Message passing for vertex covers

Martin Weigt

Institute for Scientific Interchange, Viale Settimio Severo 65, I-10133 Torino, Italy

Haijun Zhou

Institute of Theoretical Physics, The Chinese Academy of Sciences, Beijing 100080, China
(Received 9 May 2006; published 12 October 2006)

Constructing a minimal vertex cover of a graph can be seen as a prototype for a combinatorial optimization problem under hard constraints. In this paper, we develop and analyze message-passing techniques, namely, warning and survey propagation, which serve as efficient heuristic algorithms for solving these computational hard problems. We show also, how previously obtained results on the typical-case behavior of vertex covers of random graphs can be recovered starting from the message-passing equations, and how they can be extended.

I. INTRODUCTION

The minimal vertex-cover (VC) problem belongs to the most difficult class of optimization problems in graph theory [1]. It asks to mark a minimal number of vertices of a graph, such that each edge of the graph is incident to at least one of the selected vertices. This problem is known to be NP-hard, which means, in particular, that all currently known algorithms construct minimal vertex covers in a computational time, which scales exponentially with the size of the graph. The applicability of such exact algorithms is therefore restricted to pretty small graphs of few hundreds of vertices.

There are, however, applications of the vertex-cover problem and other closely related optimization problems [2,3] to a huge number of real-world network problems; examples are the monitoring of Internet traffic [4], the prevention of denial-of-service attacks [5], and immunization strategies in networks [6]. Another technically related network problem is the one of counting loops in networks, recently analyzed on the basis of statistical-physics methods [7]. The dimensions of the underlying networks easily exceed the graph sizes treatable with exact algorithms, and heuristic methods to construct as small as possible solutions are needed.

In this paper we set up two message-passing techniques based on the statistical-physics approach to combinatorial optimization problems [8,9], more precisely based on the cavity method for diluted systems [10] and its algorithmic interpretation [11,12]. The first of the message-passing techniques, the so-called warning propagation is equivalent to the Bethe-Peierls iterative scheme and therefore related to the assumption of replica symmetry; the second one is a survey propagation algorithm related to one-step replica symmetry breaking. Both algorithms have already been formulated for the vertex-cover problem by one of the authors in Ref. [8]; here we go beyond this presentation providing both a more elegant setting and a thorough analysis of the algorithmic performance.

A natural test bed for the proposed algorithms is provided by finite-connectivity random graphs [13]. The typical properties of VCs on such graphs have already been analyzed both with rigorous mathematical tools [14,15] and with statistical-physics methods [16–20]. We therefore have a pretty complete knowledge about the phase diagram of this problem, and can systematically compare the algorithmic performance of the message-passing techniques on single, finite, randomly generated graphs to the average behavior in the thermodynamic limit.

It should also be noted that the vertex-cover problem is closely related to a class of lattice glass models [21–25]. In these models, hard particles are to be positioned on a lattice under geometrical packing constraints representing hard-core interactions. These models are considered as simple lattice models for the glass transition due to geometric frustration, and their closest packings correspond to minimal VCs [16].

This paper is organized as follows: the vertex-cover problem is defined in Sec. II and the concept of cavity graph is defined in Sec. III; Sec. IV focuses on the warning propagation algorithm, and its performance and iterative stability are analyzed; Sec. V focuses on the survey propagation algorithm; Sec. VI estimates the minimal vertex-cover size for a random graph using a statistical-physics method; and finally, in Sec. VII we conclude this work.

II. THE MODEL

Let us start with the definition of vertex covers. Given is a graph $G=(V,E)$ with N vertices $i=1,\ldots,N$ and M undirected edges $\{i,j\}=\{j,i\}\in E$ connecting pairs of vertices.

Definitions. A *vertex cover* (VC) of the graph G is a subset $U\subset V$ of vertices such that for all edges $\{i,j\}\in E$, at least one end vertex is an element of U, i.e., $i\in U$ or $j\in U$. A *minimal vertex cover* of G is a vertex cover of minimal cardinality.

We also denote vertices in U as covered, as well as their incident edges: The set U is a VC iff all edges are covered. Determining a minimal vertex cover is one of the basic NP-hard combinatorial problems [1]. Its worst-case solution time is consequently expected to grow exponentially with the size of the problem instance, here measured by the vertex and edge numbers N and M. The problem is equivalent to the problem of constructing a *maximum independent set* of G, and to the problem of finding the *maximum clique* in the complementary graph of G (where edges and nonedges are exchanged).

The exponential running time of algorithms constructing minimal vertex covers is a serious limitation to practical applications: Exact algorithms are able to treat only relatively small sample graphs. It is therefore interesting to develop powerful heuristic methods, which are able to construct at least close-to-minimal VCs, which may serve as reasonable solutions in practical problems.

In the context of constraint-satisfaction problems (CSPs), recently, statistical-physics approaches have led to the proposal of so-called *survey propagation algorithms*, which are sophisticated message-passing procedures based on the cavity method of statistical physics. This type of algorithm was first proposed for the satisfiability problem [11], and then extended to graph coloring [26] and general CSPs [27], and is one of the most efficient algorithms in the hard-to-solve phase of these problems.

The vertex-cover problem is structurally different from CSPs. Whereas the computational problem of the latter results from the existence of a large number of constraints being hard to satisfy simultaneously, the constraints in a vertex cover—i.e., the need for covering each edge of the graph—can in principle be satisfied very easily by covering many vertices. The computational hardness stems from the objective of finding a *minimal* vertex cover, i.e., from the interaction between a high number of local constraints on one side, and the global minimization condition on the other side. This leads to a difference in the validation of the output of a heuristic algorithm: Whereas a solution to a CSP can be easily checked by testing all constraints, and the problem consists in finding one, it is no problem at all to construct a VC, but its minimality can hardly be shown. One can say that the hardness of solving VC stems from the fact that the landscape $X(U)=|U|$ becomes complex over the set of all VCs U (note: not over the set of all vertex subsets).

The algorithmic aim is therefore to construct a vertex cover as small as possible in polynomial time for some given graph $G=(V,E)$. The central step in this context will be the calculation (or at least approximation) of the vertex-dependent number

$$\pi_i = \frac{|\{U \subset V | U \text{ is minimal VC}, i \in U\}|}{|\{U \subset V | U \text{ is minimal VC}\}|}, \quad (1)$$

which, for every vertex $i \in V$, equals the fraction of minimal vertex covers containing vertex $i \in V$. In probabilistic terms, it can be understood as the probability that i is covered in a randomly selected minimal vertex cover.

Once we know these quantities, we can obviously exploit them algorithmically. We know, e.g., that each vertex with $\pi_i = 1$ belongs to all minimal VCs, and it has to be included into the VC we are aiming to construct. Contrarily, vertices $i \in V$ with $\pi_i = 0$ do not appear in any minimal VC, and they have to be excluded from the vertex set we are building. The problem is slightly more involved for those vertices having π values different from zero and one; they are contained in some vertex covers, but not in others. Since π_i gives only strictly local information, we do not know any possible quantitative restriction to the simultaneous assignment of pairs or even larger subsets of vertices. If we consider, e.g.,

one edge $\{i,j\} \in E$, the joint probability that both vertices are uncovered does not equal $(1-\pi_i)(1-\pi_j)$ as one might assume naively by considering the vertices to be independent. It equals obviously zero due to the vertex-cover constraint for the edge; at least one of the end vertices has to be covered. This problem can be resolved by an iterative decimation process. We select, e.g., a vertex of nonzero π and add it to the VC U to be constructed, and delete the vertex as well as all its incident edges from the graph. We then recompute the π from the decimated graph, add a new vertex to U and so on, until all edges of G are covered. The vertex set U now forms a vertex cover of the graph G.

There is an obvious algorithmic problem with evaluating the π_i: A naive calculation according to their definition would require the prior knowledge of all minimal VCs—which we do not have if we are trying to develop an algorithm finding just a single one of them. The way out will be a message-passing procedure [12,28], which only exchanges local information between neighboring vertex pairs, until these messages reach globally self-consistent values. Such message-passing procedures first need the introduction of the *cavity graph*, which will be done in the following section.

III. THE CAVITY GRAPH

A simple idea could be to determine π_i from all the π_j of the neighbors $j \in N(i)$ of vertex i. This is not directly possible. As discussed above, the π_j are single-site quantities and do not contain any information of vertex pairs. Any two $j \in N(i)$ are, however—via a path crossing i—second neighbors of each other, and thus they are highly correlated. Imagine, e.g., that vertex i is not covered, then *all* $j \in N(i)$ have to be simultaneously covered. The knowledge of the marginal cover probabilities π_j is obviously not sufficient to determine also the central π_i. The way out is to consider not the full graph, but the *cavity graphs*.

Definition. Given a graph $G=(V,E)$, and a vertex $i \in V$, the cavity graph G_i is the subgraph of G induced by the vertex set $V_i=V\backslash i$.

Said with simpler words, the cavity graph is created from the full graph G by removing vertex i as well as its incident edges $\{i,j\}$ for all $j \in N(i)$. On a tree graph, the $j \in N(i)$ would belong to pairwise-distinct connected components of the cavity graph, and they would be independent of each other. More generally, on a graph with relatively long cycles, any two of the former neighbors of vertex i will be distant on the cavity graph G_i. The basic approximation underlying message-passing algorithms consists in assuming the statistical independence of these vertices on the cavity graph (within one thermodynamic state, as will be explained in the case of survey propagation).

Having defined the cavity graphs G_i for each vertex i, we also define the generalized probabilities

$$\pi_{j|i} = \frac{|\{U \subset V_i | U \text{ is minimal VC of } G_i, j \in U\}|}{|\{U \subset V_i | U \text{ is minimal VC of } G_i\}|} \quad (2)$$

measuring the fraction of minimal vertex covers of the cavity graph G_i containing vertex $j \neq i$. Even if defined formally for

any pair of vertices i and j, these quantities will be relevant, in particular, for those vertices connected by an edge in the original graph G, i.e., for $\{i,j\} \in E$.

A comment on the statistical-independence assumption has to be included at this point: We are constructing an algorithm for real, i.e., finite graphs. This means that graph loops have finite length. The equations we are going to present in the following will therefore be only approximations to the exact values of the probabilities π_i, and the algorithm cannot guarantee to construct a true minimum-vertex cover. So, even if the presented algorithm will scale only quadratically in the graph order N, it cannot be considered as an exact polynomial algorithm, and therefore does not contribute to the solution of the P-NP problem. The importance of message-passing algorithms is related to practical applications on large graphs, where exact methods fail due to their exponential time requirements. As we will see below in numerical simulations, the procedures presented here largely outperform purely local algorithms, and therefore allow us to construct better approximations to the exact solution.

IV. WARNING PROPAGATION

A. Algorithm

The very first and simplest message-passing procedure we are going to introduce carries the name *warning propagation* (WP). In this case, we are going to calculate only the reduced quantities

$$\tilde{\pi}_i = \begin{cases} 0 & \text{if } \pi_i = 0, \\ * & \text{if } 0 < \pi_i < 1, \\ 1 & \text{if } \pi_i = 1, \end{cases} \tag{3}$$

and analogously the cavity quantities $\tilde{\pi}_{j|i}$. So these quantities are not measuring the exact probability for a vertex to be covered in a randomly selected minimal vertex cover. They only indicate whether it is always covered (value one), never covered (value zero), or sometimes covered and sometimes uncovered. For this last case we have introduced the unifying *joker state* *. Note also that this information is sufficient to be exploited algorithmically. If a vertex is assigned the joker state, it can be chosen liberally to be covered or to be uncovered during graph decimation.

As a first step, we introduce an even simpler message type, the so-called warning $u_{j \to i}$ sent from a vertex j to a neighbor i. This warning incorporates the vertex-cover constraint: If the vertex j is uncovered, it sends a warning $u_{j \to i} = 1$ to vertex i signifying, "Attention, to cover our connecting edge you should be covered, or I have to change state." If, on the other hand, vertex j is already covered, it sends the trivial message $u_{j \to i} = 0$ saying, "I have already covered our connecting edge." More formally, a set of warnings is defined for every vertex subset $U \subset V$,

$$u_{j \to i}(U) := \begin{cases} 0 & \text{if } j \in U, \\ 1 & \text{if } j \notin U, \end{cases} \tag{4}$$

with $\{i,j\} \in E$ being an arbitrary edge. Note that each edge

carries *two* messages: One sent from i to j, the other one from j to i. In a proper VC, at least one of the end vertices of each edge has to be covered, so we find that

$$U \subset V \text{ is VC of } G \leftrightarrow \forall \ \{i,j\} \in E : u_{i \to j}(U) u_{j \to i}(U) = 0, \tag{5}$$

i.e., each edge has to carry at least one trivial warning. The definition of the warning can also be extended to sets $\mathcal{M}$ of vertex subsets. We define

$$u_{j \to i}(\mathcal{M}) := \min_{U \in \mathcal{M}} u_{j \to i}(U), \tag{6}$$

i.e., a nontrivial message is sent if and only if vertex j is an element of no $U \in \mathcal{M}$. This definition obviously reproduces the warning (4) if $\mathcal{M}$ consists of only one vertex subset. The reason for selecting the minimum in the last definition will become clear below. Using the set $\mathcal{S}_i$ of all minimal vertex covers of the cavity graph G_i as a special case, the warning $u_{j \to i}(\mathcal{S}_i)$ becomes a function of $\tilde{\pi}_{j|i}$ only. For an arbitrary but fixed edge we find

$$u_{j \to i}(\mathcal{S}_i) \equiv u_{j \to i}(\tilde{\pi}_{j|i}) = \begin{cases} 1 & \text{if } \tilde{\pi}_{j|i} = 0, \\ 0 & \text{if } \tilde{\pi}_{j|i} = *, \\ 0 & \text{if } \tilde{\pi}_{j|i} = 1. \end{cases} \tag{7}$$

The required minimality of the vertex cover to be constructed leads to a simple propagation of these warnings, or equivalently of the corresponding $\tilde{\pi}_{j|i}$. This can be achieved by considering how minimal vertex covers can be extended from the cavity graph to the full graph. There are three cases (cf. Fig. 1).

(a) There exists at least one minimal vertex cover of the cavity graph G_i where *all* $j \in N(i)$ (neighbors of i in the full graph G) are simultaneously covered. These VCs are also minimal VCs of the full graph G since all edges incident to i are already covered, so i has to be uncovered to guarantee minimality. The sizes of the minimal VCs of G_i and those of G thus coincide. In this case we find $\tilde{\pi}_i = 0$, since there are no minimal VCs of G containing i.

(b) All minimal vertex covers of G_i leave at least two $j \in N(i)$ uncovered. Since all edges incident to vertex i have to be covered, we have to add i to the VC of G_i in order to extend it to the full graph. The VC of the full graph thus contains exactly one vertex more than those of the cavity graph, and $\tilde{\pi}_i$ equals one.

(c) In the last, intermediate case, there is at least one minimal VC of G_i containing all but one $j \in N(i)$, but there is none containing all $j \in N(i)$. Also in this case, we have to add exactly one vertex by going from a VC of the cavity graph G_i to one of the full graph G; the VC size grows by one. If we, however, use the VC leaving only one $j \in N(i)$ uncovered, there exists only one single uncovered edge in G. To cover it, we can select any one of its two end vertices, i.e., either i or its neighbor. In this case, we therefore find $\tilde{\pi}_i = *$, i.e., vertex i is found to be in the joker state.

At this point, the independence assumption of all $j \in N(i)$ in the cavity graph enters into the discussion. We consider their joint probability of simultaneously being cov-

ered in a minimal VC of the cavity graph G_i, and assume this quantity to factorize into $\Pi_{j \in N(i)} \pi_{j|i}$. Under this assumption, case (a) happens if and only if all $\tilde{\pi}_{j|i} \neq 0$. Case (b) happens if there are at least two vanishing $\tilde{\pi}_{j|i}$ in between the $j \in N(i)$, and the third case appears for exactly one zero $\tilde{\pi}_{j|i}$. We see that in this rule no difference between always covered and joker vertices $j \in N(i)$ exists, which explains the use of the minimum warning in definition (6). We conclude

$$
\tilde{\pi}_i = \begin{cases} 0 & \text{if } \sum_{j \in N(i)} u_{j \to i}(\tilde{\pi}_{j|i}) = 0, \\ * & \text{if } \sum_{j \in N(i)} u_{j \to i}(\tilde{\pi}_{j|i}) = 1, \\ 1 & \text{if } \sum_{j \in N(i)} u_{j \to i}(\tilde{\pi}_{j|i}) > 1. \end{cases} \tag{8}
$$

This rule is graphically represented in Fig. 1. The cavity quantities $\tilde{\pi}_{j|i}$ can now be calculated by considering the cavity graphs G_j, and by disregarding in addition the influence of vertex i,

$$
\tilde{\pi}_{j|i} = \begin{cases} 0 & \text{if } \sum_{k \in N(j) \setminus i} u_{k \to j}(\tilde{\pi}_{k|j}) = 0, \\ * & \text{if } \sum_{k \in N(j) \setminus i} u_{k \to j}(\tilde{\pi}_{k|j}) = 1, \\ 1 & \text{if } \sum_{k \in N(j) \setminus i} u_{k \to j}(\tilde{\pi}_{k|j}) > 1. \end{cases} \tag{9}
$$

Equations (7)–(9) are called *warning propagation*. The last equation, together with Eq. (7), describes a closed system of $2|E|$ equations, two for each edge due to the two different possible orientations of the messages. Note that Eqs. (7) and (9) can also be reformulated for the warning $u_{j \to i}$ itself, eliminating the cavity quantities $\tilde{\pi}_{j|i}$. The iterative equations take the particularly simple form

$$
u_{j \to i} = \delta \left(\sum_{k \in N(j) \setminus i} u_{k \to j}, 0 \right), \tag{10}
$$

where, for better readability, we have used the notation $\delta(\cdot, \cdot)$ for the Kronecker symbol.

These equations have to be solved and plugged into Eq. (8) in order to calculate the values of all $\tilde{\pi}_i$. Even if this information is not yet sufficient to immediately solve the minimal VC problem, we can already read off a lot of useful information about the properties of all minimal vertex covers. The most important quantity is an estimate for the minimal VC cardinality,

$$
X = \sum_{i \in V} \left[\delta(\tilde{\pi}_i, 1) + \frac{1}{2} \delta(\tilde{\pi}_i, *) \right]. \tag{11}
$$

The prefactor 1/2 in front of the number of joker vertices is not a direct result of WP. It can be justified using the more detailed belief propagation calculating the full single-site probability π_i [8], or via the replica method [16].

The WP equations can be used to construct a vertex cover, i.e., they can be exploited algorithmically. This is done in the following way, starting with an initial graph $G=(V,E)$ and an empty set $U=\varnothing$:
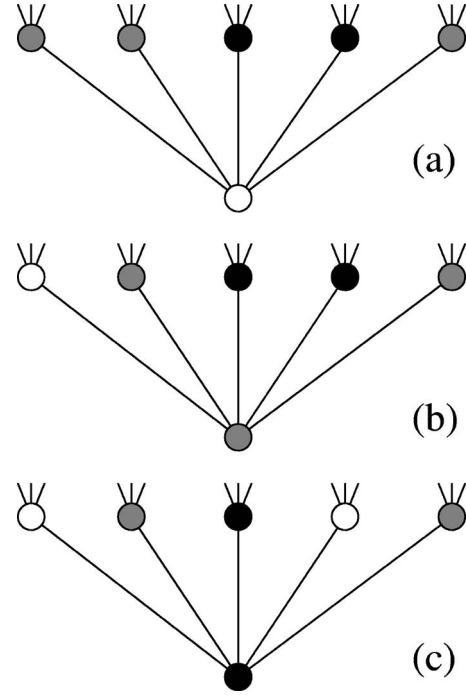

FIG. 1. Graphical representation of Eq. (8), with vertex i being identified with the lower vertex in each subfigure. The color coding of the vertices corresponds to the values of $\tilde{\pi}_i$ and $\tilde{\pi}_{j|i}$: Value zero is represented by a white dot, value one by a black dot, and the joker value * by a gray dot. In case (a), there are no white dots between the $j \in N(i)$, so the lower vertex is not to be covered and gets the color white. If there is exactly one white dot in the upper line, the lower vertex becomes gray, cf. (b). If there are two or more white dots in the upper line, as in (c), the lower vertex is black, corresponding to an always covered vertex.

(1) The $2|E|$ warnings $u_{i \to j}$ are initialized randomly.

(2) Then, sequentially, edges are selected and the warnings are updated using Eq. (10). This update is iterated until a solution of the warning-propagation equations is found.

(3) The $\tilde{\pi}_i$ are calculated from the warnings using Eq. (8).

(4) All vertices with $\tilde{\pi}_i=1$ are added to U and deleted with their incident edges from G.

(5) All vertices with $\tilde{\pi}_i=0$ are deleted from V, without changing U. Since a vertex with $\tilde{\pi}_i=0$ has only neighbors of $\tilde{\pi}_i=1$, it was already isolated after the last step. No edges therefore have to be removed from E.

(6) One remaining vertex i ($\tilde{\pi}_i=*$) is selected, and all its neighbors $j \in N(i)$ are added to U. Vertices i and $N(i)$ are removed from V and all their incident edges are subtracted from E.

(7) If uncovered edges are left, we go back to step 2 and recalculate the warnings on the decimated graph. If no edges are left, the current U is returned as a solution.

Obviously, the constructed U forms a vertex cover since only covered edges are removed from the graph. It is also a minimal one if the information provided by the $\tilde{\pi}_i$ was correct. Due to the factorization hypothesis in WP, some of the $\tilde{\pi}_i$ may, however, be erroneous, resulting possibly in a nonminimal cover. It is worth noting that after each graph decimation step followed by a reiteration of the WP equations, a

new estimate of the VC size can be calculated according to Eq. (11). This estimate is expected to be stationary only in the case where already the initial warnings were exact, and to change under the algorithm if the latter were only approximations.

B. From single samples to average results on random graphs

Starting from Eqs. (10) and (8), we can easily reconstruct the replica-symmetric typical-case results for random graphs of average vertex degree c. We start with defining the global histogram of warnings,

$$Q(u) = \frac{1}{2|E|} \sum_{(i,j) \in E} [\delta(u_{i \to j}, u) + \delta(u_{j \to i}, u)]. \quad (12)$$

Due to the binary nature of the warnings, it can be parametrized as

$$Q(u) = \rho_0 \delta(u, 0) + \rho_1 \delta(u, 1) \quad (13)$$

with $\rho_0 + \rho_1 = 1$. Consider now Eq. (10): A nontrivial warning is sent via a link $j \to i$ only if the input messages $u_{k \to j}$ from all $k \in N(j) \backslash i$ equal zero. This happens for all warnings independently with probability ρ_0, and the number d of these incoming messages is, on a random graph, distributed according to a Poissonian of mean c. We thus find

$$\rho_1 = \sum_{d=0}^{\infty} e^{-c} \frac{c^d}{d!} \rho_0^d = e^{-c\rho_1}, \quad (14)$$

which, using the Lambert-W function, is solved by

$$\rho_1 = \frac{W(c)}{c}. \quad (15)$$

Let us now reconstruct also the histogram

$$P(\tilde{\pi}) = \frac{1}{N} \sum_{i \in V} \delta(\tilde{\pi}_i, \tilde{\pi}) \quad (16)$$

of single-site marginals $\tilde{\pi}_i$. The latter are three valued; we thus parametrize the histogram as

$$P(\tilde{\pi}) = \nu_0 \delta(\tilde{\pi}, 0) + \nu_* \delta(\tilde{\pi}, *) + \nu_1 \delta(\tilde{\pi}, 1). \quad (17)$$

Using Eq. (8) and the Poissonian degree distribution, we find

$$\nu_0 = \sum_{d=0}^{\infty} e^{-c} \frac{c^d}{d!} \rho_0^d = \rho_1,$$

$$\nu_* = \sum_{d=0}^{\infty} e^{-c} \frac{c^d}{d!} d\rho_0^{d-1} \rho_1 = c\rho_1^2,$$

$$\nu_1 = 1 - \nu_* - \nu_0. \quad (18)$$

For the derivation of the second expression we have utilized the fact that the single nonzero warning among the messages reaching a joker vertex can be chosen liberally in between all d incoming edges. For the VC size we thus find

$$x(c) = \lim_{N \to \infty} \frac{X}{N} = 1 - \frac{W(c)}{c} - \frac{W(c)^2}{2c}, \quad (19)$$

which is identical to the result of a replica-symmetric calculation [17]. For average degrees $c < e$, this result was shown to be exact [18], and it can in fact already be read off from an older result by Karp and Sipser [29] on maximal matchings in random graphs (see also Refs. [30,31] for related statistical-physics approaches).

C. Bug proliferation and the stability of the WP fixed point

Besides the problem that the solution to the equations of warning propagation may be imprecise due to the existence of short loops in the graph, there can be another problem—the iteration of the warning update may fail to converge. This can happen again due to the existence of short loops, which may lead to attractive limit cycles in the iterative warning dynamics. Another problem can appear due to the existence of *many* solutions of Eqs. (9). In statistical physics we say that the replica symmetry is broken.

To be more quantitative, we study here the stability of a WP solution with respect to the introduction of a *bug* [32]: One of the warnings $u_{j \to i}$ is changed to its opposite value. After one iteration of WP, the bug itself will be cured since it depends only on unchanged messages. On the other hand, the warnings from vertex i to its neighbors $k \in N(i) \backslash j$ may be changed, i.e., new bugs may appear. The question is now if these bugs proliferate and, after some iterations, change a finite fraction of all warnings, or, if the bugs die out after a while. Only in the second case, WP is stable and can be usefully included into a decimation procedure.

Here, we perform this analysis analytically for the case of a random graph of average degree c. In this case, the number d of neighbors $k \in N(i) \backslash j$ receiving messages depending on the bug is distributed according to the Poissonian $e^{-c} c^d / d!$. They send themselves warnings $u_{k \to i}$ to vertex i which are, due to the locally treelike structure of a random graph, independent of $u_{j \to i}$, and can be considered to be randomly selected according to the global histogram $Q(u) = \rho_0 \delta(u, 0) + \rho_1 \delta(u, 1)$ of warnings introduced in Eq. (12).

We have to distinguish two cases for the introduction of a bug:

(i) We change the message $u_{j \to i}$ from *one to zero*. Prior to this change, all out messages $u_{i \to k}$ with $k \in N(i) \backslash j$ were equal to zero [cf. Eq. (10)]. Let us denote by $d = |N(i) \backslash j|$ the number of the out messages depending on $u_{j \to i}$, i.e., the degree of vertex i equals $d+1$. After the introduction of the bug, an out message $u_{i \to k}$ becomes one if and only if all other in messages $u_{l \to i}$ with $l \in N(i) \backslash \{j, k\}$ are zero. There are two subcases. First, with probability ρ_0^d, all messages $u_{k \to i}$ equal zero. In this case, all d out messages change. Second, with probability $d\rho_0^{d-1}(1 - \rho_0)$, exactly one message $u_{k \to i}$ has value one, all others are zero. In this case, only $u_{i \to k}$ changes under iteration. On average, the bug introduces thus

$$\sum_d e^{-c} \frac{c^d}{d!} [d\rho_0^d + d\rho_0^{d-1}(1 - \rho_0)] = ce^{-c(1-\rho_0)} = ce^{-c\rho_1}$$

new bugs into the graph. These bugs are of the second type.

(ii) We change the message $u_{j \to i}$ from *zero to one*. After introduction of the bug, all out messages $u_{i \to k}$ with $k \in N(i) \backslash j$ become zero under the WP update. They are bugs only if, in the initial WP solution, they had the value one. Using analogous arguments to the first case, we find that, with probability ρ_0^d, all d out messages were one, and with probability $d \rho_0^{d-1}(1-\rho_0)$, only one single message was one and becomes a new bug. The expected number of new bugs caused by the changed $u_{j \to i}$ equals again $ce^{-c\rho_1}$.

We now apply a simple percolation-type argument: If the average number of new bugs is smaller than one, the bugs are expected to be cured after a few iterations; the WP solution is stable under bug introduction. If, on the other hand, the average number of new bugs is larger than one, we expect an exponential increase in the bug number. Bugs proliferate and carry away the system from the WP fixed point. The latter is thus concluded to be unstable. Note that this argument holds only because we update out messages which, under iterated WP updates, do not interact because they all influence disjoint sets of further warnings.

The critical point can now be determined easily: The average number of new bugs is set to one, $c_{WP} e^{-c_{WP} \rho_1} = 1$. Comparing it to the self-consistent Eq. (15), we immediately conclude the

$$c_{WP} = \frac{1}{\rho_1(c_{WP})} = e. \tag{20}$$

WP converges below this critical connectivity, i.e., in the full region where replica symmetry is exact. As one would expect intuitively, it does not converge in the replica-symmetry broken phase above average degree e; there survey propagation as discussed in the following section of this work has to be applied.

D. Bug relaxation time of WP

Even if WP provides asymptotically exact results in the full replica-symmetric phase in a running time scaling quadratically with N, its convergence slows down if we approach the critical average degree. This can be seen analytically by calculating the evolution of the number of erroneous messages, or bugs, under various update schemes.

1. Parallel update

Let us start with a parallel-update scheme, where, in every iteration step, all messages are recalculated simultaneously from the old messages. Assume that there are $M_1 \ll 2M = 2|E|$ erroneous messages. These are, up to higher-order effects, isolated from each other and act thus independently under WP iteration. Each of these bugs becomes corrected in a new WP step, but causes, as seen in the last subsection, on average $ce^{-c\rho_1} = c\rho_1$ new wrong messages. Again, up to higher-order corrections, these messages do not interact. For the expected number $\overline{M_1(t)}$ we find

$$\overline{M_1(t)} = (c\rho_1)^t \overline{M_1(0)}, \tag{21}$$

i.e., below $c_{WP} = e$, this number decays exponentially with a time scale

$$\tau_{par} = -\frac{1}{\ln(c\rho_1)}. \tag{22}$$

This relaxation time diverges as we approach $c = e$ from below. To unveil the critical behavior, we set $c = e - \varepsilon$ ($0 < \varepsilon \ll 1$). With $\rho_1 = 1/e + \delta$ we find, using Eq. (15),

$$\frac{1}{e} + \delta = \exp\left\{ -1 + \frac{\varepsilon}{e} - e\delta + O(\varepsilon\delta) \right\}, \tag{23}$$

i.e., $\delta = \varepsilon/(2e^2)$, resulting in $c\rho_1 = 1 - \varepsilon/(2e) + O(\varepsilon^2)$, and thus in

$$\tau_{par} \simeq \frac{2e}{e-c} \quad \text{for } 0 < e - c \ll 1. \tag{24}$$

The critical exponent one is expected to result from the mean-field structure of the underlying graph.

2. Random update

The situation is slightly more involved in the case of a random update, where in every time step one message is selected randomly out of all $2M$ warnings, and is updated according to the WP equation. Let us denote by $p_T(M_1)$ the probability that there are M_1 erroneous messages after T random-update steps. Its evolution under WP is given by the rate equation

$$p_{T+1}(M_1) = p_T(M_1) - \frac{M_1}{2M} p_T(M_1) + \frac{M_1+1}{2M} p_T(M_1+1)$$
$$- \frac{c\rho_1 M_1}{2M} p_T(M_1) + \frac{c\rho_1(M_1-1)}{2M} p_T(M_1-1). \tag{25}$$

This is due to the fact that, with probability $M_1/(2M)$ we pick a bug and correct it—changing $M_1 \mapsto M_1 - 1$, and with probability $c\rho_1 M_1/(2M)$ we pick a "child" of a bug, which becomes changed (remember that a bug has on average c children messages, but only a fraction ρ_1 of these becomes changed when updated under WP)—changing $M_1 \mapsto M_1 + 1$.

After $2M$ random updates, each message is, on average, visited once. To obtain time scales comparable to the parallel update, we therefore have to rescale time as $t = T/(2M)$, identifying a single random update with the asymptotically infinitesimal time step $dt = 1/(2M)$. In this limit, Eq. (25) can be rewritten as a system of ordinary differential equations,

$$\frac{d}{dt} p_t(M_1) = -M_1 p_t(M_1) + (M_1+1)p_t(M_1+1)$$
$$- c\rho_1 M_1 p_t(M_1) + c\rho_1(M_1-1)p_t(M_1-1). \tag{26}$$

For the time evolution of the average number $\overline{M_1(t)} = \Sigma_{M_1} M_1 p_t(M_1)$ of bugs we thus find

$$\frac{d}{dt}\overline{M_1(t)} = -\overline{M_1^2} + \overline{M_1(M_1-1)} - c\rho_1\overline{M_1^2} + c\rho_1\overline{M_1(M_1+1)}$$

$$= -(1 - c\rho_1)\overline{M_1(t)}. \qquad (27)$$

It decays exponentially with

$$\tau_{rand} = \frac{1}{1 - c\rho_1} \qquad (28)$$

and shows the same critical behavior as the parallel update. The main difference appears for $c \to 0$: Whereas the parallel relaxation time goes to zero, τ_{rand} approaches one. This reflects the persistence time that a message is not updated at all: The fraction of variables that are not selected in N single-spin updates is e^{-1}.

Note that the algorithm as presented in Sec. IV A uses a third update scheme, namely, a sequential update, which is asynchronous but sees every message exactly once in $2M$ steps. The analytical description is more involved than the one of a simple parallel or random update, but the critical behavior is expected to remain unchanged. For small c, the behavior is further on expected to be more similar to the one of the parallel update scheme. Since every message is seen exactly once in $2M$ steps, there are no persistence effects.

E. Numerical tests

We have performed numerical tests of WP on randomly generated instances of random graphs at various connectivities $c < e$, for graph sizes up to $N = 10^5$.

To verify the results, we have also applied the *leaf-removal algorithm*, which was shown [18] to output exact results exactly in the same connectivity region, and which is the basis of the proof of correctness of the replica-symmetric result. The algorithm works as follows: In every step, a leaf (vertex of degree one) is selected, its neighbor is covered and both vertices are removed from the graph, as well as all their incident edges. If this algorithm is able to cover the full graph, the generated VC is a minimum one, but the algorithm fails if, possibly after some decimation steps, a leaf-free subgraph emerges.

We have found that both algorithms produce in almost all cases identical results, i.e., the WP output is thereby shown to be exact. Also, the initial estimate of the VC size after the first convergence of WP, before starting graph decimation, was found to coincide in the majority of all cases with the final output. As discussed above, this is a signal that already the first convergence of WP leads to exact messages.

The problem of WP is, as discussed before, the slowing down and final nonconvergence if we approach (or exceed) an average degree $c = e$. In Fig. 2, we have quantified this phenomenon. We have measured the fraction of graphs of given average degree c (and given N) which, within 1000 sequential WP updates of all $2M$ messages, are converged on more than 99% of all messages. In the figure, we see a clear drop of this probability from almost one to zero in a region concentrated close to $c = e$. This drop sharpens considerably with growing graph size N, and suggests the existence of a sharp transition in the WP behavior in the thermodynamic
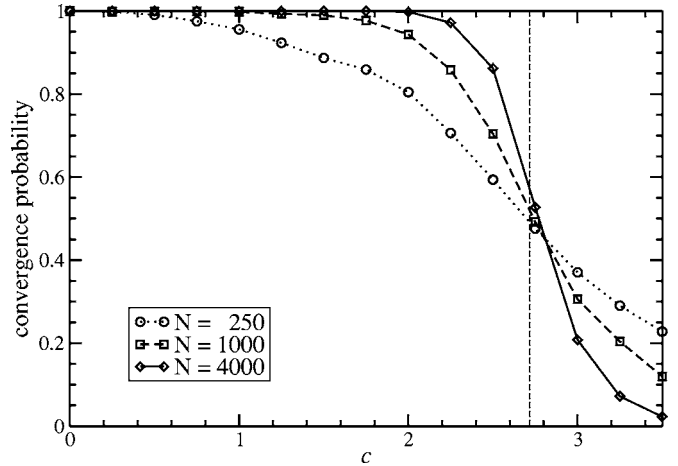

FIG. 2. Convergence probability of WP as a function of the average degree, for graphs of $N = 250$, 1000, and 4000 vertices. The symbols signify the fraction of graphs, where after 1000 sequential updates at least 99% of the warnings are converged, measured for 10 000, 3000, and 1000 sample graphs. The dashed vertical line is situated at $c = e$, where WP theoretically ceases to converge.

limit $N \to \infty$. Note that in Fig. 2, this transition seems to be at a graph degree being slightly larger than $c = e$. This is a result of the measured quantity. The transition should be found exactly in $c = e$ when for an arbitrarily large, but finite number of updates almost all messages are converged—instead of the test values used in the generation of Fig. 2.

V. SURVEY PROPAGATION

We have already mentioned the possibility that the equations of warning propagation possess a high number of solutions, and none can be found using a local iterative update scheme. The messages would try to converge to different, conflicting solutions in different regions of the graph, and global convergence cannot be achieved. In physics' language, these different solutions correspond to different thermodynamic states—to be understood as clusters of minimal VCs, as illustrated in Fig. 3. Inside such a cluster, any two
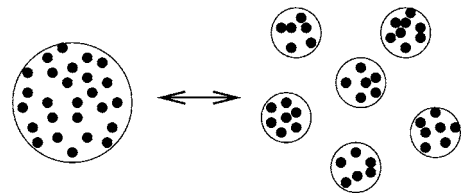

FIG. 3. Schematic graphical representation of the organization of optimal solutions for warning propagation (left side) and for survey propagation (right side). For the first case, all solutions are collected in one large unstructured cluster (or in a very small number of these clusters, as in the case of a ferromagnet), corresponding to unbroken replica symmetry. In the second case, the set of solutions is clustered into a large number of extensively separated subsets. Survey propagation corresponds to one step of replica symmetry breaking, where there is no further organization of the clusters in larger clusters.

VCs are connected by at least one path via other (almost) minimal VCs, which differ stepwise only by a small number of elements (the number of these different elements stays finite in the thermodynamic limit). For two minimal VCs selected from two different clusters, no such connecting paths exist, at least once an extensive step has to be performed. Note that this distinction is, from a mathematical point of view, not well defined for finite graphs—which are the objects of our algorithms. There can be, however, a clear separation of distance scales, which practically allows for an identification of solution clusters.

As already said, warning propagation works well only if there is a single cluster (or a very small number of clusters), corresponding to the replica symmetric solution. A breaking of the replica symmetry implies the emergence of clustering in the solution space. This effect is taken into account by the *survey propagation* (SP) algorithm, as first proposed in Refs. [11,33]. This algorithm is equivalent to the first step of replica symmetry breaking, where the solution clusters show no further organization. If there are clusters of clusters, etc., one has to go beyond survey propagation.

A. Algorithm

Let us, however, assume a clustering only on one level. Instead of defining probabilities like π_i over the full solution space, we consider for a moment only one cluster. Inside such a cluster of minimum VCs, a vertex i may either be always covered (state 1), never covered (state 0), or sometimes covered and sometimes not (joker state *). This means that for single clusters we treat the problem on the same level as WP.

However, the assignment of this three-valued vertex state may vary from cluster to cluster. We now denote by $\hat{\pi}_i^{(1)}$ the fraction of clusters where vertex i takes state one, by $\hat{\pi}_i^{(0)}$ the fraction of clusters where vertex i takes state zero, and by $\hat{\pi}_i^{(*)}$ the fraction of clusters where vertex i takes the joker state *. Analogously we define the cavity quantities $\hat{\pi}_{j|i}^{(1)}$, $\hat{\pi}_{j|i}^{(0)}$, and $\hat{\pi}_{j|i}^{(*)}$ on the cavity graph G_i. A crucial assumption of SP is that clusters do not change dramatically by eliminating one vertex from the graph, i.e., by going back and forth between the full graph and the cavity graphs for different cavities.

Again, we can distinguish the three cases in Fig. 1 of how the variable states propagate inside each solution cluster. A vertex i of state 0 has to have all neighbors in states 1 or * on the cavity graph G_i; a vertex i of state * has to have exactly one neighbor of state 0 on the cavity graph; a vertex i of state 1 has at least two neighbors which have state 0 on the cavity graph. The statistics over all clusters can now be performed in a very simple way. The fraction of clusters having vertex i in state 0, which by definition is $\hat{\pi}_i^{(0)}$, equals the fraction of solution clusters of the cavity graph G_i where all neighbors are in a state different from 0, and so on, for the other two states. This procedure guarantees the minimization *inside* each cluster. Note, however, that in clusters belonging to the first case no vertex has to be added to the minimal VC by stepping from the cavity graph to the full graph, whereas the VC size increases by one in the second and third case. The

VCs of different clusters thus grow differently. To optimize *between* clusters, we therefore introduce a penalty e^{-y} to the last two cases. The resulting equations are

$$\hat{\pi}_i^{(0)} = C_i^{-1} \prod_{j \in N(i)} (1 - \hat{\pi}_{j|i}^{(0)}),$$

$$\hat{\pi}_i^{(*)} = C_i^{-1} e^{-y} \sum_{j \in N(i)} \hat{\pi}_{j|i}^{(0)} \prod_{j' \in N(i) \backslash j} (1 - \hat{\pi}_{j'|i}^{(0)}),$$

$$\hat{\pi}_i^{(1)} = C_i^{-1} e^{-y} \left[1 - \prod_{j \in N(i)} (1 - \hat{\pi}_{j|i}^{(0)}) \right. $$
$$\left. - \sum_{j \in N(i)} \hat{\pi}_{j|i}^{(0)} \prod_{j' \in N(i) \backslash j} (1 - \hat{\pi}_{j'|i}^{(0)}) \right], \quad (29)$$

and the normalization constant is given by

$$C_i = e^{-y} \left[1 - (1 - e^y) \prod_{j \in N(i)} (1 - \hat{\pi}_{j|i}^{(0)}) \right]. \quad (30)$$

Note that we have again made an assumption of statistical independence of the vertices j on the cavity graph. This assumption enters on two levels: First, inside the cluster, when we say that j vertices of state * can be covered simultaneously in a minimum VC of the cavity graph; and second, in between clusters, when we factorize the joint probabilities in the upper expression.

Analogous equations are valid for the iteration of the cavity quantities, where again the influence of the cavity site has to be taken out;

$$\hat{\pi}_{i|l}^{(0)} = C_{i|l}^{-1} \prod_{j \in N(i) \backslash l} (1 - \hat{\pi}_{j|i}^{(0)}),$$

$$\hat{\pi}_{i|l}^{(*)} = C_{i|l}^{-1} e^{-y} \sum_{j \in N(i) \backslash l} \hat{\pi}_{j|i}^{(0)} \prod_{j' \in N(i) \backslash \{j,l\}} (1 - \hat{\pi}_{j'|i}^{(0)}),$$

$$\hat{\pi}_{i|l}^{(1)} = C_{i|l}^{-1} e^{-y} \left[1 - \prod_{j \in N(i) \backslash l} (1 - \hat{\pi}_{j|i}^{(0)}) \right. $$
$$\left. - \sum_{j \in N(i) \backslash l} \hat{\pi}_{j|i}^{(0)} \prod_{j' \in N(i) \backslash \{j,l\}} (1 - \hat{\pi}_{j'|i}^{(0)}) \right],$$

$$C_{i|l} = e^{-y} \left[1 - (1 - e^y) \prod_{j \in N(i) \backslash l} (1 - \hat{\pi}_{j|i}^{(0)}) \right]. \quad (31)$$

These are the equations for SP. To solve them, one has to first initialize the cavity quantities arbitrarily and update them iteratively according to the second set of equations. Once convergence is reached, the $\hat{\pi}_i^{(\cdot)}$ can be simply evaluated from the first set of equations. Note also that the SP equation for the cavity quantities close in the $\hat{\pi}_{j|i}^{(0)}$ alone;

$$\hat{\pi}_{i|l}^{(0)} = \frac{\displaystyle\prod_{j \in N(i) \backslash l} (1 - \hat{\pi}_{j|i}^{(0)})}{e^{-y} \left[1 - (1 - e^y) \displaystyle\prod_{j \in N(i) \backslash l} (1 - \hat{\pi}_{j|i}^{(0)}) \right]}. \quad (32)$$

A note on the selection of the reweighting parameter y is necessary: Finite values of y focus on *local minima* of the

complex landscape $X(U) = |U|$ defined over all VCs, i.e., to VCs of cardinality, which cannot be decreased by changing only a finite part of U. One would thus expect naively that minimal VCs are obtained in the limit $y \to \infty$. As we will see in the next section, the SP solution carries, however, sensible physical information only in a limited interval $y \in [0, y^*]$. It is therefore necessary to work directly with finite y values.

The knowledge of all $\hat{\pi}_i^{(\cdot)}$ does not allow us to directly create a (locally) minimal vertex cover. It is impossible to deduce a joint probability distribution of all vertices from the knowledge of the marginal single-vertex probabilities only. Nevertheless, some useful knowledge can be drawn directly from these quantities. In particular, we may estimate the VC size by

$$X(y) = \sum_{i \in V} \left(\hat{\pi}_i^{(1)} + \frac{1}{2} \hat{\pi}_i^{(*)} \right). \tag{33}$$

As in the replica-symmetric case of WP, we have assumed that vertices carrying value * are, on average, half covered and half uncovered. At this point, this is a pure conjecture which, however, will be justified in the next section [see Eq. (60)].

To actually construct a minimum vertex cover (or an approximation due to the non-exactness of SP because of, e.g., a finite value of y, cycles in the graph, or more levels of cluster organization), we have to resort again to an iterative decimation scheme. In every step, the $\hat{\pi}_i^{(\cdot)}$ are calculated; one vertex of large $\hat{\pi}_i^{(1)}$ is selected and covered. It is removed from the graph together with all incident edges, and the $\hat{\pi}_i^{(\cdot)}$ are reiterated on the decimated graph. This procedure is iterated until no uncovered edges are left, i.e., until a proper VC is constructed. Slightly different schemes of selection heuristics can be applied (select a vertex of high $\hat{\pi}_i^{(0)}$, uncover it, cover all neighbors, and decimate the graph, or take into account also the value of $\hat{\pi}_i^{(*)}$). All these heuristic rules are equally valid in the sense that, if SP is exact on a graph, they all produce one minimum VC of the same size. For real graphs, however, where the results of the SP equations are to be considered as approximations of the actual quantities defined over the set of solutions, different heuristic choices may result in VCs of different sizes. Within the numerical experiments described below, we have, however, found no preferable selection heuristic, and the fluctuations from one heuristic to another were small compared with the VC size.

B. Complexity of clusters

Different values of the reweighting parameter y lead to a concentration of the partition sum (or, equivalently, the solution of the SP equations) to clusters of vertex covers of different (locally minimal) size. The *complexity* $\Sigma(X)$, or *configurational entropy*, measures the logarithm of the number $\mathcal{N}_{cl}(X)$ of clusters of given VC size X. We introduce the generalized thermodynamic potential $\Phi(y)$ as the Legendre transform of the complexity

$$e^{-y\Phi(y)} = \sum_{X=0}^{N} \exp[-yX + \Sigma(X)]. \tag{34}$$

According to the general procedure of the cavity method in diluted systems [10], this potential can be decomposed into site and link contributions,

$$\Phi(y) = \sum_{i \in V} \Delta\Phi_i(y) - \sum_{\{i,j\} \in E} \Delta\Phi_{i,j}(y). \tag{35}$$

These contributions can be determined by adding a vertex or an edge to the graph;

$$e^{-y\Delta\Phi_i(y)} = \prod_{j \in N(i)} (1 - \hat{\pi}_{j|i}^{(0)}) + \left(1 - \prod_{j \in N(i)} (1 - \hat{\pi}_{j|i}^{(0)}) \right) e^{-y}$$

$$= e^{-y} + (1 - e^{-y}) \prod_{j \in N(i)} (1 - \hat{\pi}_{j|i}^{(0)}),$$

$$e^{-y\Delta\Phi_{i,j}(y)} = 1 - (1 - e^{-y}) \hat{\pi}_{j|i}^{(0)} \hat{\pi}_{i|j}^{(0)}, \tag{36}$$

where, as in the derivation of the SP equations, one has to take care separately of the cases where the VC size remains unchanged under vertex or link addition, or increases by one. Having solved the SP equations for the $\hat{\pi}_{i|j}^{(0)}$, the potential becomes easy to calculate;

$$-y\Phi(y) = \sum_{i \in V} \ln\left(e^{-y} + (1 - e^{-y}) \prod_{j \in N(i)} (1 - \hat{\pi}_{j|i}^{(0)}) \right)$$

$$- \sum_{\{i,j\} \in E} \ln[1 - (1 - e^{-y}) \hat{\pi}_{j|i}^{(0)} \hat{\pi}_{i|j}^{(0)}]. \tag{37}$$

Approximating the sum in Eq. (34) by the saddle-point method (valid for $N \gg 1$), we see that the complexity can be calculated via

$$\Sigma(y) = \Sigma[X(y)] = y[X(y) - \Phi(y)], \tag{38}$$

where $X(y)$ is given in Eq. (33) in dependence of the SP solution. The function $X(y)$ can also be determined directly from the potential $\Phi(y)$ via $X(y) = \Phi(y) + y\Phi'(y)$. The numerical observation that both expressions for $X(y)$ coincide is a strong justification for the ratio 2 used in Eq. (33) between the number of all unfrozen vertices and the number of simultaneously covered unfrozen vertices.

The complexity Σ is defined as the logarithm of the cluster number, i.e., in the presence of at least one cluster it takes necessarily a nonnegative value. This defines a range $y \in (0, y^*)$ where the SP equation provides a potentially sensible solution, with y^* given by the marginality condition $\Sigma(y^*) = 0$. For higher y, the predicted complexities become negative, thus corresponding to unphysical solutions of the SP equations. We see that the naive expectation that $y \to \infty$ leads to minimal VCs is thus inconsistent; the best possible estimate for the minimal VC size we can obtain at the level of SP (one-step replica symmetry breaking) is thus given by $X(y^*)$ [19]. Note that this observation, in replica theory, corresponds to the usual optimization of the replicated free energy over the replica symmetry-breaking parameter [34,35]. Note also that the existence of a finite y^* is a clear signal for the existence of more than one step of replica symmetry breaking, and the SP results can only be expected to be approximations to true minimal VCs.

C. Stability of the fixed point under SP iteration

It is, however, not clear if SP converges at all in the replica-symmetry broken phase. To investigate this question, we consider the behavior of the solution of Eq. (32) under small perturbations. Note that the situation here is different from the bug proliferation picture used in order to analyze the stability of WP fixed points: The messages now are real numbers and thus small perturbations are possible even on the level of a single message.

Let us therefore imagine that we start a set of experiments with initial conditions $\pi_{i|l}^{(0)}$ distributed around the SP solution $\hat{\pi}_{i|l}^{(0)}$ according to some narrow distribution

$$f_{i|l}(\pi_{i|l}^{(0)}) = \frac{1}{\sqrt{2\pi}\varepsilon_{i|l}} \exp\left\{-\frac{[\pi_{i|l}^{(0)} - \hat{\pi}_{i|l}^{(0)}]^2}{2\varepsilon_{i|l}^2}\right\} \quad (39)$$

of link-dependent widths $\varepsilon_{i|l} \ll 1$. After one iteration of SP, the messages are distributed according to [for simplicity we have skipped the superscript (0)]

$$f_{i|l}'(\pi_{i|l}) = \int \prod_{j \in N(i)\backslash l} [d\pi_{j|i} f_{j|i}(\pi_{j|i})] \delta[\pi_{i|l} - \tilde{\pi}_{i|l}(\{\pi_{j|i}\})], \quad (40)$$

with the update rule $\tilde{\pi}$ given by Eq. (32). We expand this update function around the SP fixed point,

$$\tilde{\pi}_{i|l}(\{\pi_{j|i}\}) = \hat{\pi}_{i|l}^{(0)} + \sum_{j \in N(i)\backslash l} \frac{\partial \hat{\pi}_{i|l}^{(0)}}{\partial \hat{\pi}_{j|i}^{(0)}} (\pi_{j|i} - \hat{\pi}_{j|i}^{(0)})$$
$$+ \frac{1}{2} \sum_{k,j \in N(i)\backslash l} \frac{\partial^2 \hat{\pi}_{i|l}^{(0)}}{\partial \hat{\pi}_{k|i}^{(0)} \partial \hat{\pi}_{j|i}^{(0)}} (\pi_{k|i} - \hat{\pi}_{k|i}^{(0)})(\pi_{j|i} - \hat{\pi}_{j|i}^{(0)})$$
$$+ O(\varepsilon^3). \quad (41)$$

The mean of the updated message is given by

$$\langle \pi_{i|l} \rangle' = \int \prod_{j \in N(i)\backslash l} [d\pi_{j|i} f_{j|i}(\pi_{j|i})] \tilde{\pi}_{i|l}(\{\pi_{j|i}\})$$
$$= \hat{\pi}_{i|l}^{(0)} + \frac{1}{2} \sum_{j \in N(i)\backslash l} \frac{\partial^2 \hat{\pi}_{i|l}^{(0)}}{(\partial \hat{\pi}_{j|i}^{(0)})^2} \varepsilon_{j|i}^2, \quad (42)$$

and its change is negligible with respect to the width of the distribution. The second moment, on the other hand, behaves as

$$\langle \pi_{i|l}^2 \rangle' = \int \prod_{j \in N(i)\backslash l} [d\pi_{j|i} f_{j|i}(\pi_{j|i})] \tilde{\pi}_{i|l}^2(\{\pi_{j|i}\})$$
$$= (\hat{\pi}_{i|l}^{(0)})^2 + \hat{\pi}_{i|l}^{(0)} \sum_{j \in N(i)\backslash l} \frac{\partial^2 \hat{\pi}_{i|l}^{(0)}}{(\partial \hat{\pi}_{j|i}^{(0)})^2} \varepsilon_{j|i}^2$$
$$+ \sum_{j \in N(i)\backslash l} \left(\frac{\partial \hat{\pi}_{i|l}^{(0)}}{\partial \hat{\pi}_{j|i}^{(0)}}\right)^2 \varepsilon_{j|i}^2. \quad (43)$$

We find thus that the variance of the updated distribution behaves as

$$\varepsilon_{i|l}'^2 = \langle \pi_{i|l}^2 \rangle' - \langle \pi_{i|l} \rangle'^2 = \sum_{j \in N(i)\backslash l} T_{i|l,j|i} \varepsilon_{j|i}^2, \quad (44)$$

with

$$T_{i|l,j|i} = \left(\frac{\partial \hat{\pi}_{i|l}^{(0)}}{\partial \hat{\pi}_{j|i}^{(0)}}\right)^2 = \left(\frac{e^{-y} \prod_{k \in N(i)\backslash\{j,l\}} (1 - \hat{\pi}_{k|i}^{(0)})}{\left[e^{-y} + (1 - e^{-y}) \prod_{k \in N(i)\backslash j} (1 - \hat{\pi}_{k|i}^{(0)})\right]^2}\right)^2. \quad (45)$$

The (in)stability of this equation is related to the largest eigenvalue λ_{max} of the matrix $(T_{i|l,j|i})$, only if λ_{max} is smaller than one the perturbations $f_{i|l}(\pi_{i|l}^{(0)})$ of the SP solution contract exponentially.

Note that this type of stability of the SP fixed point is known in the literature under the name "type-one instability" [32,36] and can be related to the appearance of more than one step of replica symmetry breaking, more precisely to the fragmentation of the solution clusters in subclusters. It is not the only type of instability of the one-step replica-symmetry-broken solution with respect to more steps; an alternative scheme would be the accumulation of clusters into clusters of clusters ("type-two instability"). The latter instability leads, however, not to an iterative instability of the SP equations itself; i.e., the latter can be used even if not being physically exact. This is what happens in the case of VC [19].

D. Numerical tests

1. Size of the constructed VC

In order to check the performance of SP, we have tested it on single samples of random graphs of medium to large size. In Fig. 4, we concentrate on a single graph of $N = 50\,000$ vertices and average degree $c = 10$. The data reported in Fig. 4 are quantitatively comparable to other graphs with the same parameters, and qualitatively with graphs of other sizes and connectivities. We show trajectories of the estimated VC size during graph reduction as a function of the number of vertices, which are still in the reduced graph, for various values of the reweighting parameter y. We see that the initial variability of the estimates is much larger than the difference in output. Even the worst-performing case $y = 0$ outputs a VC of 34 171 vertices compared to the minimal one found with $X = 34\,104$. This similarity is due to the fact that the ranking of the vertices with respect to the SP results depends only weakly on y, whereas the messages themselves change considerably and thus the corresponding predictions of the VC size. Close to the end of the curves, there are some striking fluctuations in the VC size. In this region, SP was not able to converge to a fixed point, and the nonconverged solution was used. This nonconvergence of SP may be related to the critical slowing down of SP at the phase boundary when the solution space of the minimal VC problem transits between the two schemes of Fig. 3 (see Sec. V). After an interval of these fluctuations, the SP solution collapses to the WP solution, i.e., we go from the replica-symmetry-broken clustered phase to the replica-symmetric unclustered phase. Note that
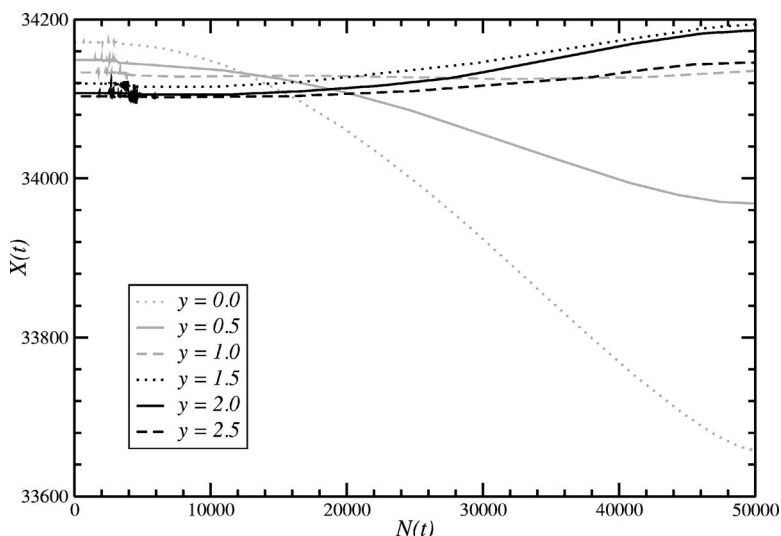
FIG. 4. Trajectories of graph decimation for a single graph with $N=50\,000$ and $c=10$. We plot the VC size as estimated by SP, as a function of the vertex number in the remaining graph. The decimation proceeds from the right to the left, i.e., from the initial $N=50\,000$ toward zero. The fact that the estimate changes under the application of the reduction process results from the approximate nature of the SP messages.

the performance of SP improves with increasing values of y, as long as it converges in the greatest majority of the decimation steps. The region of nonconvergence, however, grows with y.

To circumvent this problem, we have introduced a version of SP with adaptive y values. We start SP with a relatively large y, and whenever the convergence time exceeds a certain threshold (we have used, e.g., 100 sequential updates of all messages), the value of y is decreased (we have, e.g., multiplied it by 0.9). As a result, the trajectory of predicted VC sizes is smoothened, and the algorithm automatically tends toward the lowest-found VC sizes.

As already mentioned, the original estimate for X varies a lot with y; it is even nonmonotonous. Whereas the value for $y=0$ is substantially smaller than the smallest constructed VCs, there is a local maximum, which is larger than the constructed VCs. It is, however, astonishing that the extrapolated value at y^*, where the complexity Σ vanishes, is extremely close to the finally constructed value, $34\,090\pm10$ compared to $34\,104$. This is even more astonishing since we do not reach convergence of SP at y^* for $c=10$ (see the discussion in the next subsection).

To see the behavior of SP in the full range of average degrees, we have systematically scanned the c interval $[10, 400]$, as can be seen in Fig. 5. The graph size for these high connectivities range up to $N=6400$ (note that in this case up to $cN=2\,560\,000$ messages have to be handled). The results for fixed c and various N were extrapolated to their asymptotic value at $N\rightarrow\infty$, in order to be comparable to analytical results and to the performance of local algorithms. We see that SP performs much better than the local heuristics, and its behavior is consistent to the exact large-c behavior found by Frieze [14]. For comparison we have used two local algorithms: The first one is a simple heuristic by Gazmuri [15], where in every step a vertex is selected randomly, all its neighbors are covered, and the covered edges are removed from the graph. The second heuristic is a generalization of leaf removal [37] working also beyond average degree $c=e$, but no longer guaranteeing minimality of the constructed VC. The algorithm selects in every step a vertex of minimal degree, covers its neighbors, and removes all

considered vertices and covered edges. If the algorithm never needs to select vertices of degrees exceeding one, it reduces to leaf removal. As already said, SP outperforms the local algorithms.

A drawback for all c values is, however, that the algorithm does not work at high values of y, which, noting the derivation of the SP equations, should bring us closest to a minimal VC.

2. Iterative stability of SP

Running SP for different values of the reweighting parameter y, we observe that it converges very fast for small y (and $c\neq e$), but it does not converge at all for large y. As a first impression, it seems therefore useless to check the stability of the SP solution via the eigenvalues of the stability matrix $(T_{i|l,j|i})$. The solution itself is found via iteration of Eq. (32)
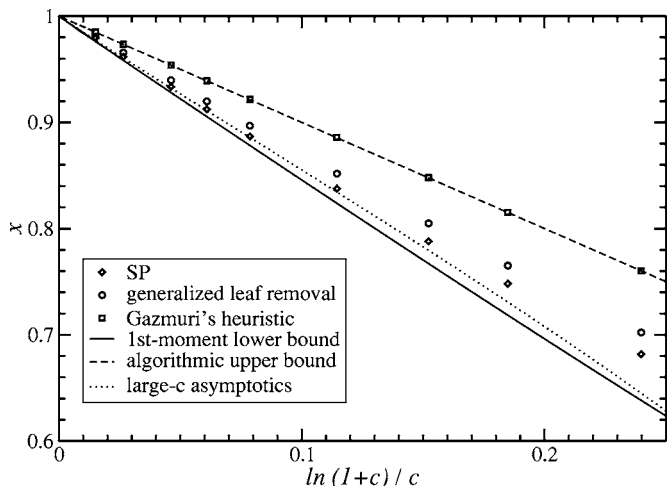

FIG. 5. Numerical results of SP run on graphs of high, but finite average degree $c \in [10, 400]$. As a comparison, we have added the results of two local algorithms (Gazmuri's heuristic and generalized leaf removal), and rigorous bounds on the asymptotic average behavior for random graphs as well as the exact large-c asymptotics. SP clearly outperforms local algorithms and is close to the asymptotically exact value.
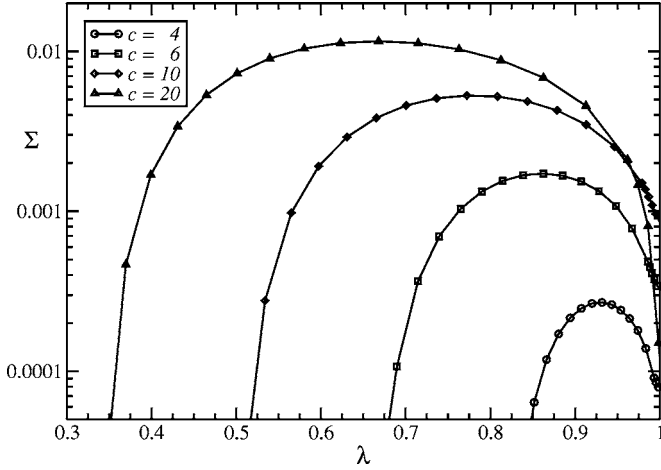
FIG. 6. Stability of the SP solution: The complexity Σ is plotted vs the largest eigenvector λ_{max} of the stability matrix $(T_{i|l,j|i})$ defined in Eq. (45), for various values of the average graph degree c. All data are produced from graphs with $N = 10\,000$ vertices averaged over 10 samples. Error bars both in λ and σ are smaller than the symbol size.

starting from a random initial condition, i.e., if this iteration converges, the solution is automatically stable. On the other hand, it is much harder to extrapolate precisely the point where the convergence time diverges instead of identifying this point by $\lambda_{max} \rightarrow 1$.

Technically the eigenvalue can be determined in a way inspired by the message-passing procedure itself: We randomly initialize all $\varepsilon_{i|j}$ to non-negative values, and update it according to Eq. (44). Then we renormalize the vector dividing it by $\sqrt{\Sigma_{(i,j) \in E}(\varepsilon_{i|j}^2 + \varepsilon_{j|i}^2)}$. This is repeated until convergence of the procedure is reached and λ_{max} equals the asymptotic renormalization factor (see Ref. [38] for an analogous approach to testing the stability of a replica symmetric solution in the problem of counting graph loops).

The results of the numerical tests are shown in Fig. 6. For the c values displayed there, we find that the SP solution is stable against small perturbations in the vicinity of $y=0$, but λ_{max} starts to grow right away. For all the displayed values, we also find that λ_{max} approaches one at positive complexity, i.e., at values $y < y^*$. At the y value, which in the ensemble average, the one step of replica-symmetry-breaking (1RSB) result is expected to be most precise compared to the exact value, SP does not even converge on the single sample.

This changes for $c > 20.4$. At this point, the instability threshold coincides precisely with the zero-complexity point corresponding to minimal VCs. At even higher value, SP thus converges at $y = y^*$. However, this does not necessarily mean that we can do all of the decimation process efficiently at the initial y^* after decimation of a c-dependent fraction of the graph SP starts to diverge even for large c.

VI. FROM SURVEY PROPAGATION TO TYPICAL PROPERTIES ON RANDOM GRAPHS

In Sec. IV B we have seen that it is possible to average the solution of warning propagation over random graphs of average vertex degree c and to recover the replica-symmetric results of Ref. [17] in the thermodynamic limit. In analogy, the equations of survey propagation can be used to reproduce and extend the results of Ref. [19] on the typical properties of minimal VCs under the assumption of 1RSB, i.e., to translate the probabilistic-algorithmic approach on single graph instances to a statistical-physics approach with the graph randomness playing the role of the quenched disorder.

As already explained above, the computational hardness of the VC problem results from the fact that the landscape of VC sizes over the space of all vertex covers may become rough. It may contain, in particular, many local minima: A VC U is considered to be locally optimal if all VCs differing only in a finite number of vertices are as least as large as U. Such local minima are expected to act as traps for many local search algorithms, which therefore are unable to find globally optimal VCs. If 1RSB is considered, not only the minimal VCs are assumed to be clustered (cf. Fig. 3), but also an exponential number of clusters of locally optimal VCs are expected to appear. The total number of such VC clusters of cardinality $X = xN$ in a given graph G is denoted as $\Omega_G(X)$. The complexity of graph G at VC density (or "energy density") x is defined as

$$\Sigma_G(X) = \frac{1}{N} \ln \Omega_G(X), \qquad (46)$$

with respect to Sec. V B we have renormalized the complexity by a factor $1/N$ to assure a sensible thermodynamic limit. The complexity $\Sigma_G(X)$ is expected to be self-averaging [34]. When N is increased, the complexity of randomly drawn graphs G approaches asymptotically the mean value averaged over the whole graph ensemble. Technically, the interesting quantity is thus

$$\Sigma(c,x) = \lim_{N \to \infty} \frac{1}{N} \langle \ln \Omega_G(xN) \rangle_G, \qquad (47)$$

where $\langle \cdots \rangle_G$ denotes the average over all random graphs G of fixed parameters N and c, since this value is found almost surely also in very large random graphs.

The partial derivative of $\Sigma(c,x)$ with respect to x is denoted as

$$y = \frac{\partial \Sigma(c,x)}{\partial x}. \qquad (48)$$

The above equation gives an implicit relationship between y and the VC density x. We can define a generalized free-energy density at given values of y and c via the Legendre transform

$$\phi(c,y) = x(c,y) - \frac{\Sigma(c,x(c,y))}{y}, \qquad (49)$$

in complete analogy to the single-graph quantity Φ in Eq. (34).

The parameter y formally corresponds to an inverse temperature β in an ordinary statistical-physics system, with the difference that microscopic configurations are replaced by clusters of locally optimal VCs. It can be used to control the

mean VC density x of our artificial statistical-physics system. This is in fact done in the SP equations; as we will see below, y is exactly the reweighting parameter introduced before.

All this holds true as long as the relative VC size $x = X/N$ is such that $\Omega_G(xN) \gg 1$ for a typical random graph G, i.e., $\Sigma(c,x) \geq 0$. When $\Sigma(c,x) < 0$, a typical random graph G has no optimal VC clusters of density x. The largest allowable value $y = y^*$ is thus located at the point where $\Sigma[c,x(c,y^*)] = 0$. This point also corresponds to the best 1RSB estimate of the globally minimal VC size $x(c,y^*)$ of a typical random graph (cf. the discussion at the end of Sec. V B).

A. Cavity equation

As already discussed, for each cluster of VCs, vertices can be decorated by a three-state variable: It assumes the value 1; if the vertex belongs to all VCs in the cluster, it takes the value *; if it belongs to some but not all VCs, and the value 0 if it belongs to no VCs of the cluster. Let us also recall the notation $\hat{\pi}_i^{(0)}$, $\hat{\pi}_i^{(*)}$, and $\hat{\pi}_i^{(1)}(=1-\hat{\pi}_i^{(0)}-\hat{\pi}_i^{(*)})$ for the probability that vertex i takes the corresponding value in a randomly selected locally optimal VC at given y. The values $\vec{\pi}_i = (\hat{\pi}_i^{(0)}, \hat{\pi}_i^{(*)}, \hat{\pi}_i^{(1)})$ fluctuate from vertex to vertex, and the main aim of the cavity method is to describe their distribution in a self-consistent way.

Suppose one already knows $\vec{\pi}_i$ for each vertex i of a random graph G with N vertices. Now add a new vertex (say vertex 0) and connect it to k randomly chosen vertices (say $j = 1, 2, \ldots, k$) of graph G. The integer k is determined according to the Poisson distribution $f_c(k) = e^{-c}c^k/k!$. After vertex 0 and the k edges are added, a new graph G' of $N+1$ vertices is constructed. Under the assumption of statistical independence of the vertices $j = 1, 2, \ldots, k$ in graph G [cf. the comment below Eq. (30)], one can write down the following equations for $\vec{\pi}_0$:

$$\hat{\pi}_0^{(0)} = \frac{\prod_{j=1}^{k}(1-\hat{\pi}_j^{(0)})}{e^{-y}+(1-e^{-y})\prod_{j=1}^{k}(1-\hat{\pi}_j^{(0)})}, \quad (50)$$

$$\hat{\pi}_0^{(*)} = \frac{e^{-y}\sum_{j=1}^{k}\hat{\pi}_j^{(0)}\prod_{l\neq j}(1-\hat{\pi}_l^{(0)})}{e^{-y}+(1-e^{-y})\prod_{j=1}^{k}(1-\hat{\pi}_j^{(0)})}, \quad (51)$$

$$\hat{\pi}_0^{(1)} = 1 - \hat{\pi}_0^{(0)} - \hat{\pi}_0^{(*)}. \quad (52)$$

Assuming furthermore that the statistical properties of the $\vec{\pi}$ do not change drastically by adding the new vertex, Eq. (50) allows us to write down the following self-consistent equation governing the probability distribution of $\hat{\pi}_0^{(0)}$:
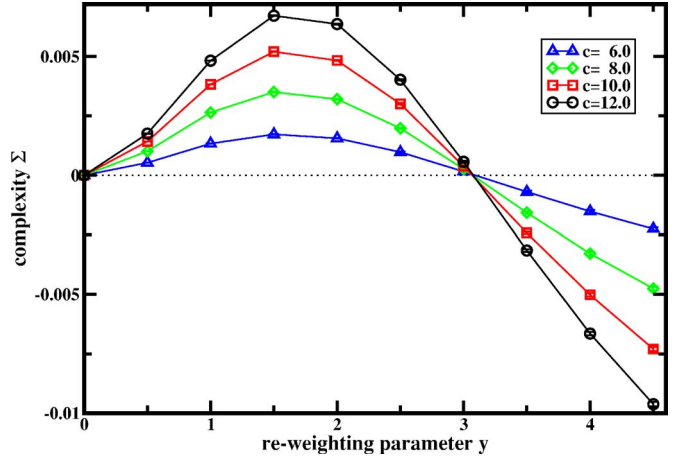

FIG. 7. (Color online) Complexity $\Sigma(c,y)$ as a function of the reweighting parameter y for fixed mean-vertex degree $c=12.0$ (circles), $c=10.0$ (squares), $c=8.0$ (diamonds), and $c=6.0$ (triangles). All these curves seem to intersect in two points, $y=0$ and $y=y^* \approx 3.1$. In these two points, $\Sigma(c,y)=0$.

$$P(\hat{\pi}_0^{(0)}) = f_c(0)\delta_{\hat{\pi}_0^{(0)},1} + \sum_{k=1}^{\infty}f_c(k)\prod_{l=1}^{k}\left[\int d\hat{\pi}_l^{(0)}P(\hat{\pi}_l^{(0)})\right]$$
$$\times\delta\left(\hat{\pi}_i^{(0)} - \frac{\prod_{l=1}^{k}(1-\hat{\pi}_l^{(0)})}{e^{-y}+(1-e^{-y})\prod_{l=1}^{k}(1-\hat{\pi}_l^{(0)})}\right). \quad (53)$$

This equation can be numerically solved with very high precision using a standard algorithm of population dynamics. Note also the equivalence of the update rule in the delta function to Eq. (32). One can in fact estimate $P(\hat{\pi}_0^{(0)})$ also by first generating a huge random graph, iterating SP on it, and than calculating the histogram of all messages.

B. VC density and complexity

Also the VC density is self-averaging. When the graph size N is sufficiently large, the VC density of a typical graph G is almost independent of the microscopic details of G; it only depends on the statistical properties of the graph ensemble represented by the mean-vertex degree c. At fixed value of the reweighting parameter y, this also means VC density $x(c,y)$ can be calculated using the cavity method. The graph G' as generated in the preceding subsection has $N+1$ vertices and mean-vertex degree $c'=2(M+k)/(N+1) = c+(2k-c)/(N+1)$. The expectation of the VC density of G' and that of the graph G are related by

$$(N+1)\,x(c',y) = Nx(c,y) + 1 - \hat{\pi}_0^{(0)}. \quad (54)$$

Expanding $x(c,y)$ around c, and keeping only the nonvanishing terms in the thermodynamic limit, we obtain

$$x(c,y) + c\frac{\partial x(c,y)}{\partial c} = 1 - \langle\hat{\pi}_0^{(0)}\rangle_{G'}. \quad (55)$$

To obtain an expression for $\partial x(c,y)/\partial c$, we add a new edge between two randomly chosen vertices (say vertex j and l) of the old graph G and thus construct a new graph G''. This new graph has mean-vertex degree $c''=c+2/N$. Averaged over all the locally optimal VC clusters at fixed reweighting parameter y, the mean increase in VC density due to the addition of edge (j,l) is

$$\frac{e^{-y}\hat{\pi}_j^{(0)}\hat{\pi}_l^{(0)}}{1-(1-e^{-y})\hat{\pi}_j^{(0)}\hat{\pi}_l^{(0)}}, \tag{56}$$

since it results from the case that both end vertices j and l are uncovered in the corresponding cluster. In other words, we have

$$Nx(c'',y) = Nx(c,y) + \frac{e^{-y}\hat{\pi}_j^{(0)}\hat{\pi}_l^{(0)}}{1-(1-e^{-y})\hat{\pi}_j^{(0)}\hat{\pi}_l^{(0)}}, \tag{57}$$

which leads to the expression

$$\frac{\partial x(c,y)}{\partial c} = \frac{1}{2}\left\langle \frac{e^{-y}\hat{\pi}_j^{(0)}\hat{\pi}_l^{(0)}}{1-(1-e^{-y})\hat{\pi}_j^{(0)}\hat{\pi}_l^{(0)}}\right\rangle_{G''}. \tag{58}$$

Combining Eq. (55) and Eq. (58) we finally obtain that

$$x(c,y) = 1 - \langle\hat{\pi}_0^{(0)}\rangle_{G'} - \frac{c}{2}\left\langle \frac{e^{-y}\hat{\pi}_j^{(0)}\hat{\pi}_l^{(0)}}{1-(1-e^{-y})\hat{\pi}_j^{(0)}\hat{\pi}_l^{(0)}}\right\rangle_{G''}$$

$$= 1 - \int_0^1 d\hat{\pi}^{(0)}P(\hat{\pi}^{(0)})\hat{\pi}^{(0)} - \frac{c}{2}\int_0^1 d\hat{\pi}_1^{(0)}P(\hat{\pi}_1^{(0)})$$

$$\times \int_0^1 d\hat{\pi}_2^{(0)}P(\hat{\pi}_2^{(0)})\frac{e^{-y}\hat{\pi}_1^{(0)}\hat{\pi}_2^{(0)}}{1-(1-e^{-y})\hat{\pi}_1^{(0)}\hat{\pi}_2^{(0)}}, \tag{59}$$

where we have used another time the argument that the change in the histogram $P[\hat{\pi}(0)]$ due to vertex or edge addition is neglectable in the thermodynamic limit.

The first line of Eq. (59) is consistent with the analog Eq. (33) for a single graph. To see this, we notice that

$$\frac{c}{2}\left\langle \frac{e^{-y}\hat{\pi}_j^{(0)}\hat{\pi}_l^{(0)}}{1-(1-e^{-y})\hat{\pi}_j^{(0)}\hat{\pi}_l^{(0)}}\right\rangle = \frac{1}{2}\left\langle \sum_{j\in N(l)}\frac{e^{-y}\hat{\pi}_j^{(0)}\hat{\pi}_l^{(0)}}{1-(1-e^{-y})\hat{\pi}_j^{(0)}\hat{\pi}_l^{(0)}}\right\rangle = \frac{1}{2}\left\langle \sum_{j\in N(l)}\frac{e^{-y}\hat{\pi}_j^{(0)}\dfrac{\prod_{i\in N(l)\backslash j}(1-\hat{\pi}_i^{(0)})}{e^{-y}+(1-e^{-y})\prod_{i\in N(l)\backslash j}(1-\hat{\pi}_i^{(0)})}}{1-(1-e^{-y})\hat{\pi}_j^{(0)}\dfrac{\prod_{i\in N(l)\backslash j}(1-\hat{\pi}_i^{(0)})}{e^{-y}+(1-e^{-y})\prod_{i\in \mathcal{N}(l)\backslash j}(1-\hat{\pi}_i^{(0)})}}\right\rangle$$

$$= \frac{1}{2}\left\langle \frac{e^{-y}\sum_{j\in\mathcal{N}(l)}\hat{\pi}_j^{(0)}\prod_{i\in\mathcal{N}(l)\backslash j}(1-\hat{\pi}_i^{(0)})}{e^{-y}+(1-e^{-y})\prod_{i\in\mathcal{N}(l)}(1-\hat{\pi}_i^{(0)})}\right\rangle = \frac{1}{2}\langle\hat{\pi}_l^{(*)}\rangle. \tag{60}$$

The mean complexity $\Sigma(c,x)$ can be calculated analogously [cf. Ref. [19] and Eqs. (37) and (38)]. The final expression reads

$$\Sigma(c,x) = yx + \sum_{k=1}^{\infty}f_c(k)\prod_{l=1}^{k}\int_0^1 d\hat{\pi}_l^{(0)}P(\hat{\pi}_l^{(0)})$$

$$\times\ln\left(e^{-y}+(1-e^{-y})\prod_{l=1}^{k}(1-\hat{\pi}_l^{(0)})\right)$$

$$-\frac{c}{2}\int_0^1 d\hat{\pi}_1^{(0)}P(\hat{\pi}_1^{(0)})\int_0^1 d\hat{\pi}_2^{(0)}P(\hat{\pi}_2^{(0)})$$

$$\times\ln[1-(1-e^{-y})\hat{\pi}_1^{(0)}\hat{\pi}_2^{(0)}]. \tag{61}$$

C. Optimal reweighting and minimal VC density

At given average degree c, Eq. (59) allows us to calculate the typical VC size as a function of the reweighting param-

eter y. It is monotonously decreasing with growing y, so naively one would expect that the minimal VC size is found in the limit of $y\to\infty$. There is, however, a problem. The complexity $\Sigma(c,y)$ reaches zero at some (a priori) c-dependent value y^*, and becomes negative for larger y. Being defined as the logarithm of the number of corresponding clusters, negative complexities correspond to VC sizes typically nonexisting in random graphs of mean degree c. Consequently, we have to determine the size of the minimal VC of a typical graph by $x(c,y^*)$ from the zero-complexity criterion $\Sigma(c,y^*)=0$.

Figure 7 shows $\Sigma(c,y)$ as a function of y at various fixed c values. At a given c value $(c>e)$, the complexity $\Sigma(c,y)$ first increases with y as y increases from zero. $\Sigma(c,y)$ attains its maximal value when y increases to $y\approx1.5$. Afterwards, $\Sigma(c,y)$ decreases with y and it reaches $\Sigma=0$ when $y=y^*\approx3.1$. Upon further increase of y, the complexity becomes negative. It is remarkable that the $\Sigma(c,y)$ curves for different c values intersect at (almost) the same point y^*, which is just
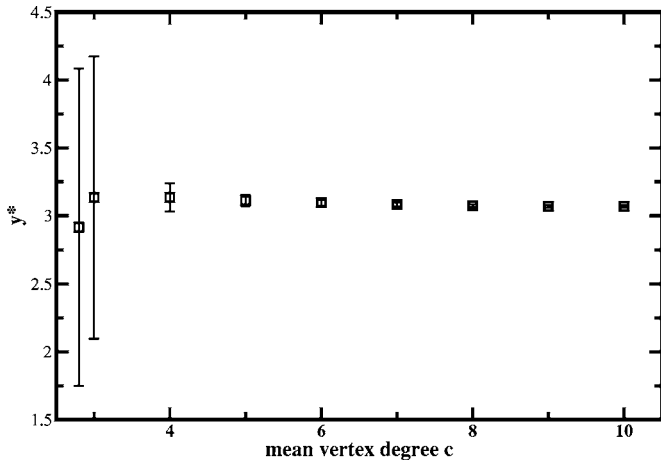
FIG. 8. Relationship between the optimal reweighting parameter y^* and the mean-vertex degree c.

FIG. 9. (Color online) The fraction of covered vertices (VC density) x in a minimum-vertex cover problem as a function of mean-vertex degree c. Typical-case statistical-physics results are given by a +, and $\times$ gives the estimates made by SP on graphs of size $N=5000$, averaged over 20 samples.

the point where the complexity vanishes, $\Sigma(c,y^*)=0$. At present we do not understand why the complexities for systems with different c values should approach zero at (almost) the same point.

At each fixed c value, the optimal y^* value can be determined from the point of $\Sigma(c,y^*)=0$. The optimal y^* value was calculated numerically by population dynamics and shown in Fig. 8 as a function of mean-vertex degree c. Figure 8 indeed demonstrates that the optimal reweighting parameter y^* is insensitive to c and stays at $y^* \approx 3.1$ over the whole range of inspected c values. This is also in agreement with Ref. [19] (note that y^* in the present article corresponds to $2y^*$ in Ref. [19]). Even when $c=2.8$ (just slightly beyond e) we have $y^*=2.9\pm1.2$, which is significantly different from zero, but consistent with a constant y^*. From Fig. 8 we thus get the impression that, as the mean-vertex degree c exceeds e, the optimal reweighting parameter jumps quickly to a value $y^* \approx 3.1$.

The minimum-vertex cover size can also be obtained. In Fig. 9 we show the relationship between the minimal vertex cover size and the mean-vertex degree c. As a comparison, Fig. 9 also includes the mean minimal vertex cover size as estimated by the SP algorithm of the last section ($N=5000$, $y=2.0$, each point averaged over 20 samples). The results obtained by SP and those obtained by the mean-field statistical physics calculations are in very good agreement. At given vertex degree c, the minimal VC density estimated by SP and the mean-field theory is lower than the corresponding value obtained through exact enumeration followed by extrapolation [17]. The reason for such a discrepancy can be understood. According to Fig. 7, at given mean-vertex degree $c \leq 10$, the maximum complexity of the system is less than 5×10^{-3}. This indicates that clustering of minimal VC solutions into distantly separated domains will only occur for random graphs with size $N \geq 10^3$. For small random graphs as used in Ref. [17], it is very likely that all the minimal VC solutions can be grouped into a single cluster (but with long-range frustrations among those vertices described by the joker state * [20]).
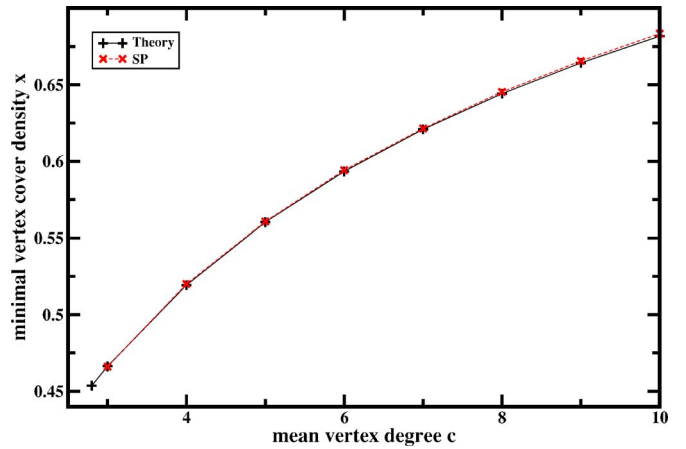
To summarize this subsection, we list in Table I the values of y^* and the minimal energy density at several c values. Theoretical and SP results are extremely close to each other, even if the latter are systematically slightly larger. This may be due to various reasons: SP uses finite size and really constructs a possibly nonoptimal VC, whereas the theory works at the 1RSB level, which again is not exact due to higher RSB effects. We expect, however, both estimates to be very close (but slightly different) from the exact result.

D. Relaxation time of the population dynamics

Let us finally analyze the mean-field population dynamics, which aims at finding a fixed-point distribution for Eq. (53). In the population dynamics, an array of $\mathcal{N}$ values $\hat{\pi}^{(0)}$ is first initialized randomly (typically we use $\mathcal{N} \sim 10^6$; this number should not be confused with the vertex number N in the single-sample analysis of the previous sections). Then in each time step, corresponding to an interval $\Delta t = 1/\mathcal{N}$, we perform the following update of the population:

(1) A natural number k is drawn from the Poisson distribution $f_c(k)$.

(2) k elements $\hat{\pi}_i^{(0)}$, $i=1,\ldots,k$, are randomly and independently chosen in the current population.

(3) A new $\hat{\pi}^{(0)}$ is calculated according to Eq. (50).

(4) A randomly selected element in the population is replaced with this new $\hat{\pi}^{(0)}$ value.

This iteration is repeated many times (typically of the order of $10^4 \mathcal{N}$), until the statistical properties of the population approach stationary values. The histogram of the population is then our estimate of the self-consistent distribution $P(\hat{\pi}_0)$ in Eq. (53).

Suppose that at time t, the histogram of $\hat{\pi}^{(0)}$ over the whole population is given by

$$P(\hat{\pi}^{(0)};t) = p_1(t)\delta(\hat{\pi}^{(0)}) + p_2(t)\delta(\hat{\pi}^{(0)}) + p_3(t)\rho(\hat{\pi}^{(0)};t),$$

(62)

TABLE I. The optimal reweighting parameter y^* and the minimal VC density x as estimated by the 1RSB ansatz and by SP ($N=5000$).

c	2.8	3.0	4.0	5.0	6.0	7.0	8.0	9.0	10.0
y^*	3(1)	3(1)	3.1(1)	3.11(4)	3.10(2)	3.08(2)	3.07(1)	3.069(9)	3.068(8)
x (theory)	0.4536290(8)	0.46632(2)	0.51934(2)	0.56033(3)	0.59341(3)	0.62088(2)	0.64416(2)	0.66423(2)	0.68175(2)
x (SP)		0.4661(5)	0.52004(4)	0.5607(3)	0.5942(3)	0.6214(3)	0.6453(2)	0.6655(3)	0.6834(2)

with $p_3(t)=1-p_1(t)-p_2(t)$, and with $\rho(\hat{\pi}^{(0)};t)$ satisfying the conditions

$$\rho(0;t) \equiv 0, \quad \rho(1;t) \equiv 0, \quad \int_0^1 d\hat{\pi}^{(0)}\rho(\hat{\pi}^{(0)};t) \equiv 1. \tag{63}$$

Since the population of $\hat{\pi}^{(0)}$ values at time $t+\Delta t$ is obtained by replacing one randomly chosen element of the population of time t with the newly calculated $\hat{\pi}^{(0)}$, we can write down the following two evolution equations for $p_1(t)$ and $p_2(t)$:

$$\mathcal{N}p_1(t+\Delta t) = \mathcal{N}p_1(t) + \sum_{k=1}^{\infty} f_c(k)\{1 - [1-p_2(t)]^k\} - p_1(t)$$

$$\rightarrow \frac{dp_1(t)}{dt} = 1 - e^{-cp_2(t)} - p_1(t), \tag{64}$$

$$\mathcal{N}p_2(t+\Delta t) = \mathcal{N}p_2(t) + \sum_{k=1}^{\infty} f_c(k)p_1^k - p_2(t) \rightarrow \frac{dp_2(t)}{dt}$$

$$= e^{-c[1-p_1(t)]} - p_2(t). \tag{65}$$

More precisely, these equations describe the average evolution over many runs of the population dynamics. For large populations $\mathcal{N} \gg 1$ the true evolution of one population is, however, expected to be closely concentrated around its expectation value, with random fluctuations of the order $O(1/\sqrt{\mathcal{N}})$. These equations can be understood easily. In Eq. (64), we describe the expected number of zero elements of the population. This number is decreased by one with probability $p_1(t)$ by replacing an old zero element, or it grows by one if a new zero element is introduced. The latter case happens if in between the k "parents" $\hat{\pi}_i^{(0)}$, $i=1,\ldots,k$, selected before, there exists at least one which equals one. Analogously, a new element equal to one is inserted in the population if all parents were equal to zero, explaining the gain term in Eq. (65).

The fixed-point solution (p_1,p_2) of Eqs. (64) and (65) is determined by

$$p_1 = 1 - e^{-cp_2}, \quad p_2 = e^{-c(1-p_1)}. \tag{66}$$

Note that for $c<e$, only one solution with $p_1+p_2=1$ exists. This solution corresponds to replica symmetry, only one solution cluster exists, and consequently no cluster-to-cluster fluctuations exist. Above $c=e$, also two other solutions with $p_1+p_2 \neq 1$ exist. Only one is iteratively stable; it fulfills $p_1+p_2<1$ and allows therefore for cluster-to-cluster variations of the value of $\hat{\pi}_i^{(0)}$ for some vertices i.

To study the convergence velocity toward this fixed-point solution, we assume that

$$p_1(t) = p_1 + \epsilon_1(t), \quad p_2(t) = p_2 + \epsilon_2(t), \tag{67}$$

where $\epsilon_1(t)$ and $\epsilon_2(t)$ are small quantities. Linearizing the dynamical equations, we find

$$\frac{d\epsilon_1(t)}{dt} = c(1-p_1)\epsilon_2(t) - \epsilon_1(t), \quad \frac{d\epsilon_2(t)}{dt} = cp_2\epsilon_1(t) - \epsilon_2(t), \tag{68}$$

and the typical relaxation time for $p_1(t)$ and $p_2(t)$ is

$$\tau_{12} = \frac{1}{1 - c\sqrt{(1-p_1)p_2}}. \tag{69}$$

When the mean-vertex degree c approaches e from below, the parameter p_2 approaches e^{-1} as

$$p_2 \simeq \frac{1}{e} + \frac{(e-c)}{2e^2}. \tag{70}$$

Consequently, the typical relaxation time τ_{12} diverges as

$$\tau_{12} \sim \frac{2e}{e-c} \quad \text{for } c < e. \tag{71}$$

Note that the same critical behavior was found for the bug-relaxation time in the purely replica-symmetric warning-propagation equations.

On the other hand, when c approaches e from above,

$$p_2 \simeq \frac{1}{e} - \left(\frac{6(c-e)}{e^3}\right)^{1/2} + \frac{(c-e)}{e^2}. \tag{72}$$

Therefore τ_{12} diverges as

$$\tau_{12} \sim \frac{e}{c-e} \quad \text{for } c > e. \tag{73}$$

Equations (71) and (73) were confirmed in a single-graph message-passing experiment. Note that the only region where this convergence slows down is close to the replica-symmetry-breaking transition at $c=e$, where also the population dynamics slows down critically. Note also that this relaxation time does not depend on the reweighting parameter y.

We now study the evolution of $\rho(\hat{\pi}^{(0)};t)$ in Eq. (62). For this purpose, in the population dynamics we can set $p_1(t) \equiv p_1$, $p_2(t) \equiv p_2$, $p_3(t)=p_3=1-p_1-p_2$ to their stationary values, and store only those $\hat{\pi}^{(0)}$ values that satisfy $0<\hat{\pi}^{(0)}<1$ in the population array. The distribution $\rho(\hat{\pi}^{(0)};t+\Delta t)$ is related with $\rho(\hat{\pi}^{(0)};t)$ by the following equation, describing

the expected number of population entries in the interval $(\hat{\pi}^{(0)}, \hat{\pi}^{(0)} + \Delta\hat{\pi}^{(0)})$:

$$
\begin{aligned}
\mathcal{N}\rho(\hat{\pi}^{(0)}&;t+\Delta t)\Delta\hat{\pi}^{(0)} \\
&= \mathcal{N}\rho(\hat{\pi}^{(0)};t)\Delta\hat{\pi}^{(0)} - \rho(\hat{\pi}^{(0)};t)\Delta\hat{\pi}^{(0)} \\
&+ \sum_{m=1}^{\infty} \frac{f_{cp_3}(m)}{1-e^{-cp_3}} \prod_{l=1}^{m} \int_0^1 d\hat{\pi}_l^{(0)}\rho(\hat{\pi}_l^{(0)};t) \\
&\times \delta\left(\hat{\pi}^{(0)} - \frac{\prod_{l=1}^{m}(1-\hat{\pi}_l^{(0)})}{e^{-y}+(1-e^{-y})\prod_{l=1}^{m}(1-\hat{\pi}_l^{(0)})} \right)\Delta\hat{\pi}^{(0)}.
\end{aligned}
\tag{74}
$$

From Eq. (74) we see that

$$
\begin{aligned}
\frac{\partial\rho(\hat{\pi}^{(0)};t)}{\partial t} &= -\rho(\hat{\pi}^{(0)};t) + \sum_{m=1}^{\infty} \frac{f_{cp_3}(m)}{1-e^{-cp_3}} \prod_{l=1}^{m} \int_0^1 d\hat{\pi}_l^{(0)}\rho(\hat{\pi}_l^{(0)};t) \\
&\times \delta\left(\hat{\pi}^{(0)} - \frac{\prod_{l=1}^{m}(1-\hat{\pi}_l^{(0)})}{e^{-y}+(1-e^{-y})\prod_{l=1}^{m}(1-\hat{\pi}_l^{(0)})} \right).
\end{aligned}
\tag{75}
$$

The fixed-point solution of Eq. (75) is

$$
\begin{aligned}
\rho(\hat{\pi}^{(0)}) &= \sum_{m=1}^{\infty} \frac{f_{cp_3}(m)}{1-e^{-cp_3}} \prod_{l=1}^{m} \int d\hat{\pi}_l^{(0)}\rho(\hat{\pi}_l^{(0)}) \\
&\times \delta\left(\hat{\pi}^{(0)} - \frac{\prod_{l=1}^{m}(1-\hat{\pi}_l^{(0)})}{e^{-y}+(1-e^{-y})\prod_{l=1}^{m}(1-\hat{\pi}_l^{(0)})} \right),
\end{aligned}
\tag{76}
$$

as can be seen also directly from Eqs. (53) and (62).

Now let us suppose that, at time t, the actual distribution $\rho(\hat{\pi}^{(0)};t)$ deviates from the fixed-point distribution only slightly;

$$
\rho(\hat{\pi}^{(0)};t) = \rho(\hat{\pi}^{(0)}) + \epsilon_3(\hat{\pi}^{(0)};t),
\tag{77}
$$

with $|\epsilon_3(\hat{\pi}^{(0)};t)| \ll 1$ for all $0 < \hat{\pi}^{(0)} < 1$ and

$$
\epsilon_3(0;t) = \epsilon_3(1;t) \equiv 0, \quad \int_0^1 d\hat{\pi}^{(0)}\epsilon_3(\hat{\pi}^{(0)};t) = 0.
\tag{78}
$$

The linearized evolution equation for $\epsilon_3(\hat{\pi}^{(0)};t)$ is

$$
\begin{aligned}
\frac{\partial\epsilon_3(\hat{\pi}^{(0)};t)}{\partial t} &= -\epsilon_3(\hat{\pi}^{(0)};t) + cp_2\frac{e^y}{[1+(e^y-1)\hat{\pi}^{(0)}]^2} \\
&\times \epsilon_3\left(\frac{1-\hat{\pi}^{(0)}}{1+(e^y-1)\hat{\pi}^{(0)}};t \right) + cp_3\int_0^1 d\hat{\pi}_1^{(0)} \\
&\times \epsilon_3(\hat{\pi}_1^{(0)};t)\int_0^1 d\hat{\pi}_2^{(0)}\rho(\hat{\pi}_2^{(0)})\delta\Bigg(\hat{\pi}^{(0)} - 1 \\
&+ \frac{\hat{\pi}_1^{(0)}(1-\hat{\pi}_2^{(0)})}{1-(1-e^{-y})(1-\hat{\pi}_1^{(0)})(1-\hat{\pi}_2^{(0)})} \Bigg).
\end{aligned}
\tag{79}
$$

The stability of Eq. (79) can be analyzed by Fourier expanding $\epsilon_3(\hat{\pi}^{(0)};t)$ in the following way:

$$
\epsilon_3(\hat{\pi}^{(0)};t) = \sum_{m=1}^{\infty} a_m(t)\sqrt{2}\,\sin(\pi m\hat{\pi}^{(0)}),
\tag{80}
$$

with coefficients $a_m(t)$ satisfying the global constraint

$$
\sum_{n=0}^{\infty} \frac{a_{2n+1}(t)}{2n+1} \equiv 0.
\tag{81}
$$

Based on Eqs. (79) and (80), one can write down the evolution equation for $a_m(t)$,

$$
\frac{da_m(t)}{dt} = \sum_{n=1}^{\infty} \Lambda_{mn}a_n(t),
\tag{82}
$$

where the elements of the matrix Λ can be easily written down.

The task now is to identify the dominant eigenmode of Eq. (82) under the constraint of Eq. (81). We have performed such an analysis for various c values in the range $3 \le c \le 30$ and various y values ranging from $y=0$ to $y=5$. In all the cases studied, the dominant eigenmode of Eq. (82) decays to zero very quickly, indicating that the mean-field population dynamics is exponentially fast converging toward its fixed point. Compared to the iterative stability of SP on single instances of random graphs, we find that the messages may converge in population even if they do not converge on the single graph any longer. This is interesting since it allows us to extend the typical-case estimates to a region, where SP applied to single samples fails to predict anything.

VII. CONCLUSION

In this paper, we have formulated two message-passing procedures for solving—or approximating—the minimal vertex cover problem, namely, warning propagation and survey propagation. We have analyzed the performance of both algorithms on the test bed of finite-connectivity random graphs, where previous statistical-physics approaches based both on the replica approach and on the cavity method provide an insight on the phase diagram. We have also discussed in detail how the message-passing approach is technically connected to these typical-case-based statistical-physics results.

For small average-vertex degrees $c < e$ replica symmetry is known to hold in the space of all minimal vertex covers.

Therefore the simpler one of the two algorithms—warning propagation, which is based on the replica-symmetric Bethe-Peierls iterative approach—is applicable. Comparing it to the exact leaf-removal algorithm, we have shown that it outputs true minimal vertex covers. Unfortunately the iterative solution of the warning propagation equations slows down critically if we approach $c=e$ from below, and it does not converge at all for higher average degrees.

In this higher-degree region, replica symmetry is known to be broken. We have therefore applied a survey propagation algorithm, which is based on the first step of replica symmetry breaking. We have identified a parameter range where the message-passing equations converge to a globally self-consistent solution, which can be used to construct small vertex covers. We have found that the provided results not only outperform simple local-search procedures, but are consistent to exact asymptotic results for high but finite average degrees. Interestingly the vertex covers produced at the end are relatively insensitive to details of the algorithms (in particular to the somewhat heuristic choice of the reweighting parameter y).

In the case of vertex cover, replica symmetry is known to be fully broken, i.e., the exact solution for $c>e$ is known to be more complicated than the one described by one-step replica-symmetry breaking. Intuitively the solutions are expected to be organized in clusters of clusters of clusters, etc.; so, even if the results of the application of survey propagation are very promising, it is expected to provide only a good approximation algorithm. It could therefore be interesting to go beyond survey propagation and to formulate an algorithm based on the second step of replica symmetry breaking (corresponding to two hierarchical clustering levels), in order to see if the higher complexity of the algorithm required is leading to even smaller vertex covers than survey propagation.

As a last point, it could be interesting to apply the algorithm to real-world covering problems, possibly extending it to the specific nature of these tasks, which may be similar but not equal to the original minimal VC problem [2,6]. These problems are frequently characterized by a broad-degree distribution of the underlying networks. Their extreme heterogeneity may result in a better performance of simple heuristic algorithms exploiting local network structures. On the other hand, it was shown in Ref. [3] that assortative degree correlations may force replica symmetry to break also in scale-free networks, and algorithms such as survey propagation are expected to become efficient. A related interesting topic is the local-network structure beyond the vertex degrees, in particular, small loops or other small dense subgraphs. It may be necessary to coarse grain the graph considering such loops as single constraints, and apply the factorization hypothesis only to larger structures. In the replica-symmetric case (warning or belief propagation), this corresponds to the region-graph method proposed in Ref. [12]; for the one-step replica-symmetry-broken case (survey propagation) it is still an open technical challenge.

ACKNOWLEDGMENTS

[1] M. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness* (Freeman, San Francisco, 1979).

[2] P. Echenique, J. Gomez-Gardenes, Y. Moreno, and A. Vazquez, Phys. Rev. E **71**, 035102(R) (2005).

[3] A. Vazquez and M. Weigt, Phys. Rev. E **67**, 027101 (2003).

[4] Y. Breitbart, C. Chan, M. Garofalakis, R. Rastogi, and A. Silverschatz, Proceedings IEEE INFOCOM 2001, Anchorage, Alaska (IEEE, 2001).

[5] K. Park and H. Lee, Proceedings ACM SIGCOM 2001, San Diego, California (ACM, 2001).

[6] J. Gomez-Gardenes, P. Echenique, and Y. Moreno, Eur. Phys. J. B **49**, 259 (2006).

[7] E. Marinari, R. Monasson, and G. Semerjian, Europhys. Lett. **73**, 8 (2006).

[8] A. Hartmann and M. Weigt, *Phase Transitions in Combinatorial Optimization Problems* (Wiley-VCH, Berlin, 2005).

[9] *Computational Complexity and Statistical Physics*, edited by A. Percus, G. Istrate, and C. Moore (Oxford University Press, New York, 2006).

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

[11] M. Mézard, G. Parisi, and R. Zecchina, Science **297**, 812 (2002).

[12] J. Yedidia, W. Freeman, and Y. Weiss, IEEE Trans. Inf. Theory **51**, 2282 (2005).

[13] B. Bollobás, *Random Graphs* (Academic Press, London, 1985).

[14] A. Frieze, Discrete Math. **81**, 171 (1990).

[15] P. Gazmuri, Networks **14**, 367 (1984).

[16] M. Weigt and A. K. Hartmann, Phys. Rev. E **63**, 056127 (2001).

[17] M. Weigt and A. K. Hartmann, Phys. Rev. Lett. **84**, 6118 (2000).

[18] M. Bauer and O. Golinelli, Eur. Phys. J. B **24**, 339 (2001).

[19] H. Zhou, Eur. Phys. J. B **32**, 265 (2003).

[20] H. Zhou, Phys. Rev. Lett. **94**, 217203 (2005).

[21] G. Biroli and M. Mézard, Phys. Rev. Lett. **88**, 025501 (2002).

[22] A. Hartmann and M. Weigt, Europhys. Lett. **62**, 533 (2003).

[23] M. P. Ciamarra, M. Tarzia, A. de Candia, and A. Coniglio, Phys. Rev. E **67**, 057105 (2003).

[24] O. Rivoire, G. Biroli, O. Martin, and M. Mézard, Eur. Phys. J. B **37**, 55 (2004).

[25] H. Hansen-Goos and M. Weigt, J. Stat. Mech.: Theory Exp. 2005, P04006.

[26] A. Braunstein, R. Mulet, A. Pagnani, M. Weigt, and R. Zecchina, Phys. Rev. E **68**, 036702 (2003).

[27] A. Braunstein, M. Mézard, M. Weigt, and R. Zecchina, in *Computational Complexity and Statistical Physics*, edited by A. Percus, G. Istrate, and C. Moore (Oxford University Press, New York, 2006).

[28] F. Kschischang, B. Frey, and H. Loeliger, IEEE Trans. Inf. Theory **47**, 498 (2001).

[29] R. Karp and M. Sipser, Proceedings of the 22nd Annual Symposium on Foundations of Computer Science 1981, Nashville, Tennessee (IEEE, 1981).

[30] H. Zhou and Z. Ou-Yang, e-print cond-mat/0309348.

[31] L. Zdeborová and M. Mézard J. Stat. Mech.: Theory Exp. 2006, P05003.

[32] S. Mertens, M. Mźard, and R. Zecchina, Random Struct. Algorithms **28**, 340 (2006).

[33] M. Mézard and R. Zecchina, Phys. Rev. E **66**, 056126 (2002).

[34] M. Mézard, G. Parisi, and M. Virasoro, *Spin Glasses and Beyond* (World Scientific, Singapore, 1987).

[35] R. Monasson, Phys. Rev. Lett. **75**, 2847 (1995).

[36] A. Montanari, G. Parisi, and F. Ricci, J. Phys. A **37**, 2073 (2004).

[37] M. Weigt, Eur. Phys. J. B **28**, 369 (2002).

[38] G. Semerjian and E. Marinari, J. Stat. Mech.: Theory Exp. 2006 P06019.