Minimum vertex cover problems on random hypergraphs: Replica symmetric solution and a leaf removal algorithm

Satoshi Takabe* and Koji Hukushima

Graduate School of Arts and Sciences, The University of Tokyo, 3-8-1 Komaba, Meguro-ku, Tokyo 153-8902, Japan

(Received 3 February 2013; revised manuscript received 3 April 2014; published 27 June 2014)

The minimum vertex-cover problems on random α-uniform hypergraphs are studied using two different approaches, a replica method in statistical mechanics of random systems and a leaf removal algorithm. It is found that there exists a phase transition at the critical average degree $e/(\alpha - 1)$, below which a replica symmetric ansatz in the replica method holds and the algorithm estimates exactly the same solution of the problem as that by the replica method. In contrast, above the critical degree, the replica symmetric solution becomes unstable and the leaf-removal algorithm fails to estimate the optimal solution because of the emergence of a large size core. These results strongly suggest a close relation between the replica symmetry and the performance of an approximation algorithm. Critical properties of the core percolation are also examined numerically by a finite-size scaling.

DOI: 10.1103/PhysRevE.89.062139 PACS number(s): 05.20.−y, 75.10.Nr, 02.60.Pn, 89.70.Eg

I. INTRODUCTION

The more crucial part of everyday life computers bear, the more significance computer science and information theory seem to have. Among them, the computational complexity theory studies the difficulty, the limit of improving algorithms, to solve theoretical computational problems. It has revealed that the problems belong to several classes such as P and NP and there are many inclusion relations between these classes. For example, two-satisfiability (2-SAT) problems belong to a class of P guaranteed to be solved in polynomial time. 3-SAT and the vertex cover problems belong to a class of NP-complete [1].

These problems are deeply related to the well-known P versus NP problem attracting the theoretical computer scientists. Then, the worst-case performance to solve the computational problems has been one of the main subjects. Among many types of combinatorial optimization problems, the minimum vertex cover (min-VC) problem to be discussed in this paper belongs to a class of NP-hard, and it applies to searching a file on a file storage [2] and to improving the group testing [3]. The approximation algorithm for the min-VC and its performance have been studied [4].

In addition to the worst-case analysis, an important alternative is the study of typical-case behavior on a class of random instances of the computational problems. Recently, statistical-mechanical methods of random spin systems have been applied to the problems such as K-SAT and other constraint-satisfaction problems [5]. These methods, developed in the spin-glass theory [6], enable us to study the typical properties of the randomized problems. For example, the statistical-mechanical approaches find a SAT/UNSAT transition of K-SAT [7] and p-XOR-SAT [8]. They are also applied to estimating a colorable transition of q coloring [9] and random-averaged optimal values of min-VC [10–13] problems. These results clarify that there is a so-called replica symmetric (RS) phase where a replica symmetry ansatz correctly provides typical properties of the problems, and a replica symmetry breaking (RSB) phase where the RS solution becomes incorrect. Together with these approaches, a typical-case performance of some approximation algorithms has been also studied [14–16], suggesting that there is a nontrivial relation between the replica symmetry and the performance of approximation algorithms.

In this paper, we study the min-VC problem on a random hypergraph. A number of vertices connected to an edge is called an edge size and a graph whose edge size is more than 3 is defined as a hypergraph. A statistical-mechanical model defined on a standard graph with edge size 2 has two-body interactions. In contrast, the model defined on a hypergraph includes multibody interactions determined by its edge size, which often change a type of phase transition and a breaking pattern of the replica symmetry as shown in the p-body spin glass model [17]. From this viewpoint, influence of an edge size on the typical estimates of random computational problems has been investigated by statistical-mechanical approaches. In fact, it has been revealed that the edge size changes the properties of some problems such as K-SAT [5,7], q-coloring [18], and min-VC problems on K-uniform regular random hypergraphs [19]. It is also found that there exists a transition related to typical-case difficulty between 2-SAT and 3-SAT [20]. Here we study the typical-case behavior of the min-VC problem, explained later, on random α-uniform hypergraphs and focus on the relation between the replica symmetry and the performance of an approximation algorithm called an extended leaf removal algorithm, proposed to solve min-VC problems on hypergraphs approximately. In the present work, we concentrate here on the replica symmetric solution, while the related model has been studied in the level of one step RSB [19], which is the current state-of-the-art technique in the statistical-mechanical approach. It is analytically shown in this paper, however, that an instability of the replica symmetric solution, not the breaking of higher step replica symmetry, is significantly related to the performance limitation of the specific approximation algorithm. We also discuss a geometric transition found in a core generated by the leaf-removal algorithm, which occurs at the same critical average degree as the instability of the RS solution.

This paper is organized as follows. In the following section, we define the min-VC problem and α-uniform random hypergraph ensembles and show the statistical-mechanical

*s_takabe@huku.c.u-tokyo.ac.jp

analysis of min-VC problems based on the replica method. We estimate an averaged minimum-cover ratio under the RS ansatz. We discuss the stability of the RS solution and find a critical average degree beyond which the RS solution becomes unstable. In Sec. III we introduce the leaf removal algorithm for solving a min-VC problem on a given graph. We study asymptotic behavior of its recursive relations in the algorithm particularly for $\alpha = 3$, which reveals emergence of the core above the same critical average degree as the instability of the RS solution. In Sec. IV we perform numerical simulations by the leaf removal algorithm and Markov-chain Monte Carlo method. We also show finite-size-scaling analyses for a critical phenomena of the core percolation. The final section is devoted to a summary and discussion. In the Appendix we describe the details of the recursive analysis of the leaf removal algorithm.

II. DEFINITION OF MIN-VC AND RANDOM HYPERGRAPHS AND STATISTICAL-MECHANICAL ANALYSIS

Let us suppose that an α-uniform hypergraph $G = (HV, HE)$ consists of N vertices $i \in HV = \{1, \ldots, N\}$ and (hyper)edges $(i_1, \ldots, i_\alpha) \in HE \subset HV^\alpha$ $(i_1 < \cdots < i_\alpha)$. We define covered vertices as a subset $HV' \subset HV$ and covered edges as a subset of edges connected to at least a covered vertex. The vertex cover problem on the hypergraph G is to find a set of the covered vertices HV' by which all edges are covered. We define the cover ratio on G as $|HV'|/N$ with $|HV'|$ being the size of the vertex cover problem. The min-VC problem on G is to search a set of the covered vertices with the minimum-cover ratio.

In the random α-uniform hypergraph all the edges are set independently from all α-tuples of vertices with probability p. The degree distribution of the graph converges to the Poisson distribution with the average degree c, which is given as $c = pN^{\alpha-1}/(\alpha-1)!$ for large N. In this paper, we focus on an average of the minimum-cover ratio x_c over the sparse random hypergraphs with the average degree c being $O(1)$.

The vertex cover problems are mapped on the lattice gas model [10,11,21] on the random hypergraphs. We define a variable ν_i on each vertex, representing the existence of a gas particle, which takes 0 if a vertex i is covered and 1 if uncovered. An covered edge has at least a vertex with $\nu_i = 0$ in its connected vertices. Thus, an indicator function for a given particle configuration $\underline{\nu} = \{\nu_i\} = \{0,1\}^N$ is defined as

$$\chi(\underline{\nu}) = \prod_{(i_1,\ldots,i_\alpha)\in HE} (1 - \nu_{i_1}\cdots\nu_{i_\alpha}), \tag{1}$$

which takes 1 if $\underline{\nu}$ is a solution of the vertex cover problem on the hypergraph, and 0 otherwise. Using the indicator function, the grand canonical partition function of the model reads

$$\Xi = \sum_{\underline{\nu}} \exp\left(\mu \sum_{i=1}^{N} \nu_i\right) \chi(\underline{\nu}), \tag{2}$$

where μ is a chemical potential and the sum is over all configurations of $\underline{\nu}$. In this formulation, only the solutions of the vertex cover problem contribute the partition function and its ground states in a large μ limit are given by the solutions of the min-VC problem. To study the typical case, we need to take the average over the random hypergraphs and the limit as $N \to \infty$. Then, the average minimum-cover ratio is represented as

$$x_c(c) = 1 - \lim_{\mu\to\infty}\lim_{N\to\infty}\frac{1}{N}\mathsf{E}\left\langle \sum_i \nu_i \right\rangle_\mu, \tag{3}$$

where $\langle\cdots\rangle_\mu$ is the grand canonical average and E is the average over the random hypergraph ensemble. Our aim is to obtain the theoretical estimate of the average minimum-cover ratio as a function of the average degree c.

The average minimum-cover ratio is derived from the averaged grand potential density $-(\mu N)^{-1}\mathsf{E}\ln\Xi$, which is obtained by using the replica method for finite connectivity graphs [22]. Following the standard procedure of the replica method, the original problem is reduced to solving a saddle-point equation of a replicated order parameter functional. To proceed the calculation, we assume the RS ansatz that the solution of the saddle-point equation has a replica symmetry. Introducing a local field on a vertex associated to the order parameter and its distribution function, we obtain the saddle-point equation of the distribution. Finally, under the RS ansatz, the average minimum-cover ratio is obtained as a function of the average degree c,

$$x_c(c) = 1 - \left[\frac{W[(\alpha-1)c]}{(\alpha-1)c}\right]^{1/(\alpha-1)}\left(1 + \frac{W[(\alpha-1)c]}{\alpha}\right), \tag{4}$$

where $W(x)$ is the Lambert W function defined as $W(x)\exp[W(x)] = x$. We call this estimate the RS solution of min-VC problems. This solution is also obtained by an alternative cavity method [12]. Although the instability of the RS solution such as the de Almeida-Thouless instability [23] must be examined to validate the solution, we here naively study an instability condition of the saddle-point equation against a perturbation of the local field distribution within the RS sector. The analysis leads to a critical value of the average degree $c_* = e/(\alpha-1)$ above which the RS solution becomes unstable. These results, x_c and c_*, include the case of $\alpha = 2$ [10]. The obtained x_c gives a correct value below the critical average degree, while a RSB solution for x_c is required above it.

III. LEAF REMOVAL ALGORITHM

Here we turn our attention to the estimate of x_c by using an approximation algorithm. The leaf removal algorithm has been proposed as an approximation algorithm to solve a min-VC problem on a graph with $\alpha = 2$ [24] and has also been applied to search for a k-core [25] and a 3-XOR-SAT solution [15]. For a min-VC problem on a given graph, this algorithm consists of iterative steps, where vertices called a leaf, as well as the edges connected to the leaves, are removed from the graph with covered vertices appropriately assigned to those vertices. This removal step makes new leaves and the algorithm continues in an iterative way until the leaf is empty. By this procedure, the minimum-cover ratio is estimated correctly at least for the removed part of the graph. We consider the global leaf removal (GLR) algorithm [14], which removes simultaneously all the leaves found in a recursive step. We focus on the expansion of this algorithm for the min-VC problem on a hypergraph with $\alpha = 3$, while it is straightforward to extend it to that on a hypergraph with $\alpha \geqslant 4$. A crucial point in our algorithm is in definition of a leaf, where a leaf $\{i,j,k\} \in HV^3\,(i < j < k)$ is defined as a 3-tuple of vertices connected to an edge (i,j,k), at least two of which has degree 1. As in [14], we define a bunch of leaves as vertices in a maximal family of leaves with the same vertex. For example, if there are two edges $\{i,j_1,j_2\}$ and $\{i,k_1,k_2\}$ and the degree of vertices j_1,j_2,k_1 and k_2 is 1, a set of vertices $\{i,j_1,j_2,k_1,k_2\}$ is a bunch of leaves.

The definition of the GLR algorithm is as follows:

Step 1: The initial graph G is named $G^{(0)}$. Set $k = 0$.

Step 2: Search all leaves from the graph $G^{(k)}$. If there is no leaf, go to Step 6.

Step 3: Remove all the leaves except for a bunch of leaves. In each bunch of leaves, remove only one of leaves. Then, the others are isolated with degree 0.

Step 4: Assign covered vertices to the one with the maximal degree in each removed leaf from $G^{(k)}$.

Step 5: The left graph is named $G^{(k+1)}$, and return to Step 2 with k increased by 1.

Step 6: If there exist connected vertices in the left graph, assign all of them to covered vertices. Stop the algorithm.

Figure 1 is an example of the GLR algorithm on a 3-uniform hypergraph. It is proven that the result of the algorithm is independent of order of removal and a selection of a leaf out of a bunch of leaves in the removal process. In the example, the resultant numbers of covered and isolated vertices are determined regardless of the selection of a removed leaf, {1,2,5} or {3,4,5}. As shown in Fig. 1, when the recursive steps stop, the left graph consists of isolated vertices and a core, which is defined as a set of vertices connected to edges without leaves. Vertices which are not selected for the removal

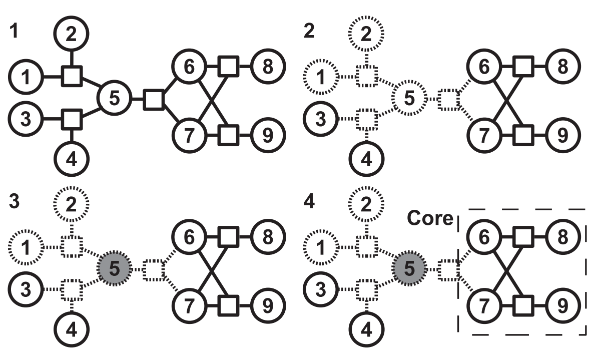

FIG. 1. A series of GLR procedures on a 3-uniform hypergraph. 1) Step 2 ($k = 0$): The initial graph $G^{(0)}$ has a bunch of leaves {1,2,3,4,5}, where {1,2,5} and {3,4,5} are leaves. 2) Step 3: By the "bunch of leaves" rule, either leaf must be removed. Here, a leaf {1,2,5} is removed (drawn by dotted lines) and vertices 3 and 4 become isolated. 3) Step 4: As an optimal solution, the GLR algorithm assigns "covered" to vertex 5 with degree 3 and "uncovered" to vertices 1,2,3, and 4. Step 5: The left graph with vertices 3,4,6,7,8 and 9 becomes $G^{(1)}$. The GLR algorithm goes to the next iteration as $k = 1$. 4) Step 1 ($k = 1$): $G^{(1)}$ has no leaves. Then, the GLR algorithm goes to Step 6. Step 6: Vertices 6,7,8, and 9, which belong to a core (component surrounded by a broken line), are assigned to covered vertices.

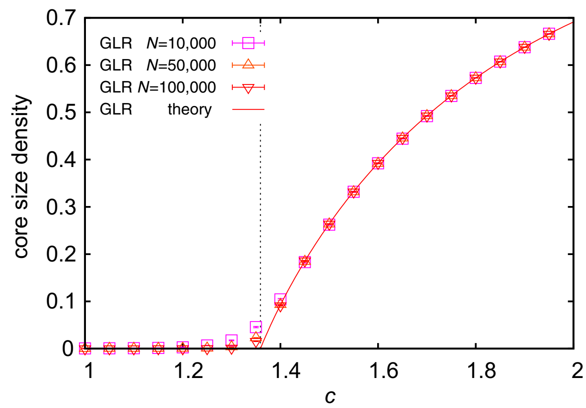

FIG. 2. (Color online) The core size density in the GLR algorithm as a function of the average degree c. Open marks are the data obtained by the GLR algorithm with the vertex size 10^4, 5×10^4, and 10^5, which are taken an average over 10^4 random hypergraphs. The solid line is the core size density predicted by the asymptotic analysis of the recursive formula in the GLR algorithm. The vertical dotted line represents the critical average degree $c_* = e/2$.

in Step 3 become isolated and the core with $O(N)$ vertices exists in large c. We note that Step 4 can be omitted if one is interested only in the minimum-cover ratio, not the covered vertices. Because the algorithm covers all vertices in the core without searching an optimal solution of the min-VC problem as shown in Step 6, the existence of the core with $O(N)$ vertices leads to overestimation of the average minimum-cover ratio. We study the core size at the end of the GLR algorithm by numerically performing the above-mentioned procedure for finite-size random hypergraphs with $\alpha = 3$. While the computational time for the GLR algorithm is proportional to the number of vertices, it takes time of the order of N^3 for generating a random graph. To avoid it, we use the microcanonical ensemble [14] with fixing the number of edges to the expectation number of edges $cN/3$, ignoring fluctuation of the average degree. We expect that such fluctuation is irrelevant in a large-N limit. In Fig. 2, the core size density obtained by numerical simulations is presented as a function of the average degree c up to $N = 10^5$. The data averaged over 10^4 random graphs converge well for large sizes and a giant core with $O(N)$ emerges above a certain value of c.

We discuss the asymptotic behavior of the recursive procedure in the GLR algorithm. We introduce the fractions of the core c_n, the isolated vertices i_n, and the edges in the core l_n averaged over random hypergraphs after the nth step of the algorithm, and find

$$\begin{aligned} i_n &= e_{2n+1} + 2e_{2n} + 2ce_{2n}e_{2n-1}^2 - 2,\\ c_n &= e_{2n} - e_{2n+1} - 2ce_{2n}e_{2n-1}^2 + 2ce_{2n-1}^3,\\ l_n &= \frac{c}{3}(e_{2n} - e_{2n-1})^2(e_{2n} + 2e_{2n-1}), \end{aligned} \tag{5}$$

where a parameter e_n $(n \geqslant -1)$ obeys a recursion relation,

$$e_n = \exp\left(-ce_{n-1}^2\right), \quad e_{-1} = 0. \tag{6}$$

A detailed derivation of the formulas is shown in the Appendix. By definition, the average fraction of the removed vertices r_n

up to the nth step is given by $r_n = 1 - i_n - c_n$. These fractions are governed by the sequence of e_n and their values at the end of the algorithm are determined by the asymptotic behavior of the recursion relation of $\{e_n\}$.

It is found that there exists a critical average degree $c_* = e/2$ for the recursion relation. Below the critical value, the sequence $\{e_n\}$ converges to the unique value $[W(2c)/(2c)]^{1/2}$ and consequently the core size c_∞ is zero. Above the critical value, however, a bifurcation occurs in the recursion relation and the sequence has a cycle with period 2. This type of transition would occur above $\alpha = 3$ at the critical average degree $c_* = e/(\alpha - 1)$. Because $e_{-1} = 0$, an even term e_{2n} is larger than that at one step later, that is e_{2n+1}. We compute the limiting values $\lim_{n\to\infty} e_{2n+1}$ and $\lim_{n\to\infty} e_{2n}$ numerically as a function of c. The difference between them yields the emergence of the $O(N)$ core. We present the core size density obtained from the asymptotic analysis of the recursion relation by the solid line in Fig. 2, which coincides with numerical GLR data. We thus confirm that a core percolation occurs at the critical average degree in the GLR algorithm, which coincides with that of the RS instability. From the analysis near the critical degree, we find

$$\begin{aligned} c_\infty &= \frac{1}{\sqrt{e}}\left(6\epsilon - 3\sqrt{6}\epsilon^{3/2} + \frac{51}{10}\epsilon^2\right) + O(\epsilon^{5/2}), \\ l_\infty &= \frac{1}{\sqrt{e}}\left(3\epsilon - \frac{\sqrt{6}}{2}\epsilon^{3/2} + \frac{27}{20}\epsilon^2\right) + O(\epsilon^{5/2}), \end{aligned} \tag{7}$$

where $\epsilon = (c - c_*)/c_* (\geqslant 0)$. The core thus emerges linearly near above the critical average degree. These findings, the bifurcation in the recursion relation and the core percolation, are common in the min-VC problems on random graphs with $\alpha = 2$ [14].

As mentioned above, the GLR algorithm estimates the minimum-cover ratio by the size of the removed part in the graph during the recursive procedure, which is given as $r_\infty = 1 - i_\infty - c_\infty$.

Taking one-third of r_∞ and adding c_∞ to the value, we obtain the estimate of the average minimum-cover ratio by the algorithm. Thus, we find that below the critical average degree $e/2$ the estimate $r_\infty/3$ coincides with the RS solution Eq. (4) estimated by the replica method. In contrast, the sequence $\{e_n\}$ of the algorithm does not converge to a unique value above the critical value and the GLR algorithm could not give a precise estimate of x_c there.

Here, we show the asymptotic analysis of the recursion relation in the GLR algorithm on 3-uniform hypergraphs. It revealed that the $O(N)$ core emerges at the critical average degree $c_* = e/2$, where the RS solution becomes unstable. While this analysis offers thermodynamic properties of the core, it is still hard to predict other critical properties of the core percolation, especially finite-size effects. In the next section, we show some numerical results according to typical performance of the GLR algorithm and critical properties of the core.

IV. NUMERICAL RESULTS

In order to confirm whether these analyses in the previous section estimate the average minimum-cover ratio x_c for $\alpha = 3$ correctly, we study the min-VC problems by a Markov chain Monte Carlo (MC) method. In particular, we use the replica exchange Monte Carlo method [26,27] for accelerating the equilibriation of the system. In our simulations, the total number of replicas is fixed to be 50 and each has a different value of chemical potential ranging from −2 to 10. The value of x_c for each random graph is estimated as the smallest cover ratio found during MC simulations with 2^{17} Monte Carlo steps, which is averaged over 800 hypergraphs randomly generated. The average minimum-cover ratio is extrapolated from the numerical results for finite sizes N up to $N = 512$ by fitting a second-order polynomial of $1/N$. Figure 3 shows the obtained minimum-cover ratio as a function of the average degree c. Below the critical average degree $e/2$ where the RS solution is considered to be correct, the MC result is consistent with those by the two approaches, the replica method and the GLR algorithm. Above the critical value, on the other hand, the MC estimate stays slightly above that by the replica method and considerably deviates from that by the GLR algorithm. The former is due to the instability of the RS solution and the latter is the existence of the core with $O(N)$ vertices.

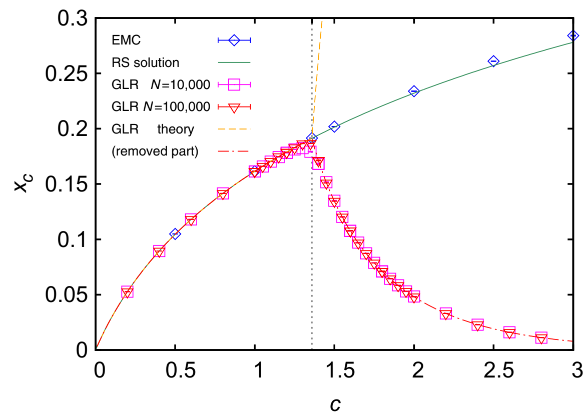

FIG. 3. (Color online) The average minimum-cover ratio on random α-uniform hypergraphs with $\alpha = 3$ as a function of the average degree c. Open marks are numerical results by the exchange MC (diamonds) and by the GLR algorithm for $N = 10^4$ (squares) and 10^5 (triangles). Lines represent analytic results by the replica method (solid), by the GLR algorithm (dashed), and on the removed part of the graphs by the GLR algorithm (dashed-dotted). The vertical dotted line is the critical average degree $c_* = e/2$, below which all lines merge into a single line.

Next, we study the critical properties of the core percolation in the GLR algorithm. While the critical threshold of the core percolation and the exponent of the core size are determined in the previous section, some critical exponents are not obtained analytically from the asymptotic analysis of the GLR algorithm. Thus, we study the critical nature of the core percolation by numerically performing the GLR algorithm. We measure the number of vertices in the core, N_c, and that of edges in the core, L_c, for a given hypergraph with the vertex size N. At the critical average degree $c = c_*$ of the core percolation, N_c, L_c and its average connectivity $c_{\mathrm{eff}} = 3L_c/N_c$ are expected to exhibit power-law behavior as

$$\begin{aligned} N_c(c_*) \sim L_c(c_*) \sim N^{\omega}, \\ c_{\mathrm{eff}}(c_*) - \tfrac{3}{2} \sim N^{-\phi}, \end{aligned} \tag{8}$$

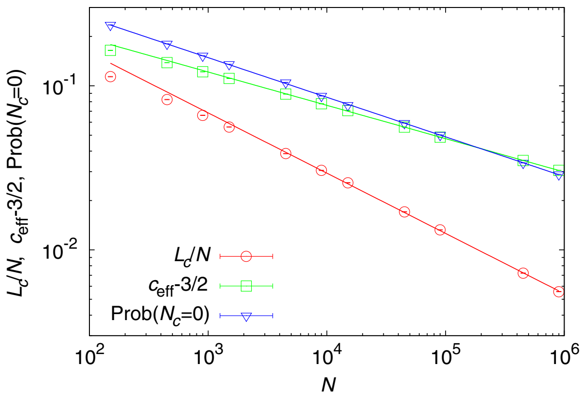

FIG. 4. (Color online) Finite-size scaling plot of some quantities of interest for the core percolation in the GLR algorithm at the critical average degree $c = e/2$. Open circles are an average ratio of edges L_c/N in the core, open squares are $c_{\rm eff} - 3/2$ with $c_{\rm eff}$ being an effective average degree in the core, and open triangles are the probability $\text{Prob}(N_c = 0)$ of the random graph for completely removable in the GLR algorithm. The lines represent results obtained by a fitting to power-law decay in N.

where the average connectivity 3/2 in the large N limit is derived from Eq. (7). We also evaluate probability of random hypergraphs removed completely by the GLR algorithm, which is also assumed to be a power law in N with another exponent η,

$$\text{Prob}(N_c(c_*) = 0) \sim N^{-\eta}. \tag{9}$$

These quantities are shown in Fig. 4, where the average is taken over 10^5 random hypergraphs. The data in Fig. 4 are fitted by the power laws with

$$\omega = 0.63(1), \quad \phi = 0.20(1), \quad \eta = 0.241(1). \tag{10}$$

We then perform the finite-size-scaling analyses of the fluctuation of N_c and L_c near the critical average degree.

According to the standard finite-size scaling ansatz, the singular part of a quantity $Q(c,N)$ which behaves as $Q(c,\infty) \sim |c - c_*|^{\omega_Q}$ follows the leading scaling form

$$Q(c,N) = N^{-\omega_Q \theta} f_Q[(c - c_*)N^\theta], \tag{11}$$

where ω_Q and θ are critical exponents and $f_Q(x)$ is a scaling function. In order to obtain the data for the finite-size scaling analysis, we perform the GLR algorithm for some values of c around c_* on 10^4 hypergraphs whose sizes are $N = 9 \times 10^3$, 9×10^4, and 9×10^5. All the finite-size scalings shown below are done by using a Bayesian scaling analysis [28]. As discussed in the previous section, the fraction N_c/N of vertices and that L_c/N of edges in the core are proportional to $c - c_*$ as $c \to c_* + 0$, leading that $\omega_Q = 1$ for L_c. Thus, the finite-size scaling form for L_c/N has an unknown value of the exponent θ. Figure 5 shows the finite-size scaling plot for L_c/N with $\theta = 0.38(1)$. Here, other finite-size scaling analyses are performed to deal with the fluctuation of fractions L_c/N and $c_{\rm eff}$, and confirm scaling relations with ω and ϕ.

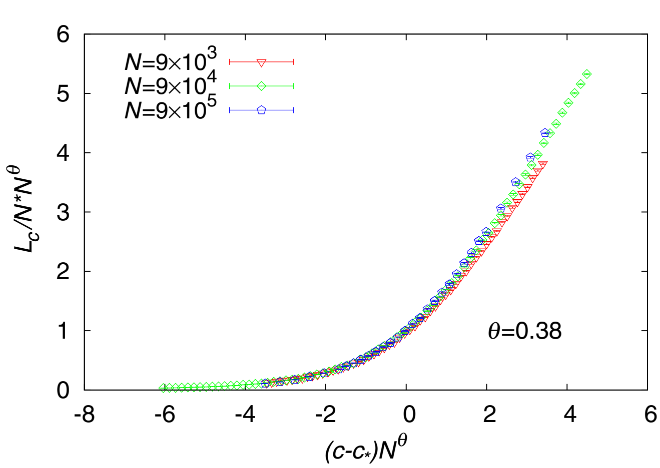

FIG. 5. (Color online) Finite-size scaling plot of the number of edges L_c in the core with the exponent $\theta = 0.38(1)$.

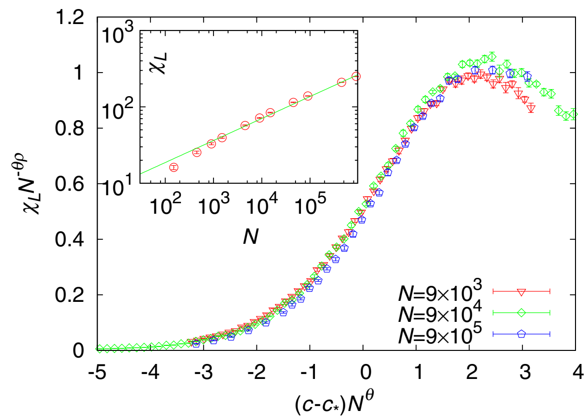

FIG. 6. (Color online) Finite-size scaling plot of the variance $\chi_L(c,N)$ of the edge number in the core. The obtained scaling parameters are $\theta = 0.37(2)$ and $\rho = 0.77(2)$. The inset shows the size dependence of χ_L at the critical average degree. The slope represents the asymptotic power law as $N^{\theta\rho}$ expected from the finite-size scaling analysis.

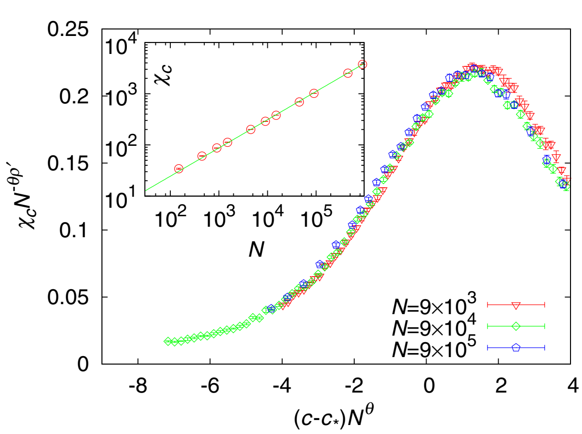

FIG. 7. (Color online) Finite-size scaling plot of the variance $\chi_c(c,N)$ of the core size. The obtained scaling parameters are $\theta = 0.39(2)$ and $\rho' = 1.39(2)$. The inset shows the size dependence of χ_c at the critical average degree. The slope represents the asymptotic power law as $N^{\theta\rho'}$ expected from the finite-size scaling analysis.

TABLE I. Critical exponents of a percolation transition of the core generated by the GLR algorithm. The first line presents our numerical results for $\alpha = 3$. The second one is those for $\alpha = 2$ and the bottom line displays those conjectured for $\alpha = 2$ in Ref. [14].

	θ	ω	ϕ	ρ	ρ'	η
$\alpha = 3$	0.38(1)	0.63(1)	0.20(1)	0.7(1)	1.4(1)	0.241(1)
$\alpha = 2$	0.36(3)	0.63(1)	0.21(1)		1.5(1)	0.25(1)
Conjecture	2/5	3/5	1/5	1/2	3/2	24/100

We define normalized variances χ_L and χ_c as $\mathrm{Var}(L_c)/N$ and $N\mathrm{Var}(c_{\mathrm{eff}})$, respectively, where $\mathrm{Var}(\cdots)$ means a variance, and the corresponding critical exponents ρ and ρ' are defined by

$$\chi_L(c,\infty) \sim |c - c_*|^{-\rho}, \quad \chi_c(c,\infty) \sim |c - c_*|^{-\rho'}. \tag{12}$$

They are examined using the finite-size scaling hypothesis (11) and the scaling plots are presented in Figs. 6 and 7, respectively. From these numerical observations, we evaluate the critical exponents associated with the core percolation in the first line of Table I.

Bauer and Golinelli have evaluated those critical exponents in the case of $\alpha = 2$ which are significantly different from those of the bond percolation on a random graph and discussed their scaling relations among the exponents [14].

These relations are appropriate if the scaling relations (8), (11), and (12) are also valid for $\alpha = 3$. Our finite-size scaling analyses are marginally compatible with the scaling relations, and the evaluated critical exponents are sufficiently close to numerical results and their conjectures in the case of $\alpha = 2$ are shown in Table I. These results suggest that there is a universal "GLR core" class, which is independent to the uniform edge degree α.

V. SUMMARY AND DISCUSSION

To summarize, we have discussed the minimum vertex cover problems on random α-uniform hypergraphs by the statistical-mechanical replica method and the approximation algorithm. The former estimates the average minimum-cover ratio x_c as a function of the average degree c under the replica symmetric assumption. We find that an RS/RSB phase transition occurs at the critical average degree $c_* = e/(\alpha - 1)$, which is well above a percolation threshold $c = 1/(\alpha - 1)$ in the random graph. We also extend the global leaf removal algorithm to the problem with $\alpha \geqslant 3$ and study the asymptotic behavior of the recursive procedure of the algorithm, particularly in the case of $\alpha = 3$. If the average degree is below the critical value which coincides with that in the replica theory, the cores at the end of the GLR algorithm remain to be $o(N)$ and they do not affect the estimate of the minimum-cover ratio. In contrast, above the critical value, the $O(N)$ core emerges, leading to a wrong estimation of the minimum-cover ratio. Comparing the results obtained by MC simulations, we confirm that these estimates are correct below the critical average degree, but this is not the case above it. These results strongly suggest that there is a close relation between the replica symmetry in statistical physics and the performance limitation of the leaf removal algorithm even when the edge size α is larger than 2, although the similar relation has been pointed out for $\alpha = 2$ [14]. We also perform the finite-size scaling analysis of the core generated by the GLR algorithm and find that the critical exponents estimated are consistent with those of the GLR core percolation with $\alpha = 2$ and that they are different from those of a normal percolation transition on the random graph. This implies that there exists a universal class of the GLR core percolation irrespective of the edge size. After the submission of this paper, we became aware of a recent work [29] in which the GLR algorithm for the maximum set packing (MSP) problem also generates a core suggesting the existence of the RS/RSB transition. As the MSP problem is the dual problem of the min-VC problem, it is expected that the core found in the MSP has the same critical properties reported in this paper.

It is noted that the relation is not always true for all types of random graphs. For instance, the GLR algorithm is not able to remove any vertex on regular random graphs with $c \geqslant 2$ because no leaf is found there while, from the point of the statistical-mechanical view, the min-VC problems on regular random graphs with degree 2 are described by the RS solution [19]. Thus, the relation depends on a type of random graphs and approximation algorithms to be used to solve the problem. In addition to the leaf removal algorithm, a recent work for the min-VC problem with $\alpha = 2$ [16,30] suggests that the linear programming method, which is one of the most commonly used techniques for solving optimization problems, has the relation discussed in the present work.

Further study will be needed to establish the relation between the replica symmetry and the performance of numerous algorithms.

ACKNOWLEDGMENTS

This research was supported by Grants-in-Aid for Scientific Research from MEXT, Japan (Grants No. 22340109, No. 25120010, and No. 25610102), and by the JSPS Core-to-Core program "Non-equilibrium dynamics of soft-matter and information."

APPENDIX: RECURSIVE ANALYSIS OF GLR

In this Appendix we show details of recursive relations of the GLR algorithm for $\alpha \geqslant 3$. Following Ref. [14], we consider the GLR algorithm on a three-uniform random rooted (hyper)tree with the average degree c. On the rooted tree we distinguish a root from other vertices. We perform the GLR algorithm on it except that we do not remove the root before the degree of its two neighbors becomes 1. Then we define p_n ($n \geqslant 0$) and q_n ($n \geqslant 1$) as a generating function of rooted trees whose root is removed exactly at the nth removal step and whose root becomes isolated at the nth steps, respectively. To simplify our notation we also use p_n and q_n to express a state of a vertex. We define the sum of each state as

$$P_n = \sum_{k=0}^{n} p_k, \quad Q_n = \sum_{k=1}^{n} q_k, \tag{A1}$$

where $P_\infty + Q_\infty = 1$ because all the roots are identified with either p_n or q_n.

These quantities are to be evaluated as a function of a recursive step n. When the root is connected to a pair of neighbors by an edge, we denote their states by a pair of p_n and q_n. A state of the neighboring vertex is determined on a new graph rooted by the neighboring vertex without the original root and edges connected to them. For example, a couple of states $(p_n, \cdot)$ means that a state of a neighbor is p_n and the other is in any states ("$\cdot$" represents an arbitrarily state). The root with degree 0 is denoted by p_0 and its existence probability is e^{-c}. The root of p_n should have $k(\geqslant 1)$ edges connected to at least a vertex of q_n out of two vertices which are not Q_{n-1}. Except for those edges it can also be connected to $(Q_{n-1}, \cdot)$. We express the number of the pairs as $l(\geqslant 0)$. Hence, we find

$$\begin{aligned} p_n &= e^{-c}\sum_{k\geqslant 1}\frac{\{c[q_n^2+2q_n(1-Q_n)]\}^k}{k!}\\ &\quad\times\sum_{l\geqslant 0}\frac{\{c[1-(1-Q_n)^2]\}^l}{l!}\\ &= \exp\left[-c(1-Q_n)^2\right]-\exp\left[-c(1-Q_{n-1})^2\right]. \end{aligned} \tag{A2}$$

The root of q_n should have $k(\geqslant 1)$ pairs of neighbors with (p_{n-1}, P_{n-1}) and other $l(\geqslant 0)$ pairs without P_{n-1}. Then q_n reads

$$\begin{aligned} q_n &= e^{-c}\sum_{k\geqslant 1}\frac{\left[c\left(P_{n-1}^2-P_{n-2}^2\right)\right]^k}{k!}\sum_{l\geqslant 0}\frac{\left[c\left(1-P_n^2\right)\right]^l}{l!}\\ &= \exp\left[-cP_{n-2}^2\right]-\exp\left[-cP_{n-1}^2\right]. \end{aligned} \tag{A3}$$

Rewriting Eqs. (A2) and (A3), they read

$$p_n = e_{2n+1}-e_{2n-1}\ (n\geqslant 0),\quad q_n = e_{2n-2}-e_{2n}\ (n\geqslant 1). \tag{A4}$$

These relations lead to Eq. (6). This analysis on rooted trees can be straightforwardly expanded to the case of general uniform hypergraphs.

Here we discuss fractions i_n and r_n of isolated and removed vertices at the nth recursive step of the GLR algorithm. Using a trivial relation $i_n + c_n + r_n = 1$, the core fraction c_n is obtained from i_n and r_n. While p_n and q_n mentioned above are clues to estimate those fractions, they should be reclassified because of the exception rule on the rooted trees. We divide p_n into two fractions p_n' and p_n'' for calculating i_n. Defining p_n' as a generating function of vertices which connect to a pair of neighbors (q_n, P_{n-1}) and p_n'' as the others, we find that the vertices of p_n'' eventually are isolated at the nth step. It reads

$$\begin{aligned} p_n'' &= p_n - e^{-c}c(2q_nP_{n-1})\sum_l\frac{\left\{c\left[1-(1-Q_{n-1})^2\right]\right\}^l}{l!}\\ &= p_n - 2ce_{2n-1}^2(e_{2n-2}-e_{2n}). \end{aligned} \tag{A5}$$

In contrast, p_n' is potentially removed at the nth step except for a bunch-of-leaves rule. If a vertex of q_n connects to more than two couples of (p_{n-1}, P_{n-1}), the GLR algorithm removes only one of them and leaves others isolated by the rule. We note that these neighbors are all in the p' state. The isolated fraction reads

$$\begin{aligned} &e^{-c}\sum_{k\geqslant 1}2(k-1)\frac{\left[c\left(P_{n-1}^2-P_{n-2}^2\right)\right]^k}{k!}\sum_l\frac{\left[c\left(1-P_{n-1}^2\right)\right]^l}{l!}\\ &= 2c\left(e_{2n-1}^2-e_{2n-3}^2\right)e_{2n-2}-2q_n. \end{aligned} \tag{A6}$$

We therefore find the fraction Δi_n of isolated vertices at the nth step,

$$\Delta i_n = p_n - 2q_n + 2c\left(e_{2n}e_{2n-1}^2 - e_{2n-2}e_{2n-3}^2\right). \tag{A7}$$

Starting with the initial condition $i_0 = e^{-c} = e_1$, the fraction of isolated vertices after the nth recursive step is obtained as

$$i_n = e_{2n+1} + 2e_{2n} + 2ce_{2n}e_{2n-1}^2 - 2. \tag{A8}$$

Next, we evaluate the removed fraction r_n. The basic strategy is to count a fraction of q_n and its neighbors at the nth step, but it requires classification to avoid a multicounting of the removed vertices. In the calculation of Eq. (A3), we denote the number of pairs (p_{n-1}, P_{n-1}) and that of others by k and l, respectively. Vertices in the q_n state are classified by its neighbors as follows:

(1) q_n': $k\geqslant 2$.

(2) q_n'': $k = 1$ and $l\geqslant 0$ pairs of $(Q_{n-2}, \cdot)$.

(2a) $\tilde{q}_n$: Only one pair with (p_{n-1}, p_{n-1}).

(2b) $\hat{q}_n$: Without $\tilde{q}_n$ out of q_n''.

(3) q_n^*: $k = 1$ and $l\geqslant 1$ pairs with $(Q_{n-1}, \cdot)$ at least one of which needs to be $(q_{n-1}, \cdot)$.

(4) q_n''': Others.

A careful observation shows that the GLR algorithm removes three vertices from vertices with q_n' and q_n''' and a vertex from those with q_n^* and $\tilde{q}_n$ at the nth step. Then the increment of the fraction Δr_n of removed vertices in the nth step is given by

$$\begin{aligned} \Delta r_n &= 3q_n' + q_n^* + \tilde{q}_n + 3q_n'''\\ &= 3q_n - 2ce_{2n-1}^3 + 2ce_{2n-3}^3. \end{aligned} \tag{A9}$$

Combined with the initial condition $r_0 = 0$, the expression of r_n is obtained as

$$r_n = 3 - 3e_{2n} - 2ce_{2n} - 2ce_{2n-1}^3, \tag{A10}$$

and eventually the core fraction c_n at the nth step is obtained as in Eq. (5).

A ratio l_n of unremoved edges per the number of vertices N and a ratio t_n of removed edges at the nth step are also estimated. They obey a simple relation $l_n + t_n = c/3$. It is rather easy to estimate all edges connected to q_n vertices as

$$\begin{aligned} &e^{-c}\sum_{k\geqslant 1}\sum_l(k+l)\frac{\left[c\left(P_{n-1}^2-P_{n-2}^2\right)\right]^k}{k!}\frac{\left[c\left(1-P_{n-1}^2\right)\right]^l}{l!}\\ &= ce_{2n-2} - ce_{2n} + ce_{2n}e_{2n-1}^2 - ce_{2n-2}e_{2n-3}^2. \end{aligned} \tag{A11}$$

It is, however, obviously overestimated because some edges connect to other vertices in q_n and some are already removed before the nth step. Recalling the classification of vertices in q_n, we notice that some type of edges depending on the indices

k and l causes this overcounting. They are estimated as

$$q_n'' + \frac{1}{2}\tilde{q}_n + e^{-c}\sum_{k\geqslant 1}\frac{\left[c\left(P_{n-1}^2 - P_{n-2}^2\right)\right]^k}{k!} \times \sum_{l,l'}\left(l+\frac{l'}{2}\right)\frac{(cq_n^2)^l}{l!}\frac{[2cq_n(1-Q_n)]^{l'}}{l'!} \times \sum_m \frac{\left\{c\left[1-P_{n-1}^2 - q_n^2 - 2q_n(1-Q_n)\right]\right\}^m}{m!} = \frac{2}{3}ce_{2n-1}^3 - \frac{2}{3}ce_{2n-3}^3 + \frac{2}{3}cq_n^3 + cq_n^2 e_{2n}. \tag{A12}$$

Edges connected to (Q_{n-1},Q_{n-1}) or $(Q_{n-1},1-Q_{n-1})$ are removed before the nth step and the ratio reads

$$e^{-c}\sum_{k\geqslant 1}\frac{\left[c\left(P_{n-1}^2 - P_{n-2}^2\right)\right]^k}{k!} \times \sum_l l\frac{\left\{c\left[Q_{n-1}^2 + 2Q_{n-1}(1-Q_{n-1})\right]\right\}^l}{l!} \times \sum_m \frac{\left\{c\left[1-P_{n-1}^2 - Q_{n-1}^2 - 2Q_{n-1}(1-Q_{n-1})\right]\right\}^m}{m!} = cq_n(1-e_{2n-2}^2). \tag{A13}$$

These relations allow us to lead the incremental of t_n as

$$\Delta t_n = -\frac{c}{3}(e_{2n}-e_{2n-1})^2(e_{2n}+2e_{2n-1}) + \frac{c}{3}(e_{2n-2}-e_{2n-3})^2(e_{2n-2}+2e_{2n-3}). \tag{A14}$$

Then, t_n and l_n are derived as

$$t_n = \frac{c}{3} - \frac{c}{3}(e_{2n}-e_{2n-1})^2(e_{2n}+2e_{2n-1}),\quad l_n = \frac{c}{3}(e_{2n}-e_{2n-1})^2(e_{2n}+2e_{2n-1}), \tag{A15}$$

with $t_0 = 0$.

[1] R. M. Karp, *Complexity of Computer Computations*, edited by R. E. Miller and J. W. Thatcher (Plenum, New York, 1972), p. 85.
[2] W. H. Kautz and R. C. Singleton, IEEE Trans. Inf. Theory **10**, 363 (1964).
[3] R. Dorfman, Ann. Math. Stat. **14**, 436 (1943).
[4] S. Khot and O. Regev, J. Comput. Syst. Sci. **74**, 335 (2008).
[5] M. Mézard and A. Montanari, *Information, Physics, and Computation* (Oxford University Press, Oxford, 2009).
[6] M. Mézard, G. Parisi, and M. Á. Virasoro, *Spin Glass Theory and Beyond* (World Scientific, Singapore, 1987).
[7] S. Kirkpatrick and B. Selman, Science **264**, 1297 (1994).
[8] S. Franz, M. Leone, F. Ricci-Tersenghi, and R. Zecchina, Phys. Rev. Lett. **87**, 127209 (2001).
[9] L. Zdeborová and F. Krząkała, Phys. Rev. E **76**, 031131 (2007).
[10] M. Weigt and A. K. Hartmann, Phys. Rev. Lett. **84**, 6118 (2000).
[11] M. Weigt and A. K. Hartmann, Phys. Rev. E **63**, 056127 (2001).
[12] H. Zhou, Eur. Phys. J. B **32**, 265 (2003).
[13] M. Weigt and H. Zhou, Phys. Rev. E **74**, 046110 (2006).
[14] M. Bauer and O. Golinelli, Eur. Phys. J. B **24**, 339 (2001).
[15] M. Mézard, F. Ricci-Tersenghi, and R. Zecchina, J. Stat. Phys. **111**, 505 (2003).
[16] T. Dewenter and A. K. Hartmann, Phys. Rev. E **86**, 041128 (2012).
[17] E. Gardner, Nucl. Phys. B **257**, 747 (1985).
[18] J. van Mourik and D. Saad, Phys. Rev. E **66**, 056120 (2002).
[19] M. Mézard and M. Tarzia, Phys. Rev. E **76**, 041124 (2007).
[20] R. Monasson, R. Zecchina, S. Kirkpatrick, B. Selman, and L. Troyansky, Nature (London) **400**, 133 (1999).
[21] A. K. Hartmann and M. Weigt, *Phase Transitions in Combinatorial Optimization Problems*, (Wiley-VCH Verlag GmbH & Co. KGaA, Weinheim, 2005).
[22] R. Monasson, J. Phys. A **31**, 513 (1998).
[23] J. R. L. de Almeida and D. J. Thouless, J. Phys. A **11**, 983 (1978).
[24] R. M. Karp and M. Sipser, in *Proceedings of 22nd Annual Symposium on Foundations of Computer Science* (IEEE Computer Society, Los Alamitos, 1981), p. 364.
[25] B. Pittel, J. Spencer, and N. Wormald, J. Comb. Theory B **67**, 111 (1996).
[26] C. J. Geyer, in *Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface*, edited by E. M. Keramidas (Interface Foundation of North America, Fairfax Station, 1991), p. 156.
[27] K. Hukushima and K. Nemoto, J. Phys. Soc. Jpn. **65**, 1604 (1996).
[28] K. Harada, Phys. Rev. E **84**, 056704 (2011); http://kenjiharada.github.io/BSA/.
[29] C. Lucibello and F. Ricci-Tersenghi, Int. J. Stat. Mech. (2014) 136829.
[30] S. Takabe and K. Hukushima, J. Phys. Soc. Jpn. **83**, 043801 (2014).