PHYSICAL REVIEW B **89**, 214421 (2014)

Belief-propagation-guided Monte-Carlo sampling

Aurélien Decelle[1,*] and Florent Krzakala[2,3]

[1]*Dipartimento di Fisica, Università La Sapienza, Piazzale Aldo Moro 5, I-00185 Roma, Italy*

[2]*CNRS and ESPCI ParisTech, 10 rue Vauquelin, UMR 7083 Gulliver, Paris 75000, France*

[3]*Laboratoire de Physique Statistique, CNRS UMR 8550, Université P. et M. Curie Paris 6 et École Normale Supérieure, 24, rue Lhomond, 75005 Paris, France*

(Received 23 August 2013; revised manuscript received 5 June 2014; published 26 June 2014)

A Monte-Carlo algorithm for discrete statistical models that combines the full power of the belief-propagation algorithm with the advantages of a heat-bath approach fulfilling the detailed balance is presented. First we extract randomly a subtree inside the interaction graph of the system. Second, given the boundary conditions, belief propagation is used as a perfect sampler to generate a configuration on the tree, and finally, the procedure is iterated. This approach is best adapted for locally treelike graphs and we therefore tested it on random graphs for hard models such as spin glasses, demonstrating its state-of-the art status in those cases.

DOI: 10.1103/PhysRevB.89.214421 PACS number(s): 02.50.Ng, 75.50.Lk, 89.70.Eg

Sampling a distribution of strongly correlated variables is a central task in many fields such as statistical mechanics, machine learning, and statistical analysis. Indeed, there are many problems where an exact treatment is impossible due to the large number of strongly correlated variables. Furthermore, in many cases analytical approximations lead to imprecise results if compared to the one obtained in numerical simulation. The Markov-chain Monte-Carlo (MCMC) approach for sampling is a fundamental component of modern physics [1,2] playing a central role in inference and learning problems (e.g., computational biology [3,4], machine learning [5], simulated annealing [6]). A drawback of MCMC methods, such as Metropolis, is the long runtime needed to obtain high-precision estimates. In addition, this time can be affected by local energy or entropy barriers and ergodicity breaking. A large scientific effort has been devoted to the design of faster MCMC schemes [1,2].

A particular family of discrete statistical models considers systems where the underlying graph of interaction is a tree. Those cases have been widely studied both in physics, where they form the basis of the Bethe approximation [7], and in computer science [8,9]. In these problems, an *exact* marginalization in linear time is possible—i.e., in $O(N)$ steps, where N is the system size—by using an algorithm called belief propagation (BP). As we shall see, it implies the possibility of perfect sampling in linear time as well (i.e., extracting a configuration from the Boltzmann measure). This has been a fundamental breakthrough allowing gigantic possibilities in machine learning [10]. A natural extension is to consider graphs with few (or large) loops. Those graphs are typically present in many concrete problems [11]. But when the loops appear it becomes much harder to sample perfectly the measure. It would be indeed very useful to be able to sample efficiently graphs that have *locally* a tree structure but which are *not* tree. Among such graphs, random ones are commonly used in statistical physics as mean-field models [12], in combinatorial optimization as archetypes of hard benchmarks [13], and in many types of sparse networks encountered in clustering problems [14,15]. They are also used for error-correcting codes [16] and in inference problems [11], where a good sampler is mandatory if one hopes to deal with large-size problems.

Belief-propagation-guided MCMC. In this paper we present an algorithm respecting the detailed balance which combines the perfect sampling ability of BP on trees with the traditional heat-bath strategy using MCMC approaches. A method close to ours has been studied on 2D lattices in [17]. However our algorithm is able, on random graphs, to flip random trees of huge sizes which would have been impossible with their method. Those very large clusters allow us to avoid local traps that one can encounter with the local Metropolis dynamics. We expect in addition that our algorithm unleashes its full potential on graphs without too many short loops where the subtree extraction is facilitated. Finally, our algorithm can be adapted for any kind of graph. A similar approach has been used in [18] but where only deterministic subtrees were considered.

In what follows, we present the algorithm in detail and apply it to various difficult benchmarks. We focus mainly on systems having the random graph topology, as they are typical examples of networks without short loops, and we demonstrate the state-of-the-art nature of our approach.

Our method is based on a heat-bath procedure: we repeatedly select a random subpart of the problem and equilibrate it given its interaction with the rest of the system. When it is applied to a single spin, it leads to Glauber dynamics. A common strategy to improve the convergence of the dynamics is to apply it to a group of two or more spins. However, the difficulty to perform a perfect sampling increases dramatically with the number of spins in general. Our strategy will be to select sub-parts of the network having the topology of a tree and to use BP to guide our sampling process.

Sampling on a tree. We describe first how using BP we can sample efficiently the Boltzmann measure on a tree given the boundary conditions. Starting from the leaves of the graph (the nodes with only one neighbor), we can compute sequentially the BP messages (or “partial” marginals) toward the center. Let us consider first a “leaf” spin s_l and compute its “partial” marginal toward k, that is, the probability of s_l given that its only neighbor k has been removed. When the only neighbor has been removed, the “leaf” spin is only sensitive to the boundary condition (which we denote as an effective magnetic

*Corresponding author: adecelle@yahoo.fr

field h_i^{bound}) and the partial marginal reads

$$\psi_{s_l}^{l\to k} = \frac{e^{\beta h_l^{\text{bound}}(s_l)}}{Z^{l\to k}}, \tag{1}$$

where $Z^{l\to k}$ is the normalization constant and β the inverse temperature. In the following, we should also consider the partial marginal $\psi_{s_i}^{i\to j}$ of a site i when the link (ij) has been removed (also called BP message from i to j). Starting from the leaves, these partial marginals can be propagated toward the center of the tree using the BP equations: for any i, $\psi^{i\to j}$ can be computed when the $\psi^{k\to i}$ are known for all neighbors $k \neq j$ of i. This propagation is exact on trees and reads for $\psi^{i\to j}$

$$\psi_{s_i}^{i\to j} = \frac{1}{Z^{i\to j}} \prod_{k\in\partial i\setminus j} \sum_{s_k} e^{-\beta\mathcal{H}(s_i,s_k)} \psi_{s_k}^{k\to i}, \tag{2}$$

where $\mathcal{H}$ encodes the interaction between neighbors, and ∂i denote the neighbors of i. Iterating this procedure allows us to go deeper and deeper in the tree, until we reach a spin where *all* incoming messages have been computed. For this very last spin, we can now compute the correct "complete" marginal using

$$p_{\text{marg}}(s_c) = \frac{1}{Z^c} \prod_{k\in\partial c} \sum_{s_k} \psi_{s_k}^{k\to c} e^{-\beta\mathcal{H}(s_k,s_c)}. \tag{3}$$

Thus, we can use this marginal to choose a new state for the spin s_c. Given this new assigned value for the spin's state, it is possible to compute the complete marginal for its neighbors. From them, we choose again a new state for these spins, and this procedure is iterated back to the boundaries of the tree. At this point, the whole tree has been updated with a new configuration sampled from the Boltzmann measure in $O(N)$ steps, where N is the size of the tree graph.

Sampling on a graph. We now explain how to use this procedure to perform a heat bath on any graph. The procedure follows three different steps: (i) Extract randomly a tree subgraph from the interaction graph; see Fig. 1, left panel. (ii) Cancel the states of the spins inside the tree. All spins *immediately outside* of the tree will be used as the boundary conditions. (iii) Use the BP perfect sampler described above to extract a new configuration of the spins *inside* the tree given the boundary conditions. In Fig. 1 the middle panel illustrates the propagation of BP messages toward the central spin. The right panel illustrates how new states are drawn and used to iterate BP messages.

In our implementation of the algorithm, we construct the subtree by taking a node at random and adding its neighbors in random order. The neighbors are added unless it creates a loop in the subgraph. In such case we put it in the list of spins at the border of the tree. This list will be used as boundary conditions. The procedure is repeated on all newly added nodes until all of them have been treated. This construction is particularly efficient on random graphs and we illustrate it in the left panel of Fig. 1. It is however important to point out that it might not be efficient on finite-dimensional systems. In that case one should design a different procedure to extract a tree from the graph.

The creation of the subtree is dominating the algorithm's complexity and a complete update of the graph scales as $O(c^2N)$, where c is the average degree of the graph. Our algorithm is therefore faster on diluted graphs. However, we should be careful that in some cases this construction might update less frequently some sites. Indeed, since the root of the tree is chosen randomly, small disconnected clusters will be updated more rarely. To counterbalance this effect, we always alternate our algorithm with a Metropolis move so that all the sites are updated frequently. It is also important to point out that this is not a problem of ergodicity: all states of the phase space can be reached with a nonzero probability. In our tests of the algorithm, the results were mainly independent of the amount of randomness used during the subtree creation. We also observed that the local MCMC moves were quite important when dealing with Poissonian random graphs.

Numerical tests. We shall now discuss the performance of our algorithm. In the following we consider three different examples. First we focus our attention on the relaxation of the energy as a function of the time. We compare three different methods on two Ising models and we investigate the effect of the subtree maximum size on the convergence. Second we study the autocorrelation time for both Metropolis and our

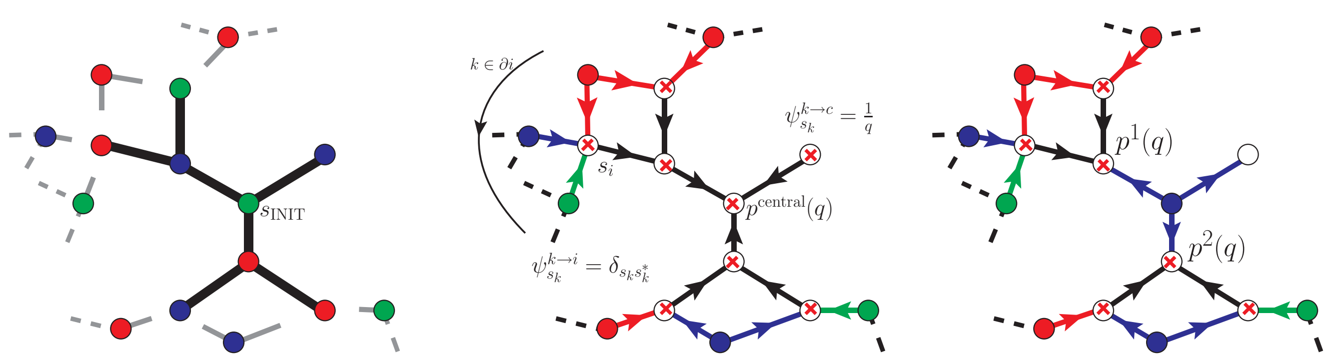

FIG. 1. (Color online) A schematic illustration of the BP-guided MCMC on the coloring problem. Left: Starting from a random node s_{INIT} we create a random tree by adding the neighbors of each node in the tree progressively (but without creating a loop). The (plain) links in gray have been cut on the picture to emphasis the constructed tree structure. Middle: The tree nodes have been reset and marked by a red X on the figure. We illustrate by arrows how the BP messages propagate to compute the partial marginals until the central spin is reached. For the latter we can then compute the complete marginal. Right: The central node has been put in one state according its marginal p^{central}. Propagating this information backward on the tree, this allows us to compute the complete marginal for each variable and to sample a new configuration for all the variables on the tree.

method on a p spin. Finally we compare the same algorithms on an annealing experiment. Note that for our algorithm one time step corresponds to an update of N spins (N being the system's size) such that after T time steps all sites are updated T times on average. This definition is chosen in order to make a fair comparison with MC where a time step corresponds to the update of N randomly chosen spins. We should also mention that the results presented here do not depend on the system sizes. We controlled that using larger (or smaller) system sizes, the conclusions were not affected.

We consider first the energy relaxation after a quench in the low-temperature phase of two systems. We consider an easy case—a standard ferromagnetic Ising model on a random graph with connectivity $c = 3$, $T_c \approx 2.94$—and a harder one—an antiferromagnet on the same random graph, $T_c \approx 1.52$. This later, due to the presence of loops of various sizes, behaves as a spin-glass model. For both, we start from a random configuration and cool the system at $T = 0.1$. A quench from a random configuration at this temperature get stuck into relatively high energy states due to the presence of local energy barriers.

In our simulation, we compare the relaxation time of our algorithm where we add a parameter controlling the subtrees's maximum size and we report the results obtained by varying this parameter (when the maximum size is 1 we recover the Glauber MCMC). In addition we implemented the Wolff algorithm [19] to confront a cluster method to the BP-guided one. In Fig. 2 we plot the energy as a function of the iteration time for both systems. As we increase the maximum size of the subtrees, we update larger and larger clusters and the barriers no longer block the dynamics. One can observe on the figure that the convergence time is drastically improved. In fact, perhaps not surprisingly, the algorithm converges faster when using the maximum cluster size—thus avoiding larger and larger local minima—in both the ferromagnet and the spin-glass case. It is instructive to compare it with the standard Wolff algorithm. For the ferromagnet, Wolff is able to avoid the barriers and in fact performs even better than the BP-guided MCMC. This is hardly a surprise as for ferromagnetic systems without frustration the Wolff approach is always very efficient. For the spin glass problem, however, even the Wolff method remains stuck in some subspace of the phase space (see Fig. 2). In these cases, the advantage of our approach is evident.

As a second example, we move to a more complicated model: the Ising p-spin glass on random graphs, also known as the random XORSAT model. The Hamiltonian reads

$$\mathcal{H}(\{s\}) = -\sum_{a=1}^{M} J_a \prod_{i\in\partial a} s_i, \tag{4}$$

where M groups of p spins are chosen randomly and coupled according to the random coupling J_a which is equal to $\{\pm 1\}$ with equal probability. This model has been widely studied in the literature both as a model for glasses [20,21] and for error-correcting codes [22] and as a toy model for the satisfiability problem [23]. It exhibits a dynamical glass transition at $T_d = 0.510$ [24] at which the relaxation time diverges, thus making difficult to take independent measures. We study here the relaxation time of the magnetization when $T \to T_d$ for

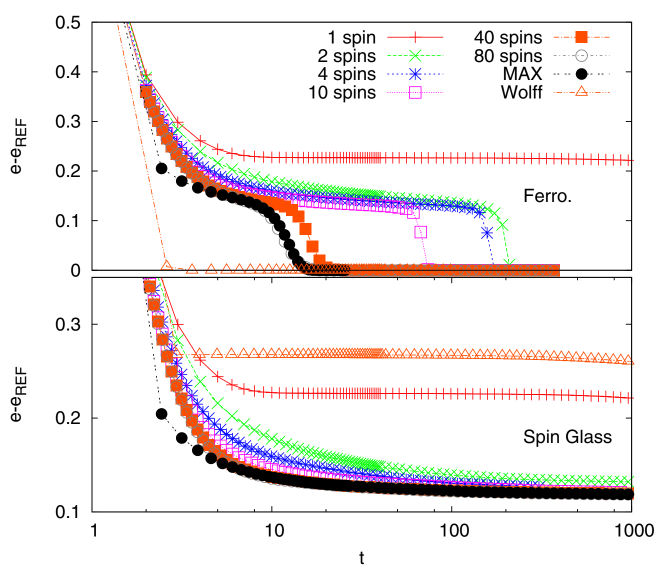

FIG. 2. (Color online) Energy density after a quench (using $e_{\rm REF} = c/2$ for the ferromagnet and $e_{\rm REF} = 0$ for the antiferromagnet) starting from a random initial configuration for an Ising model on a large ($N = 10^6$) regular random graph with connectivity $c = 3$ at temperature $T = 0.1$. Top panel: The convergence of the energy is shown versus the iteration time in the ferromagnetic case using different thresholds for the largest possible tree-cluster move on each curve. While standard Metropolis (1 spin) gets trapped in configurations with large energy for infinitely long time, increasing the size of the trees systematically increases the efficiency of the algorithm. With large enough trees, one equilibrates the system in less than 20 iterations. Bottom panel: A similar study for an antiferromagnet that behaves as a spin glass model due to the frustration. Similar performances to those in the ferromagnetic case are observed. Note that while the Wolff cluster approach is able to perfectly sample the ferromagnet, it fails in the spin glass case where the best performances are obtained by the BP-guided algorithm.

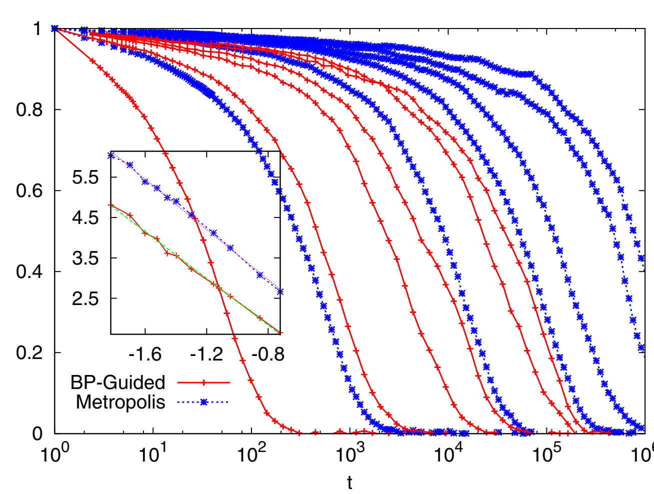

FIG. 3. (Color online) Autocorrelation $m(t) = C_{\rm eq}(t)$ for the p spin on regular graph with $N = 10^5$ and $c = 3$ starting from equilibrium. The temperatures go from $0.7 \to 0.525$ and the dynamic glass transition arises at $T_d = 0.510$ [24]. In the inset the melting time diverges as a power law. The BP-guided MCMC method is more than 10 times faster than the traditional one.

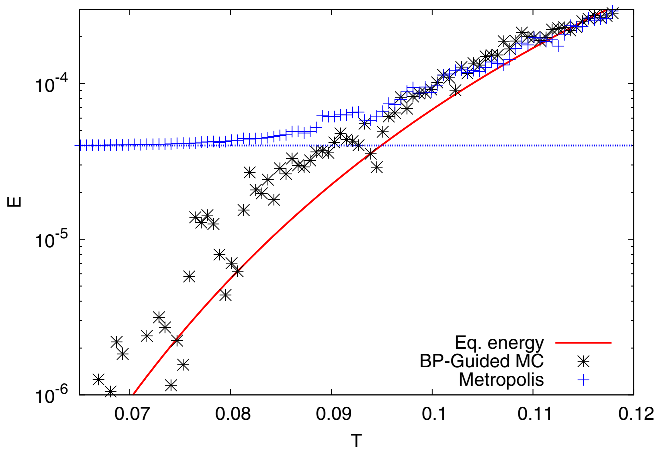

FIG. 4. (Color online) Simulated annealing starting from an equilibrated configuration at $T = 0.12$ in a 4-coloring problem on a large $N = 10^5$ 9-regular random graph. The annealing is performed by decreasing the temperature by $\Delta T = 10^{-7}$ every ten time steps. While the traditional Metropolis approach is stuck at finite energy, our BP-guided algorithm shows no sign of such blocking and actually reaches the ground state.

both Metropolis and our algorithm. By using the approach of [25,26], it is possible to create perfectly thermalized initial conditions which allow us to measure the equilibrium spin-glass correlation time easily. The time-relaxation of the correlation as $T \to T_d$ can be directly observed starting from this thermalized configuration. In Fig. 3 we illustrate the alpha relaxation starting from an equilibrium configuration for different temperatures close to T_d. Our algorithm improves the decorrelation time by one order of magnitude while the exponent for the diverging relaxation time was, however, not significantly smaller. Hence, even in models as complicated as the XORSAT one, the BP-guided approach improves the mixing property of the dynamics by taking advantage of the local-tree structure.

A third example is given by the q-coloring problem. This is an NP-complete constraint optimization problem that aims to color a graph with q colors such that all variables have a different color than their neighbors. It is equivalent to an antiferromagnetic Potts model:

$$\mathcal{H}(\{s\}) = \sum_{(i,j)\in\mathcal{E}} \delta(s_i,s_j), \quad \text{with } s_i = 1,\ldots,q. \qquad (5)$$

We consider this model on random graphs for which, in some region of the parameter q and the average degree c, a coloring configuration exists with probability one but can be hard to find. In a recent work [27], it has been shown that the 9-regular graph with $q = 4$ has a dynamical transition below which the equilibrium states possess a colorable configuration. We therefore perform an annealing experiment from an equilibrium configuration at $T < T_d$ using MCMC dynamics on one part and the BP-guided approach on the other one. As can be seen in Fig. 4, the MCMC dynamics gets stuck in some local minima as the temperature is cooled down. However, under the same condition, our algorithm manages to escape such minima and to reach the ground state of the system.

Conclusion. We have presented an algorithm for exact sampling in complex systems, illustrated its performance, and compared it to those of more traditional Metropolis dynamics. We show different examples where our method performs better than local move MCMC. In addition we demonstrate that our algorithm out-competes (some) cluster rejection-free methods and is immediately adaptable to many types of systems. We have made all the tests on random graphs since the extraction method of subtrees we are using is particularly adapted for this topology. On the other hand, the algorithm can be applied to any kind of graph (except fully connected ones), but one should be careful when choosing the subgraph construction. Indeed for a typical Euclidean graph the first step of the algorithm has to be optimized in order to construct quickly a subtree for the network considered. Many developments could be considered: combining our algorithm together with parallel tempering; testing the performances on finite-dimensional models, as for instance the diluted spin models of [28] where large trees could be constructed; or studying the zero-temperature version as an optimization tool.

Acknowledgments. This work has been supported in part by the ERC under the European Union's 7th Framework Programme Grant Agreement No. 307087-SPARCS. A.D. has been supported by FIRB Project No. RBFR086NN1.

[1] K. Binder, *Monte Carlo Methods in Statistical Physics*, Topics in Current Physics (Springer-Verlag, Berlin, 1986).

[2] W. Krauth, *Statistical Mechanics: Algorithms and Computations*, Oxford Master Series in Physics (Oxford University Press, Oxford, 2006).

[3] B. Mau, M. A. Newton, and B. Larget, Biometrics **55**, 1 (1999).

[4] P. D. O'Neill *et al.*, Math. Biosci. **180**, 103 (2002).

[5] D. Gamerman and H. Lopes, *Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference*, Chapman & Hall Texts in Statistical Science Series (Taylor & Francis, New York, 2006).

[6] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, Science **220**, 671 (1983).

[7] M. Mézard, G. Parisi, and M. A. Virasoro, *Spin-Glass Theory and Beyond*, Lecture Notes in Physics, Vol. **9** (World Scientific, Singapore, 1987).

[8] D. J. MacKay, *Information Theory, Inference, and Learning Algorithms* (Cambridge University Press, Cambridge, 2003).

[9] J. Pearl, in *Proceedings of the American Association of Artificial Intelligence National Conference on AI* (AAAI, Pittsburgh, 1982), pp. 133–136.

[10] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference* (Morgan Kaufmann Publishing, San Francisco, CA, 1988).

[11] M. J. Wainwright and M. I. Jordan, Foundations and Trends in Machine Learning **1**, 1 (2008).

[12] M. Mézard and G. Parisi, Eur. Phys. J. B **20**, 217 (2001).

[13] R. Monasson, R. Zecchina, S. Kirkpatrick, B. Selman, and L. Troyansky, Nature (London) **400**, 133 (1999).
[14] A. Decelle, F. Krzakala, C. Moore, and L. Zdeborová, Phys. Rev. Lett. **107**, 065701 (2011).
[15] M. Girvan and M. E. Newman, Proc. Natl. Acad. Sci. **99**, 7821 (2002).
[16] T. J. Richardson and R. L. Urbanke, IEEE Trans. Inf. Theory **47**, 599 (2001).
[17] F. Matsubara, A. Sato, O. Koseki, and T. Shirakura, Phys. Rev. Lett. **78**, 3237 (1997).
[18] F. Hamze and N. de Freitas, in *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence* (AUAI Press, Arlington, Virginia, 2004), pp. 243–250.
[19] U. Wolff, Phys. Lett. B **228**, 379 (1989).
[20] B. Derrida, Phys. Rev. Lett. **45**, 79 (1980).
[21] S. Franz, M. Mézard, F. Ricci-Tersenghi, M. Weigt, and R. Zecchina, Europhys. Lett. **55**, 465 (2001).
[22] M. Mézard and A. Montanari, *Information, Physics, and Computation* (World Scientific, Oxford, 2010).
[23] F. Ricci-Tersenghi, M. Weigt, and R. Zecchina, Phys. Rev. E **63**, 026702 (2001).
[24] A. Montanari and G. Semerjian, J. Stat. Phys. **125**, 23 (2006).
[25] A. Montanari and G. Semerjian, J. Stat. Phys. **124**, 103 (2006).
[26] F. Krzakala and L. Zdeborová, J. Chem. Phys. **134**, 034513 (2011).
[27] L. Zdeborová and F. Krzakala, Phys. Rev. B **81**, 224205 (2010).
[28] T. Jörg, H. G. Katzgraber, and F. Krząkała, Phys. Rev. Lett. **100**, 197202 (2008).