growth. The two curves are indistinguishable for the analyzed sizes, and we cannot
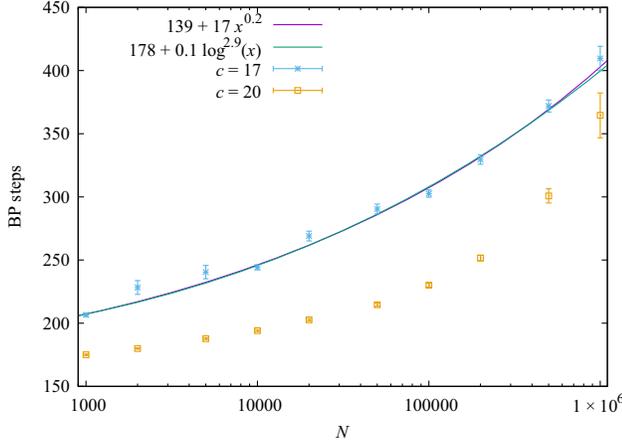
FIG. 13. Number of iterations that the BP algorithm needs before convergence to the planted fixed point as a function of the size of the system N for two values of the connectivities $c = 17$ and $c = 20$. The growth in the number of iterations can be fitted both by a power law $t(N) = a + bN^c$ with a very small exponent $c \simeq 0.2$, and with a logarithmic growth $t(N) = a + b\log^c(N)$ with $c \simeq 2.9$. The two curves are indistinguishable for the analyzed sizes, and we cannot conclude which is the right scaling.

say which is the right scaling. The times grow for smaller connectivities, but the scaling with N remains the same.

APPENDIX C: REPLICATED BELIEF PROPAGATION

In this appendix, we derive the BP equations for the coloring problem with y-coupled replicas at inverse temperature β, which we call RBP. For this model, the probability distribution of the variables follows the Gibbs-Boltzmann probability associated with the Hamiltonian in Eq. (8). As for the original model, one would like to estimate the
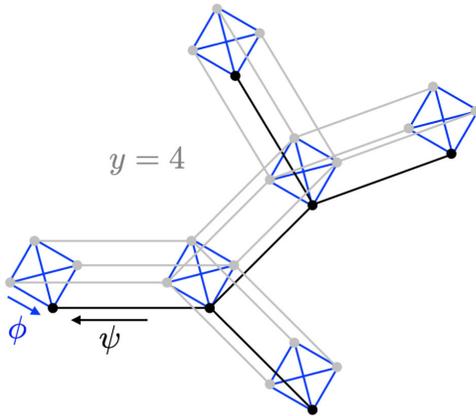
FIG. 14. A schematic view of the replicated graph for $y = 4$. The black graph is the original one, while the gray graphs are the $y - 1$ replicas. Blue ferromagnetic couplings are present among every pair of replicated vertices with the same site index. Replicated BP uses two kinds of messages, ψ and ϕ, represented by black and blue arrows.

marginal probabilities through the fixed point of some BP-like equations. However, in the replicated case, the graph is no more locally treelike, because of the presence of small loops between the y-coupled replicas, and the BP equations are not guaranteed to provide the correct solution to the problem. We see that in the limit $y \gg 1$, the interaction among replicas is weak enough, and the critical temperatures predicted by RBP will describe perfectly the transitions observed in the RSA in the limit of large y.

Making reference to Fig. 14, where every site of the replicated graph is identified by a pair (i, a), with i being the vertex of the original graph, taking values in $[1, N]$, and a is the replica index running in $[1, y]$, the RBP equations for the coloring model with y-coupled replicas at inverse temperature β can be written straightforwardly [58]

$$\psi_s^{(i,a)\to(j,a)} = \frac{1}{Z^{(i,a)\to(j,a)}} \prod_{b(\neq a)} \left(1 - (1 - e^{\gamma\beta/y})\phi_s^{(i,b)\to(i,a)}\right)$$
$$\times \prod_{k \in \partial i \backslash j} \left(1 - (1 - e^{-\beta})\psi_s^{(k,a)\to(i,a)}\right),$$

$$\phi_s^{(i,a)\to(i,b)} = \frac{1}{Z_\phi^{(i,a)\to(i,b)}} \prod_{c(\neq a,b)} \left(1 - (1 - e^{\gamma\beta/y})\phi_s^{(i,c)\to(i,a)}\right)$$
$$\times \prod_{k \in \partial i}(1 - (1 - e^{-\beta})\psi_s^{(k,a)\to(i,a)}), \qquad (C1)$$

with Z and Z_ϕ being the normalization factors enforcing $\sum_s \psi_s^{(i,a)\to(j,a)} = \sum_s \phi_s^{(i,a)\to(i,b)} = 1$ for every edge of the replicated graph. We call ψ the cavity messages passed between nodes of the same replica, and ϕ the cavity messages passed between different replicas on the same vertex of the original graph. In the limit $y = 1$, the ϕ messages are irrelevant, and the above equations reduce to the standard BP equations for the original (unreplicated) model.

To simplify Eqs. (C1) we assume a RS ansatz, where cavity messages do not depend on the replica indices, that is, $\psi_s^{(i,a)\to(j,a)} = \psi_s^{i\to j}$ and $\phi_s^{(i,a)\to(i,b)} = \phi_s^i$, and then we get

$$\psi_s^{i\to j} = \frac{1}{Z^{i\to j}}(1 - (1 - e^{\gamma\beta/y})\phi_s^i)^{y-1} \prod_{k \in \partial i \backslash j}(1 - (1 - e^{-\beta})\psi_s^{k\to i}),$$

$$\phi_s^i = \frac{1}{Z_\phi^i}(1 - (1 - e^{\gamma\beta/y})\phi_s^i)^{y-2} \prod_{k \in \partial i}(1 - (1 - e^{-\beta})\psi_s^{k\to i}).$$
$$(C2)$$

We verify numerically that fixing $\gamma = 1$ and solving Eqs. (C1) and (C2) on any given graph, we get exactly the same result, both in the high-temperature PM state and in the low-temperature FM state. So, the RS assumption is valid in the present case, and we focus only on Eqs. (C2) hereafter.

APPENDIX D: COMPUTATION OF THE KESTEN-STIGUM TRANSITION TEMPERATURE IN THE REPLICATED MODEL

The Kesten-Stigum transition temperature corresponds to the spinodal point of the PM state, that is, to the lowest temperature such that the PM state is locally stable. If the trivial fixed point of the RBP equations ($\psi_s = \phi_s = (1/q)$) is correctly describing the PM state of the replicated model, we can compute $T_{KS}(y, \gamma)$ via the study of the local stability of the RBP trivial fixed point with respect to fluctuations leading to the planted FM state.

We perform a linear expansion around the PM trivial solution in the variables

$$\epsilon_s^{i \to j} \equiv \psi_s^{i \to j} - \frac{1}{q}, \qquad \delta_s^i \equiv \phi_s^i - \frac{1}{q}. \tag{D1}$$

At leading order in this expansion, Eqs. (C2) become

$$
\begin{aligned}
\epsilon_s^{i \to j} &= \sum_{k \in \partial i \backslash j} \sum_t \frac{\partial \psi_s^{i \to j}}{\partial \psi_t^{k \to i}}\bigg|_{\text{PM}} \epsilon_t^{k \to i} + \sum_t \frac{\partial \psi_s^{i \to j}}{\partial \phi_t^i}\bigg|_{\text{PM}} \delta_t^i \\
&= A \sum_{k \in \partial i \backslash j} \left[\sum_t \epsilon_t^{k \to i} - q\epsilon_s^{k \to i} \right] + B(y-1)\left[\sum_t \delta_t^i - q\delta_s^i \right],
\end{aligned}
$$

$$
\begin{aligned}
\delta_s^i &= \sum_{k \in \partial i} \sum_t \frac{\partial \phi_s^i}{\partial \psi_t^{k \to i}}\bigg|_{\text{PM}} \epsilon_t^{k \to i} + \sum_t \frac{\partial \phi_s^i}{\partial \phi_t^i}\bigg|_{\text{PM}} \delta_t^i \\
&= A \sum_{k \in \partial i} \left[\sum_t \epsilon_t^{k \to i} - q\epsilon_s^{k \to i} \right] + B(y-2)\left[\sum_t \delta_t^i - q\delta_s^i \right],
\end{aligned}
\tag{D2}
$$

where the subscript PM indicates that the function should be computed at the PM trivial solution $\psi_s = \phi_s = (1/q)$ and

$$A = \frac{1 - e^{-\beta}}{q(q - 1 + e^{-\beta})}, \qquad B = \frac{1 - e^{\gamma \beta / y}}{q(q - 1 + e^{\gamma \beta / y})}.$$

Equations (D2) define a $q(N + M) \times q(N + M)$ matrix associated with a given graph of N vertices and M edges. To study the stability of the paramagnetic solution, one should look at its maximum eigenvalue $\lambda_{\max}$. When $\lambda_{\max} = 1$, the PM solution loses its local stability. In this way, we can locate $T_{KS}(y, \gamma)$ that we show (fixing $\gamma = 1$) in Fig. 3 for $y = 1$, in Fig. 5 for $y > 1$, and in Fig. 4 with a vertical dashed line in the $y \gg 1$ limit. The latter value is a very good estimate of the jump temperature for large y, where the planted signal can be detected.

At this point, one could think to obtain an analytic estimation for $T_{KS}(y, \gamma)$ computing the eigenvalues of the linear transformation in Eqs. (D2), assuming that in the large-N limit the original graph becomes locally a tree, and so the cavity messages on the right-hand sides are uncorrelated. This is a standard argument that indeed works for $y = 1$ and provides the analytical value $T_{KS}(y = 1) = -1/\log\left[(c - (q - 1)^2/q - 1 + c)\right]$ of Ref. [27].

In the following, we show that this kind of computation for $y > 1$ does not give the correct result compared to the large-N limit estimation of T_{KS} from the single graphs illustrated before. Let us introduce $P_{s,t}(\epsilon^{i \to j})$ and $Q_{s,t}(\delta^{(i,a) \to (i,b)})$ as the probability distributions of the two kinds of perturbations (between the nodes and between the replicas) in the direction of color s if the spin i is planted in the color t. In the large-N limit, assuming a locally treelike replicated graph, we can write closed equations for these distributions

$$
\begin{aligned}
P_{s,t}(\epsilon) &= \sum_{d=0}^{\infty} \frac{d}{c} r_d \int \sum_{\{\sigma_j\},\{\tau_j\},\{\sigma_a\}} \prod_{j=1}^{d-1} d\epsilon_j P_{\sigma_j, \tau_j}(\epsilon_j) p(\tau_j | t) \\
&\quad \times \prod_{a=1}^{y-1} d\delta_a Q_{\sigma_a, t}(\delta_a) \delta(\epsilon - f_1(\{\epsilon_j\}, \{\delta_a\})),
\end{aligned}
$$

$$
\begin{aligned}
Q_{s,t}(\delta) &= \sum_{d=0}^{\infty} r_d \int \sum_{\{\sigma_j\},\{\tau_j\},\{\sigma_a\}} \prod_{j=1}^{d} d\epsilon_j P_{\sigma_j, \tau_j}(\epsilon_j) p(\tau_j | t) \\
&\quad \times \prod_{a=1}^{y-2} d\delta_a Q_{\sigma_a, t}(\delta_a) \delta(\delta - f_2(\{\epsilon_j\}, \{\delta_a\})), \tag{D3}
\end{aligned}
$$

where r_d is the probability of having a vertex of degree d in the original graph (a Poisson distribution of mean c in the model we study in this work), $(d/c)r_d$ is the probability that, taken a link $i \to j$ at random, i has degree d, $p(\tau | t) = [I(\tau \neq t)/q - 1]$ is the probability that a neighbor of site i is planted in the state τ given that i is planted in the state t, and finally the functions f_1 and f_2 are defined, respectively, by the right-hand sides of the first and second equation in (D2). Please note that we are allowing δ_a to take different values for each replica. We are thus analyzing the linear expansion of Eq. (C1) rather than Eq. (D2) which is the linear expansion of Eq. (C2).

At this point, given that we are looking for a transition toward a "ferromagnetic" state, we can restrict our attention to the first moments of the above distributions

$$\langle \epsilon_{s,t} \rangle \equiv \int d\epsilon P_{s,t}(\epsilon) \epsilon = \sum_{d=0}^{\infty} \frac{d}{c} r_d \left\{ \sum_{\{\tau_j\} \neq t} \frac{A}{q-1} \sum_{j=1}^{d-1} \left[\sum_{\sigma_j} \langle \epsilon_{\sigma_j, \tau_j} \rangle - q \langle \epsilon_{s, \tau_j} \rangle \right] + B \sum_{a=1}^{y-1} \left[\sum_{\sigma_a} \langle \delta_{\sigma_a, t} \rangle - q \langle \delta_{s,t} \rangle \right] \right\},$$

$$\langle \delta_{s,t} \rangle \equiv \int d\delta Q_{s,t}(\delta) \delta = \sum_{d=0}^{\infty} r_d \left\{ \sum_{\{\tau_j\} \neq t} \frac{A}{q-1} \sum_{j=1}^{d} \left[\sum_{\sigma_j} \langle \epsilon_{\sigma_j, \tau_j} \rangle - q \langle \epsilon_{s, \tau_j} \rangle \right] + B \sum_{a=1}^{y-2} \left[\sum_{\sigma_a} \langle \delta_{\sigma_a, t} \rangle - q \langle \delta_{s,t} \rangle \right] \right\}. \tag{D4}$$

Exploiting the symmetry under permutation of the colors, we can identify the only four relevant observables: $\epsilon^e \equiv \langle \epsilon_{s,s} \rangle$, $\epsilon^d \equiv \langle \epsilon_{s,t} \rangle$ with $s \neq t$, $\delta^e \equiv \langle \delta_{s,s} \rangle$ and $\delta^d \equiv \langle \delta_{s,t} \rangle$ with $s \neq t$. Our goal is to look at fluctuations toward the planted state; thus, it is convenient to look at the differences $\epsilon^d - \epsilon^e$ and $\delta^d - \delta^e$ that satisfy the following closed equations:

$$\epsilon^d - \epsilon^e = (\epsilon^d - \epsilon^e)Ac\frac{q}{q-1} - qB(y-1)(\delta^d - \delta^e),$$

$$\delta^d - \delta^e = (\epsilon^d - \epsilon^e)Ac\frac{q}{q-1} - qB(y-2)(\delta^d - \delta^e). \quad \text{(D5)}$$

The analysis of the stability of the PM solution in the tree limit of the replicated graph is thus reduced to the computation of the largest eigenvalues of the matrix

$$\begin{bmatrix} Ac\frac{q}{q-1} & -qB(y-1) \\ Ac\frac{q}{q-1} & -qB(y-2) \end{bmatrix}. \quad \text{(D6)}$$

The value of T_{KS} resulting from this computation for $y = 64$ is shown in Fig. 15 with the name "RPopDyn," as we check it coincides with the result found by running the population dynamic algorithm to solve Eqs. (D2). The comparison with the values of T_{KS} obtained from the solution of the RBP equations on several graph realizations for different sizes N reveals that in the large-N limit the two computations coincide only for $\gamma = 0$, that is when replicas do not interact. As long as $\gamma > 0$ and the replicas are coupled, the large-N limit of the single graph computation
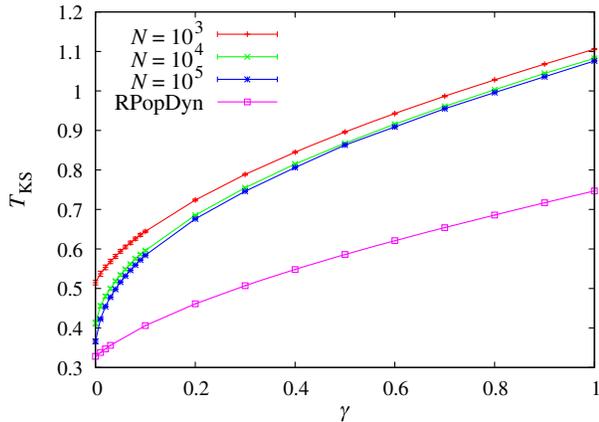
FIG. 15. $T_{\text{KS}}(y = 64, \gamma)$ at $c = 17$ as a function of the coupling γ among replicas. The upper curves are obtained running RBP on several graph of a given size N, while the lower curve is obtained running population dynamics to solve Eqs. (D2), or equivalently, computing the largest eigenvalue of matrix in Eq. (D6). The limit $N \to \infty$ of the former computation coincides with the latter only at $\gamma = 0$. As soon as $\gamma > 0$, the assumption of the replicated graph being locally treelike is wrong, and the latter computation fails, while running RBP on a given graph still provides a reliable value for T_{KS}.

does not coincide with the analytical estimation obtained under the assumption of a locally treelike replicated graph.

The reason for the mismatch between the two computations is that for $y > 1$ and $\gamma > 0$ the replicated graph is not locally treelike and cavity messages are correlated. While running RBP on a given graph these correlations can be taken into account, the population dynamics algorithm, by construction, ignores any correlation among cavity messages. The correlations among messages coming from different replicas are positive, thanks to the ferromagnetic coupling among replicas, and this, in turn, produces a much larger value for T_{KS} than under the assumption of uncorrelated messages.

The attentive reader could be puzzled by the fact that population dynamics gives the wrong result even in the $y \to \infty$ limit (the curves shown in Fig. 15 are for $y = 64$, which is quite large, and their value is almost the asymptotic one for $y = \infty$). In fact, in the limit $y \to \infty$, the y replicas form a fully connected ferromagnetic Potts model, and we know that the cavity method becomes exact for fully connected graphs. However, there is a simple explanation for the failure of population dynamics even when $y \to \infty$ in this case. The Bethe approximation between the y replicas is exact in the limit $y \to \infty$ inside a single pure state. In fact, the clustering property holds inside a pure state. Our model has a permutation symmetry: When the system is magnetized, there are in principle different pure states that can be obtained one from the other just by the permutation of the colors. Depending on which pure state is chosen by the y replicas on the site i, also the neighboring spins $j \in \partial i$ will choose their color. In the replicated graph, there are many small loops of length 4, between the spins $(i, a) - (i, b) - (j, b) - (j, a)$. While in the limit $y \to \infty$, the connected correlation functions between (i, a) and (i, b) and between (j, a) and (j, b) decay to zero, the full correlation functions do not decay to zero, because the average magnetization is not zero inside a pure state. The BP algorithm for a single instance of the graph takes into account this correlation of the magnetizations, just because, when applied to a fixed instance of the graph, BP spontaneously breaks the permutation of the colors just dynamically choosing a given pure state, and thus, it is exact in the limit of $y \to \infty$. On the other hand, the population dynamics algorithm [Eqs. (C3)] does not consider the correlation in the magnetizations, picking each time different neighboring spins i and j as if they were uncorrelated, and this random choice is wrong even in the limit $y \to \infty$.

Willing to improve the analytical computation (or the solution via the population dynamics algorithm), one should consider a supervariable at every node of the original graph, taking q^y values. In this case, the graph connecting the supervariables is indeed locally treelike, and a population dynamical computation would give the correct estimates for the thermodynamical thresholds both for

finite and infinite y (we leave this computation for future work).

APPENDIX E: COMPUTATION OF THE DYNAMICAL TRANSITION TEMPERATURE

The dynamical temperature T_d is the temperature below which glassy states are locally stable (and also attractive for the out-of-equilibrium dynamics). T_d is not modified by the addition of the planted state, which is roughly orthogonal to the glassy states. For this reason, the usual procedure to identify T_d is as follows: One plants a configuration typical of the Gibbs measure at temperature T constructing a graph compatible with it. This is a generalization of the planting procedure at $T = 0$ that we describe in the main text. In practice, to plant a configuration of temperature T when $y = 1$, one first decides the assigned color s_i for each node i, in such a way that there are N/q nodes for each color. Then, one extracts two nodes i and j at random among the N possible ones, and an edge is put between them with probability $p = e^{-\beta(1-\delta_{s_i,s_j})}$. In practice, nonmonocromatic edges are always accepted, while monochromatic edges are accepted with probability $e^{-\beta}$ that tends to zero if $T \to 0$. One then repeats this operation until $M = (Nc/2)$ edges are put in the graph. The chosen planted configuration is in this way an equilibrium configuration of the constructed graph at temperature T. One then initializes BP near enough to the planted glassy configuration and runs the BP equations at temperature T. If the fixed point reached by BP is the PM one, then glassy states are locally unstable and thus $T > T_d$, while if the fixed point is correlated with the planted glassy state, then $T < T_d$.

In this way we locate T_d for $y = 1$ (see data in Fig. 3) [59].

In principle, one could use the same planting technique to locate T_d in the $y > 1$ case, paying attention to plant a state according to the replicated Hamiltonian in Eq. (8). In practice, we do the following: We first decide the assigned color s_i for each node i in such a way that there are N/q nodes for each color. We put all the y replicas associated with the node i in the color s_i at the beginning. Then we run a short Monte Carlo simulation at fixed temperature T for the y replicas of node i with associated Hamiltonian $H_i = -(\gamma/2y) \sum_{a\neq b=1}^{y} \delta_{s_i^a, s_i^b}$. In this way, we obtain an equilibrium configuration for the y replicas of each of the N nodes (that at this point are still disconnected). Then we construct the graph: We extract two nodes i and j at random among the N possible ones, and an edge is put between them with probability $p = e^{-\beta \sum_{a=1}^{y}(1-\delta_{s_i^a, s_j^a})}$. We then repeat this operation until $M = (Nc/2)$ edges are put in the graph. To test the local stability of this planted configuration, one really needs to run the BP algorithm with y different replicas, according to Eqs. (C1), because the different replicas will not be planted necessarily with the same color (being the temperature positive $T > 0$ each replica will have a nonzero

probability to differ from the planted configuration). Computing T_d in this way, we discover that the transition becomes a continuous one as soon as $y > 1$ (at least for $\gamma = 1$). This is signaled by the fact that the overlap between the fixed point reached by BP initialized near the planted glassy state at T and the planted glassy configuration tends to zero continuously in the limit $T \to T_d^-$. One can also look at another order parameter defined as

$$m^2 = \frac{1}{yN} \sum_{i=1}^{N} \sum_{a=1}^{y} \sum_{s=1}^{q} (\psi_s^{(i,a)})^2. \tag{E1}$$

In the paramagnetic state, $\psi_s^{(i,a)} = (1/q) \ \forall s, i, a$, and thus, $m^2 = (1/q)$. When $y = 1$, m^2 jumps discontinuously to the value $m^2 = (1/q)$ at $T = T_d$, while for $y > 1$, m^2 reaches its paramagnetic value continuously at T_d.

Both the behavior of the overlap and the squared magnetization summarized in Figs. 16 and 17 imply that for $y > 1$ the dynamical transition of the random model changes nature and becomes continuous, at difference with the case $y = 1$, where this transition is discontinuous. This fact implies that we can compute T_d in the $y > 1$ case just by looking at the instability of the paramagnetic solution through the computation of the maximal eigenvalue of Eqs. (D2) on a given graph without the $T = 0$ planted solution (exactly as we did to compute T_{KS} on a graph with the $T = 0$ planted solution). We check that this computation gives exactly the same result for T_d as the previously described planting procedure. However, the study of the
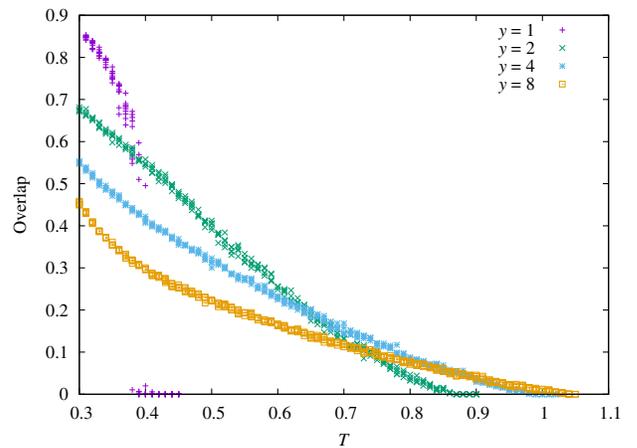
FIG. 16. Overlap between the fixed point reached by BP initialized near a planted glassy state at T and the glassy configuration planted at T, as a function of the planting temperature T for different values of the replicas y at $c = 17$, $N = 10^4$, $\gamma = 1$. At each temperature, the results for five different planted graphs are shown. T_d is located as the lowest temperature at which the overlap reaches 0. As long as $y > 1$, the transition becomes continuous.
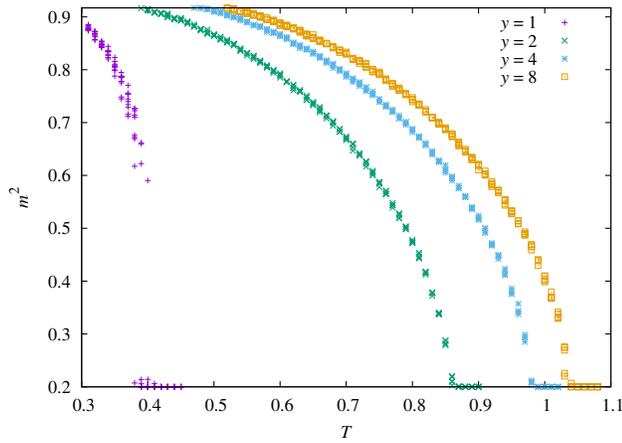
FIG. 17.   Squared magnetization defined as in Eq. (E1) of the fixed point reached by BP initialized near a planted glassy state at T, as a function of the planting temperature T for different values of the replicas y at $c = 17$, $N = 10^4$, $\gamma = 1$. At each temperature, the results for five different planted graphs are shown. T_d is located as the lowest temperature at which $m^2 = (1/q)$. As long as $y > 1$, the transition becomes continuous.

linearized equations is much faster, the convergence of BP initialized around the planted glassy state being very slow in the vicinity of T_d. With this linearization method, we compute the T_d values for $y > 1$ shown in Fig. 5.

[1] Paul W. Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt, *Stochastic Blockmodels: First Steps*, Soc. Networks **5,** 109 (1983).

[2] Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborová, *Inference and Phase Transitions in the Detection of Modules in Sparse Networks*, Phys. Rev. Lett. **107,** 065701 (2011).

[3] Emile Richard and Andrea Montanari, *A Statistical Model for Tensor PCA*, Adv. Neural Inf. Process. Syst. **27** (2014), https://proceedings.neurips.cc/paper_files/paper/2014/file/b5488aeff42889188d03c9895255cecc-Paper.pdf.

[4] Emmanuel Abbe and Colin Sandon, *Detection in the Stochastic Block Model with Multiple Clusters: Proof of the Achievability Conjectures, Acyclic BP, and the Information-Computation Gap*, arXiv:1512.09080.

[5] Yash Deshpande and Andrea Montanari, *Finding Hidden Cliques of Size $\sqrt{N/e}$ in Nearly Linear Time*, Found. Comput. Math. **15,** 1069 (2015).

[6] Thibault Lesieur, Florent Krzakala, and Lenka Zdeborová, *MMSE of Probabilistic Low-Rank Matrix Estimation: Universality with Respect to the Output Channel*, in *Proceedings of the 2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton)* (IEEE, New York, 2015), pp. 680–687.

[7] Yudong Chen and Jiaming Xu, *Statistical-Computational Tradeoffs in Planted Problems and Submatrix Localization with a Growing Number of Clusters and Submatrices*, J. Mach. Learn. Res. **17,** 1 (2016), https://www.jmlr.org/papers/volume17/14-330/14-330.pdf.

[8] Samuel B. Hopkins, Tselil Schramm, Jonathan Shi, and David Steurer, *Fast Spectral Algorithms from Sum-of-Squares Proofs: Tensor Decomposition and Planted Sparse Vectors*, in *Proceedings of the Forty-Eighth Annual ACM Symposium on Theory of Computing* (2016), pp. 178–191, 10.1145/2897518.2897529.

[9] Afonso S. Bandeira, Amelia Perry, and Alexander S. Wein, *Notes on Computational-to-Statistical Gaps: Predictions Using Statistical Physics*, Portugaliae mathematica **75,** 159 (2018).

[10] Boaz Barak, Samuel Hopkins, Jonathan Kelner, Pravesh K. Kothari, Ankur Moitra, and Aaron Potechin, *A Nearly Tight Sum-of-Squares Lower Bound for the Planted Clique Problem*, SIAM J. Comput. **48,** 687 (2019).

[11] Elizabeth Gardner and Bernard Derrida, *Three Unfinished Works on the Optimal Storage Capacity of Networks*, J. Phys. A **22,** 1983 (1989).

[12] There are problems in which the data are not just the graph, and the SNR is more complicated than just c. However, the following reasoning based on Bayes's theorem can be generalized to other situations [13].

[13] Lenka Zdeborová and Florent Krzakala, *Statistical Physics of Inference: Thresholds and Algorithms*, Adv. Phys. **65,** 453 (2016).

[14] P Erdős and A Renyi, *On Random Graphs*, Publ. Math. Debrecen **6,** 290 (1959), https://publi.math.unideb.hu/load_doi.php?pdoi=10_5486_PMD_1959_6_3_4_12.

[15] Marc Mezard and Andrea Montanari, *Information, Physics, and Computation* (Oxford University Press, New York, 2009).

[16] There are some counterexamples like planted XORSAT [17,18] or noiseless sign retrieval with Gaussian matrices [19,20] in which there exist algorithms that do beat BP (even though these examples are very brittle and break down when adding a very small noise).

[17] S. Franz, M. Mezard, F. Ricci-Tersenghi, M. Weigt, and R. Zecchina, *A Ferromagnet with a Glass Transition*, Europhys. Lett. **55,** 465 (2001)

[18] L. Zdeborová and F. Krzakala, *Quiet Planting in the Locked Constraint Satisfaction Problems*, SIAM J. Discrete Math. **25,** 750 (2011).

[19] M. J. Song, I. Zadik, J. Bruna, *On the Cryptographic Hardness of Learning Single Periodic Neurons*, Adv. Neural Inf. Process. Syst. **34,** 29602 (2021).

[20] D. Gamarnik, E. C. Kızıldağ, and I. Zadik, *Inference in High-Dimensional Linear Regression via Lattice Basis Reduction and Integer Relation Detection*, IEEE Trans. Inf. Theory **67,** 8109 (2021).

[21] Jonathan S. Yedidia, William T. Freeman, Yair Weiss *et al.*, *Understanding Belief Propagation and Its Generalizations*, in *Exploring Artificial Intelligence in the New Millennium*, edited by G. Lakemeyer and B. Nebel (Morgan Kaufmann, San Francisco, 2003), Chap. 8, pp. 239–268.

[22] Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborová, *Asymptotic Analysis of the Stochastic Block Model for Modular Networks and Its Algorithmic Applications*, Phys. Rev. E **84,** 066106 (2011).

[23] David Gamarnik and Ilias Zadik, *The Landscape of the Planted Clique Problem: Dense Subgraphs and the Overlap Gap Property*, arXiv:1904.07174.

[24] Maria Chiara Angelini, Paolo Fachin, and Simone de Feo, *Mismatching as a Tool to Enhance Algorithmic Performances of Monte Carlo Methods for the Planted Clique Model*, J. Stat. Mech. (2021) 113406.

[25] Zongchen Chen, Elchanan Mossel, and Ilias Zadik, *Almost-Linear Planted Cliques Elude the Metropolis Process*, in *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)* (SIAM, 2023), pp. 4504–4539, 10.1137/1.9781611977554.ch171.

[26] Fabrizio Antenucci, Silvio Franz, Pierfrancesco Urbani, and Lenka Zdeborová, *Glassy Nature of the Hard Phase in Inference Problems*, Phys. Rev. X **9,** 011020 (2019).

[27] Florent Krzakala and Lenka Zdeborová, *Hiding Quiet Solutions in Random Constraint Satisfaction Problems*, Phys. Rev. Lett. **102,** 238701 (2009).

[28] Roberto Mulet, Andrea Pagnani, Martin Weigt, and Riccardo Zecchina, *Coloring Random Graphs*, Phys. Rev. Lett. **89,** 268701 (2002).

[29] Lenka Zdeborova and Florent Krzakala, *Phase Transitions in the Coloring of Random Graphs*, Phys. Rev. E **76,** 031131 (2007).

[30] For more complicated scenarios, we refer the reader to the thorough survey of phases in Ref. [31].

[31] Federico Ricci-Tersenghi, Guilhem Semerjian, and Lenka Zdeborová, *Typology of Phase Transitions in Bayesian Inference Problems*, Phys. Rev. E **99,** 042109 (2019).

[32] Marc Mézard and Giorgio Parisi, *The Bethe Lattice Spin Glass Revisited*, Eur. Phys. J. B **20,** 217 (2001).

[33] Hidetoshi Nishimori, *Statistical Physics of Spin Glasses and Information Processing: An Introduction* (Clarendon Press, Oxford, 2001), p. 111, https://global.oup.com/academic/product/statistical-physics-of-spin-glasses-and-information-processing-9780198509400.

[34] Scott Kirkpatrick, C. Daniel Gelatt, Jr., and Mario P. Vecchi, *Optimization by Simulated Annealing*, Science **220,** 671 (1983).

[35] M. Chams, A. Hertz, and D. De Werra, *Some Experiments with Simulated Annealing for Coloring Graphs*, Eur. J. Oper. Res. **32,** 260 (1987).

[36] David S. Johnson, Cecilia R. Aragon, Lyle A. McGeoch, and Catherine Schevon, *Optimization by Simulated Annealing: A Experimental Evaluation; Part II, Graph Coloring and Number Partitioning*, Oper. Res. **39,** 378 (1991).

[37] Carlo Baldassi, Christian Borgs, Jennifer T. Chayes, Alessandro Ingrosso, Carlo Lucibello, Luca Saglietti, and Riccardo Zecchina, *Unreasonable Effectiveness of Learning Neural Networks: From Accessible States and Robust Ensembles to Basic Algorithmic Schemes*, Proc. Natl. Acad. Sci. U.S.A. **113,** E7655 (2016).

[38] Dandan Hu, Peter Ronhovde, and Zohar Nussinov, *Phase Transitions in Random Potts Systems and the Community Detection Problem: Spin-Glass Type and Dynamic Perspectives*, Philos. Mag. **92,** 406 (2012).

[39] Dandan Hu, Peter Ronhovde, and Zohar Nussinov, *Replica Inference Approach to Unsupervised Multiscale Image Segmentation*, Phys. Rev. E **85,** 016101 (2012).

[40] Matteo Bellitti, Federico Ricci-Tersenghi, and Antonello Scardicchio, *Entropic Barriers as a Reason for Hardness in Both Classical and Quantum Algorithms*, Phys. Rev. Res. **3,** 043015 (2021).

[41] Haim Sompolinsky, Naftali Tishby, and H. S. Seung, *Learning from Examples in Large Neural Networks*, Phys. Rev. Lett. **65,** 1683 (1990).

[42] Stefano Sarao Mannelli, Giulio Biroli, Chiara Cammarota, Florent Krzakala, Pierfrancesco Urbani, and Lenka Zdeborová, *Marvels and Pitfalls of the Langevin Algorithm in Noisy High-Dimensional Inference*, Phys. Rev. X **10,** 011057 (2020).

[43] Stefano Sarao Mannelli, Florent Krzakala, Pierfrancesco Urbani, and Lenka Zdeborova, *Passed & Spurious: Descent Algorithms and Local Minima in Spiked Matrix-Tensor Models*, in *Proceedings of the International Conference on Machine Learning* (PMLR, 2019), pp. 4333–4342, https://proceedings.mlr.press/v97/mannelli19a.html.

[44] Stefano Sarao Mannelli, Giulio Biroli, Chiara Cammarota, Florent Krzakala, and Lenka Zdeborová, *Who Is Afraid of Big Bad Minima? Analysis of Gradient-Flow in Spiked Matrix-Tensor Models*, Adv. Neural Inf. Process. Syst. **32** (2019), https://proceedings.neurips.cc/paper_files/paper/2019/file/fbad540b2f3b5638a9be9aa6a4d8e450-Paper.pdf.

[45] Maria Chiara Angelini and Federico Ricci-Tersenghi, *Monte Carlo Algorithms Are Very Effective in Finding the Largest Independent Set in Sparse Random Graphs*, Phys. Rev. E **100,** 013302 (2019).

[46] Florent Krzakała, Andrea Montanari, Federico Ricci-Tersenghi, Guilhem Semerjian, and Lenka Zdeborová, *Gibbs States and the Set of Solutions of Random Constraint Satisfaction Problems*, Proc. Natl. Acad. Sci. U.S.A. **104,** 10318 (2007).

[47] Andrea Montanari, Federico Ricci-Tersenghi, and Guilhem Semerjian, *Clusters of Solutions and Replica Symmetry Breaking in Random k-Satisfiability*, J. Stat. Mech. (2008) P04004.

[48] In addition to the results shown in the present work about the planted random graph coloring, we have preliminary results also for the planted bicoloring of random hypergraphs supporting our statements.

[49] Maria Chiara Angelini, *Parallel Tempering for the Planted Clique Problem*, J. Stat. Mech. (2018) 073404.

[50] Farzad Pourkamali and Nicolas Macris, *Mismatched Estimation of Rank-One Symmetric Matrices under Gaussian Noise*, arXiv:2107.08927.

[51] Andrea Crisanti and H.-J. Sommers, *The Spherical p-Spin Interaction Spin Glass Model: The Statics*, Z. Phys. B **87,** 341 (1992).

[52] In previous works [37], BP in the replicated model was found to converge to the planted state only if helped by focusing, that is, an effective slow change in γ or in T. We observe that standard BP does converge to the planted state if the right choice is made for γ and T to be in the blue region of the corresponding phase diagram.

[53] Louise Budzynski, Federico Ricci-Tersenghi, and Guilhem Semerjian, *Biased Landscapes for Random Constraint Satisfaction Problems*, J. Stat. Mech. (2019) 023302.

[54] Angelo Giorgio Cavaliere, Thibault Lesieur, and Federico Ricci-Tersenghi, *Optimization of the Dynamic Transition in the Continuous Coloring Problem*, J. Stat. Mech. (2021) 113302.

[55] Giulio Biroli, Chiara Cammarota, and Federico Ricci-Tersenghi, *How to Iron Out Rough Landscapes and Get Optimal Performances: Averaged Gradient Descent and Its Application to Tensor PCA*, J. Phys. A **53**, 174003 (2020).

[56] David Gamarnik, *The Overlap Gap Property: A Topological Barrier to Optimizing over Random Structures*, Proc. Natl. Acad. Sci. U.S.A. **118**, e2108492118 (2021).

[57] This is the function expected if we consider the distribution of times-to-solution being a log-normal. If we put a threshold, the probability that the time is above the threshold is described by an error function.

[58] In principle, when writing (linearized) BP equations, one should specify the iteration time indices τ. However, having already many indices in Eqs. (C1), (C2), (D2), and following, we omit the time indices, reminding the careful reader that time index is $\tau + 1$ on the lhs and τ on the rhs.

[59] One could locate T_d also performing a standard 1RSB computation, looking when a 1RSB solution is stable and different from the PM one; however, this procedure is much more expensive than the planting procedure we use.