The Peculiar Phase Structure of Random Graph Bisection

Allon G. Percus[*]   Gabriel Istrate[†]   Bruno Gonçalves[‡]

Robert Z. Sumi[§]   Stefan Boettcher[¶]

Abstract

The mincut graph bisection problem involves partitioning the n vertices of a graph into disjoint subsets, each containing $n/2$ vertices, while minimizing the number of "cut" edges with an endpoint in each subset. When considered over sparse random graphs, the phase structure of the graph bisection problem displays certain familiar properties, but also some surprises. It is known that when the mean degree is below the critical value of $2\log 2$, the cutsize is zero with high probability. We study how the minimum cutsize increases with mean degree above this critical threshold, finding a new analytical upper bound that improves considerably upon previous bounds. Combined with recent results on expander graphs, our bound suggests the unusual scenario that random graph bisection is replica symmetric up to and beyond the critical threshold, with a replica symmetry breaking transition possibly taking place *above* the threshold. An intriguing algorithmic consequence is that although the problem is NP-hard, we can find cutsizes that are asymptotically within a factor 1 of optimal — and possibly even the optimum itself — in polynomial time for typical instances near the phase transition.

1 Introduction

The graph bisection problem arises in a wide range of contexts, ranging from VLSI design [1] to computer vision [9, 21] to protecting networks from attack [18]. The basic mincut graph bisection (or graph bipartitioning) problem is defined as follows. Given an undirected,

[*]Information Sciences Group, Computer, Computational and Statistical Sciences Division, Los Alamos National Laboratory, Los Alamos, NM 87545, USA and Department of Mathematics, UCLA, Los Angeles, CA 90095, USA, e-mail: percus@ipam.ucla.edu

[†]eAustria Research Institute, Bd. V. Pârvan 4, cam. 045B, RO-300223 Timişoara, Romania, e-mail: gabrielistrate@acm.org

[‡]Department of Physics, Emory University, Atlanta, GA 30322–2430, USA, e-mail: bgoncalves@physics.emory.edu

[§]Department of Physics, Babes-Bolyai University, RO-400884 Cluj, Romania and eAustria Research Institute, Bd. V. Pârvan 4, cam. 045B, RO-300223 Timişoara, Romania

[¶]Department of Physics, Emory University, Atlanta, GA 30322–2430, USA, e-mail: stb@physics.emory.edu

unweighted graph, partition its n nodes into two disjoint subsets of equal size, while minimizing the number w of "cut" edges with an endpoint in each subset. This is an NP-hard optimization problem [14].

From a statistical physics perspective, graph bisection is also equivalent to finding the ground state of a finite-range spin glass constrained to have zero magnetization [13, 20, 25, 23, 22]. Over the past decade, the study of combinatorial optimization problems within the physics community as well as the study of statistical physics approaches within the computer science community has led to remarkable successes [30]. By considering optimization problems over appropriately parametrized ensembles of random instances, physicists have developed methods leading both to analytical predictions of global optima and to algorithmic approaches for finding these. One particularly rich source of insight has been the investigation of phase transition behavior: for many combinatorial optimization problems, both the nature of the solution space and the average-case complexity of algorithms solving the problem are closely related to an underlying phase structure.

In this paper, we consider graph bisection over Erdős-Rényi (Bernoulli) random graphs [12] generated from the $\mathcal{G}_{np}$ ensemble: given n nodes, place an edge between each of the $\binom{n}{2}$ pairs of nodes independently with probability p. We are interested in the regime of sparse random graphs, where p scales as $1/n$ and so a vertex's mean degree $p(n-1)$ is some finite α. On such graphs, it is known that a phase transition occurs at $\alpha_c = 2 \log 2$ [16, 25, 22, 24]. For $\alpha < \alpha_c$, typical graphs can be bisected without cutting any edges: the *bisection width w* is zero. For $\alpha > \alpha_c$, the graph typically contains a connected component (giant component) of size greater than $n/2$ and so a bisection requires cutting edges, giving $w > 0$. In other combinatorial optimization problems such as graph coloring or satisfiability, a similar sharp threshold coincides with a rapid escalation of average-case computational complexity [11, 28, 30].

Additionally, for many of these other problems, at some $\alpha_d \leq \alpha_c$ the solution space undergoes a structural transition that corresponds to replica symmetry breaking [4, 27]. Define two optimal solutions to be *adjacent* if the Hamming distance between them is small: an infinitesimal fraction of the variables differ in value. Then below α_d, any two solutions are connected by a path of adjacent solutions. The solution space may be thought of as a single "cluster". This is a replica symmetric (RS) phase. But above α_d, the cluster fragments into multiple non-adjacent clusters, differing by an extensive number of variables. This is a replica symmetry breaking (RSB) phase. In cases where $\alpha_d < \alpha_c$, such as 3-COL and 3-SAT, this structural transition may contribute to the rise in algorithmic complexity as the main threshold α_c is approached from below [26].

We find that the scenario is quite different for graph bisection. Rather than the RSB transition taking place below α_c, our arguments suggest that it actually occurs somewhere above α_c. Graph bisection is the first combinatorial optimization problem we are aware of that displays this behavior. Furthermore, in 1987, Liao proposed an analytical solution for bisection width [22] that assumes replica symmetry above α_c, so our results help validate this prediction. Finally, the same arguments that indicate replica symmetry also suggest an algorithm for finding extremely close approximations to the optimal solution, within $\alpha \in (\alpha_c, \alpha_d)$, in polynomial time.

The rest of the paper is organized as follows. In Section 2 we study the bisection width (or "cutsize") w as a function of mean degree α. We present numerical results and derive a new upper bound by analyzing a simple greedy heuristic. In Section 3 we study the solution

cluster structure, and argue that the problem is replica symmetric through and beyond the critical threshold $\alpha_c = 2\log 2$. In Section 4 we conclude with a discussion of the physical and algorithmic consequences of this unusual phase structure.

2 Bisection Width

Formally, the problem is as follows. Given an undirected, unweighted graph $G = (V, E)$ where $|V| = n$ is even, a valid solution is a partition of the vertex set into two disjoint subsets V_1 and V_2 such that $|V_1| = |V_2| = n/2$. An optimal solution is one that minimizes the *bisection width* $w = |E(V_1, V_2)|$, i.e., the number of "cut" edges with an endpoint in each subset. We consider graphs G drawn uniformly at random from $\mathcal{G}_{np}$, and take our control parameter to be the graph's mean degree $\alpha = p(n - 1)$.

The $\mathcal{G}_{np}$ ensemble [12, 8] has been studied very extensively since its introduction by Erdős and Rényi in 1959, and many of its structural properties are known. These have a crucial effect on the bisection width.

- For $\alpha < 1$, the graph consists only of small components, the largest being asymptotically of size $O(\log n)$. The expected fraction of monomers (isolated vertices) is itself at least e^{-1}. Consequently, with high probability a perfect bisection ($w = 0$) exists.

- For $\alpha > 1$, there is a unique giant component containing $O(n)$ vertices. The second-largest component is asymptotically of size $O(\log n)$. As long as α is below the threshold $\alpha_c = 2\log 2$, the giant component contains a fraction $g < 1/2$ of the vertices, and the fraction of monomers is still above $1/4$. This allows one to prove [16, 24] that with high probability, $w = 0$ for all $\alpha < 2\log 2$.

- For $\alpha > 2\log 2$, $g > 1/2$, and so bisecting the graph requires cutting some of its edges. It is known [24] that w is extensive (scales as n) in this regime, as long as α is finite.

How does the bisection width constant w/n behave for $\alpha > \alpha_c$? Figure 1 shows simulation results using the extremal optimization [6] heuristic to find the presumed optimal bisection. These suggest that the bisection width undergoes a continuous transition at α_c, and that its first derivative is continuous as well. However, neither of these properties has yet been shown analytically. We will do so now, by deriving a straightforward upper bound on w. We give the main elements of the derivation here; a more rigorous mathematical treatment will be provided elsewhere.

We obtain the upper bound by "stripping" trees off of the giant component. For random graphs with mean degree α, it is known that the expected number of nodes in trees (the *mantle* [31]) is bn where

$$b = \alpha g(1 - g) \tag{1}$$

and the size of the giant component, gn, is given by [12]

$$g = 1 - e^{-\alpha g}. \tag{2}$$

Now imagine a greedy process (Figure 2) of removing trees, starting from the largest one and decreasing in size, until the giant component has been pruned to size $\leq n/2$. In order
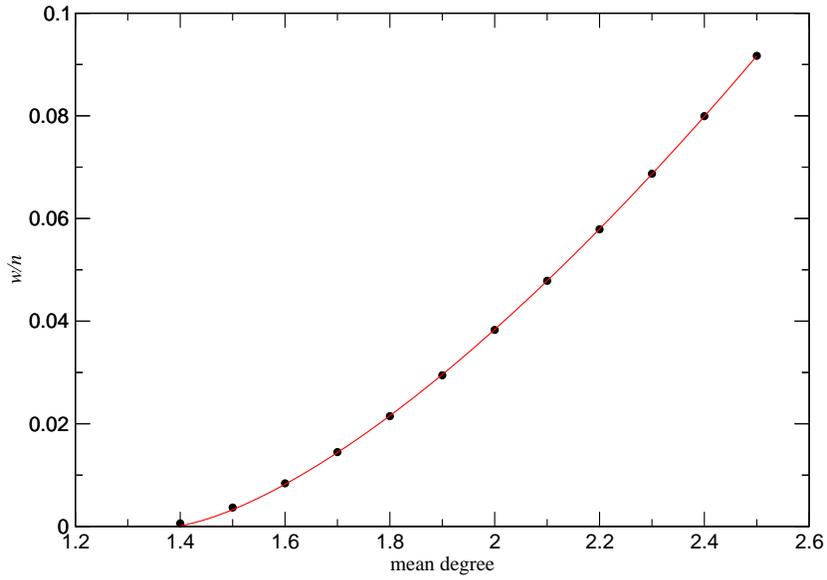
3

Figure 1: Presumed optimal bisection width, found by extremal optimization on random graphs for $n = 2000$. Line shows fit to $\gamma(\alpha - \alpha_c)^\beta$, with $\gamma \approx 0.0783$ and $\beta \approx 1.46$. See also [5].
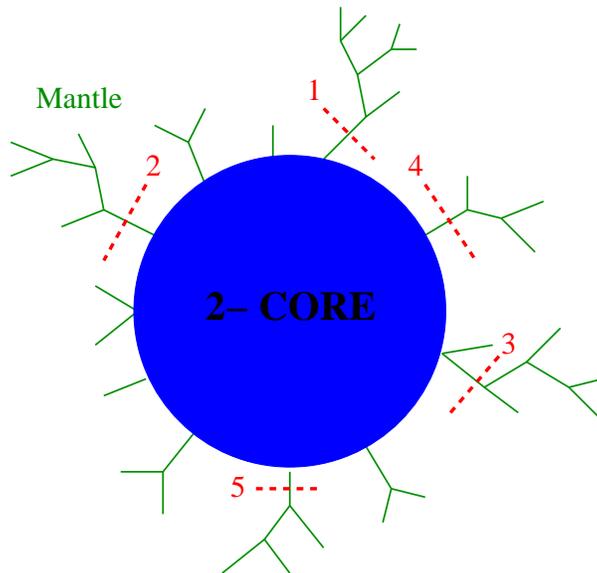

Figure 2: Schematic of giant component consisting of a *mantle* of trees and a remaining *2-core*. Greedy upper bound algorithm cuts trees of decreasing size. Numbers show order of cuts, depicted by dashed lines.

to know how many trees need to be removed, consider the distribution of tree sizes, $P(t)$. A fortunate consequence of the probabilistic independence in $\mathcal{G}_{np}$ is that $P(t)$ is simply given by the number of ways of constructing a tree of size t from q roots and r other nodes, where $q = (g - b)n$ is the number of nodes in the *2-core* of the giant component (all nodes *not* in the mantle), and $r = bn$ is the number of nodes in the mantle [19, 2]. A basic combinatorial argument then gives

$$P(t) = \binom{r}{t} t^t \frac{q}{r} \frac{(q + r - t)^{r-t+1}}{(q + r)^{r-1}}, \tag{3}$$

and for large n, letting $\rho = b/g$,

$$P(t) \approx \frac{(\rho t)^t e^{-\rho t}}{t!} \frac{1 - \rho}{\rho} \tag{4}$$

for $t \geq 1$.

Now let z be the total number of trees. Then, given $P(t)$, the total number of nodes in all trees is $\sum_{t=1}^{\infty} tP(t)z$. As this is simply the size of the mantle,

$$z = \frac{bn}{\sum_{t=1}^{\infty} tP(t)}. \tag{5}$$

We need to find the largest t_0 such that the number of nodes living on trees of size $\geq t_0$ is at least the total number of nodes we need to cut from the giant component. Let $\delta = g - 1/2$ be precisely this "excess" of the giant component. Then, we ask for the t_0 satisfying

$$\sum_{t=t_0+1}^{\infty} tP(t)z < \delta n \leq \sum_{t=t_0}^{\infty} tP(t)z. \tag{6}$$

Bisecting the graph is now simply a matter of cutting each tree of size $\geq t_0$. Any resulting imbalance in the partitions (from "overstripping") can easily be corrected by transferring enough monomers: there is an extensive number of these, whereas the largest trees are only of size $O(\log n)$. Thus,

$$w \leq \sum_{t=t_0}^{\infty} P(t)z. \tag{7}$$

To evaluate this sum, first consider z. Normalization of $P(t)$, from Eq. (4), gives,

$$\sum_{t=1}^{\infty} \frac{(\rho t)^t e^{-\rho t}}{t!} = \frac{\rho}{1 - \rho}, \tag{8}$$

and differentiating with respect to ρ,

$$\left(\frac{1}{\rho} - 1\right) \sum_{t=1}^{\infty} \frac{(\rho t)^t e^{-\rho t}}{(t - 1)!} = \frac{1}{(1 - \rho)^2}. \tag{9}$$

The left-hand expression is simply $\sum_{t=1}^{\infty} tP(t)$, so from Eq. (5),

$$z = bn(1 - \rho)^2. \tag{10}$$

Now in Eq. (7), we approximate the sum by an integral. Stirling's formula gives the inequality $t! > t^t e^{-t}\sqrt{2\pi t}$, from which

$$w \;\leq\; \frac{bn(1-\rho)^3}{\sqrt{2\pi\rho}} \int_{t_0}^{\infty} (\rho e^{1-\rho})^t t^{-1/2} dt \tag{11}$$

$$\leq\; \frac{bn(1-\rho)^3}{\sqrt{2\pi t_0 \rho}} \int_{t_0}^{\infty} (\rho e^{1-\rho})^t dt \tag{12}$$

$$\leq\; \frac{bn(1-\rho)^3}{\sqrt{2\pi t_0 \rho}} \frac{(\rho e^{1-\rho})^{t_0}}{\rho - 1 - \log\rho}. \tag{13}$$

Similarly, in Eq. (6), we approximate the sum by an integral but use a different (easily verified) inequality from Stirling's formula, $t! < t^t e^{-t}\sqrt{2\pi t}/(1 - 1/t)$. Then,

$$\delta \;>\; \frac{b(1-\rho)^3}{\sqrt{2\pi\rho}} \int_{t_0+1}^{\infty} (\rho e^{1-\rho})^t (t^{1/2} - t^{-1/2}) dt \tag{14}$$

$$>\; \frac{b(1-\rho)^3}{\sqrt{2\pi\rho}} \sqrt{t_0}\left(1 - \frac{1}{t_0}\right)\frac{(\rho e^{1-\rho})^{t_0+1}}{\rho - 1 - \log\rho}, \tag{15}$$

so

$$\frac{w}{n} \;<\; \frac{\delta(\rho e^{1-\rho})^{-1}}{t_0 - 1}. \tag{16}$$

From Eq. (2), it is not difficult to show that $\delta \leq \alpha - \alpha_c$. Furthermore, since the mantle must contain enough vertices to strip (or else this procedure cannot work), we know that $b \geq \delta$. One can solve Eqs. (1) and (2) numerically to find that this imposes the condition $(\rho e^{1-\rho})^{-1} \leq c_1$ with constant $c_1 \approx 1.40$ (which corresponds to $\alpha \lesssim 2.06$). Therefore,

$$\frac{w}{n} \;<\; c_1 \frac{\alpha - \alpha_c}{t_0 - 1}. \tag{17}$$

Finally, consider t_0 itself. From Eq. (15),

$$\delta > \frac{b(1-\rho)^3}{\sqrt{2\pi\rho}} \frac{(\rho e^{1-\rho})^{t_0+1}}{\rho - 1 - \log\rho}. \tag{18}$$

Since $\alpha > \alpha_c$, $g > 1/2$ and so $\rho = -(1-g)\log g/g < \log 2$. This leads to

$$\delta > b(\rho e^{1-\rho})^{t_0+1}/c_2, \tag{19}$$

where $c_2 = \sqrt{2\pi}\log 2(\log 2 - 1 - \log\log 2)/(1 - \log 2)^3 \approx 3.59$. Thus,

$$(\rho e^{1-\rho})^{t_0+1} \;<\; c_2\frac{\delta}{b}, \text{ so} \tag{20}$$

$$t_0 \;>\; \frac{\log b/\delta - \log c_2}{\rho - 1 - \log\rho} - 1 \tag{21}$$

$$>\; \frac{\log b/\delta - \log c_2}{\log c_1} - 1. \tag{22}$$

6

From Eq. (1),

$$\frac{b}{\delta} = \frac{\alpha(1/2 + \delta)(1/2 - \delta)}{\delta} \tag{23}$$

$$= \frac{\alpha}{4\delta}(1 - 4\delta^2), \text{ and so} \tag{24}$$

$$\log\frac{b}{\delta} = \log\frac{1}{\delta} + \log\frac{\alpha}{4} + \log(1 - 4\delta^2). \tag{25}$$

An additional consequence of the mantle condition $b \geq \delta$ discussed above is that $1 - 4\delta^2 \geq 1/c_3$ with constant $c_3 \approx 1.65$. Using this together with $\delta \leq \alpha - \alpha_c$,

$$t_0 > \frac{\log 1/(\alpha - \alpha_c) + \log \alpha/4c_2c_3}{\log c_1} - 1. \tag{26}$$

Inserting the t_0 bound into Eq. (17) gives

$$\frac{w}{n} < c_1 \log c_1 \frac{\alpha - \alpha_c}{\log 1/(\alpha - \alpha_c) + \log \alpha/4c_2c_3 - 2\log c_1}. \tag{27}$$

Since $\alpha > \alpha_c = 2\log 2$, if we define the constant $c_0 = -\log(\log 2/2) + \log c_2 c_3 + 2\log c_1 \approx 3.51$, we end up with the bisection width bound

$$\frac{w}{n} < c_1 \log c_1 \frac{\alpha - \alpha_c}{\log 1/(\alpha - \alpha_c) - c_0}. \tag{28}$$

Thus, defining $\epsilon = \alpha - \alpha_c$, when ϵ is small the bound behaves as

$$\frac{w}{n} \lesssim \frac{\epsilon}{\log(1/\epsilon)}. \tag{29}$$

We stress several points regarding this bound. First of all, the result closes the gap between upper and lower bounds on w/n at $\alpha = \alpha_c$, showing that both do indeed go to zero and the bisection width is continuous at the transition. Second of all, the first derivative is also continuous: w/ϵ itself goes to zero. This will be significant for the arguments in the next section. Finally, given the numerical value we have obtained for c_0, the denominator of Eq. (28) is only positive when ϵ is quite small indeed: our bound only holds for $\alpha - \alpha_c < 0.030$. However, our derivation uses very weak bounds on t_0, which can almost certainly be strengthened. The behavior in Eq. (29) would then extend to larger values of α.

3   Solution Cluster Structure

We now consider the geometry of the solution space for random graph bisection. For a graph $G \in \mathcal{G}_{np}$, given two distinct optimal bisections A and B, define the relative Hamming distance $\mu(A, B)$ to be the fraction of nodes that are in one of the subsets (V_1 or V_2) in solution A but in the opposite subset in solution B. Since there is a global symmetry under exchange of V_1 and V_2, we are interested in $d(A, B) = \min(\mu(A, B), 1 - \mu(A, B))$.

Define A and B to be r-*adjacent* if $d(A, B) \leq r$. For some small r, any two solutions that are connected by a chain of r-adjacent solutions are said to be in the same *solution cluster*.

Asymptotically, for a given mean degree α, we would like to know what the limiting cluster structure is when r becomes an arbitrarily small constant. One scenario that is common for sufficiently underconstrained combinatorial optimization problems is *replica symmetry*: for any finite $r > 0$, all solutions are with high probability within a single cluster. Another scenario, common for more constrained instances, is *one-step replica symmetry breaking*: for sufficiently small r, there is a large number of disconnected clusters, separated from each other by at least some finite relative Hamming distance.

In what follows, if solutions constructed in a particular way are with high probability r-adjacent for any finite $r > 0$, we will refer to them as being simply *adjacent*.

3.1 $\quad \alpha < 2 \log 2$

Consider graphs where optimal bisections require no cuts in the giant component. We first argue that we can almost surely create such a bisection by populating one of the two subsets only with the largest component and with monomers and dimers (components of size 2). We then show that having created a bisection in this way, it must be in the same cluster as *all* optimal bisections.

As discussed in Section 2, in $\mathcal{G}_{np}$ graphs for large n, a fraction $e^{-\alpha}$ of nodes are monomers. What about dimers? For a pair of nodes to be a dimer, it must contain an edge (probability $p \approx \alpha/n$), and all other pairs involving one of its two endpoints ($2n - 4$ of these) must not contain an edge (probability $1 - p$): asymptotically, this occurs with probability $\alpha e^{-2\alpha}/n$. Thus the expected number of dimer pairs is $\approx \alpha e^{-2\alpha} n/2$, meaning that the fraction of nodes in dimers is $\alpha e^{-2\alpha}$. In expectation, then, the fraction s of nodes in monomers or dimers is $s = e^{-\alpha}(1 + \alpha e^{-\alpha})$.

For $\alpha \leq 1$, $s > 1/2$. For $1 < \alpha < 2 \log 2$, s falls below $1/2$, but one may verify that $g + s > 1/2$ where g is the size of the giant component given in Eq. (2). Therefore, as long as $\alpha < 2 \log 2$, we almost surely have enough monomers and dimers to fill up a subset that otherwise contains only the largest component (or one of them, if not unique). Call such optimal solutions *clean bisections*.

Next, we show that any two distinct clean bisections A and B (and hence all clean bisections) must be in the same solution cluster. If the largest component is unique, A and B can differ at most by the choice of which monomers and dimers are selected to be in its subset. If the largest component is not unique, it cannot be a giant component, so two clean bisections can then differ at most by: 1) a component consisting of a vanishing fraction of nodes, and 2) monomers and dimers. In both cases, we can take any one of the differing components in A and swap it with monomers and dimers from the other subset, producing an adjacent clean bisection. By iterating the process, we will necessarily arrive at B through a chain of adjacent clean bisections, since we have already shown that there is a sufficient supply of monomers and dimers for this purpose.

Finally, it is straightforward to see that any optimal solution must be in the same cluster as some clean bisection. This holds for almost the same reason as in the previous paragraph. If the subset containing the largest component also contains any components other than monomers and dimers, each of these asymptotically consists of a vanishing fraction of nodes. They may be iteratively swapped with monomers and dimers to produce a chain of adjacent solutions, leading to a clean bisection.

Since all clean bisections are in the same cluster, and any solution is in the same cluster as some clean bisection, all solutions must lie within a single cluster for $\alpha < 2\log 2$. This is a replica symmetric phase.

3.2 $\quad \alpha \geq 2\log 2$

When the bisection width is nonzero, the situation is more complicated. Now we must consider the different ways in which optimal bisections can cut the giant component. We will show that up to some $\alpha_d > \alpha_c$, optimal bisections avoid a central core of the giant component and are restricted to a well-defined periphery. We then give arguments suggesting that for those graphs, all optimal bisections are in a single cluster.

3.2.1 The expander core

Call the giant component S, and imagine that for any $\alpha \geq 2\log 2$, some identifiable part of it, $X \subseteq S$, is an *expander*. This means that cutting off any piece X_0 from X requires (asymptotically) at least some finite *expansion*

$$h = \frac{|E(X \setminus X_0, X_0)|}{\min(|X \setminus X_0|, |X_0|)}, \tag{30}$$

i.e., the number of cut edges per vertex in the broken piece. The expander X would then be precisely the central core that optimal bisections avoid, for $\epsilon = \alpha - \alpha_c$ less than some constant. The reason is as follows. A slightly weaker version of our bisection width upper bound Eq. (29) is

$$\frac{w}{n} \lesssim \frac{\delta}{\log(1/\epsilon)} \tag{31}$$

where we recall that $\delta = g - 1/2$ is the excess of the giant component. But this means that cutting trees *only*, as in the greedy algorithm in Section 2, gives an expansion of

$$h = \frac{w}{\delta n} \lesssim \frac{1}{\log(1/\epsilon)}. \tag{32}$$

For sufficiently small ϵ, this must fall below the (finite) value of h for the expander. Thus, below some finite $\epsilon_0 > 0$, trimming the giant component's excess requires fewer cuts if these are performed exclusively within the mantle than if any part of the cuts are performed within the core.

One might hope that the 2-core is an expander, in which case greedily trimming trees would in fact be the optimal way to bisect a graph. As it happens, this is almost certain not true. Consider the 2-core's spectral gap, $\lambda_1 - \lambda_2$, which via Cheeger's inequality [10] yields an upper bound on expansion. Figure 3 suggests that the spectral gap likely vanishes as $n \to \infty$, consistent with the 2-core's expansion vanishing. Certain "bottlenecks" in the 2-core could then provide more efficient cuts than trees in the mantle.

But some further intuition comes from considering *random* cuts in the 2-core. Figure 4 shows cutsizes from a number of ways of slicing the 2-core into two pieces C_1 and C_2 ($|C_1| \leq |C_2|$). These numerical results, which we originally reported in [17], suggest that regardless
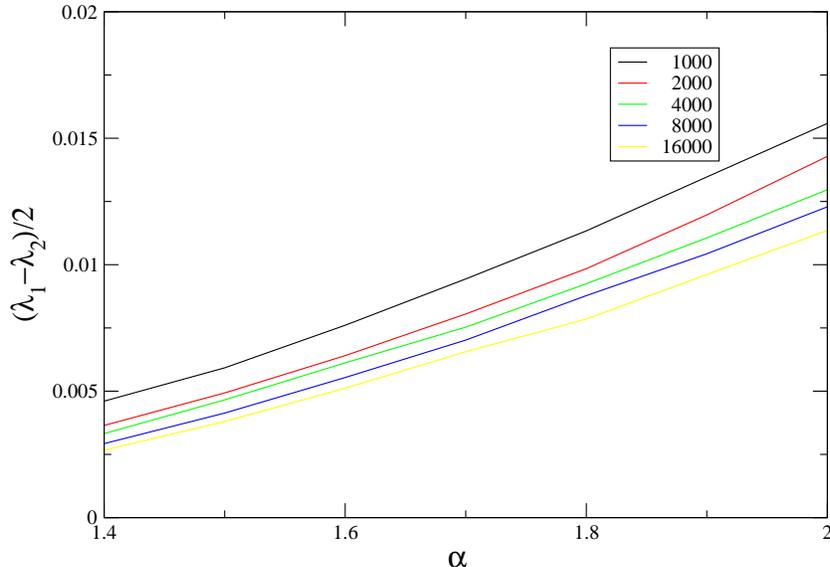
Figure 3: Spectral gap for mean degree $\alpha = 1.4$ through $\alpha = 2.0$, on graphs from size $n = 1000$ up to $n = 16000$.

of the size of $|C_1|$, the minimum number of cut edges (over an admittedly small sample of cuts) never falls below $|C_1|$. This means that any bottlenecks that do exist in the 2-core are not immediately apparent, and some significant part of the 2-core could indeed be an expander.

Subsequent results by Benjamini et al. [2] have confirmed that this is indeed the case: for all $\alpha > 1$, they give an explicit procedure for stripping the giant component down to an expander core. The giant component is a "decorated expander," with many small (at most $O(\log n)$) "decorations" D_i attached to the expander core X, as shown schematically in Figure 5. Therefore, by our previous argument, below some finite ϵ_0, the cuts in all optimal bisections are restricted to the decorations D_i.

3.2.2 Consequences for solution structure

We now use arguments related to the ones in Section 3.1 to suggest that for $\epsilon < \epsilon_0$ ($\alpha < \alpha_d = 2 \log 2 + \epsilon_0$), all optimal solutions are contained within a single cluster. We first show how to construct a *neat* bisection. By *neat*, we mean that one of the two subsets contains only the expander core, some attached decorations, and monomers. We then present arguments suggesting that this bisection, even if not optimal, is adjacent to a neat optimal bisection that is in the same cluster as *all* optimal bisections, thus replica symmetry holds for all $\alpha < \alpha_d$.

The construction procedure is as follows. Consider the giant component S as being composed of the expander core X and decorations D_i as in Figure 5. These are disjoint subsets, with some edges connecting the D_i to X but none connecting the D_i to each other [2]. For each decoration D_i, find the nonempty subset $D_i^1 \subseteq D_i$ that minimizes the expansion

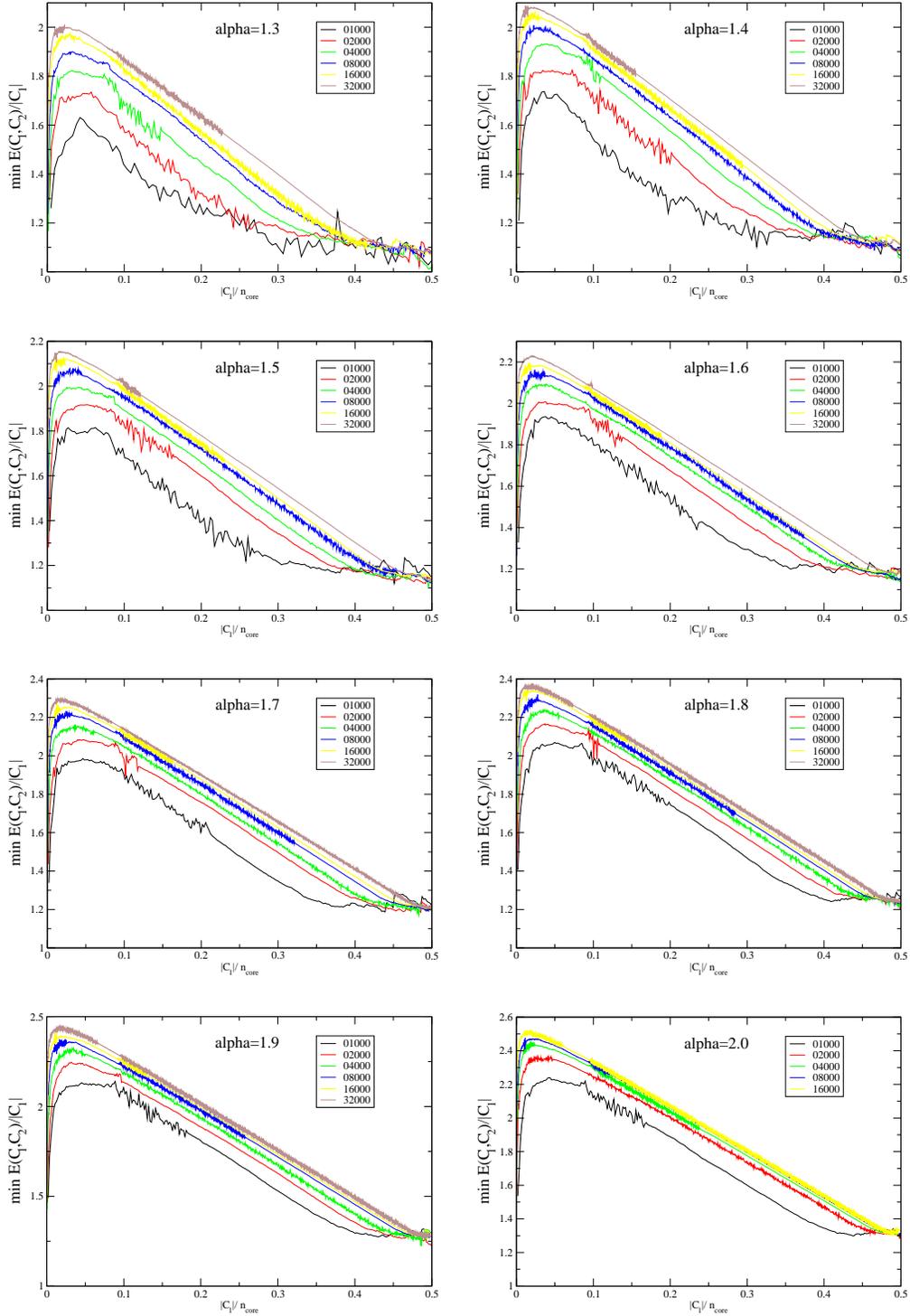$$h_i^1 = \frac{|E(S \setminus D_i^1, D_i^1)|}{|D_i^1|} \tag{33}$$

10

Figure 4: Smallest cuts over a random sample of ways of slicing the 2-core into two pieces C_1 and C_2 ($|C_1| \leq |C_2|$). For each value of mean degree α, and for each size from $n = 1000$ to $n = 32000$, results are shown for 100 graphs, in each of which 10 different random partitions of the 2-core are chosen. Even when partitions are perfectly balanced (right side of plots), expansion never falls below 1.
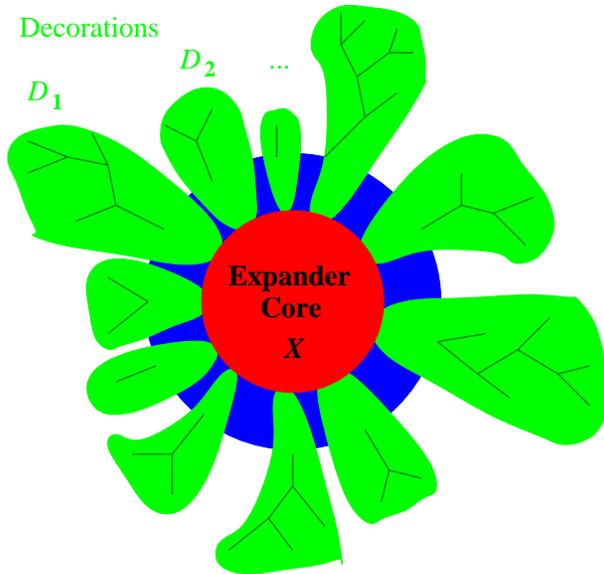
11

Figure 5: A different view of the giant component in Figure 2, shown here as an identifiable *expander core* X within the 2-core. The trees are now contained within numerous small *decorations* D_i that are attached to the expander core.

resulting from cutting off the piece D_i^1. The value of h_i^1 may be interpreted as a "cost" per node that we attribute to each node in D_i^1. Now repeat this procedure on $D_i \setminus D_i^1$, resulting in a new piece D_i^2, except that this time we find the D_i^2 minimizing the *marginal* cost of cutting off D_i^2 given that D_i^1 is already cut off, i.e.,

$$h_i^2 = \frac{|E(S \setminus (D_i^1 \cup D_i^2), D_i^1 \cup D_i^2)| - |E(S \setminus D_i^1, D_i^1)|}{|D_i^2|}. \tag{34}$$

Recursively perform the procedure, letting

$$h_i^j = \frac{|E(S \setminus \bigcup_{l=1}^{j} D_i^l, \bigcup_{l=1}^{j} D_i^l)| - h_i^{j-1}|D_i^{j-1}|}{|D_i^j|}, \tag{35}$$

until decoration D_i has been completely partitioned into disjoint subsets $D_i^1, \ldots, D_i^m$. By construction, these are now ordered in increasing marginal cost $h_i^1 \leq \cdots \leq h_i^m$, where the total cost $|E(S, D_i)|$ of cutting the entire decoration is simply equal to the sum of the marginal costs over all nodes,

$$|E(S, D_i)| = \sum_{j=1}^{m} h_i^j |D_i^j|. \tag{36}$$

Note that h_i^j is the *minimum possible* cost of cutting a node in D_i^j. Since cuts in optimal solutions are restricted to decorations, generating a neat optimal bisection requires an aggregated ranking of all subsets of all decorations, D_i^j, according to their costs h_i^j. Analogously to the greedy algorithm in Section 2, where we stripped trees in increasing order of expansion, we now strip the pieces of the decorations until the giant component's excess

12

has been removed. As with the greedy algorithm, this can result in overstripping. There may also be numerous pieces with an equal ranking, leaving the choice of which ones to pick in order to overstrip as little as possible. But the largest of these pieces is $O(\log n)$. So regardless of which we choose, as with tree-stripping, we can compensate for the imbalance by transferring some of the extensive supply of monomers. The result is a neat bisection, albeit not necessarily an optimal one.

In principle, by appropriately selecting the final pieces of equal rank and thereby minimize the cost of overstripping, we can even form any neat *optimal* bisection in this way. In practice, no matter how we select those final pieces, our construction seems highly likely to give a solution that differs from optimal ones by at most the number of variables contained in a finite number of these pieces, which is $O(\log n)$. Neat optimal bisections that use the identical choice of compensating monomers are then all in a single cluster. But in that case, *every* neat optimal bisection must be in the same cluster, since successively swapping monomers gives a chain of adjacent solutions connecting them.

Finally, for similar reasons as in Section 2, any optimal bisection must be in the same cluster as some neat optimal bisection. If the subset containing the expander core also contains components other than monomers, each one can have size at most $O(\log n)$ and can be swapped with monomers from the other subset to produce an adjacent optimal bisection. Iterating this process leads to a neat optimal bisection. Consequently, the argument implies that for $2\log 2 < \alpha < \alpha_d$, all optimal bisections lie within a single cluster, and the replica symmetric phase extends up to $\alpha_d > \alpha_c$.

Above α_d, the solution structure is less clear. When α is large enough, optimal cuts are no longer necessarily restricted to the decorations: the expander core likely needs to be cut as well. In that case, the sizes of the cut pieces could well be extensive. One could easily imagine extensive gaps in the solution structure, leading to multiple clusters and replica symmetry breaking. Although we have no direct evidence of it, we speculate that this is the case for α greater than some finite value, suggesting an RSB phase boundary at or above α_d, i.e., above the critical threshold α_c.

4   Discussion and Conclusions

Our study points to an unusual kind of phase structure for random graph bisection. The typical scenario, in random combinatorial optimization problems displaying a sharp threshold in the objective function's behavior, is that an RSB transition takes place at or below the main critical threshold: $\alpha_d \leq \alpha_c$. By contrast, graph bisection appears to be replica symmetric up to *and beyond* the threshold, with $\alpha_d > \alpha_c$. We note that this lends support to an analytical bisection width prediction proposed by Liao in 1987 [22]. Let us now comment on several further aspects of α_d, as well as on the intriguing algorithmic consequences of this phase picture.

Although we have argued that α_d is strictly greater than α_c, it may not be much greater. As mentioned at the end of Section 2, our bisection width bound is only valid for $\alpha - \alpha_c < 0.030$. That bound is needed for being able to restrict optimal cuts to the giant component's decorations. And even if the arguments could be strengthened to extend it to larger α — which should be possible, given that we have settled for very weak bounds in our derivation
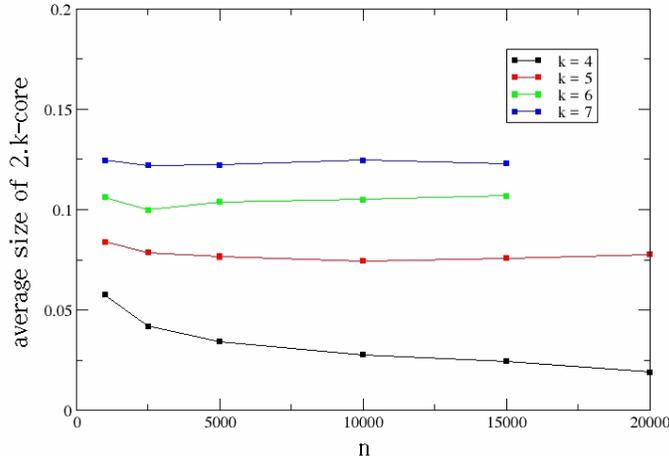
13

Figure 6: Fraction of nodes in the 2.k-core, $k = 4$ through $k = 7$, at $\alpha = 1.4$. Numerical results are averaged over 200 instances, for increasing values of n.

— we do not presently have any other quantitative bound or estimate on the specific value of α_d. However, some further qualitative observations come from examining the procedure for obtaining the expander core. Roughly speaking, Benjamini et al. [2] strip the giant component down to what we call a *2.k-core*, namely a connected subgraph of the giant component where all vertices have degree at least 2 (no trees) *and* where the largest chain is of length k. The 2.k-core is an expander as long as k is a sufficiently large finite constant. The construction of [2] does not explicitly state how large k must be, or what the minimum expansion value h is for the 2.k-core at a given mean degree α. But decreasing values of k clearly result in smaller 2.k-cores (Figure 6), and presumably in larger expansion as more long chains are removed. This would lead to a higher α_d. Thus, by picking the smallest possible k (for every α) that still yields an expander core, the resulting expansion gives, via Eq. (32), the largest possible α_d for which we can be sure that graph bisection is replica symmetric at all $\alpha < \alpha_d$.

Algorithmically, the striking consequence of our study is that we may obtain near-optimal bisections and perhaps even optimal ones, for all $\alpha < \alpha_d$, in polynomial time. By near-optimal, we mean that the ratio of its bisection width to the optimal value approaches 1 asymptotically. We do this by stripping the giant component down to the expander core (using any sufficiently large k), and then following the construction in Section 3.2.2. The stripping process and decoration identification [2] takes polynomial time. The next subtlety is breaking a decoration D_i into the correct disjoint pieces D_i^j: an exhaustive approach to finding D_i^1 involves comparing all of the $2^{|D_i|}$ ways of partitioning D_i into two subsets. But since no decoration D_i has more than $O(\log n)$ nodes, this can be done in $n^{O(1)}$ steps, and then all pieces D_i^j can be found and ranked ($h_i^1 \leq \cdots \leq h_i^m$) in $n^{O(1)}$ steps. For the final pieces of equal rank, even if we simply pick them in arbitrary order we will overstrip by at most $O(\log n)$. Thus, in polynomial time we reach a solution whose (extensive) bisection width is within $O(\log n)$ of the optimal one.

Of course, the $O(1)$ constant in the exponent may be quite large, which could make this result largely a theoretical one. But it is worth noting that we have some freedom to decrease

the size of the decorations, and thus lower the constant. If, say, instead of using the smallest possible k that gives an expander core, we use the *largest* possible k that gives expansion greater than our tree-cutting bound (Eq. (32)), this will strip the minimum possible from the giant component. In this way, we may be able to minimize the size of the largest decoration.

It would be particularly remarkable if some version of our construction could give a true optimal bisection, with high probability, in polynomial time. For this, we need to pick the final pieces of equal rank so as to minimize overstripping. The difficulty is that there is likely to be an infinite number of such pieces: an extensive number of nodes is contained collectively within the decorations, while there are at most $O(\log^2 n)$ possible marginal costs ($O(\log n)$ possible integers for the numerator and $O(\log n)$ possible integers for the denominator). On the other hand, it could well be that of that infinite number of final pieces, a finite fraction of them have any given finite size $|D_i^j|$. In that case, a simple greedy procedure will find a collection of pieces that minimizes overstripping. Having minimized overstripping, one still needs to check whether it is possible to lower the bisection width further by restoring a small fragment of a cut piece. If the amount we overstrip is $O(\log n)$, then exhaustively checking this could require $\sim \binom{n}{\log n}$ steps. But more likely, one can strengthen our analysis to show that the final pieces are in fact of size $O(1)$, in which case the last operation could also be accomplished in polynomial time. Clearly, this intuition is far from a proof of any sort. But if indeed the optimal bisection can be found in polynomial time for $\alpha < \alpha_d$, we would have a very unusual example of an NP-hard optimization problem whose typical-case complexity is polynomial at *and above* the main threshold α_c.

We offer two final remarks about other versions of the problem. First, our approach suggests that the peculiarities of random graph bisection's phase structure are closely related to the problem's global constraint — the fact that a subset cannot be larger than $n/2$ — combined with the specific topology of the giant component in $\mathcal{G}_{np}$. This means that the problem of balanced multiway graph partitioning on $\mathcal{G}_{np}$ graphs should have very similar characteristics, and can likely be analyzed through similar means. Second, the situation is very different if instead of $\mathcal{G}_{np}$ we use a graph ensemble that has local geometric structure. An example is the random geometric graph ensemble $\mathcal{G}_{nr}$ [29], where nodes are placed uniformly at random in a unit square and pairs are connected by an edge if separated by distance $\leq r$. In this case, the giant component is very unlikely to have any expander core due to its large diameter. Rather than having a tree-like periphery, it contains numerous local cliques, and minimizing the bisection width involves locating the bottlenecks in between those cliques. But at the same time, $\mathcal{G}_{nr}$ graphs have useful lattice-like properties [15], with the optimal bisection width scaling [29, 7] as $w \sim \sqrt{n}$, precisely as one finds on a regular 2-dimensional lattice structure. We expect that the structure of bisections is closely related to the geometry of the giant component in bond percolation on a lattice, which has been studied [3]. Adapting that analysis could lead to nontrivial bounds on w and rapid algorithms finding very close approximations to the optimal cut in $\mathcal{G}_{nr}$.

5   Acknowledgments

References

[1] C. J. Alpert and A. B. Kahng. Recent directions in netlist partitioning: a survey. *Integration, the VLSI Journal*, 19:1–81, 1995.

[2] I. Benjamini, G. Kozma, and N. Wormald. The mixing time of the giant component of a random graph. arXiv:math.PR/0610459, 2006.

[3] I. Benjamini and E. Mossel. On the mixing time of a simple random walk on the supercritical percolation cluster. *Probability Theory and Related Fields*, 125:408–420, 2003.

[4] G. Biroli, R. Monasson, and M. Weigt. A variational description of the ground state structure in random satisfiability problems. *European Physical Journal B*, 14:551–568, 2000.

[5] S. Boettcher. Extremal optimization of graph partitioning at the percolation threshold. *Journal of Physics A: Mathematical and General*, 32:5201–5211, 1999.

[6] S. Boettcher and A. G. Percus. Nature's way of optimizing. *Artifical Intelligence*, 119:275–286, 2000.

[7] S. Boettcher and A. G. Percus. Extremal optimization for graph partitioning. *Physical Review E*, 64:026114, 2001.

[8] B. Bollobas. *Random Graphs*. Cambridge University Press, Cambridge, UK, 2001.

[9] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23:1222–1239, 2001.

[10] J. Cheeger. A lower bound for the smallest eigenvalue of the Laplacian. In *Problems in analysis (Papers dedicated to Salomon Bochner, 1969)*, pages 195–199. Princeton Univ. Press, Princeton, N. J., 1970.

[11] P. Cheeseman, B. Kanefsky, and W. F. Taylor. Where the *really* hard problems are. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI'91)*, pages 331–337, San Mateo, CA, 1991. Morgan Kaufmann.

[12] P. Erdős and A. Rényi. On random graphs. *Publ. Math. Inst. Hungar. Acad. Sci.*, 1959.

[13] Y. Fu and P. Anderson. Application of statistical mechanics to np-complete problems in combinatorial optimisation. *Journal of Physics A: Mathematical and General*, 19(9):1605–1620, 1986.

[14] M. R. Garey and D. S. Johnson. *Computers and Intractability.* W.H. Freeman, New York, 1979.

[15] A. Goel, S. Rai, and B. Krishnamachari. Monotone properties of random geometric graphs have sharp thresholds. *Annals of Applied Probability*, 15:2535–2552, 2005.

[16] M. Goldberg and J. Lynch. Lower and upper bounds for the bisection width of a random graph. *Congr Numer*, 49, 1985.

[17] B. Gonçalves, G. Istrate, A. Percus, and S. Boettcher. The core peeling algorithm for graph bisection: an experimental evaluation. Technical Report LA-UR 06-6863, Los Alamos National Laboratory, 2006.

[18] P. Holme, B. J. Kim, C. N. Yoon, and S. K. Han. Attack vulnerability of complex networks. *Phys. Rev. E*, 65:056109, 2002.

[19] S. Janson, T. Łuczak, and A. Ruciński. *Random Graphs.* John Wiley & Sons, Inc., New York, 2000.

[20] I. Kanter and H. Sompolinsky. Mean-field theory of spin-glasses with finite coordination number. *Physical Review Letters*, 58(2):164–167, 1987.

[21] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:147–169, 2004.

[22] W. Liao. Graph bipartitioning problem. *Physical Review Letters*, 59(15):1625–1628, 1987.

[23] W. Liao. Towards the solution of the graph bipartitioning problem. *Journal of Physics A: Mathematical and General*, 20(11):L695–L699, 1987.

[24] M. J. Łuczak and C. McDiarmid. Bisecting sparse random graphs. *Random Struct. Algorithms*, 18:31–38, 2001.

[25] M. Mézard and G. Parisi. Mean-field theory of randomly frustrated systems with finite connectivity. *Europhysics Letters*, 3(10):1067–1074, 1987.

[26] M. Mézard, G. Parisi, and R. Zecchina. Analytic and algorithmic solutions of random satisfiability problems. *Science*, 297:812–815, 2002.

[27] M. Mézard and R. Zecchina. Random K-satisfiability problem: from an analytical solution to an efficient algorithm. *Physical Review E*, 66:056126, 2002.

[28] D. G. Mitchell, B. Selman, and H. J. Levesque. Hard and easy distributions for SAT problems. In P. Rosenbloom and P. Szolovits, editors, *Proceedings of the Tenth National Conference on Artificial Intelligence*, pages 459–465, Menlo Park, California, 1992. AAAI Press.

[29] M. D. Penrose. *Random Geometric Graphs.* Oxford University Press, Oxford, 2003.

[30] A. G. Percus, G. Istrate, and C. Moore. *Computational Complexity and Statistical Physics*. Oxford University Press, New York, 2006.

[31] B. Pittel. On tree census and the giant component in sparse random graphs. *Random Struct. Algorithms*, 1:311–342, 1990.